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

H8/36912 Group, H8/36902 Group

Hardware Manual

Renesas 16-Bit Single-Chip Microcomputer
H8 Family/H8/300H Tiny Series

H8/36912F	HD64F36912G
H8/36902F	HD64F36902G
H8/36912	HD64336912G
H8/36911	HD64336911G
H8/36902	HD64336902G
H8/36901	HD64336901G
H8/36900	HD64336900G

Renesas Electronics
www.renesas.com

Rev.3.00 2006.09

a third party.

2. Renesas Technology Corp. assumes no responsibility for any damage, or infringement of any party's rights, originating in the use of any product data, diagrams, charts, programs, algorithm circuit application examples contained in these materials.
3. All information contained in these materials, including product data, diagrams, charts, program algorithms represents information on products at the time of publication of these materials, and subject to change by Renesas Technology Corp. without notice due to product improvements other reasons. It is therefore recommended that customers contact Renesas Technology Co an authorized Renesas Technology Corp. product distributor for the latest product information before purchasing a product listed herein.
   The information described here may contain technical inaccuracies or typographical errors.
   Renesas Technology Corp. assumes no responsibility for any damage, liability, or other loss 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.com
4. When using any or all of the information contained in these materials, including product data, diagrams, charts, programs, and algorithms, please be sure to evaluate all information as a to system before making a final decision on the applicability of the information and products. Re Technology Corp. assumes no responsibility for any damage, liability or other loss resulting fr information contained herein.
5. Renesas Technology Corp. semiconductors are not designed or manufactured for use in a de system that is used under circumstances in which human life is potentially at stake. Please c Renesas Technology Corp. or an authorized Renesas Technology Corp. product distributor w considering the use of a product contained herein for any specific purposes, such as apparat 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 co 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 produc contained therein.

are in their open states, intermediate levels are induced by noise in the vicinity, a 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 throughou chip and a low level is input on the reset pin. During the period where the states a undefined, the register settings and the output state of each pin are also undefined your system so that it does not malfunction because of processing while it is in th undefined state. For those products which have a reset function, reset the LSI imr 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 reg may have been be allocated to these addresses. Do not access these registers; the operation is not guaranteed if they are accessed.

- CPU and System-Control Modules
- On-Chip Peripheral Modules

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

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

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

7. List of Registers
8. Electrical Characteristics
9. Appendix
10. Main Revisions and Additions in this Edition (only for revised versions)

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

11. Index

microcomputers.

Objective: This manual was written to explain the hardware functions and electrical characteristics of the H8/36912 Group and H8/36902 Group to the target Refer to the H8/300H Series Software Manual for a detailed description o instruction set.

Notes on reading this manual:

- In order to understand the overall functions of the chip

  Read the manual according to the contents. This manual can be roughly categorized in on the CPU, system control functions, peripheral functions and electrical characteristi
- In order to understand the details of the CPU's functions

  Read the H8/300H Series Software Manual.
- In order to understand the details of a register when its name is known

  Read the index that is the final part of the manual to find the page number of the entry register. The addresses, bits, and initial values of the registers are summarized in secti List of Registers.

  Example: Register name: The following notation is used for cases when the sa similar function, e.g. serial communication interface implemented on more than one channel:
  XXX_N (XXX is the register name and N is the cha number)

  Bit order: The MSB is on the left and the LSB is on the right.

5. When the E7 or E8 is used, $\overline{\text{NMI}}$ is an input/output pin (open-drain in output mode).

Related Manuals: The latest versions of all related manuals are available from our we
Please ensure you have the latest versions of all documents you rec
http://www.renesas.com/

H8/36912 Group and H8/36902 Group manuals:

Document Title	Docume
H8/36912 Group, H8/36902 Group Hardware Manual	This ma
H8/300H Series Software Manual	REJ09B

User's manuals for development tools:

Document Title	Docume
H8S, H8/300 Series C/C++ Compiler, Assembler, Optimizing Linkage Editor User's Manual	REJ10B
H8S, H8/300 Series Simulator/Debugger User's Manual	REJ10B
H8S, H8/300 Series High-Performance Embedded Workshop 3 Tutorial	REJ10B
H8S, H8/300 Series High-Performance Embedded Workshop 3 User's Manual	REJ10B

Application notes:

Document Title	Docume
H8S, H8/300 Series C/C++ Compiler Package Application Note	REJ05B
Single Power Supply F-ZTAT™ On-Board Programming	REJ05B

Section 2 CPU

Section 3 Exception Handling

Section 4 Address Break

Section 15 I^2C Bus Interface 2 (IIC2)

Appendix

— Timer B1* (8-bit timer)
— Timer V (8-bit timer)
— Timer W (16-bit timer)
— Watchdog timer
— SCI3 (Asynchronous or clocked synchronous serial communication interface)
— 10-bit A/D converter
— I^2C bus interface* (conforms to the Philips I^2C bus interface functions)
— POR/LVD (Power-on reset and low-voltage detection circuits)
— Address break

Note: * Available for the H8/36912 Group only.

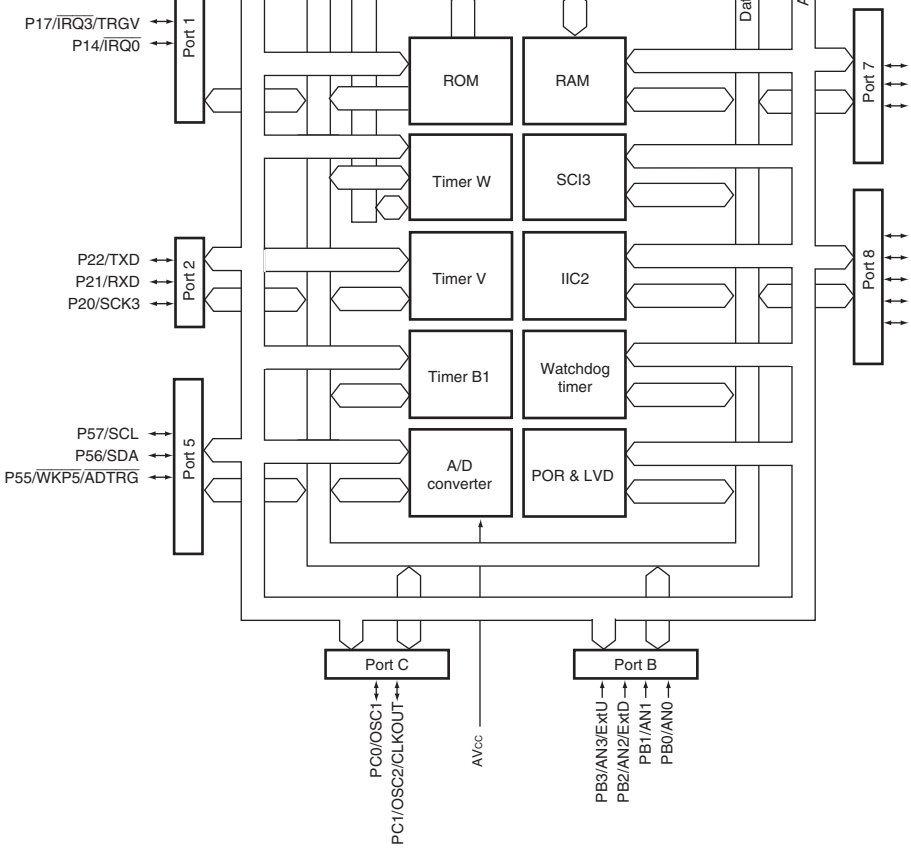
- On-chip memory

Product Classification		Type	ROM	RAM	Remar
Flash memory version (F-ZTAT™ version)	H8/36912F	HD64F36912G	8 kbytes	1,536 bytes	
	H8/36902F	HD64F36902G	8 kbytes	1,536 bytes	
Masked ROM version	H8/36912	HD64336912G	8 kbytes	512 bytes	
	H8/36911	HD64336911G	4 kbytes	256 bytes	
	H8/36902	HD64336902G	8 kbytes	512 bytes	
	H8/36901	HD64336901G	4 kbytes	256 bytes	
	H8/36900	HD64336900G	2 kbytes	256 bytes	

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

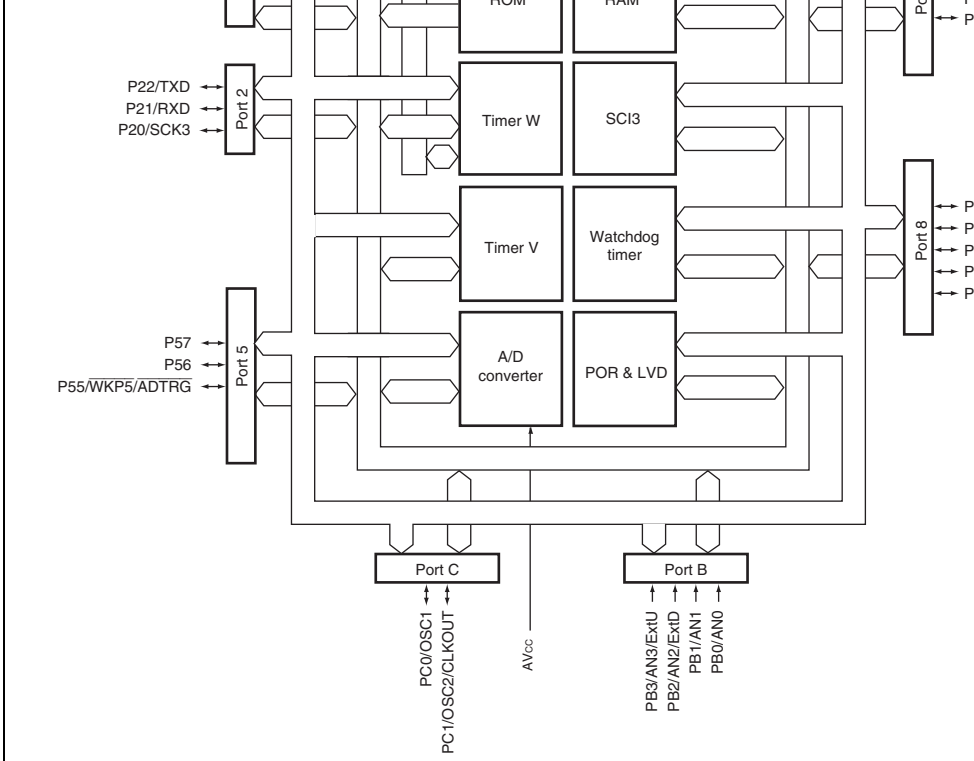
Package	Code	Body Size	Pin Pitch	Remarks
LQFP-32	FP-32A	7.0 × 7.0 mm	0.8 mm	
SOP-32	FP-32D	11.3 × 20.45 mm	1.27 mm	
SDIP-32*	32P4B	400 mil	1.78 mm	

Note: * Flash memory version only



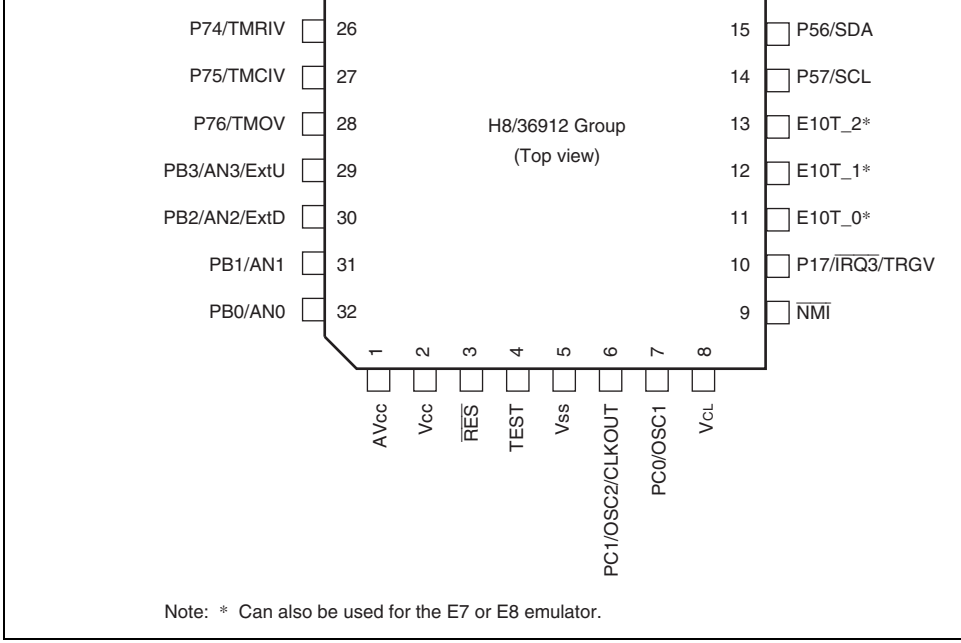
Note: * Can also be used for the E7 or E8 emulator.

Figure 1.1 Internal Block Diagram of H8/36912 Group



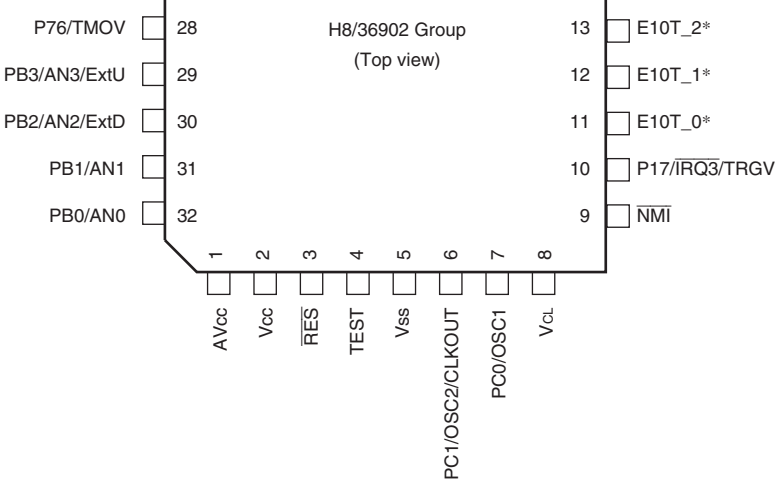
Note: * Can also be used for the E7 or E8 emulator.

Figure 1.2 Internal Block Diagram of H8/36902 Group



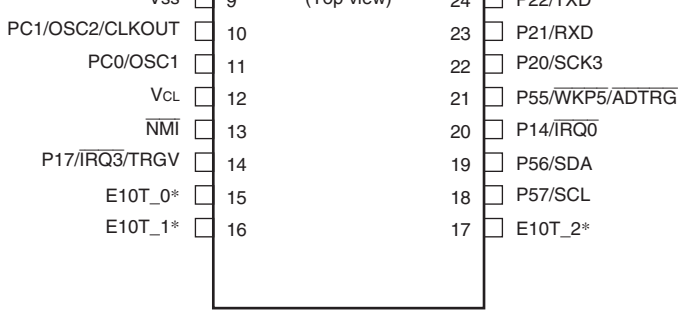
Note: * Can also be used for the E7 or E8 emulator.

Figure 1.3 Pin Arrangement of H8/36912 Group (FP-32A)



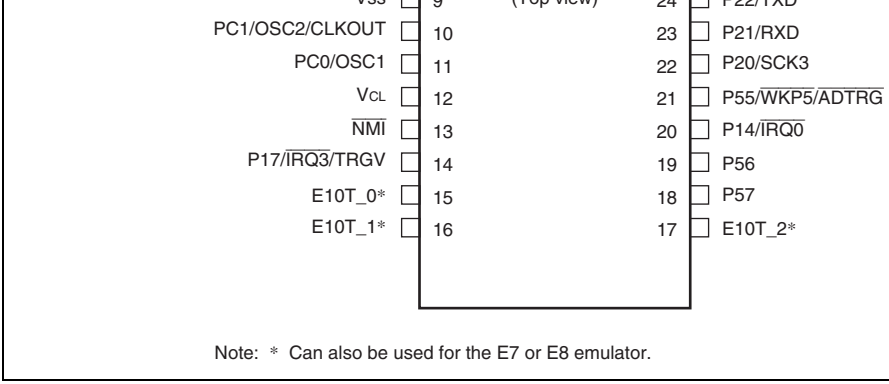
Note: * Can also be used for the E7 or E8 emulator.

Figure 1.4 Pin Arrangement of H8/36902 Group (FP-32A)



Note: * Can also be used for the E7 or E8 emulator.

Figure 1.5 Pin Arrangement of H8/36912 Group (FP-32D, 32P4B)



Note: * Can also be used for the E7 or E8 emulator.

Figure 1.6 Pin Arrangement of H8/36902 Group (FP-32D, 32P4B)

	AV_{CC}	5	1	Input	Analog power supply pin A/D converter. When the converter is not used, co pin to the system power s
	V_{CL}	12	8	Input	Internal step-down power pin. Connect a capacitor 0.1 μF between this pin a pin for stabilization.
Clock	OSC1	11	7	Input	These pins are connecte crystal or ceramic resona system clocks, or can be input an external clock. W on-chip oscillator is used, clocks can be output to C section 5, Clock Pulse Ge for a typical connection.
	OSC2/ CLKOUT	10	6	Output	
System control	$\overline{RES}$	7	3	Input	Reset pin. The pull-up res 150 kΩ) is incorporated. W driven low, the chip is res
	TEST	8	4	Input	Test pin. Connect this pin
External interrupt	$\overline{NMI}$	13	9	Input	Non-maskable interrupt r input pin. Be sure to pull- pull-up resistor.
	$\overline{IRQ0}$, $\overline{IRQ3}$	20, 14	16, 10	Input	External interrupt request pins. Can select the rising edge.
	$\overline{WKP5}$	21	17	Input	External interrupt request Can select the rising or fa

	FTIOA to FTIOD	26 to 29	22 to 25	I/O	Output compare output/ inp capture input/ PWM output pins
I^2C bus interface 2*	SDA	19	15	I/O	I^2C data I/O pin. NMOS op output can directly drive th
	SCL	18	14	I/O	I^2C clock I/O pin. NMOS op output can directly drive th
Serial communi-cation interface	TXD	24	20	Output	Transmit data output pin
	RXD	23	19	Input	Receive data input pin
	SCK3	22	18	I/O	Clock I/O pin
A/D converter	AN3 to AN0	1 to 4	29 to 32	Input	Analog input pin
	$\overline{\text{ADTRG}}$	21	17	Input	A/D converter trigger input
I/O ports	P17, P14	14, 20	10, 16	I/O	2-bit I/O port
	P22 to P20	24 to 22	20 to 18	I/O	3-bit I/O port
	P57 to P55	18, 19, 21	14, 15, 17	I/O	3-bit I/O port
	P76 to P74	32 to 30	28 to 26	I/O	3-bit I/O port
	P84 to P80	29 to 25	25 to 21	I/O	5-bit I/O port
	PB3 to PB0	1 to 4	29 to 32	Input	4-bit input port
	PC1, PC0	10, 11	6, 7	I/O	2-bit I/O port
Low voltage detection circuit	ExtU, ExtD	1, 2	29, 30	Input	External input pins for the voltage used in the low-vol detection circuit
E7, E8	E10T_0, E10T_1, E10T_2	15, 16, 17	11, 12, 13	—	Interface pins for the E7 or emulator

Note: * Available for the H8/36912 Group only.

- General-register architecture
  - Sixteen 16-bit general registers also usable as sixteen 8-bit registers or eight 32-b
- Sixty-two basic instructions
  - 8/16/32-bit data transfer and 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:24,ERn)]
  - Register indirect with post-increment or pre-decrement [@ERn+ or @–ERn]
  - Absolute address [@aa:8, @aa:16, @aa:24]
  - Immediate [#xx:8, #xx:16, or #xx:32]
  - Program-counter relative [@(d:8,PC) or @(d:16,PC)]
  - Memory indirect [@@aa:8]
- 64-kbyte address space
- High-speed operation
  - All frequently-used instructions execute in two or four states
  - 8/16/32-bit register-register add/subtract : 2 state
  - 8 × 8-bit register-register multiply : 14 states
  - 16 ÷ 8-bit register-register divide : 14 states
  - 16 × 16-bit register-register multiply : 22 states
  - 32 ÷ 16-bit register-register divide : 22 states
- Power-down state
  - Transition to power-down state by SLEEP instruction

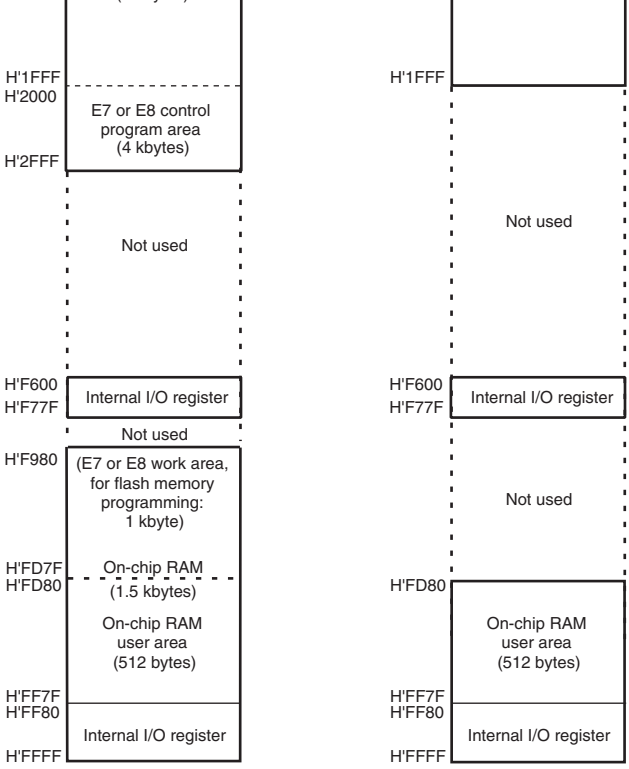


Figure 2.1 Memory Map (1)

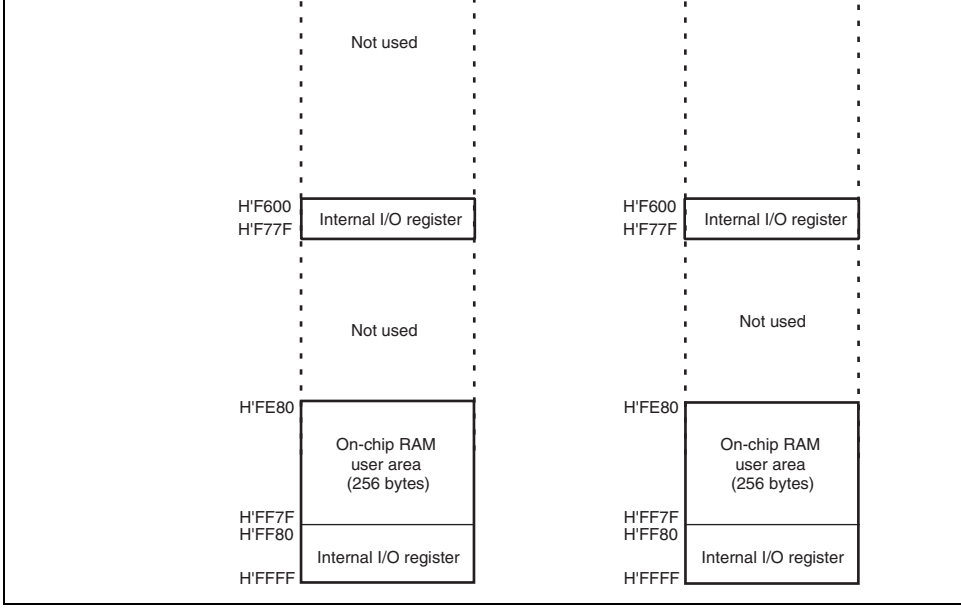
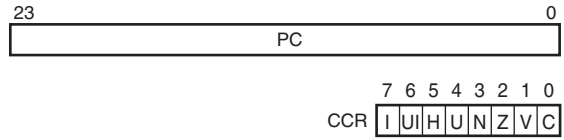


Figure 2.1 Memory Map (2)

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
CCR:	Condition-code register	N:	Negative flag
I:	Interrupt mask bit	Z:	Zero flag
UI:	User bit	V:	Overflow flag
		C:	Carry flag

Figure 2.2 CPU Registers

The R registers divide into 8-bit registers designated by the letters RH (R0H to R7H) an
to R7L). These registers are functionally equivalent, providing a maximum of sixteen 8-
registers.

The usage of each register can be selected independently.

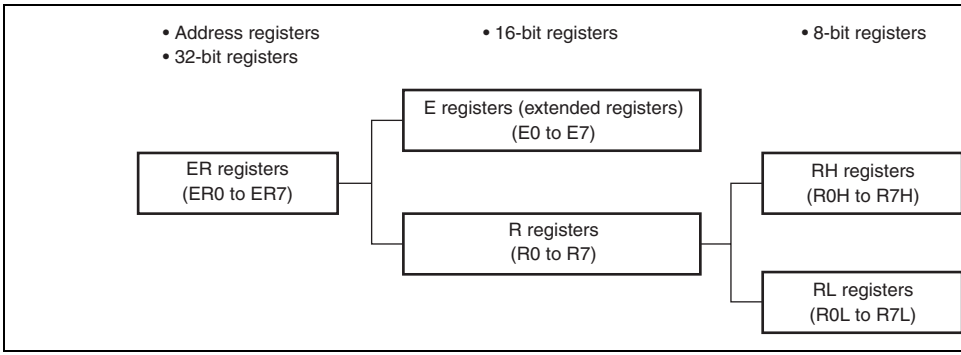


Figure 2.3 Usage of General Registers

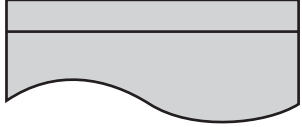


Figure 2.4 Relationship between Stack Pointer and Stack Area

2.2.2 Program Counter (PC)

This 24-bit counter indicates the address of the next instruction the CPU will execute. Th
of all CPU instructions is 2 bytes (one word), so the least significant PC bit is ignored. (W
instruction is fetched, the least significant PC bit is regarded as 0). The PC is initialized w
start address is loaded by the vector address generated during reset exception-handling se

2.2.3 Condition-Code Register (CCR)

This 8-bit register contains internal CPU status information, including an interrupt mask b
half-carry (H), negative (N), zero (Z), overflow (V), and carry (C) flags. The I bit is initia
by reset exception-handling sequence, but other bits are not initialized.

Some instructions leave flag bits unchanged. Operations can be performed on the CCR bi
LDC, STC, ANDC, ORC, and XORC instructions. The N, Z, V, and C flags are used as b
conditions for conditional branch (Bcc) instructions.

For the action of each instruction on the flag bits, see appendix A.1, Instruction List.

				When the ADD.B, ADDX.B, SUB.B, SUBX.B, C NEG.B instruction is executed, this flag is set to is a carry or borrow at bit 3, and cleared to 0 oth When the ADD.W, SUB.W, CMP.W, or NEG.W instruction is executed, the H flag is set to 1 if th carry or borrow at bit 11, and cleared to 0 other When the ADD.L, SUB.L, CMP.L, or NEG.L ins executed, the H flag is set to 1 if there is a carry borrow at bit 27, and cleared to 0 otherwise.
4	U	Undefined	R/W	User Bit Can be written and read by software using the l ANDC, ORC, and XORC instructions.
3	N	Undefined	R/W	Negative Flag Stores the value of the most significant bit of da sign bit.
2	Z	Undefined	R/W	Zero Flag Set to 1 to indicate zero data, and cleared to 0 t non-zero data.
1	V	Undefined	R/W	Overflow Flag Set to 1 when an arithmetic overflow occurs, an to 0 at other times.
0	C	Undefined	R/W	Carry Flag Set to 1 when a carry occurs, and cleared to 0 o Used by: • Add instructions, to indicate a carry • Subtract instructions, to indicate a borrow • Shift and rotate instructions, to indicate a ca The carry flag is also used as a bit accumulator manipulation instructions.

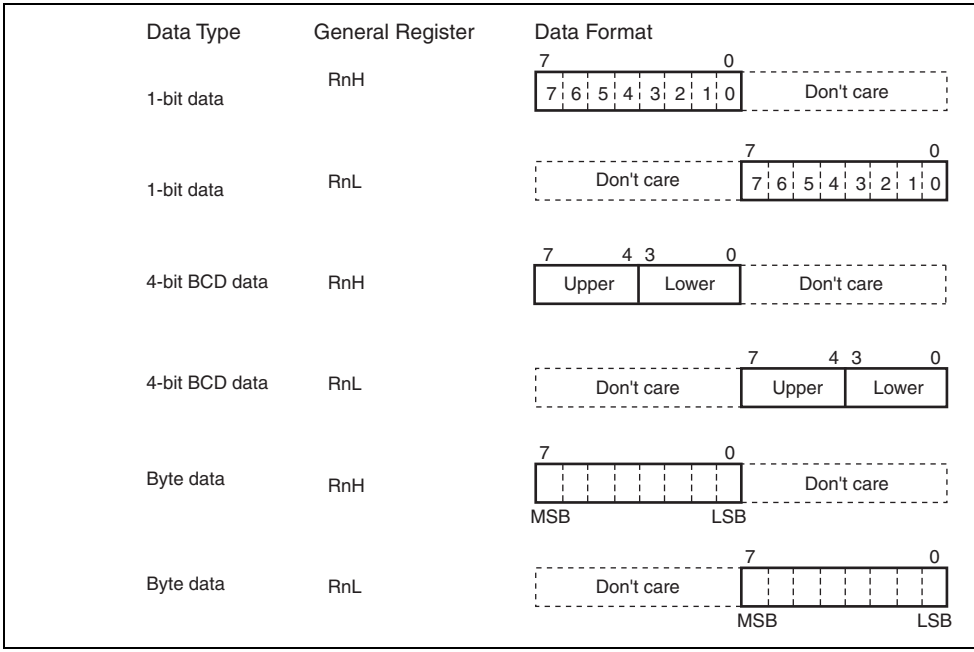
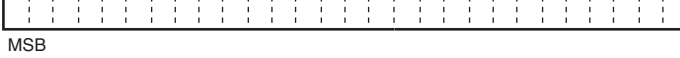


Figure 2.5 General Register Data Formats (1)



[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 2.5 General Register Data Formats (2)

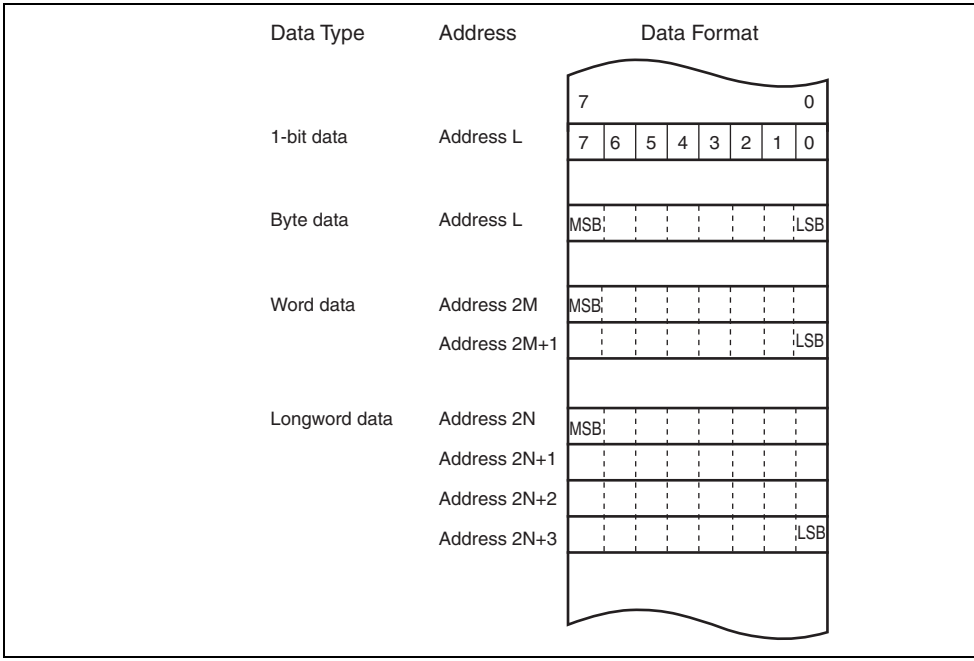


Figure 2.6 Memory Data Formats

Rs	General register (source)*
Rn	General register*
ERn	General register (32-bit register or address register)
(EAd)	Destination operand
(EAs)	Source operand
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 XOR
→	Move
¬	NOT (logical complement)

MOVFPE	B	(EAs) → Rd, Cannot be used in this LSI.
MOVTPE	B	Rs → (EAs) Cannot be used in this LSI.
POP	W/L	@SP+ → Rn Pops a general register from the stack. POP.W Rn is identical to @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, @

Note: * Refers to the operand size.
B: Byte
W: Word
L: Longword

DEC		Increments or decrements a general register by 1 or 2. (Byte o can be incremented or decremented by 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 from data in a 32-bi
DAA DAS	B	Rd decimal adjust → Rd Decimal-adjusts an addition or subtraction result in a general r referring to the CCR to produce 4-bit BCD data.
MULXU	B/W	Rd × Rs → Rd Performs unsigned multiplication on data in two general registe 8 bits × 8 bits → 16 bits or 16 bits × 16 bits → 32 bits.
MULXS	B/W	Rd × Rs → Rd Performs signed multiplication on data in two general registers bits × 8 bits → 16 bits or 16 bits × 16 bits → 32 bits.
DIVXU	B/W	Rd ÷ Rs → Rd Performs unsigned division on data in two general registers: ei bits ÷ 8 bits → 8-bit quotient and 8-bit remainder or 32 bits ÷ 1 16-bit quotient and 16-bit remainder.

Note: * Refers to the operand size.
B: Byte
W: Word
L: Longword

		general register.
EXTU	W/L	Rd (zero extension) → Rd Extends the lower 8 bits of a 16-bit register to word size, or the l bits of a 32-bit register to longword size, by padding with zeros o left.
EXTS	W/L	Rd (sign extension) → Rd Extends the lower 8 bits of a 16-bit register to word size, or the l bits of a 32-bit register to longword size, by extending the sign b

Note: * Refers to the operand size.
B: Byte
W: Word
L: Longword

		another general register or immediate data.
NOT	B/W/L	¬ (Rd) → (Rd) Takes the one's complement of general register contents.

Note: * Refers to the operand size.
B: Byte
W: Word
L: Longword

Table 2.5 Shift Instructions

Instruction	Size*	Function
SHAL SHAR	B/W/L	Rd (shift) → Rd Performs an arithmetic shift on general register contents.
SHLL SHLR	B/W/L	Rd (shift) → Rd Performs a logical shift on general register contents.
ROTL ROTR	B/W/L	Rd (rotate) → Rd Rotates general register contents.
ROTXL ROTXR	B/W/L	Rd (rotate) → Rd Rotates general register contents through the carry flag.

Note: * Refers to the operand size.
B: Byte
W: Word
L: Longword

		Inverts a specified bit in a general register or memory operand. number is specified by 3-bit immediate data or the lower three b general register.
BTST	B	¬ (<bit-No.> of <EAd>) → Z Tests a specified bit in a general register or memory operand ar clears the Z flag accordingly. The bit number is specified by 3-b immediate data or the 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 general register or n operand and stores the result in the carry flag.
BIAND	B	C ∧ ¬ (<bit-No.> of <EAd>) → C ANDs the carry flag with the inverse of a specified bit in a gener register or memory operand and stores the result in the carry fla The bit number is specified by 3-bit immediate data.
BOR	B	C ∨ (<bit-No.> of <EAd>) → C ORs the carry flag with a specified bit in a general register or me operand and stores the result in the carry flag.
BIOR	B	C ∨ ¬ (<bit-No.> of <EAd>) → C ORs the carry flag with the inverse of a specified bit in a genera or memory operand and stores the result in the carry flag. The bit number is specified by 3-bit immediate data.

Note: * Refers to the operand size.
B: Byte

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

Note: * Refers to the operand size.
B: Byte

		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 ⊕ V = 0
		BLT	Less than	N ⊕ V = 1
		BGT	Greater than	Z∨(N ⊕ V) = 0
		BLE	Less or equal	Z∨(N ⊕ V) = 1
JMP	—	Branches unconditionally to a specified address.		
BSR	—	Branches to a subroutine at a specified address.		
JSR	—	Branches to a subroutine at a specified address.		
RTS	—	Returns from a subroutine		

Note: * Bcc is the general name for conditional branch instructions.

		register size is one byte, but in transfer to memory, data is writ word access.
ANDC	B	CCR ∧ #IMM → CCR, EXR ∧ #IMM → EXR Logically ANDs the CCR with immediate data.
ORC	B	CCR ∨ #IMM → CCR, EXR ∨ #IMM → EXR Logically ORs the CCR with immediate data.
XORC	B	CCR ⊕ #IMM → CCR, EXR ⊕ #IMM → EXR Logically XORs the CCR with immediate data.
NOP	—	PC + 2 → PC Only increments the program counter.

Note: * Refers to the operand size.
B: Byte
W: Word

	else next; Transfers a data block. Starting from the address set in ER5, tra data for the number of bytes set in R4L or R4 to the address loc in ER6. Execution of the next instruction begins as soon as the transfer completed.

2.4.2 Basic Instruction Formats

H8/300H CPU instructions consist of 2-byte (1-word) units. An instruction consists of an operation field (op field), a register field (r field), an effective address extension (EA field condition field (cc).

Figure 2.7 shows examples of instruction formats.

(1) Operation Field

Indicates the function of the instruction, the addressing mode, and the operation to be carr on the operand. The operation field always includes the first four bits of the instruction. S instructions have two operation fields.

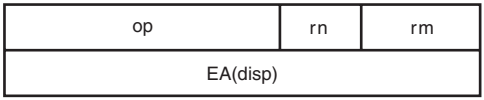
(2) Register Field

Specifies a general register. Address registers are specified by 3 bits, and data registers by 4 bits. Some instructions have two register fields. Some have no register field.

(2) Operation field and register fields

ADD.B Rn, Rm, etc.

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



MOV.B @(d:16, Rn), Rm

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

BRA d:8

Figure 2.7 Instruction Formats

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

Table 2.10 Addressing Modes

No.	Addressing Mode	Symbol
1	Register direct	Rn
2	Register indirect	@ERn
3	Register indirect with displacement	@(d:16,ERn)/@(d:24,ERn)
4	Register indirect with post-increment Register indirect with pre-decrement	@ERn+ @–ERn
5	Absolute address	@aa:8/@aa:16/@aa:24
6	Immediate	#xx:8/#xx:16/#xx:32
7	Program-counter relative	@(d:8,PC)/@(d:16,PC)
8	Memory indirect	@@aa:8

(1) Register Direct—Rn

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

- Register indirect with post-increment—@ERn+

  The register field of the instruction code specifies an address register (ERn) the lowe of which contains the address of a memory operand. After the operand is accessed, 1 added to the address register contents (32 bits) and the sum is stored in the address re The value added is 1 for byte access, 2 for word access, or 4 for longword access. Fo or longword access, the register value should be even.

- Register indirect with pre-decrement—@-ERn

  The value 1, 2, or 4 is subtracted from an address register (ERn) specified by the reg in the instruction code, and the lower 24 bits of the result is the address of a memory The result is also stored in the address register. The value subtracted is 1 for byte acc word access, or 4 for longword access. For the word or longword access, the register should be even.

(5) Absolute Address—@aa:8, @aa:16, @aa:24

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

For an 8-bit absolute address, the upper 16 bits are all assumed to be 1 (H'FFFF). For a 1 absolute address the upper 8 bits are a sign extension. A 24-bit absolute address can acc entire address space.

The access ranges of absolute addresses for the group of this LSI are those shown in tab because the upper 8 bits are ignored.

operand.

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

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

This mode is used in the BSR instruction. An 8-bit or 16-bit displacement contained in th
instruction is sign-extended and added to the 24-bit PC contents to generate a branch addr
PC value to which the displacement is added is the address of the first byte of the next ins
so the possible branching range is –126 to +128 bytes (–63 to +64 words) or –32766 to +3
bytes (–16383 to +16384 words) from the branch instruction. The resulting value should b
even number.

(8) Memory Indirect—@@aa:8

This mode can be used by the JMP and JSR instructions. The instruction code contains an
absolute address specifying a memory operand. This memory operand contains a branch a
The memory operand is accessed by longword access. The first byte of the memory opera
ignored, generating a 24-bit branch address. Figure 2.8 shows how to specify branch addr
memory indirect mode. The upper bits of the absolute address are all assumed to be 0, so
address range is 0 to 255 (H'0000 to H'00FF).

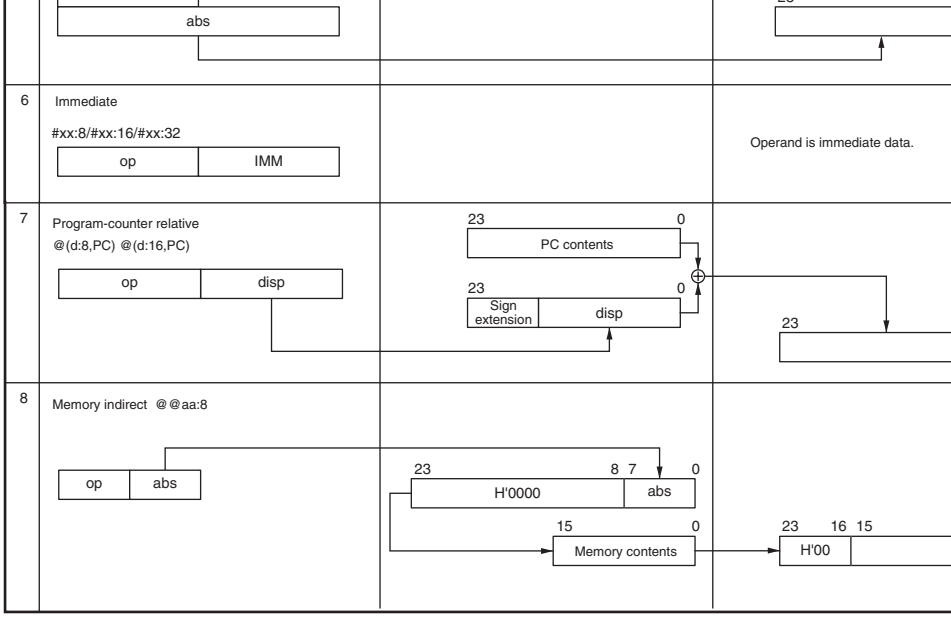
Note that the first part of the address range is also the exception vector area.

2.5.2 Effective Address Calculation

Table 2.12 indicates how effective addresses are calculated in each addressing mode. In the upper 8 bits of the effective address are ignored in order to generate a 16-bit effectiv

Table 2.12 Effective Address Calculation (1)

No	Addressing Mode and Instruction Format	Effective Address Calculation	Effective Address (EA
1	Register direct(Rn) op | rm | rn		Operand is general register con
2	Register indirect(@ERn) op | r	31 0 General register contents	23
3	Register indirect with displacement @(d:16,ERn) or @(d:24,ERn) op | r | disp	31 0 General register contents 31 0 Sign extension | disp	23
4	Register indirect with post-increment or pre-decrement •Register indirect with post-increment @ERn+ op | r •Register indirect with pre-decrement @-ERn op | r	31 0 General register contents 1, 2, or 4 31 0 General register contents 1, 2, or 4 The value to be added or subtracted is 1 when the operand is byte size, 2 for word size, and 4 for longword size.	23 23



[Legend]
r, rm,rn : Register field
op : Operation field
disp : Displacement
IMM : Immediate data
abs : Absolute address

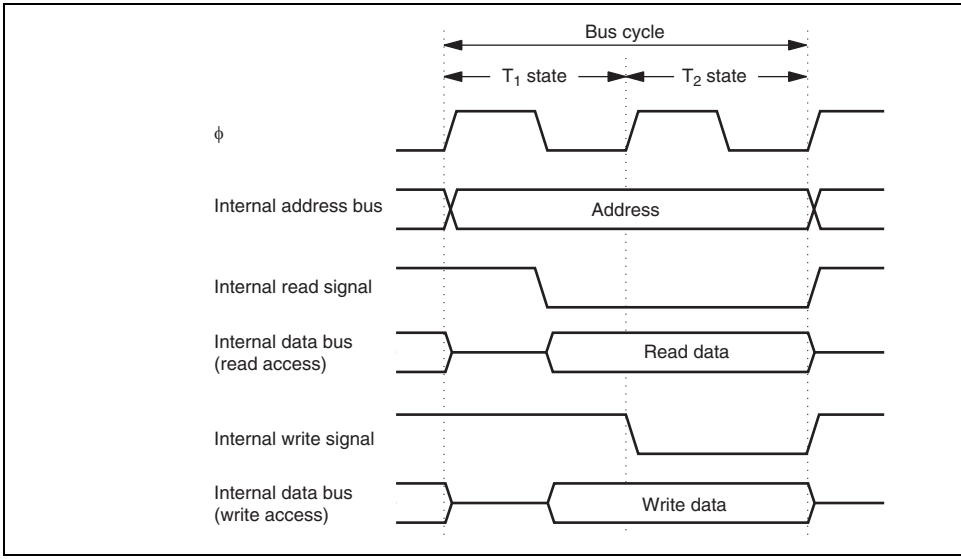


Figure 2.9 On-Chip Memory Access Cycle

module.

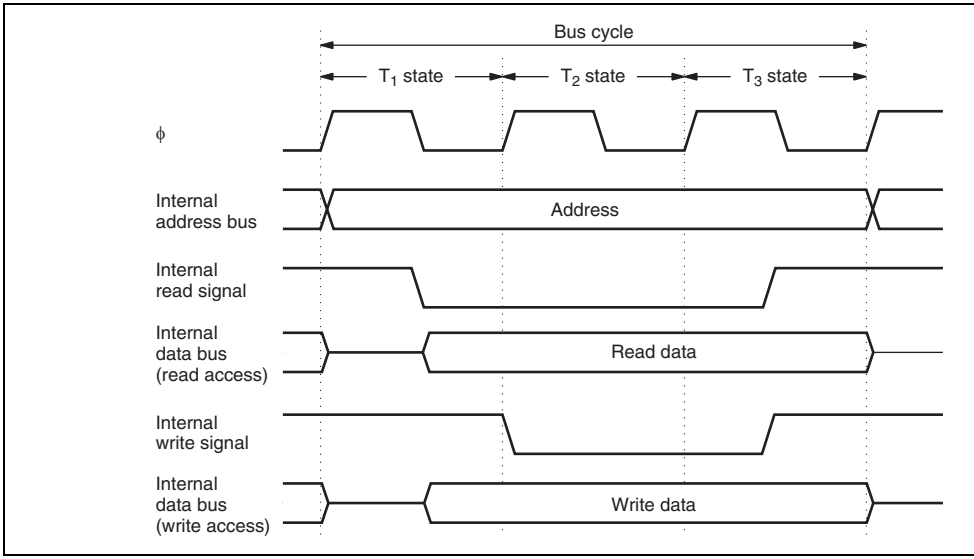


Figure 2.10 On-Chip Peripheral Module Access Cycle (3-State Access)

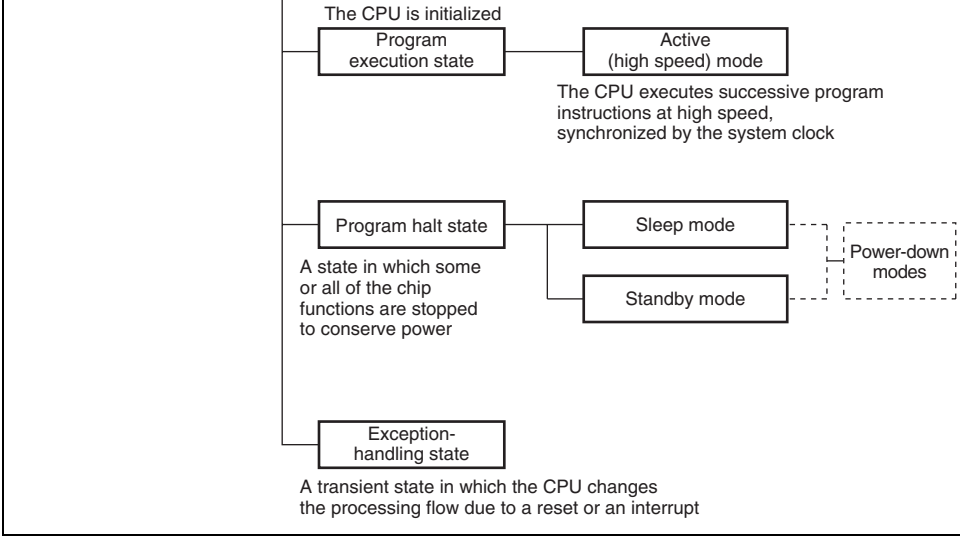


Figure 2.11 CPU Operation States

Figure 2.12 State Transitions

2.8 Usage Notes

2.8.1 Notes on Data Access to Empty Areas

The address space of this LSI includes empty areas in addition to the ROM, RAM, and on
I/O registers areas available to the user. When data is transferred from CPU to empty area
transferred data will be lost. This action may also cause the CPU to malfunction. When d
transferred from an empty area to CPU, the contents of the data cannot be guaranteed.

2.8.2 EEPMOV Instruction

EEPMOV is a block-transfer instruction and transfers the byte size of data indicated by R
which starts from the address indicated by R5, to the address indicated by R6. Set R4L an
that the end address of the destination address (value of R6 + R4L) does not exceed H'FF
value of R6 must not change from H'FFFF to H'0000 during execution).

2.8.3 Bit Manipulation Instruction

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

2. The CPU sets or resets the bit to be manipulated with the bit manipulation instructio
3. The written data is written again in byte units to the timer load register.

The timer is counting, so the value read is not necessarily the same as the value in the ti
register. As a result, bits other than the intended bit in the timer counter may be modifie
modified value may be written to the timer load register.

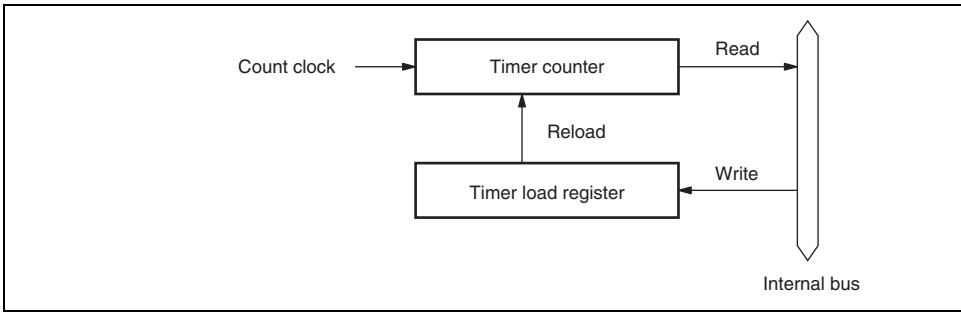


Figure 2.13 Example of Timer Configuration with Two Registers Allocated Same Address

	level	level	level	level	level	level	level
PCR5	0	0	1	1	1	1	1
PDR5	1	0	0	0	0	0	0

[BSET instruction executed]

```
BSET    #0,     @PDR5
```

The BSET instruction is executed for port 5.

[After executing BSET]

	P57	**P56**	**P55**	**P54**	**P53**	**P52**	**P51**
Input/output	Input	Input	Output	Output	Output	Output	Output
Pin state	Low level	High level	Low level	Low level	Low level	Low level	Low level
PCR5	0	0	1	1	1	1	1
PDR5	0	1	0	0	0	0	0

[Description on operation]

1. When the BSET instruction is executed, first the CPU reads port 5. Since P57 and P5 input pins, the CPU reads the pin states (low-level and high-level input).
   P55 to P50 are output pins, so the CPU reads the value in PDR5. In this example PDR value of H'80, but the value read by the CPU is H'40.
2. Next, the CPU sets bit 0 of the read data to 1, changing the PDR5 data to H'41.
3. Finally, the CPU writes H'41 to PDR5, completing execution of BSET.

	P57	P56	P55	P54	P53	P52	P51
Input/output	Input	Input	Output	Output	Output	Output	Output
Pin state	Low level	High level	Low level	Low level	Low level	Low level	Low level
PCR5	0	0	1	1	1	1	1
PDR5	1	0	0	0	0	0	0
RAM0	1	0	0	0	0	0	0

[BSET instruction executed]

```
BSET    #0,     @RAM0
```

The BSET instruction is executed designating the
work area (RAM0).

[After executing BSET]

```
MOV.B   @RAM0, R0L
MOV.B   R0L,   @PDR5
```

The work area (RAM0) value is written to PDR5.

	P57	P56	P55	P54	P53	P52	P51
Input/output	Input	Input	Output	Output	Output	Output	Output
Pin state	Low level	High level	Low level	Low level	Low level	Low level	Low level
PCR5	0	0	1	1	1	1	1
PDR5	1	0	0	0	0	0	0
RAM0	1	0	0	0	0	0	0

Input/output	Input	Input	Output	Output	Output	Output	Output
Pin state	Low level	High level	Low level	Low level	Low level	Low level	Low level
PCR5	0	0	1	1	1	1	1
PDR5	1	0	0	0	0	0	0

[BCLR instruction executed]

```
BCLR    #0,     @PCR5
```

The BCLR instruction is executed for PCR5.

[After executing BCLR]

	P57	**P56**	**P55**	**P54**	**P53**	**P52**	**P51**
Input/output	Output	Output	Output	Output	Output	Output	Output
Pin state	Low level	High level	Low level	Low level	Low level	Low level	Low level
PCR5	1	1	1	1	1	1	1
PDR5	1	0	0	0	0	0	0

[Description on operation]

1. When the BCLR instruction is executed, first the CPU reads PCR5. Since PCR5 is a v register, the CPU reads a value of H'FF, even though the PCR5 value is actually H'3F
2. Next, the CPU clears bit 0 in the read data to 0, changing the data to H'FE.
3. Finally, H'FE is written to PCR5 and BCLR instruction execution ends.

	P57	P56	P55	P54	P53	P52	P51
Input/output	Input	Input	Output	Output	Output	Output	Output
Pin state	Low level	High level	Low level	Low level	Low level	Low level	Low level
PCR5	0	0	1	1	1	1	1
PDR5	1	0	0	0	0	0	0
RAM0	0	0	1	1	1	1	1

[BCLR instruction executed]

```
BCLR    #0,     @RAM0
```

The BCLR instructions executed for the PCR5 w
(RAM0).

[After executing BCLR]

```
MOV.B   @RAM0, R0L
MOV.B   R0L,   @PCR5
```

The work area (RAM0) value is written to PCR5.

	P57	P56	P55	P54	P53	P52	P51
Input/output	Input	Input	Output	Output	Output	Output	Output
Pin state	Low level	High level	Low level	Low level	Low level	Low level	Low level
PCR5	0	0	1	1	1	1	1
PDR5	1	0	0	0	0	0	0
RAM0	0	0	1	1	1	1	1

Exception handling starts when a trap instruction (TRAPA) is executed. The TRAPA instruction generates a vector address corresponding to a vector number from 0 to 3, specified in the instruction code. Exception handling can be executed at all times in t program execution state.

- Interrupts

  External interrupts other than NMI and internal interrupts other than address break a by the I bit in CCR, and kept masked while the I bit is set to 1. Exception handling s the current instruction or exception handling ends, if an interrupt request has been iss

3.1 Exception Sources and Vector Address

Table 3.1 shows the vector addresses and priority of each exception handling. When mo one interrupt is requested, handling is performed from the interrupt with the highest prio

Table 3.1 Exception Sources and Vector Address

Relative Module	Exception Sources	Vector Number	Vector Address
$\overline{\text{RES}}$ pin Watchdog timer	Reset	0	H'0000 to H'0001
—	Reserved for system use	1 to 6	H'0002 to H'000D
External interrupt pin	NMI	7	H'000E to H'000F
CPU	Trap instruction #0	8	H'0010 to H'0011
	Trap instruction #1	9	H'0012 to H'0013
	Trap instruction #2	10	H'0014 to H'0015
	Trap instruction #3	11	H'0016 to H'0017

—	Reserved for system use	19, 20	H'0026 to H'0029
Timer W	Timer W input capture A/ compare match A Timer W input capture B/ compare match B Timer W input capture C/ compare match C Timer W input capture D/ compare match D Timer W overflow	21	H'002A to H'002B
Timer V	Timer V compare match A Timer V compare match B Timer V overflow	22	H'002C to H'002D
SCI3	SCI3 receive data full SCI3 transmit data empty SCI3 transmit end SCI3 receive error	23	H'002E to H'002F
IIC2*	IIC_2 transmit data empty IIC_2 transmit end IIC_2 receive error	24	H'0030 to H'0031
A/D converter	A/D conversion end	25	H'0032 to H'0033
—	Reserved for system use	26 to 28	H'0034 to H'0039
Timer B1*	Timer B1 overflow	29	H'003A to H'003B
—	Reserved for system use	30 to 33	H'003C to H'0043
Clock switch	Clock switch (external clock to on-chip oscillator clock)	34	H'0044 to H'0045

Note: * Available for the H8/36912 Group only.

- Wakeup interrupt flag register (IWPR)

3.2.1 Interrupt Edge Select Register 1 (IEGR1)

IEGR1 selects the direction of an edge that generates interrupt requests of the $\overline{\text{IRQ3}}$ and pins.

Bit	Bit Name	Initial Value	R/W	Description
7	—	0	–	Reserved This bit is always read as 0.
6 to 4	—	All 1	—	Reserved These bits are always read as 1.
3	IEG3	0	R/W	IRQ3 Edge Select 0: Falling edge of $\overline{\text{IRQ3}}$ pin input is detected 1: Rising edge of $\overline{\text{IRQ3}}$ pin input is detected
2, 1	—	All 0	—	Reserved These bits are always read as 0.
0	IEG0	0	R/W	IRQ0 Edge Select 0: Falling edge of $\overline{\text{IRQ0}}$ pin input is detected 1: Rising edge of $\overline{\text{IRQ0}}$ pin input is detected

				0: Falling edge of WKP5 (ADTRG) pin input is de
				1: Rising edge of WKP5 (ADTRG) pin input is de
4 to 0	—	All 0	—	Reserved
				These bits are always read as 0.

3.2.3 Interrupt Enable Register 1 (IENR1)

IENR1 enables direct transition interrupts and external pin interrupts.

Bit	Bit Name	Initial Value	R/W	Description
7	IENDT	0	R/W	Direct Transfer Interrupt Enable
				When this bit is set to 1, direct transition interrup requests are enabled.
6	—	0	—	Reserved
				This bit is always read as 0.
5	IENWP	0	R/W	Wakeup Interrupt Enable
				This bit is an enable bit of the WKP5 pin. When t set to 1, interrupt requests are enabled.
4	—	1	—	Reserved
				This bit is always read as 1.
3	IEN3	0	R/W	IRQ3 Interrupt Enable
				When this bit is set to 1, interrupt requests of the pin are enabled.
2, 1	—	All 0	—	Reserved
				These bits are always read as 0.

Bit	Bit Name	Initial Value	R/W	Description
7	—	0	—	Reserved This bit is always read as 0.
6	—	0	R/W	Reserved Although this bit is readable/writable, it should to 1.
5	IENTB1	0	R/W	Timer B1 Interrupt Enable When this bit is set to 1, overflow interrupt req timer B1 are enabled.
4 to 0	—	All 1	—	Reserved These bits are always read as 1.

When disabling interrupts by clearing bits in an interrupt enable register, or when clearin an interrupt flag register, always do so while interrupts are masked (I = 1). If the above o operations are performed while I = 0, and as a result a conflict arises between the clear i and an interrupt request, exception handling for the interrupt will be executed after the c instruction has been executed.

				[Clearing condition] • When IRRDT is cleared by writing 0
6	—	0	—	Reserved This bit is always read as 0.
5, 4	—	All 1	—	Reserved These bits are always read as 1.
3	IRRI3	0	R/W	IRQ3 Interrupt Request Flag [Setting condition] • When $\overline{\text{IRQ3}}$ pin is designated for interrupt inp the designated signal edge is detected [Clearing condition] • When IRRI3 is cleared by writing 0
2, 1	—	All 0	—	Reserved These bits are always read as 0.
0	IRRI0	0	R/W	IRQ0 Interrupt Request Flag [Setting condition] • When $\overline{\text{IRQ0}}$ pin is designated for interrupt inp the designated signal edge is detected [Clearing condition] • When IRRI0 is cleared by writing 0

				[Setting condition] • When timer B1 overflows [Clearing condition] • When IRRTB1 is cleared by writing 0
4 to 0	—	All 1	—	Reserved These bits are always read as 1.

3.2.7 Wakeup Interrupt Flag Register (IWPR)

IWPR is a status flag register for $\overline{\text{WKP5}}$ interrupt requests.

Bit	Bit Name	Initial Value	R/W	Description
7, 6	—	All 1	—	Reserved These bits are always read as 1.
5	IWPF5	0	R/W	WKP5 Interrupt Request Flag [Setting condition] • When $\overline{\text{WKP5}}$ pin is designated for interrupt the designated signal edge is detected. [Clearing condition] • When IWPF5 is cleared by writing 0
4 to 0	—	All 0	—	Reserved These bits are always read as 0.

1. Set the I bit in the condition code register (CCR) to 1.
2. The CPU generates a reset exception handling vector address (from H'0000 to H'0001
   data in that address is sent to the program counter (PC) as the start address, and progra
   execution starts from that address.

(2) IRQ3 and IRQ0 Interrupts

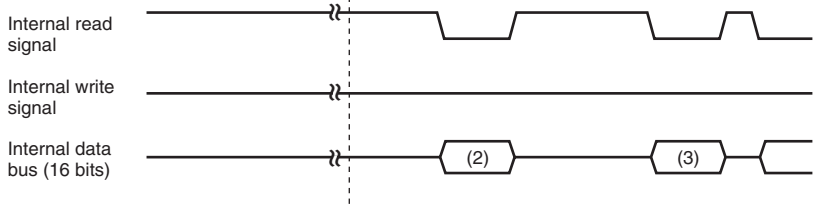
IRQ3 and IRQ0 interrupts are requested by input signals to the $\overline{\text{IRQ3}}$ and $\overline{\text{IRQ0}}$ pins. Th
interrupts are given different vector addresses, and are detected individually by either ris
sensing or falling edge sensing, depending on the settings of the IEG3 and IEG0 bits in

When the $\overline{\text{IRQ3}}$ and $\overline{\text{IRQ0}}$ pins are designated for interrupt input in PMR1 and the desig
signal edge is input, the corresponding bit in IRR1 is set to 1, requesting the CPU of an i
These interrupts can be masked by setting the IEN3 and IEN0 bits in IENR1.

(3) WKP Interrupt

WKP interrupt is requested by an input signal to the $\overline{\text{WKP5}}$ pin. This interrupt is detecte
rising edge sensing or falling edge sensing, depending on the setting of the WPEG5 bit i

When the $\overline{\text{WKP5}}$ pin is designated for interrupt input in PMR5 and the designated signa
input, the corresponding bit in IWPR is set to 1, requesting the CPU of an interrupt. This
can be masked by setting the IENWP bit in IENR1.



(1) Reset exception handling vector address (H'0000)
(2) Program start address
(3) Initial program instruction

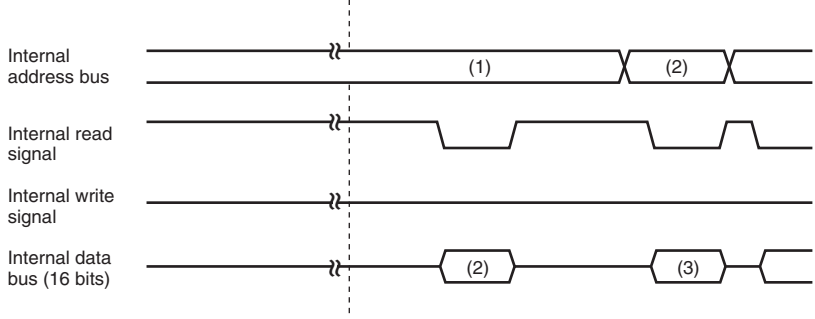
Figure 3.1 Reset Sequence

3.4.2 Internal Interrupts

Each on-chip peripheral module has a flag to show the interrupt request status and the ena
enable or disable the interrupt. For direct transfer interrupt requests generated by executio
SLEEP instruction, this function is included in IRR1 and IENR1.

When an on-chip peripheral module requests an interrupt, the corresponding interrupt req
status flag is set to 1, requesting the CPU of an interrupt. When this interrupt is accepted,
is set to 1 in CCR. These interrupts can be masked by writing 0 to clear the corresponding
bit.

3. The CPU accepts the NMI or address break without depending on the I bit value. Ot interrupt requests are accepted, if the I bit is cleared to 0 in CCR; if the I bit is set to interrupt request is held pending.
4. If the CPU accepts the interrupt after processing of the current instruction is complet interrupt exception handling will begin. First, both PC and CCR are pushed onto the state of the stack at this time is shown in figure 3.2. The PC value pushed onto the st address of the first instruction to be executed upon return from interrupt handling.
5. Then, the I bit of CCR is set to 1, masking further interrupts excluding the NMI and break. Upon return from interrupt handling, the values of I bit and other bits in CCR restored and returned to the values prior to the start of interrupt exception handling.
6. Next, the CPU generates the vector address corresponding to the accepted interrupt, transfers the address to PC as a start address of the interrupt handling-routine. Then starts executing from the address indicated in PC.



(1) Reset exception handling vector address (H'0000)
(2) Program start address
(3) Initial program instruction

Figure 3.2 Stack Status after Exception Handling

Vector fetch	2
Instruction fetch	4
Internal processing	4

Note: * EEPMOV instruction is not included.

Figure 3.3 Interrupt Sequence

3.5.2 Notes on Stack Area Use

When word data is accessed, the least significant bit of the address is regarded as 0. Acc
stack always takes place in word size, so the stack pointer (SP: R7) should never indicat
address. Use PUSH Rn (MOV.W Rn, @–SP) or POP Rn (MOV.W @SP+, Rn) to save
register values.

3.5.3 Notes on Rewriting Port Mode Registers

When a port mode register is rewritten to switch the functions of external interrupt pins,
$\overline{\text{IRQ0}}$, and $\overline{\text{WKP5}}$, the interrupt request flag may be set to 1.

When switching a pin function, mask the interrupt before setting the bit in the port mode
After accessing the port mode register, execute at least one instruction (e.g., NOP), then
interrupt request flag from 1 to 0.

Figure 3.4 shows a port mode register setting and interrupt request flag clearing procedu

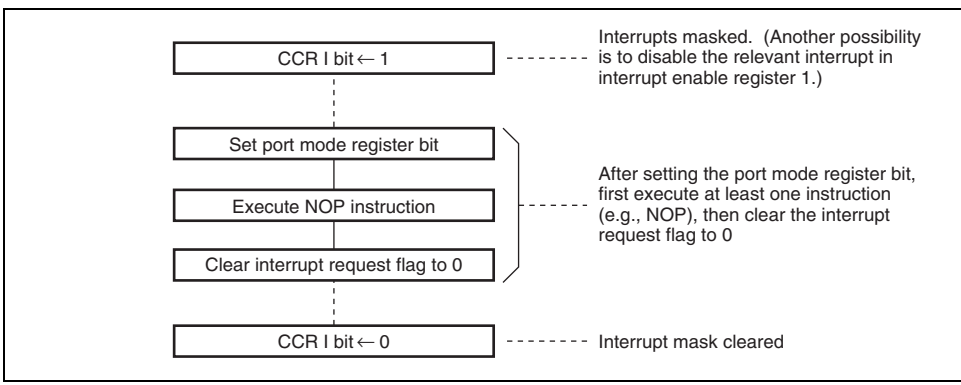
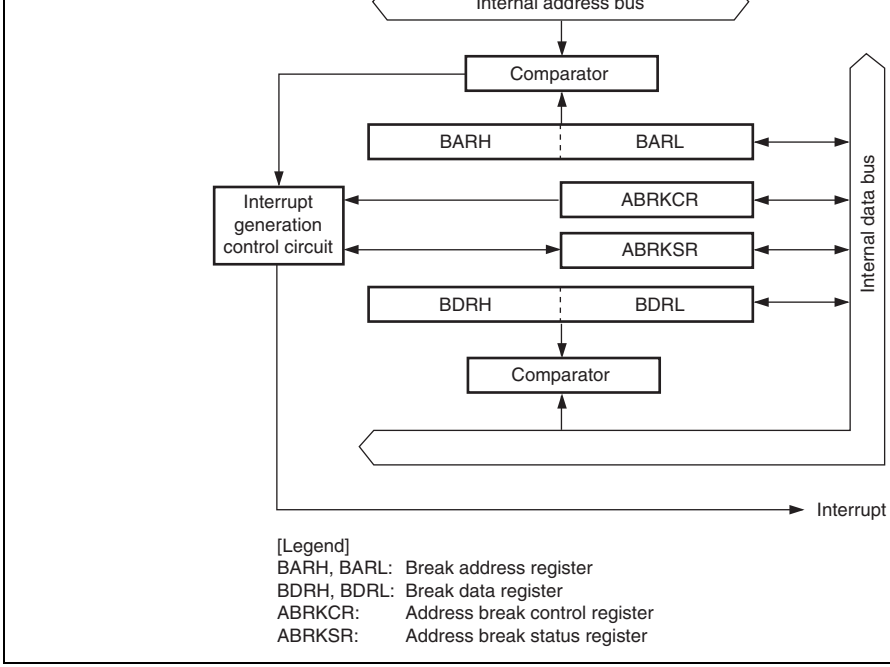


Figure 3.4 Port Mode Register Setting and Interrupt Request Flag Clearing Pr



[Legend]
BARH, BARL: Break address register
BDRH, BDRL: Break data register
ABRKCR: Address break control register
ABRKSR: Address break status register

Figure 4.1 Block Diagram of Address Break

ABRKCR sets address break conditions.

Bit	Bit Name	Initial Value	R/W	Description
7	RTINTE	1	R/W	RTE Interrupt Enable When this bit is 0, the interrupt immediately after executing RTE is masked and then one instructi be executed. When this bit is 1, the interrupt is n masked.
6	CSEL1	0	R/W	Condition Select 1 and 0 These bits set address break conditions. 00: Instruction execution cycle 01: CPU data read cycle 10: CPU data write cycle 11: CPU data read/write cycle
5	CSEL0	0	R/W	
4	ACMP2	0	R/W	Address Compare Condition Select 2 to 0 These bits comparison condition between the ad in BAR and the internal address bus. 000: Compares 16-bit addresses 001: Compares upper 12-bit addresses 010: Compares upper 8-bit addresses 011: Compares upper 4-bit addresses 1XX: Reserved (setting prohibited)
3	ACMP1	0	R/W	
2	ACMP0	0	R/W	

[Legend]
X: Don't care

When an address break is set in the data read cycle or data write cycle, the data bus used depend on the combination of the byte/word access and address. Table 4.1 shows the ac data bus used. When an I/O register space with an 8-bit data bus width is accessed in wo byte access is generated twice. For details on data widths of each register, see section 19 Register Addresses (Address Order).

Table 4.1 Access and Data Bus Used

	Word Access		Byte Acces	
	Even Address	Odd Address	Even Address	Odd
ROM space	Upper 8 bits	Lower 8 bits	Upper 8 bits	Upp
RAM space	Upper 8 bits	Lower 8 bits	Upper 8 bits	Upp
I/O register with 8-bit data bus width	Upper 8 bits	Upper 8 bits	Upper 8 bits	Upp
I/O register with 16-bit data bus width	Upper 8 bits	Lower 8 bits	—	—

				• When 0 is written after ABIF=1 is read
6	ABIE	0	R/W	Address Break Interrupt Enable When this bit is 1, an address break interrupt rec enabled.
5 to 0	—	All 1	—	Reserved These bits are always read as 1.

4.1.3 Break Address Registers (BARH, BARL)

BARH and BARL are 16-bit read/write registers that set the address for generating an ad break interrupt. When setting the address break condition to the instruction execution cyc first byte address of the instruction. The initial value of this register is H'FFFF.

4.1.4 Break Data Registers (BDRH, BDRL)

BDRH and BDRL are 16-bit read/write registers that set the data for generating an addres interrupt. BDRH is compared with the upper 8-bit data bus. BDRL is compared with the bit data bus. When memory or registers are accessed by byte, the upper 8-bit data bus is u even and odd addresses in the data transmission. Therefore, comparison data must be set for byte access. For word access, the data bus used depends on the address. See section 4. Address Break Control Register (ABRKCR), for details. The initial value of this register undefined.

When the address break is specified in instruction execution cycle

Register setting
- ABRKCR = H'80
- BAR = H'025A

Program

```
  0258  NOP
* 025A  NOP
  025C  MOV.W @H'025A,R0
  0260  NOP
  0262  NOP
   :     :
```

Underline indicates the address to be stacked.

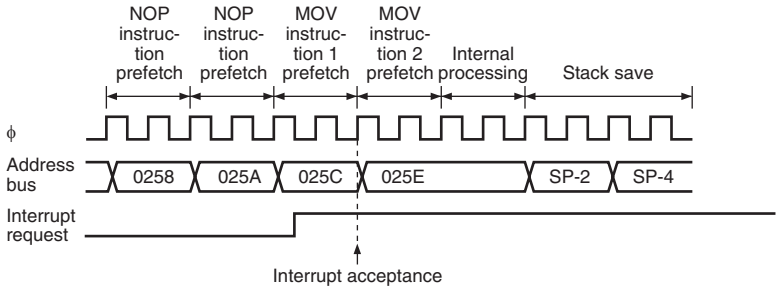


Figure 4.2 Address Break Interrupt Operation Example (1)

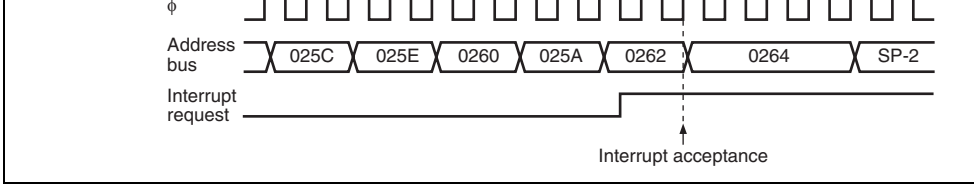


Figure 4.2 Address Break Interrupt Operation Example (2)

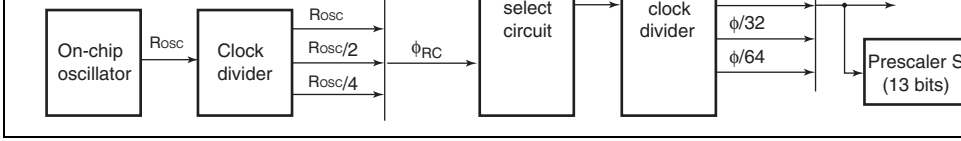


Figure 5.1 Block Diagram of Clock Pulse Generators

The system clock (ϕ) is a basic clock on which the CPU and on-chip peripheral modules The system clock is divided into ϕ/2 to ϕ/8192 by prescaler S and the divided clocks are to respective peripheral modules.

Users can adjust the on-chip oscillation frequency by rewriting the trimming registers

- Interrupt can be requested to the CPU when the system clock is changed from the exte clock to the on-chip oscillator clock.

RCCR controls the on-chip oscillator.

Bit	Bit Name	Initial Value	R/W	Description
7	RCSTP	0	R/W	On-chip Oscillator Standby The internal on-chip oscillator standby state is e setting this bit to 1.
6	FSEL	0	R/W	Frequency Select for On-chip Oscillator 0: 8MHz 1: 10MHz
5	VCLSEL	0	R/W	Power Supply Select for On-chip Oscillator 0: Selects VBGR 1: Selects VCL When VCL is selected, the accuracy of the on-c oscillator frequency cannot be guaranteed.
4 to 2	—	All 0	—	Reserved These bits are always read as 0.
1	RCPSC1	0	R/W	Division Ratio Select for On-chip Oscillator
0	RCPSC0	0	R/W	The division ratio of R_{osc} changes right after rev bit. These bits can be written to only when the CKS CKCSR is 0. 0X: R_{osc} (not divided) 10: $R_{osc}/2$ 11: $R_{osc}/4$

				Bits 5 and 4 can be written to when this bit is set [Setting condition] • When writing 0 to the WRI bit and writing 1 to PRWE bit [Clearing conditions] • Reset • When writing 0 to the WRI bit and writing 0 to PRWE bit
5	LOCKDW	0	R/W	Trimming Data Register Lock Down The RC trimming data register (RCTRMDR) can written to when this bit is set to 1. Once this bit is this register cannot be written to until a reset is in if 0 is written to this bit. [Setting condition] • When writing 0 to the WRI bit and writing 1 to LOCKDW bit while the PRWE bit is 1 [Clearing condition] • Reset

				• When writing 0 to the WRI bit and writing 0 TRMDRWE bit while the PRWE bit is 1
3 to 0	—	All 1	—	Reserved These bits are always read as 1

5.2.3 RC Trimming Data Register (RCTRMDR)

RCTRMDR stores the trimming data of the on-chip oscillator frequency.

Bit	Bit Name	Initial Value	R/W	Description
7	TRMD7	(0)*	R/W	Trimming Data
6	TRMD6	(0)*	R/W	In the flash memory version, the trimming data from the flash memory to this register right after These bits are always read as undefined value.
5	TRMD5	(0)*	R/W	
4	TRMD4	(0)*	R/W	
3	TRMD3	(0)*	R/W	As for the masked ROM version, the on-chip os frequency can be trimmed by rewriting these bi
2	TRMD2	(0)*	R/W	frequency generated in the on-chip oscillator ch
1	TRMD1	0	R	right after rewriting these bits. These bits are in H'00.
0	TRMD0	0	R	Frequency variation is expressed as follows (th bit is a sign bit):
				(Min.) H'80 ← H'FC ← H'00 →H'04 → H'7C (Ma

Note: * The initial value differs from product to product in the flash memory version.

				0	1	I/O	OSC1 (external cloc
				1	1	OSC2	OSC1
5	—	0	R/W	Reserved Although this bit is readable/writable, it should nc to 1.			
4	OSCSEL	0	R/W	LSI Operation Clock Select This bit forcibly selects the system clock of this L 0: Selects the on-chip oscillator clock as the syst 1: Selects the external clock as the system clock			
3	CKSWIE	0	R/W	Clock Switch Interrupt Enable Setting this bit to 1 enables the clock switch inter request.			
2	CKSWIF	0	R/W	Clock Switch Interrupt Request Flag [Setting condition] • When the external clock is switched to the on oscillator clock [Clearing condition] • When writing 0 after reading 1			
1	—	1	R	Reserved This bit is always read as 1.			
0	CKSTA	0	R	LSI Operating Clock Status 0: This LSI operates on on-chip oscillator clock. 1: This LSI operates on external clock.			

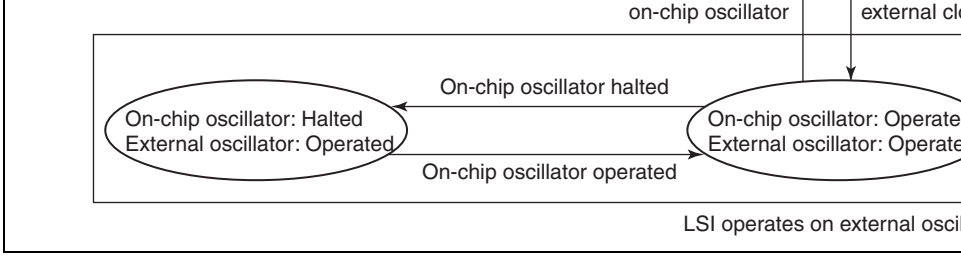
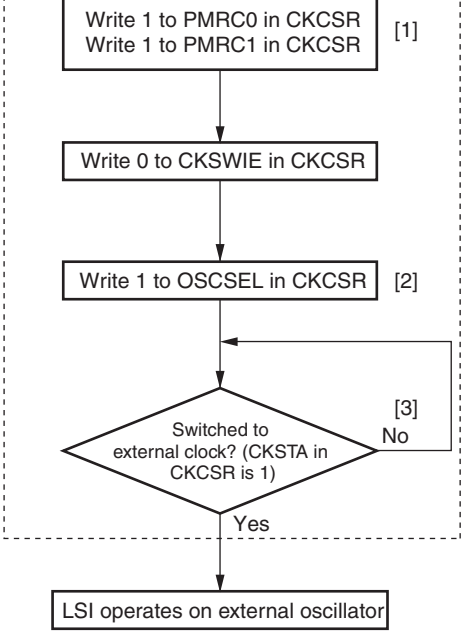


Figure 5.2 State Transition of System Clock



settles. The correspondence between Nwa
which is the number of wait cycles for oscil
settling, and Nstby, which is the number of
cycles for oscillation settling when returnin
standby mode, is as follows:
Nstby ≤ Nwait ≤ 2 × Nstby
Nstby is set by bits STS 2 to 0 in SYSCR1.
For details, see section 6.1.1, System Con
Register 1 (SYSCR1).

[3] While the system is waiting for the external
oscillation settling, this LSI is not halted bu
continues to operate on the on-chip oscilla
Read the value of the CKSTA bit in CKCSR
ensure that the system clocks are switched

Figure 5.3 Flowchart of Clock Switching On-chip Oscillator Clock to External Clock (1)

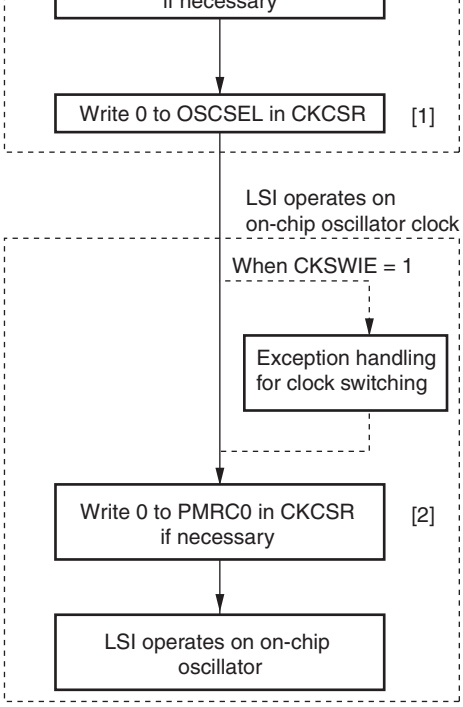
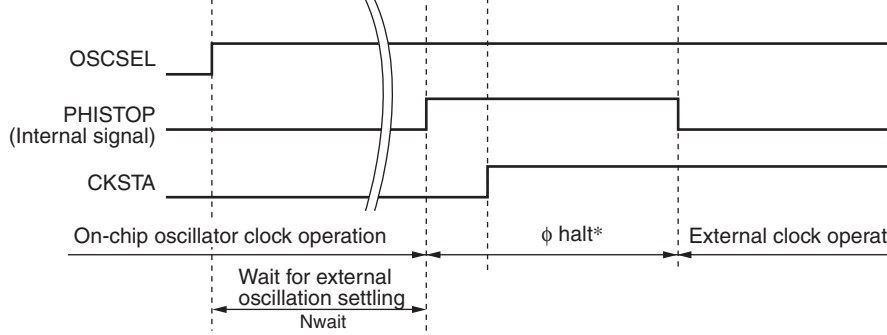


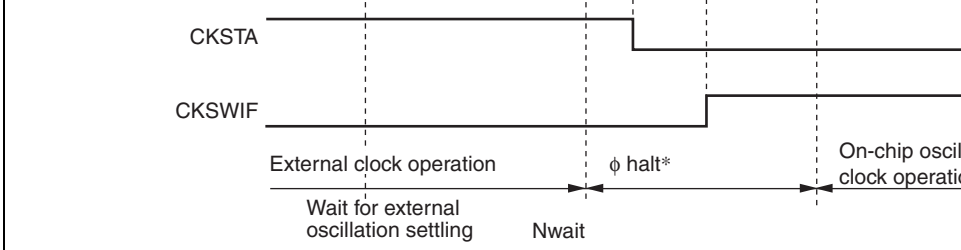
Figure 5.4 Flowchart of Clock Switching External Clock to On-chip Oscillator Clock (2)



[Legend]
ϕOSC: External clock
ϕRC: On-chip oscillator clock
ϕ: System clock
OSCSEL: Bit 4 in CKCSR
PHISTOP: System clock stop control signal
CKSTA: Bit 0 in CKCSR

Note: * The ϕ halt duration is the duration from the timing when the ϕ clock stops to the fi
rising edge of the ϕ_{OSC} clock after six clock cycles of the ϕ_{RC} clock have elapsed.

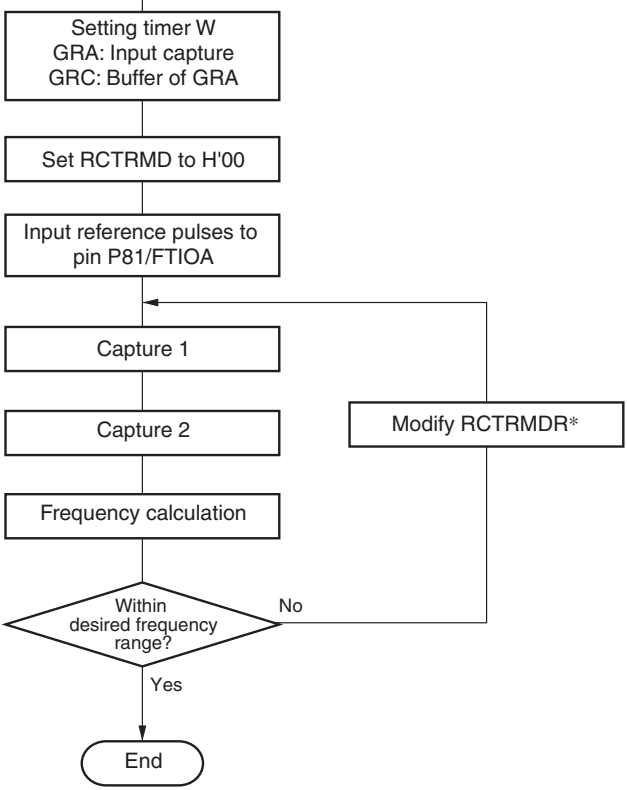
Figure 5.5 Timing Chart of Switching On-chip Oscillator Clock to External C



[Legend]
φOSC: External clock
φRC: On-chip oscillator clock
φ: System clock
OSCSEL: Bit 4 in CKCSR
PHISTOP: System clock stop control signal
CKSTA: Bit 0 in CKCSR
CKSWIF: Bit 2 in CKCSR

Note: * The φ halt duration is the duration from the timing when the φ clock stops to the seventh rising edge of the ϕ_{RC} clock.

Figure 5.6 Timing Chart to Switch External Clock to On-chip Oscillator Cl



Note: * Comparing the difference between the measured frequency and the desired frequer
individual bits of RCTRMDR are decided from the MSB bit by bit.

Figure 5.7 Example of Trimming Flow for On-chip Oscillator Frequency

Figure 5.8 Timing Chart of Trimming of On-chip Oscillator Frequency

The on-chip oscillator frequency is gained by the expression below. Since the input-capt
is sampled by the ϕ_{RC} clock, the calculated result may include a sampling error of ±1 cyc
ϕ_{RC} clock.

$$\phi RC = \frac{(M + \alpha) - M}{t_A} \quad (MHz)$$

ϕRC: Frequency of on-chip oscillator (MHz)
t_A: Period of reference clock (µs)
M: Timer W counter value

crystal resonator should be used. Figure 5.12 shows the equivalent circuit of a crystal reso
resonator having the characteristics given in table 5.1 should be used.

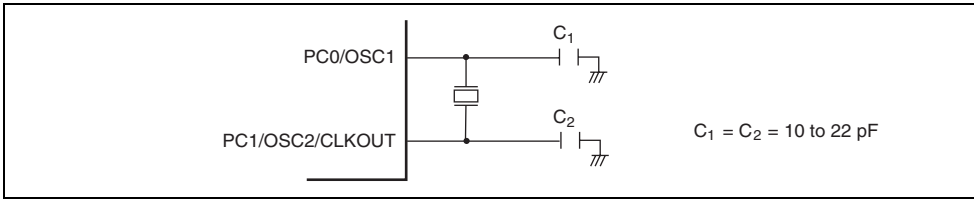


Figure 5.9 Example of Connection to Crystal Resonator

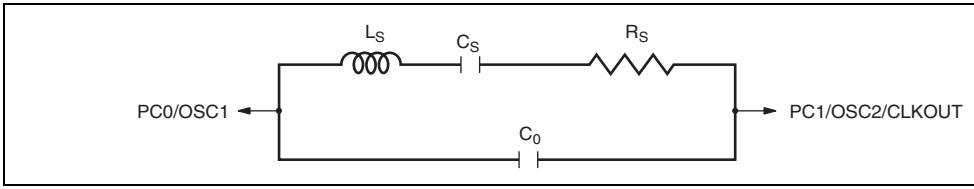


Figure 5.10 Equivalent Circuit of Crystal Resonator

Table 5.1 Crystal Resonator Parameters

Frequency (MHz)	2	4	8	10	12
R_s (Max.)	500 Ω	120 Ω	80 Ω	60 Ω	50 Ω
C_0 (Max.)			70 pF		

5.5.3 External Clock Input Method

To use the external clock, input the external clock on pin OSC1. Figure 5.12 shows an e
connection. The duty cycle of the external clock signal must be 45 to 55%.

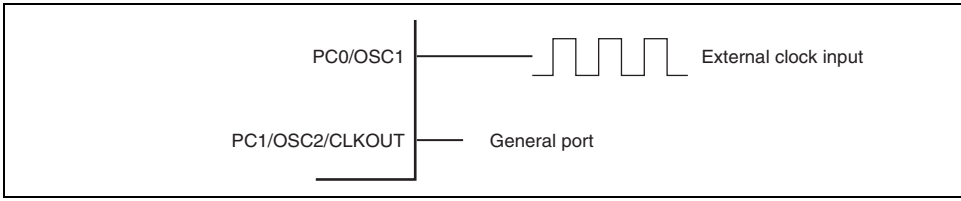


Figure 5.12 Example of External Clock Input

5.6 Prescaler

5.6.1 Prescaler S

Prescaler S is a 13-bit counter using the system clock (ϕ) as its input clock. The outputs,
divided clocks, are used as internal clocks by the on-chip peripheral modules. Prescaler
initialized to H'0000 by a reset, and starts counting on exit from the reset state. In standb
and subsleep mode, the system clock pulse generator stops. Prescaler S also stops and is
to H'0000. It cannot be read from or written to by the CPU.

The outputs from prescaler S is shared by the on-chip peripheral modules. The division
be set separately for each on-chip peripheral module. In active mode and sleep mode, th
input to prescaler S is a system clock with the division ratio specified by bits MA2 to M
SYSCR2.

5.7.2 Notes on Board Design

When using a crystal resonator (ceramic resonator), place the resonator and its load capac
close as possible to pins OSC1 and OSC2. Other signal lines should be routed away from
oscillator circuit to prevent induction from interfering with correct oscillation (see figure

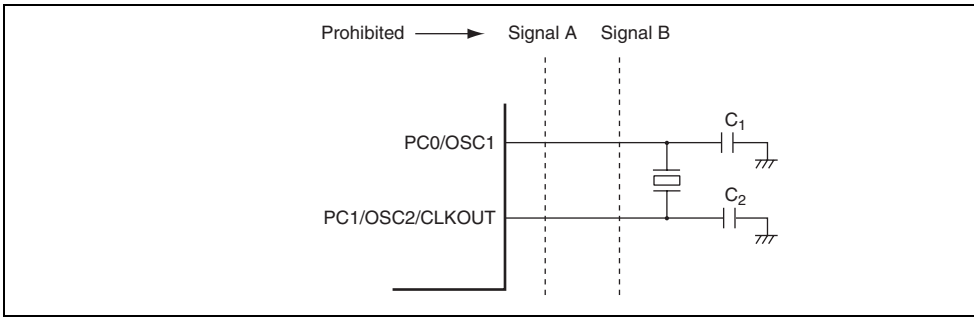


Figure 5.13 Example of Incorrect Board Design

- Sleep mode

  The CPU halts. On-chip peripheral modules are operable on the system clock.
- Standby mode

  The CPU and all on-chip peripheral modules halt.
- Subsleep mode

  The CPU and all on-chip peripheral modules halt. I/O ports keep the same states as b
  transition.
- Module standby function

  Independent of the above modes, power consumption can be reduced by halting on-c
  peripheral modules that are not used in module units.

6.1 Register Descriptions

The registers related to power-down modes are listed below.

- System control register 1 (SYSCR1)
- System control register 2 (SYSCR2)
- Module standby control register 1 (MSTCR1)
- Module standby control register 2 (MSTCR2)

				1: Shifts to standby mode. For details, see table 6.2.
6	STS2	0	R/W	Standby Timer Select 2 to 0
5	STS1	0	R/W	These bits set the wait time from when the syste
4	STS0	0	R/W	oscillator starts functioning until the clock is supp shifting from standby mode, to active mode or sl mode. During the wait time, this LSI automaticall the on-chip oscillator clock as its system clock a the number of wait states. Select a wait time of 6 (oscillation stabilization time) or longer, dependir operating frequency. Table 6.1 shows the relatio between the STS2 to STS0 values and the wait When using an external clock, set the wait time t 100 μs or longer in the F-ZTAT version. In the m ROM version, the minimum value (STS2 = STS1 = 1) is recommended. These bits also set the wait states for external os stabilization when system clock is switched from chip oscillator clock to the external clock by user software. The relationship between Nwait (number of wait oscillation stabilization) and Nstby (number of wa for recovering to the standby mode) is as follows Nstby ≤ Nwait ≤ 2 × Nstby
3 to 0	—	All 0	—	Reserved These bits are always read as 0.

1	1	0	128 states	0.01	0.02	0.03	0.03	0.05
1	1	1	16 states	0.00	0.00	0.00	0.00	0.00

Notes: 1. Time unit is ms.

2. The on-chip oscillator clock counts the wait states, even when the external cl
as system clock.

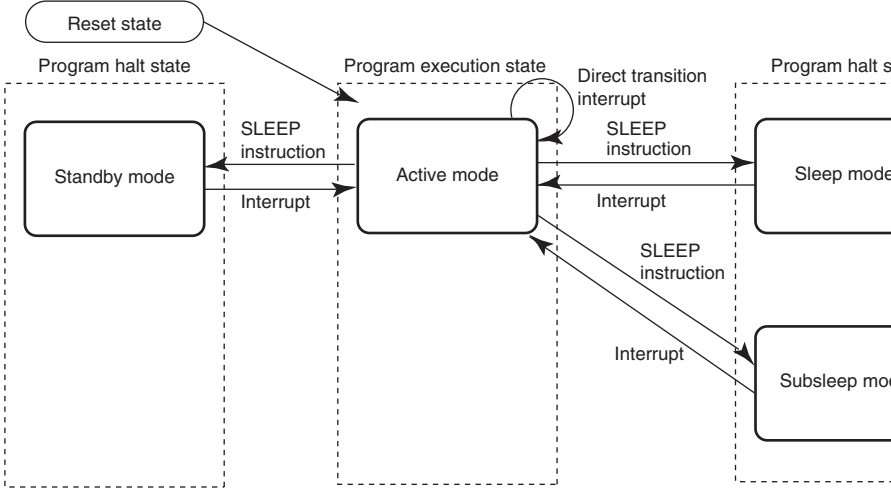
				This bit is always read as 0.
5	DTON	0	R/W	Direct Transfer on Flag This bit specifies the mode to be entered after ex the SLEEP instruction, as well as the SSBY bit in SYSCR1. For details, see table 6.2.
4	MA2	0	R/W	Active Mode Clock Select 2 to 0
3	MA1	0	R/W	These bits select the operating clock frequency i
2	MA0	0	R/W	and sleep modes. The operating clock frequency changes to the set frequency after the SLEEP in is executed. 0XX: ϕ 100: ϕ /8 101: ϕ/16 110: ϕ/32 111: ϕ/64
1, 0	—	All 0	—	Reserved These bits are always read as 0.

[Legend]

X: Don't care

5	MSTS3	0	R/W	SCI3 Module Standby SCI3 enters standby mode when this bit is set t
4	MSTAD	0	R/W	A/D Converter Module Standby A/D converter enters standby mode when this b to 1.
3	MSTWD	0	R/W	Watchdog Timer Module Standby Watchdog timer enters standby mode when this to 1. (When the internal oscillator is selected fo watchdog timer clock, the watchdog timer opera regardless of the setting of this bit.)
2	MSTTW	0	R/W	Timer W Module Standby Timer W enters standby mode when this bit is s
1	MSTTV	0	R/W	Timer V Module Standby Timer V enters standby mode when this bit is s
0	—	0	—	Reserved This bit is always read as 0.

3 to 0	—	All 0	—	Reserved These bits are always read as 0.



Notes: 1. To make a transition to another mode by an interrupt, make sure interrupt handling is after the in is accepted.
2. Details on the mode transition conditions are given in table 6.2.

Figure 6.1 Mode Transition Diagram

Note: When a state transition is performed while SMSEL is 1, timer V, SCI3, and the converter are reset, and all registers are set to their initial values. To use these functions after entering active mode, reset the registers.

Table 6.3 Internal State in Each Operating Mode

Function		Active Mode	Sleep Mode	Subsleep Mode	Standby Mode
System clock oscillator		Functioning	Functioning	Halted	Halted
CPU operations	Instructions	Functioning	Halted	Halted	Halted
	Registers	Functioning	Retained	Retained	Retained
RAM		Functioning	Retained	Retained	Retained
IO ports		Functioning	Retained	Retained	Register content retained, but outp high-impedance
External interrupts	IRQ3, IRQ0	Functioning	Functioning	Functioning	Functioning
	WKP5	Functioning	Functioning	Functioning	Functioning
Peripheral modules	Timer B1	Functioning	Functioning	Retained	Retained
	Timer V	Functioning	Functioning	Reset	Reset
	Timer W	Functioning	Functioning	Retained	Retained
	Watchdog timer	Functioning	Functioning	Retained (Functioning if the internal oscillator is as a count clock.)	
	SCI3	Functioning	Functioning	Reset	Reset
	IIC2	Functioning	Functioning	Retained	Retained
	A/D converter	Functioning	Functioning	Reset	Reset
	LVD	Functioning	Functioning	Functioning	Functioning

In standby mode, the system clock oscillator is halted, and operation of the CPU and on-
peripheral modules is halted. However, as long as the rated voltage is supplied, the conte
CPU registers, on-chip RAM, and some on-chip peripheral module registers are retained
RAM contents will be retained as long as the voltage set by the RAM data retention volt
provided. The I/O ports go to the high-impedance state.

Standby mode is cleared by an interrupt. When an interrupt is requested, the on-chip osc
starts functioning. The external oscillator also starts functioning when used. After the tir
the STS2 to STS0 bits in SYSCR1 has elapsed, standby mode is cleared and the CPU sta
interrupt exception handling. Standby mode is not cleared if the I bit in the condition co
(CCR) is set to 1 or the requested interrupt is disabled by the interrupt enable bit.

When the $\overline{\text{RES}}$ pin is driven low in standby mode, the on-chip oscillator starts functionin
system clock is supplied to the entire chip as soon as the on-chip oscillator starts functio
$\overline{\text{RES}}$ pin must be kept low for the rated period. On driving the $\overline{\text{RES}}$ pin high, after the os
stabilization time set by the power-on reset circuit has elapsed, the internal reset signal i
and the CPU starts reset exception handling.

register (CCR) is set to 1 or the requested interrupt is disabled by the interrupt enable bit.

When the $\overline{\text{RES}}$ pin is driven low in subsleep mode, the on-chip oscillator starts functionin system clock is supplied to the entire chip as soon as the on-chip oscillator starts function $\overline{\text{RES}}$ pin must be kept low for the rated period. On driving the $\overline{\text{RES}}$ pin high, after the osc stabilization time set by the power-on reset circuit has elapsed, the internal reset signal is and the CPU starts reset exception handling.

6.3 Operating Frequency in Active Mode

Operation in active mode is clocked at the frequency designated by the MA2 to MA0 bits SYSCR2. The operating frequency changes to the set frequency after SLEEP instruction execution.

6.4 Direct Transition

The CPU can execute programs in active mode. The operating frequency can be changed making a transition directly from active mode to active mode. A direct transition can be r executing a SLEEP instruction while the DTON bit in SYSCR2 is set to 1. If the direct tr interrupt is disabled by the interrupt enable register 1, a transition is made instead to sleep subsleep mode. Note that if a direct transition is attempted while the I bit in condition coc register (CCR) is set to 1, sleep mode or subsleep mode will be entered though that mode be cleared by means of an interrupt.

— The flash memory can be reprogrammed up to 1,000 times.

- On-board programming
  — On-board programming/erasing can be done in boot mode, in which the boot pro into the chip is started to erase or program of the entire flash memory. In normal program mode, individual blocks can be erased or programmed.
- Automatic bit rate adjustment
  — For data transfer in boot mode, this LSI's bit rate can be automatically adjusted to the transfer bit rate of the host.
- Programming/erasing protection
  — Sets software protection against flash memory programming/erasing.

7.1 Block Configuration

Figure 7.1 shows the block configuration of 12-kbyte flash memory. The thick lines indi erasing units, the narrow lines indicate programming units, and the values are addresses. memory is divided into 1 kbyte × 4 blocks and 4 kbytes × 2 blocks. Erasing is performed units. Programming is performed in 64-byte units starting from an address with lower ei H'00, H'40, H'80, or H'C0.

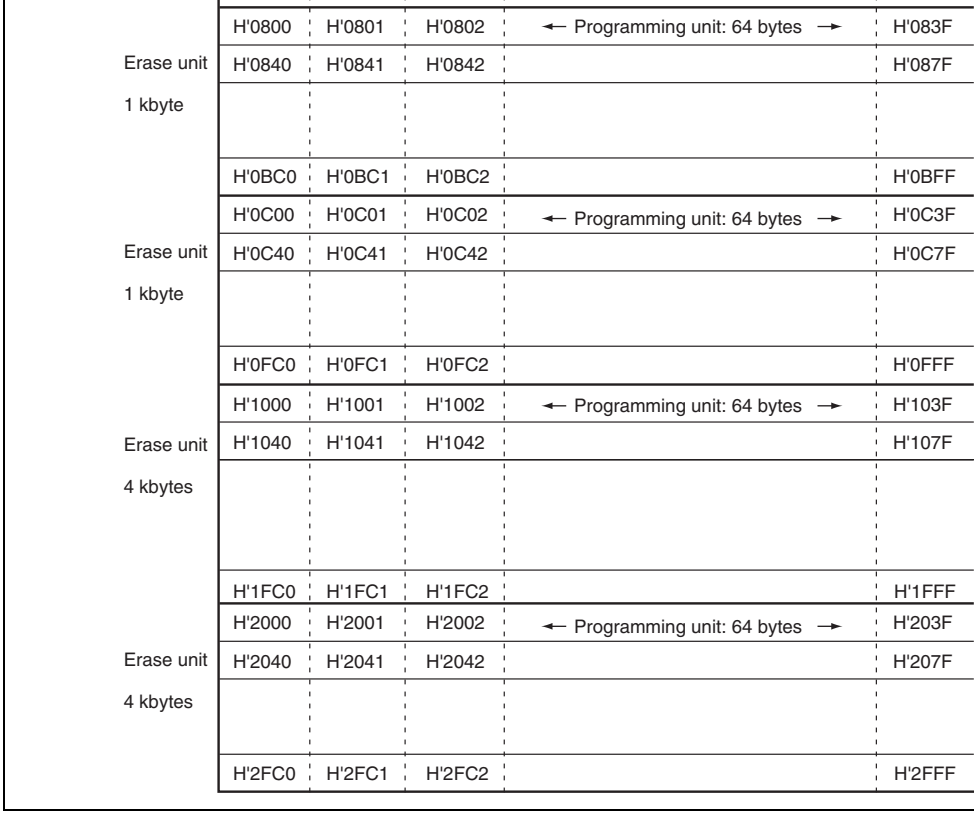


Figure 7.1 Flash Memory Block Configuration

FLMCR1 is a register that makes the flash memory change to program mode, program-v
mode, erase mode, or erase-verify mode. For details on register setting, refer to section 7
Memory Programming/Erasing.

Bit	Bit Name	Initial Value	R/W	Description
7	—	0	—	Reserved This bit is always read as 0.
6	SWE	0	R/W	Software Write Enable When this bit is set to 1, flash memory programming/erasing is enabled. When this bit to 0, other FLMCR1 register bits and all EBR1 b be set.
5	ESU	0	R/W	Erase Setup When this bit is set to 1, the flash memory char erase setup state. When it is cleared to 0, the e state is cancelled. Set this bit to 1 before settin to 1 in FLMCR1.
4	PSU	0	R/W	Program Setup When this bit is set to 1, the flash memory char program setup state. When it is cleared to 0, th setup state is cancelled. Set this bit to 1 before P bit in FLMCR1.
3	EV	0	R/W	Erase-Verify When this bit is set to 1, the flash memory char erase-verify mode. When it is cleared to 0, eras mode is cancelled.

				When this bit is set to 1 while SWE = 1 and PSU flash memory changes to program mode. When cleared to 0, program mode is cancelled.

7.2.2 Flash Memory Control Register 2 (FLMCR2)

FLMCR2 is a register that displays the state of flash memory programming/erasing. FLM read-only register, and should not be written to.

Bit	Bit Name	Initial Value	R/W	Description
7	FLER	0	R	Flash Memory Error Indicates that an error has occurred during an op on flash memory (programming or erasing). Whe is set to 1, flash memory goes to the error-protec state. See section 7.5.3, Error Protection, for details.
6 to 0	—	All 0	—	Reserved These bits are always read as 0.

5	EB5	0	R/W	When this bit is set to 1, 4 kbytes of H'2000 to H be erased.
4	EB4	0	R/W	When this bit is set to 1, 4 kbytes of H'1000 to H be erased.
3	EB3	0	R/W	When this bit is set to 1, 1 kbyte of H'0C00 to H be erased.
2	EB2	0	R/W	When this bit is set to 1, 1 kbyte of H'0800 to H be erased.
1	EB1	0	R/W	When this bit is set to 1, 1 kbyte of H'0400 to H be erased.
0	EB0	0	R/W	When this bit is set to 1, 1 kbyte of H'0000 to H be erased.

7.2.4 Flash Memory Enable Register (FENR)

Bit 7 (FLSHE) in FENR enables or disables the CPU access to the flash memory control FLMCR1, FLMCR2, and EBR1.

Bit	Bit Name	Initial Value	R/W	Description
7	FLSHE	0	R/W	Flash Memory Control Register Enable Flash memory control registers can be accesse this bit is set to 1. Flash memory control registe be accessed when this bit is set to 0.
6 to 0	—	All 0	—	Reserved These bits are always read as 0.

This can be used for programming initial values in the on-board state or for a forcible retu
programming/erasing can no longer be done in user program mode. In user program mod
individual blocks can be erased and programmed by branching to the user program/erase
program prepared by the user.

Table 7.1 Setting Programming Modes

TEST	$\overline{\text{NMI}}$	E10T_0	LSI State after Reset End
0	1	X	User mode
0	0	1	Boot mode

[Legend]
X: Don't care

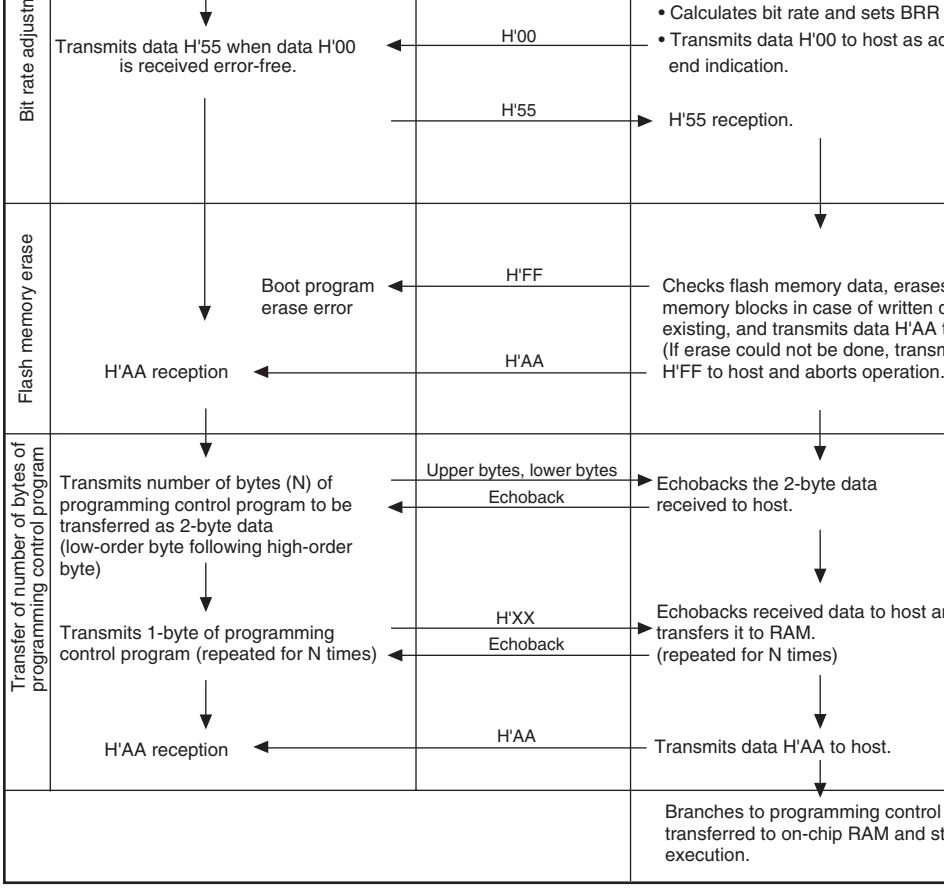
7.3.1 Boot Mode

Table 7.2 shows the boot mode operations between reset end and branching to the progra
control program.

1. When boot mode is used, the flash memory programming control program must be pr
   the host beforehand. Prepare a programming control program in accordance with the
   description in section 7.4, Flash Memory Programming/Erasing.
2. SCI3 should be set to asynchronous mode, and the transfer format as follows: 8-bit da
   bit, and no parity.

the bit rates of the host and the chip. To operate the SCI properly, set the host's trans
and system clock frequency of this LSI within the ranges listed in table 7.3.

5. In boot mode, a part of the on-chip RAM area is used by the boot program. The area
H'FEEF is the area to which the programming control program is transferred from th
The boot program area cannot be used until the execution state in boot mode switche
programming control program.
6. Before branching to the programming control program, the chip terminates transfer o
by SCI3 (by clearing the RE and TE bits in SCR3 to 0), however the adjusted bit rat
remains set in BRR. Therefore, the programming control program can still use it for
write data or verify data with the host. The TxD pin is high (PCR22 = 1, P22 = 1). T
contents of the CPU general registers are undefined immediately after branching to t
programming control program. These registers must be initialized at the beginning o
programming control program, as the stack pointer (SP), in particular, is used implic
subroutine calls, etc.
7. Boot mode can be cleared by a reset. End the reset after driving the reset pin low, wa
least 20 states, and then setting the TEST pin and $\overline{\text{NMI}}$ pin. Boot mode is also cleare
WDT overflow occurs.
8. Do not change the TEST pin and $\overline{\text{NMI}}$ pin input levels in boot mode.


Bit rate adjustm
Transmits data H'55 when data H'00 is received error-free.
H'00
• Calculates bit rate and sets BRR
• Transmits data H'00 to host as ac
end indication.
H'55
H'55 reception.
Flash memory erase
Boot program erase error
H'FF
Checks flash memory data, erases
memory blocks in case of written d
existing, and transmits data H'AA t
(If erase could not be done, transm
H'FF to host and aborts operation.
H'AA reception
H'AA
Transfer of number of bytes of programming control program
Transmits number of bytes (N) of programming control program to be transferred as 2-byte data (low-order byte following high-order byte)
Upper bytes, lower bytes
Echoback
Echobacks the 2-byte data received to host.
Transmits 1-byte of programming control program (repeated for N times)
H'XX
Echoback
Echobacks received data to host a
transfers it to RAM.
(repeated for N times)
H'AA reception
H'AA
Transmits data H'AA to host.
Branches to programming control
transferred to on-chip RAM and st
execution.

Flash Memory Programming/Erasing.

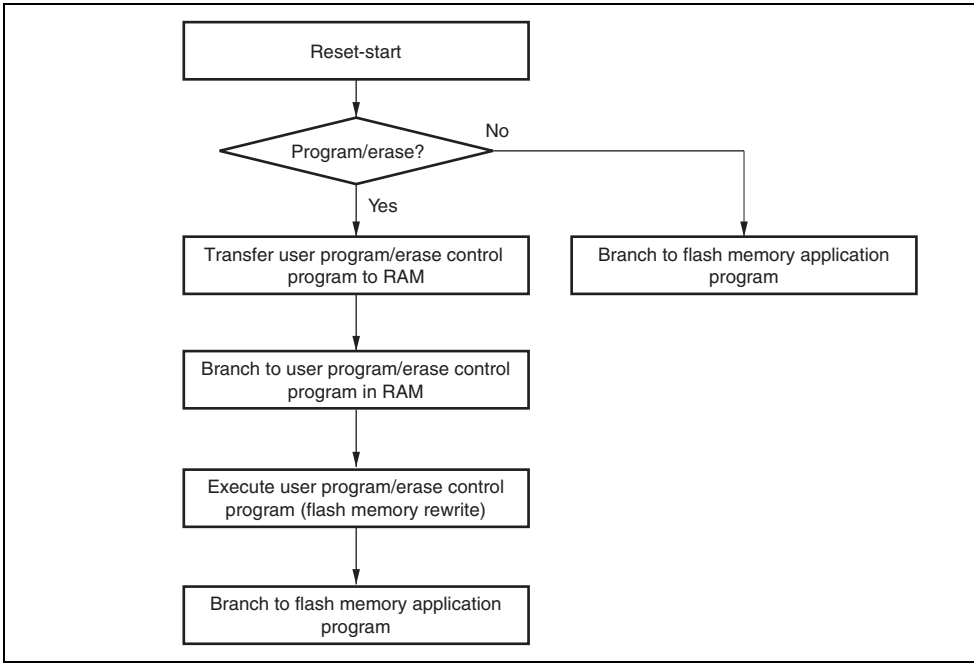
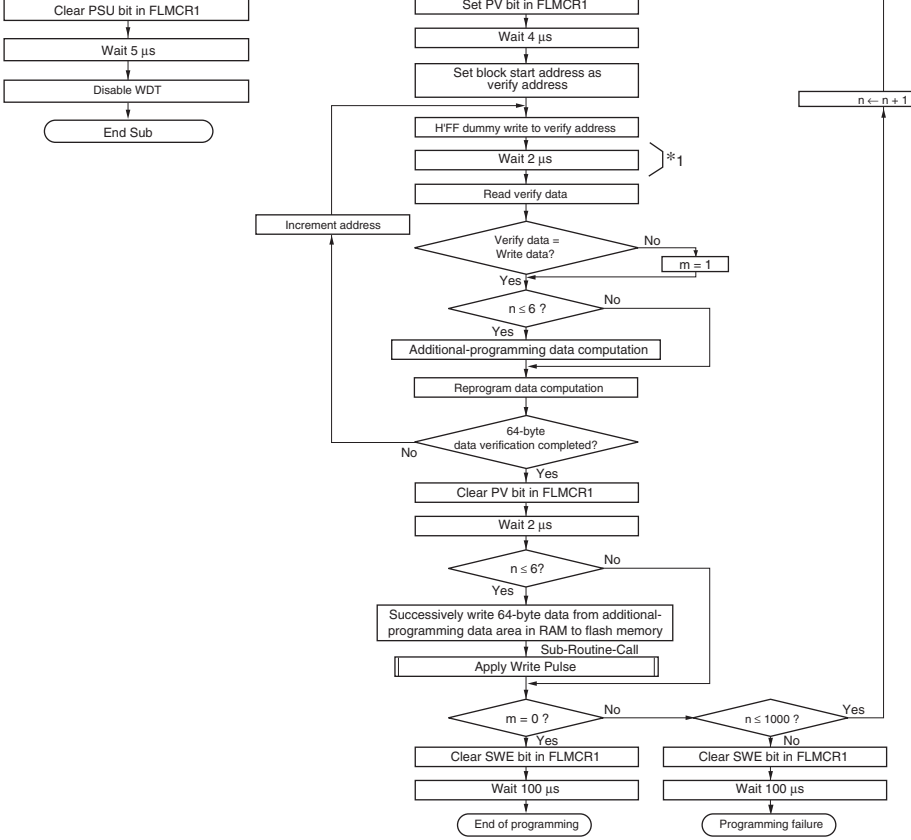


Figure 7.2 Programming/Erasing Flowchart Example in User Program Mo

7.4.1 Program/Program-Verify

When writing data or programs to the flash memory, the program/program-verify flowch in figure 7.3 should be followed. Performing programming operations according to this will enable data or programs to be written to the flash memory without subjecting the ch voltage stress or sacrificing program data reliability.

1. Programming must be done to an empty address. Do not reprogram an address to wh programming has already been performed.
2. Programming should be carried out 64 bytes at a time. A 64-byte data transfer must b performed even if writing fewer than 64 bytes. In this case, H'FF data must be writte extra addresses.
3. Prepare the following data storage areas in RAM: A 64-byte programming data area, reprogramming data area, and a 64-byte additional-programming data area. Perform reprogramming data computation according to table 7.4, and additional programmin computation according to table 7.5.
4. Consecutively transfer 64 bytes of data in byte units from the reprogramming data ar additional-programming data area to the flash memory. The program address and 64 are latched in the flash memory. The lower 8 bits of the start address in the flash mem destination area must be H'00, H'40, H'80, or H'C0.
5. The time during which the P bit is set to 1 is the programming time. Table 7.6 shows allowable programming times.
6. The watchdog timer (WDT) is set to prevent overprogramming due to program runaw An overflow cycle of approximately 6.6 ms is allowed.
7. For a dummy write to a verify address, write 1-byte data H'FF to an even address. Ve can be read in words from the address to which a dummy write was performed.
8. The maximum number of repetitions of the program/program-verify sequence of the is 1,000.



Notes: 1. The RTS instruction must not be used during the following (1) and (2) periods.
(1) A period between 64-byte data programming to flash memory and the P bit clearing
(2) A period between dummy writing of H'FF to a verify address and verify data reading
2. When WDT is in use, disable it once.

Figure 7.3 Program/Program-Verify Flowchart

Reprogram Data	Verify Data	Additional-Program Data	Comments
0	0	0	Additional-program
0	1	1	No additional progra
1	0	1	No additional progra
1	1	1	No additional progra

Table 7.6 Programming Time

n (Number of Writes)	Programming Time	In Additional Programming	Comments
1 to 6	30	10	
7 to 1,000	200	—	

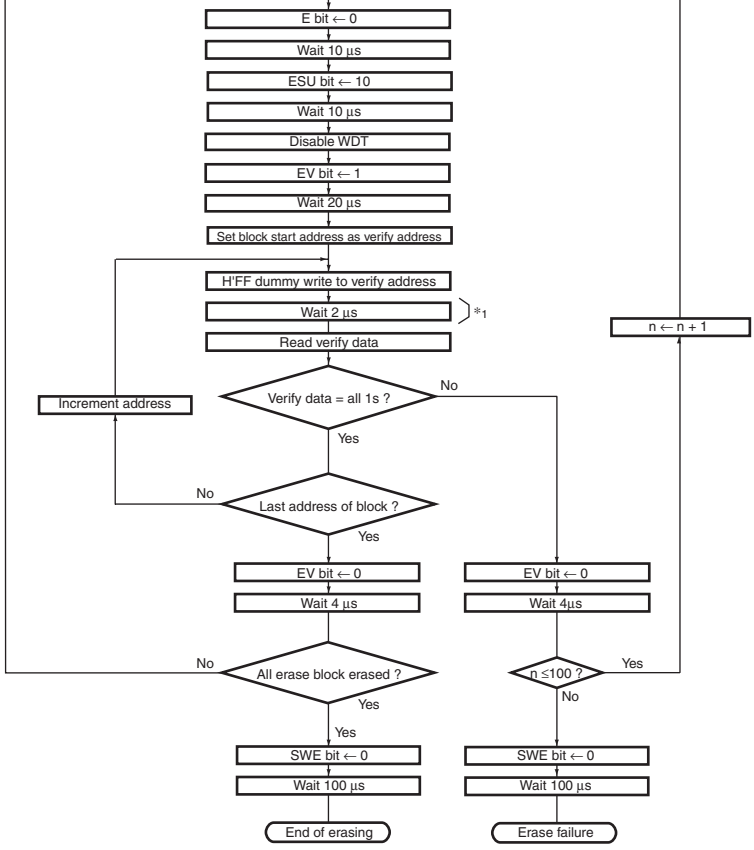
Note: Time shown in µs.

7.4.2 Erase/Erase-Verify

When erasing flash memory, the erase/erase-verify flowchart shown in figure 7.4 should followed.

1. Prewriting (setting erase block data to all 0s) is not necessary.
2. Erasing is performed in block units. Make only a single-bit specification in the erase register (EBR1). To erase multiple blocks, each block must be erased in turn.
3. The time during which the E bit is set to 1 is the flash memory erase time.
4. The watchdog timer (WDT) is set to prevent overerasing due to program runaway, e overflow cycle of approximately 19.8 ms is allowed.
5. For a dummy write to a verify address, write 1-byte data H'FF to an even address. V can be read in words from the address to which a dummy write was performed.

2. If interrupt exception handling starts before the vector address is written or during programming/erasing, a correct vector cannot be fetched and the CPU malfunctions.
3. If an interrupt occurs during boot program execution, normal boot mode sequence can carried out.



Notes: 1. The RTS instruction must not be used during a period between dummy writing of H'FF to a verify address and verify data readin
2. When WDT is in use, disable it once.

Figure 7.4 Erase/Erase-Verify Flowchart

unless the $\overline{\text{RES}}$ pin is held low until oscillation stabilizes after powering on. In the case of during operation, hold the $\overline{\text{RES}}$ pin low for the $\overline{\text{RES}}$ pulse width specified in the AC Characteristics section.

7.5.2 Software Protection

Software protection can be implemented against programming/erasing of all flash memor by clearing the SWE bit in FLMCR1. When software protection is in effect, setting the P in FLMCR1 does not cause a transition to program mode or erase mode. By setting the er register 1 (EBR1), erase protection can be set for individual blocks. When EBR1 is set to erase protection is set for all blocks.

7.5.3 Error Protection

In error protection, an error is detected when CPU runaway occurs during flash memory programming/erasing, or operation is not performed in accordance with the program/eras algorithm, and the program/erase operation is aborted. Aborting the program/erase operat prevents damage to the flash memory due to overprogramming or overerasing.

When the following errors are detected during programming/erasing of flash memory, the bit in FLMCR2 is set to 1, and the error protection state is entered.

- When the flash memory of the relevant address area is read during programming/eras (including vector read and instruction fetch)
- Immediately after exception handling excluding a reset during programming/erasing
- When a SLEEP instruction is executed during programming/erasing

Masked ROM version	H8/36912, H8/36902	512 bytes	H'FD80 to H'FF7F
	H8/36911, H8/36901	256 bytes	H'FE80 to H'FF7F
	H8/36900	256 bytes	H'FE80 to H'FF7F

Note: * When the E7 or E8 is used, area H'F980 to H'FD7F must not be accessed.

units. For functions in each port, see appendix B.1, I/O Port Block Diagrams. For the ex
bit manipulation instructions to the port control register and port data register, see sectio
Bit Manipulation Instruction.

9.1 Port 1

Port 1 is a general I/O port also functioning as an IRQ interrupt input pin and timer V in
Figure 9.1 shows its pin configuration.

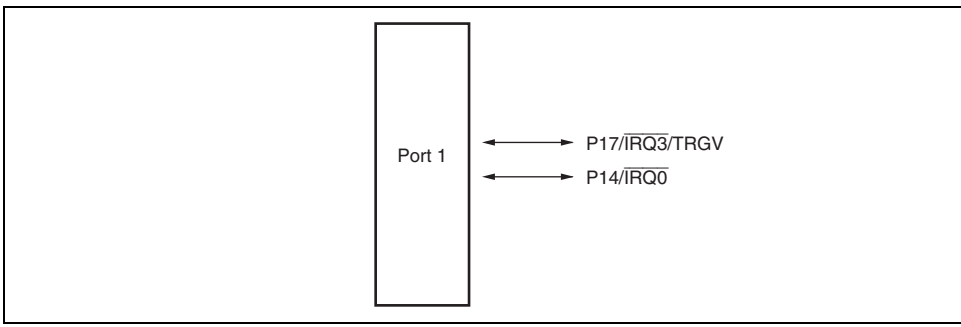


Figure 9.1 Port 1 Pin Configuration

Port 1 has the following registers.

- Port mode register 1 (PMR1)
- Port control register 1 (PCR1)
- Port data register 1 (PDR1)
- Port pull-up control register 1 (PUCR1)

				1: IRQ3/TRGV input pin
6, 5	—	All 0	—	Reserved These bits are always read as 0.
4	IRQ0	0	R/W	P14/$\overline{\text{IRQ0}}$ Pin Function Switch Selects whether pin P14/$\overline{\text{IRQ0}}$ is used as P14 or $\overline{\text{IRQ0}}$. 0: General I/O port 1: $\overline{\text{IRQ0}}$ input pin
3, 2	—	All 0	—	Reserved These bits are always read as 0.
1	TXD	0	R/W	P22/TXD Pin Function Switch Selects whether pin P22/TXD is used as P22 or 0: General I/O port 1: TXD output pin
0	—	0	—	Reserved This bit is always read as 0.

3	—	—	—	
2	—	—	—	
1	—	—	—	
0	—	—	—	

9.1.3 Port Data Register 1 (PDR1)

PDR1 is a general I/O port data register of port 1.

Bit	Bit Name	Initial Value	R/W	Description
7	P17	0	R/W	These bits store output data for port 1 pins.
6	—	1	—	If PDR1 is read while PCR1 bits are set to 1, th
5	—	1	—	stored in PDR1 are read. If PDR1 is read while
4	P14	0	R/W	are cleared to 0, the pin states are read regardl value stored in PDR1.
3	—	1	—	Bits 6, 5, and 3 to 0 are reserved. These bits ar
2	—	1	—	read as 1.
1	—	1	—	
0	—	1	—	

3	—	1	—	read as 1.
2	—	1	—	
1	—	1	—	
0	—	1	—	

9.1.5 Pin Functions

The correspondence between the register specification and the port functions is shown be

- P17/IRQ3/TRGV pin

Register	PMR1	PCR1	
Bit Name	**IRQ3**	**PCR17**	**Pin Function**
Setting value	0	0	P17 input pin
		1	P17 output pin
	1	X	$\overline{\text{IRQ3}}$ input/TRGV input pin

[Legend]
X: Don't care

9.2 Port 2

Port 2 is a general I/O port also functioning as a SCI3 I/O pin. Each pin of the port 2 is s
figure 9.2. The register settings of PMR1 and SCI3 have priority for functions of the pin
uses.

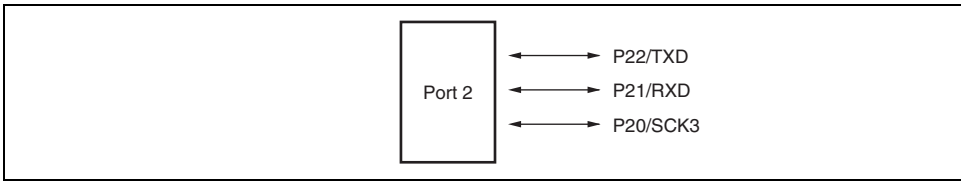


Figure 9.2 Port 2 Pin Configuration

Port 2 has the following registers.

- Port control register 2 (PCR2)
- Port data register 2 (PDR2)

9.2.2 Port Data Register 2 (PDR2)

PDR2 is a general I/O port data register of port 2.

Bit	Bit Name	Initial Value	R/W	Description
7 to 3	—	All 1	—	Reserved These bits are always read as 1.
2	P22	0	R/W	These bits store output data for port 2 pins.
1	P21	0	R/W	If PDR2 is read while PCR2 bits are set to 1, the stored in PDR2 is read. If PDR2 is read while PC are cleared to 0, the pin states are read regardle value stored in PDR2.
0	P20	0	R/W	

		1	P22 output pin
	1	X	TXD output pin

[Legend]
X: Don't care

- P21/RXD pin

Register	SCR3	PCR2	
Bit Name	**RE**	**PCR21**	**Pin Function**
Setting value	0	0	P21 input pin
		1	P21 output pin
	1	X	RXD input pin

[Legend]
X: Don't care

- P20/SCK3 pin

Register	SCR3		SMR	PCR2	
Bit Name	**CKE1**	**CKE0**	**COM**	**PCR20**	**Pin Function**
Setting value	0	0	0	0	P20 input pin
				1	P20 output pin
	0	0	1	X	SCK3 output pin
	0	1	X	X	SCK3 output pin
	1	X	X	X	SCK3 input pin

[Legend]
X: Don't care

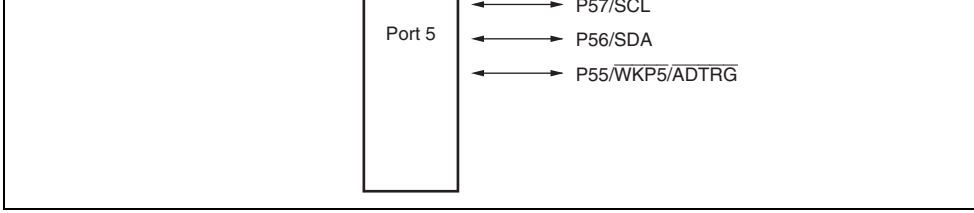


Figure 9.3 Port 5 Pin Configuration

Port 5 has the following registers.

- Port mode register 5 (PMR5)
- Port control register 5 (PCR5)
- Port data register 5 (PDR5)
- Port pull-up control register 5 (PUCR5)

				or as WKP5/ADTRG. 0: General I/O port 1: $\overline{\text{WKP5}}$/$\overline{\text{ADTRG}}$ input pin
4 to 0	—	All 0	—	Reserved These bits are always read as 0.

9.3.2 Port Control Register 5 (PCR5)

PCR5 selects inputs/outputs in bit units for pins to be used as general I/O ports of port 5

Bit	Bit Name	Initial Value	R/W	Description
7	PCR57	0	W	When each of the port 5 pins, P57 to P55, func
6	PCR56	0	W	general I/O port, setting a PCR5 bit to 1 makes
5	PCR55	0	W	corresponding pin an output port, while clearing 0 makes the pin an input port.
4 to 0	—	—	—	Reserved

4 to 0	—	All 1	—	Reserved These bits are always read as 1.

9.3.4 Port Pull-Up Control Register 5 (PUCR5)

PUCR5 controls the pull-up MOS in bit units of the pins set as the input ports.

Bit	Bit Name	Initial Value	R/W	Description
7, 6	—	All 0	—	Reserved These bits are always read as 0.
5	PUCR55	0	R/W	Only bits for which PCR5 is cleared are valid. The pull-up MOS of the corresponding pins ente state when this bit is set to 1, while they enter th state when this bit is cleared to 0.
4 to 0	—	All 0	—	Reserved These bits are always read as 0.

	1	X	SCL I/O pin*

[Legend]

X: Don't care

Note: As the SCL output form is NMOS open-drain, direct bus drive is enabled.

* Supported only by the H8/36912 Group.

- P56/SDA pin

Register	**ICCR**	**PCR5**	
Bit Name	**ICE**	**PCR56**	**Pin Function**
Setting value	0	0	P56 input pin
		1	P56 output pin
	1	X	SDA I/O pin*

[Legend]

X: Don't care

Note: As the SDA output form is NMOS open-drain, direct bus drive is enabled.

* Supported only by the H8/36912 Group.

- P55/$\overline{\text{WKP5}}$/$\overline{\text{ADTRG}}$ pin

Register	**PMR5**	**PCR5**	
Bit Name	**WKP5**	**PCR55**	**Pin Function**
Setting value	0	0	P55 input pin
		1	P55 output pin
	1	X	$\overline{\text{WKP5}}$/$\overline{\text{ADTRG}}$ input pin

[Legend]

X: Don't care

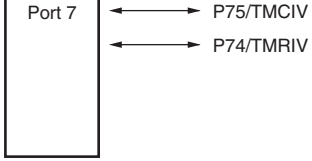


Figure 9.4 Port 7 Pin Configuration

Port 7 has the following registers.

- Port control register 7 (PCR7)
- Port data register 7 (PDR7)

9.4.1 Port Control Register 7 (PCR7)

PCR7 selects inputs/outputs in bit units for pins to be used as general I/O ports of port 7.

Bit	Bit Name	Initial Value	R/W	Description
7	—	—	—	Reserved
6	PCR76	0	W	Setting a PCR7 bit to 1 makes the correspondin
5	PCR75	0	W	output port, while clearing the bit to 0 makes the input port. Note that the TCSRV setting of the tin
4	PCR74	0	W	priority for deciding input/output direction of the P76/TMOV pin.
3 to 0	—	—	—	Reserved

4	P74	0	R/W	stored in PDR7 is read. If PDR7 is read while P are cleared to 0, the pin states are read regardl value stored in PDR7.
3 to 0	—	All 1	—	Reserved These bits are always read as 1.

9.4.3 Pin Functions

The correspondence between the register specification and the port functions is shown b

- P76/TMOV pin

Register	TCSRV	PCR7	
Bit Name	**OS3 to OS0**	**PCR76**	**Pin Function**
Setting value	0000	0	P76 input pin
		1	P76 output pin
	Other than the above values	X	TMOV output pin

[Legend]

X: Don't care

Bit Name	PCR74	Pin Function
Setting value	0	P74 input/TMRIV input pin
	1	P74 output/TMRIV input pin

9.5 Port 8

Port 8 is a general I/O port also functioning as a timer W I/O pin. Each pin of the port 8 is in figure 9.5. The register setting of the timer W has priority for functions of the P84/FTI P83/FTIOC, P82/FTIOB, and P81/FTIOA pins. The P80/FTCI pin also functions as a tim input port that is connected to the timer W regardless of the register setting of port 8.

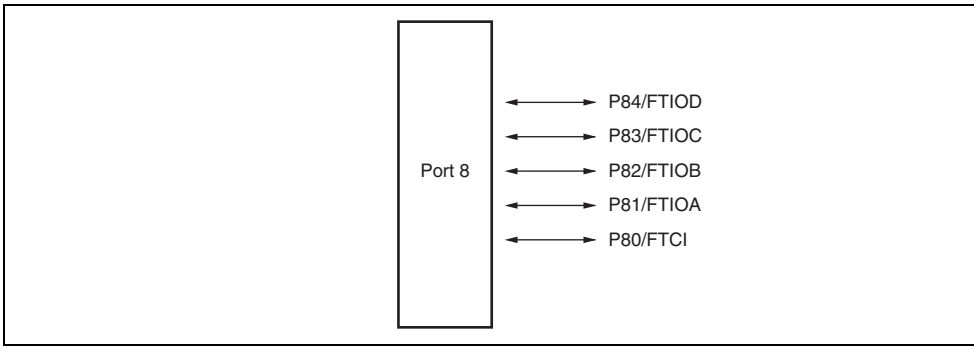


Figure 9.5 Port 8 Pin Configuration

Port 8 has the following registers.

- Port control register 8 (PCR8)
- Port data register 8 (PDR8)

1	PCR81	0	W
0	PCR80	0	W

9.5.2 Port Data Register 8 (PDR8)

PDR8 is a general I/O port data register of port 8.

Bit	Bit Name	Initial Value	R/W	Description
7 to 5	—	All 0	—	Reserved
4	P84	0	R/W	These bits store output data for port 8 pins.
3	P83	0	R/W	If PDR8 is read while PCR8 bits are set to 1, th
2	P82	0	R/W	stored in PDR8 is read. If PDR8 is read while P
1	P81	0	R/W	are cleared to 0, the pin states are read regardl
0	P80	0	R/W	value stored in PDR8.

				1	P84 output/FTIOD inp
		0	0	1	X	FTIOD output pin
		0	1	X	X	FTIOD output pin
		1	X	X	0	P84 input/FTIOD inpu
					1	P84 output/FTIOD inp
	1	X	X	X	X	PWM output pin

[Legend]
X: Don't care

- P83/FTIOC pin

Register	**TMRW**	**TIOR1**			**PCR8**	
Bit Name	**PWMC**	**IOC2**	**IOC1**	**IOC0**	**PCR83**	**Pin Function**
Setting value	0	0	0	0	0	P83 input/FTIOC inpu
					1	P83 output/FTIOC inp
		0	0	1	X	FTIOC output pin
		0	1	X	X	FTIOC output pin
		1	X	X	0	P83 input/FTIOC inpu
					1	P83 output/FTIOC inp
	1	X	X	X	X	PWM output pin

[Legend]
X: Don't care

				1	P82 output/FTIOB in
	1	X	X	X	X	PWM output pin

[Legend]
X: Don't care

- P81/FTIOA pin

Register		TIOR0		PCR8	
Bit Name	**IOA2**	**IOA1**	**IOA0**	**PCR81**	**Pin Function**
Setting value	0	0	0	0	P81 input/FTIOA input pin
				1	P81 output/FTIOA input pin
	0	0	1	X	FTIOA output pin
	0	1	X	X	FTIOA output pin
	1	X	X	0	P81 input/FTIOA input pin
				1	P81 output/FTIOA input pin

[Legend]
X: Don't care

- P80/FTCI pin

Register	PCR8	
Bit Name	**PCR80**	**Pin Function**
Setting value	0	P80 input/FTCI input pin
	1	P80 output/FTCI input pin

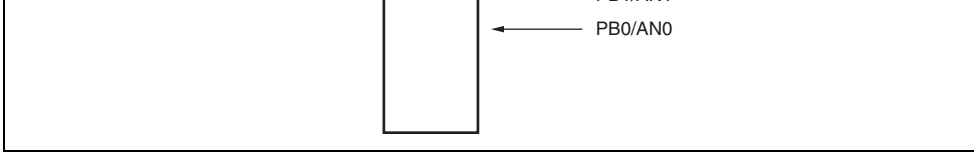


Figure 9.6 Port B Pin Configuration

Port B has the following register.

- Port data register B (PDRB)

9.6.1 Port Data Register B (PDRB)

PDRB is a general input-only port data register of port B.

Bit	Bit Name	Initial Value	R/W	Description
7 to 4	—	—	—	Reserved
3	PB3	—	R	The input value of each pin is read by reading th register.
2	PB2	—	R	
1	PB1	—	R	However, if a port B pin is designated as an anal channel by ADCSR in A/D converter or external comparison voltage input pin by LVDCR in low-v detection circuit, 0 is read.
0	PB0	—	R	

	Other than the above values				1	PB3 input pin
					0	PB3 input/ExtU input

- PB2/AN2/ExtD pin

Register	ADCSR				LVDCR	
Bit Name	**CH2**	**SCAN**	**CH1**	**CH0**	**VDDII**	**Pin Function**
Setting value	0	0	1	0	1	AN2 input pin
	0	1	1	X	0	AN2 input/ExtD input
	Other than the above values				1	PB2 input pin
					0	PB2 input/ExtD input

[Legend]
X: Don't care

- PB1/AN1 pin

Register	ADCSR				
Bit Name	**CH2**	**SCAN**	**CH1**	**CH0**	**Pin Function**
Setting value	0	X	0	1	AN1 input pin
	0	1	1	X	
	Other than the above values				PB1 input pin

[Legend]
X: Don't care

9.7 Port C

Port C is a general I/O port also functioning as an external oscillation pin and clock outpu
Each pin of the port C is shown in figure 9.7. The register setting of CKCSR has priority
functions of the pins for both uses.

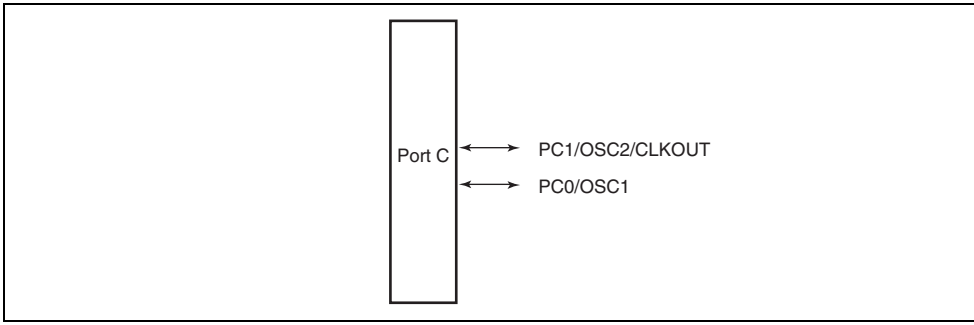


Figure 9.7 Port C Pin Configuration

Port C has the following registers.

- Port control register C (PCRC)
- Port data register C (PDRC)

9.7.2 Port Data Register C (PDRC)

PDRC is a general I/O port data register of port C.

Bit	Bit Name	Initial Value	R/W	Description
7 to 2	—	—	—	Reserved
1	PC1	0	R/W	These bits store output data for port C pins.
0	PC0	0	R/W	If PDRC is read while PCRC bits are set to 1, th stored in PDRC is read. If PDRC is read while P are cleared to 0, the pin states are read regardl value stored in PDRC.

	1	0	X	CLKOUT output pin
		1	X	OSC2 oscillation pin

[Legend]
X: Don't care

- PC0/OSC1 pin

Register	CKCSR	PCRC	
Bit Name	**PMRC0**	**PCRC0**	**Pin Function**
Setting value	0	0	PC0 input pin
		1	PC0 output pin
	1	X	OSC1 oscillation pin

[Legend]
X: Don't care

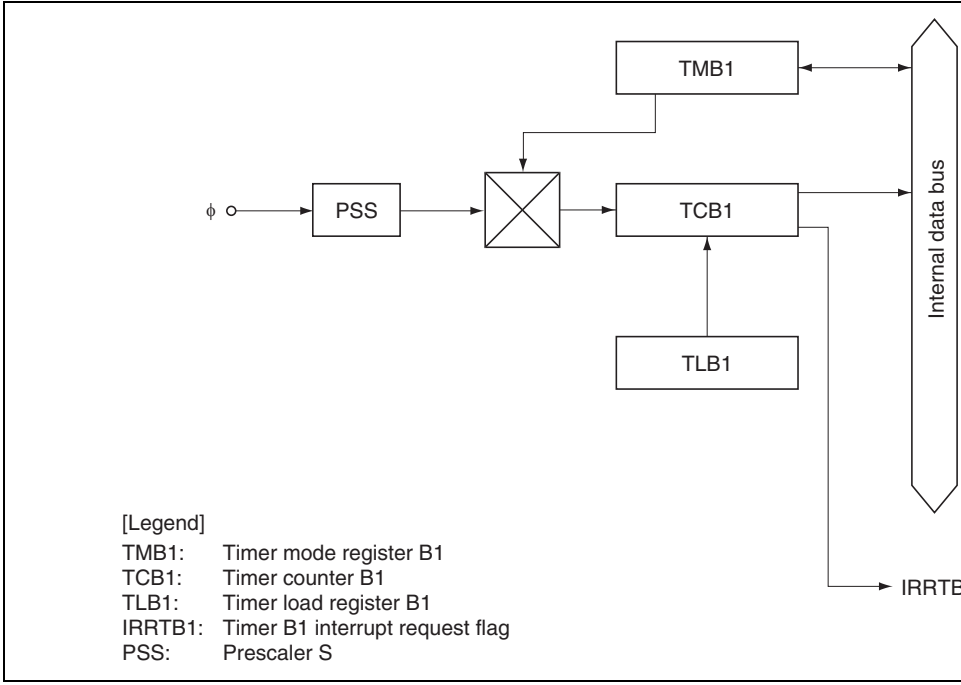


Figure 10.1 Block Diagram of Timer B1

TMB1 selects the auto-reload function and input clock.

Bit	Bit Name	Initial Value	R/W	Description
7	TMB17	0	R/W	Auto-Reload Function Select 0: Interval timer function selected 1: Auto-reload function selected
6	—	1	R/W	Reserved Although this bit is readable/writable, it should no to 0.
5 to 3	—	All 1	—	Reserved These bits are always read as 1.
2	TMB12	0	R/W	Clock Select
1	TMB11	0	R/W	000: Internal clock: φ/8192
0	TMB10	0	R/W	001: Internal clock: φ/2048 010: Internal clock: φ/512 011: Internal clock: φ/256 100: Internal clock: φ/64 101: Internal clock: φ/16 110: Internal clock: φ/4 111: Reserved (setting prohibited)

set in TLB1, the same value is loaded into TCB1 as well, and TCB1 starts counting up f
value. When TCB1 overflows during operation in auto-reload mode, the TLB1 value is l
into TCB1. Accordingly, overflow periods can be set within the range of 1 to 256 input
TLB1 is allocated to the same address as TCB1.

overflow, setting flag IRRTB1 in IRR2 to 1. If IENTB1 in IENR2 is 1, an interrupt is req
the CPU.

At overflow, TCB1 returns to H'00 and starts counting up again. During interval timer op
(TMB17 = 0), when a value is set in TLB1, the same value is set in TCB1.

10.3.2 Auto-Reload Timer Operation

Setting bit TMB17 in TMB1 to 1 causes timer B1 to function as an 8-bit auto-reload time
a reload value is set in TLB1, the same value is loaded into TCB1, becoming the value fr
TCB1 starts its count. After the count value in TCB1 reaches H'FF, the next clock signal
causes timer B1 to overflow. The TLB1 value is then loaded into TCB1, and the count co
from that value. The overflow period can be set within a range from 1 to 256 input clocks
depending on the TLB1 value.

The clock sources and interrupts in auto-reload mode are the same as in interval mode. In
reload mode (TMB17 = 1), when a new value is set in TLB1, the TLB1 value is also load
TCB1.

- Choice of seven clock signals is available.
  Choice of six internal clock sources (ϕ/128, ϕ/64, ϕ/32, ϕ/16, ϕ/8, ϕ/4) or an external
- Counter can be cleared by compare match A or B, or by an external reset signal. If th
  stop function is selected, the counter can be halted when cleared.
- Timer output is controlled by two independent compare match signals, enabling puls
  with an arbitrary duty cycle, PWM output, and other applications.
- Three interrupt sources: compare match A, compare match B, timer overflow
- Counting can be initiated by trigger input at the TRGV pin. The rising edge, falling e
  both edges of the TRGV input can be selected.

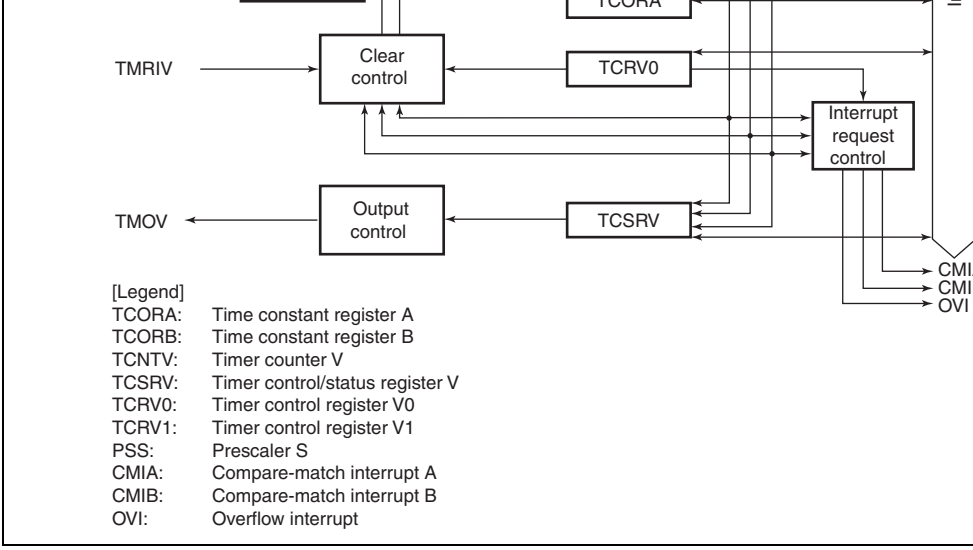


Figure 11.1 Block Diagram of Timer V

11.3 Register Descriptions

Time V has the following registers.

- Timer counter V (TCNTV)
- Timer constant register A (TCORA)
- Timer constant register B (TCORB)
- Timer control register V0 (TCRV0)
- Timer control/status register V (TCSRV)
- Timer control register V1 (TCRV1)

11.3.1 Timer Counter V (TCNTV)

TCNTV is an 8-bit up-counter. The clock source is selected by bits CKS2 to CKS0 in ti
control register V0 (TCRV0). The TCNTV value can be read and written by the CPU at
TCNTV can be cleared by an external reset input signal, or by compare match A or B. T
clearing signal is selected by bits CCLR1 and CCLR0 in TCRV0.

When TCNTV overflows, OVF is set to 1 in timer control/status register V (TCSRV).

TCNTV is initialized to H'00.

and the settings of bits OS3 to OS0 in TCSRV.

TCORA and TCORB are initialized to H'FF.

11.3.3 Timer Control Register V0 (TCRV0)

TCRV0 selects the input clock signals of TCNTV, specifies the clearing conditions of TC and controls each interrupt request.

Bit	Bit Name	Initial Value	R/W	Description
7	CMIEB	0	R/W	Compare Match Interrupt Enable B When this bit is set to 1, interrupt request from th bit in TCSRV is enabled.
6	CMIEA	0	R/W	Compare Match Interrupt Enable A When this bit is set to 1, interrupt request from th bit in TCSRV is enabled.
5	OVIE	0	R/W	Timer Overflow Interrupt Enable When this bit is set to 1, interrupt request from th bit in TCSRV is enabled.

2	CKS2	0	R/W	Clock Select 2 to 0
1	CKS1	0	R/W	These bits select clock signals to input to TCNT counting condition in combination with ICKS0 in
0	CKS0	0	R/W	Refer to table 11.2.

Table 11.2 Clock Signals to Input to TCNTV and Counting Conditions

TCRV0			TCRV1	
Bit 2	**Bit 1**	**Bit 0**	**Bit 0**	
CKS2	**CKS1**	**CKS0**	**ICKS0**	**Description**
0	0	0	—	Clock input prohibited
		1	0	Internal clock: counts on φ/4, falling e
			1	Internal clock: counts on φ/8, falling e
	1	0	0	Internal clock: counts on φ/16, falling
			1	Internal clock: counts on φ/32, falling
		1	0	Internal clock: counts on φ/64, falling
			1	Internal clock: counts on φ/128, fallin
1	0	0	—	Clock input prohibited
		1	—	External clock: counts on rising edge
	1	0	—	External clock: counts on falling edge
		1	—	External clock: counts on rising and f edge

				• After reading CMFB = 1, cleared by writing 0
6	CMFA	0	R/W	Compare Match Flag A [Setting condition] • When the TCNTV value matches the TCORA [Clearing condition] • After reading CMFA = 1, cleared by writing 0
5	OVF	0	R/W	Timer Overflow Flag [Setting condition] • When TCNTV overflows from H'FF to H'00 [Clearing condition] • After reading OVF = 1, cleared by writing 0 to
4	—	1	—	Reserved This bit is always read as 1.
3	OS3	0	R/W	Output Select 3 and 2
2	OS2	0	R/W	These bits select an output method for the TMO the compare match of TCORB and TCNTV. 00: No change 01: 0 output 10: 1 output 11: Output toggles

OS3 and OS2 select the output level for compare match B. OS1 and OS0 select the outp
for compare match A. The two output levels can be controlled independently. After a re
timer output is 0 until the first compare match.

11.3.5 Timer Control Register V1 (TCRV1)

TCRV1 selects the edge at the TRGV pin, enables TRGV input, and selects the clock in
TCNTV.

Bit	Bit Name	Initial Value	R/W	Description
7 to 5	—	All 1	—	Reserved These bits are always read as 1.
4	TVEG1	0	R/W	TRGV Input Edge Select
3	TVEG0	0	R/W	These bits select the TRGV input edge. 00: TRGV trigger input is prohibited 01: Rising edge is selected 10: Falling edge is selected 11: Rising and falling edges are both selected
2	TRGE	0	R/W	TCNT starts counting up by the input of the edg selected by TVEG1 and TVEG0. 0: Disables starting counting-up TCNTV by the the TRGV pin and halting counting-up TCNT TCNTV is cleared by a compare match. 1: Enables starting counting-up TCNTV by the i the TRGV pin and halting counting-up TCNT TCNTV is cleared by a compare match.

11.4 Operation

11.4.1 Timer V Operation

1. According to table 11.2, six internal/external clock signals output by prescaler S can b selected as the timer V operating clock signals. When the operating clock signal is sel TCNTV starts counting-up. Figure 11.2 shows the count timing with an internal clock selected, and figure 11.3 shows the count timing with both edges of an external clock selected.
2. When TCNTV overflows (changes from H'FF to H'00), the overflow flag (OVF) in T will be set. The timing at this time is shown in figure 11.4. An interrupt request is sen CPU when OVIE in TCRV0 is 1.
3. TCNTV is constantly compared with TCORA and TCORB. Compare match flag A or (CMFA or CMFB) is set to 1 when TCNTV matches TCORA or TCORB, respectivel compare-match signal is generated in the last state in which the values match. Figure shows the timing. An interrupt request is generated for the CPU when CMIEA or CM TCRV0 is 1.
4. When a compare match A or B is generated, the TMOV responds with the output valu selected by bits OS3 to OS0 in TCSRV. Figure 11.6 shows the timing when the outpu toggled by compare match A.
5. When CCLR1 or CCLR0 in TCRV0 is 01 or 10, TCNTV can be cleared by the corres compare match. Figure 11.7 shows the timing.
6. When CCLR1 or CCLR0 in TCRV0 is 11, TCNTV can be cleared by the rising edge input of TMRIV pin. A TMRIV input pulse-width of at least 1.5 system clocks is nec Figure 11.8 shows the timing.
7. When a counter-clearing source is generated with TRGE in TCRV1 set to 1, the coun halted as soon as TCNTV is cleared. TCNTV resumes counting-up when the edge sel TVEG1 or TVEG0 in TCRV1 is input from the TGRV pin.

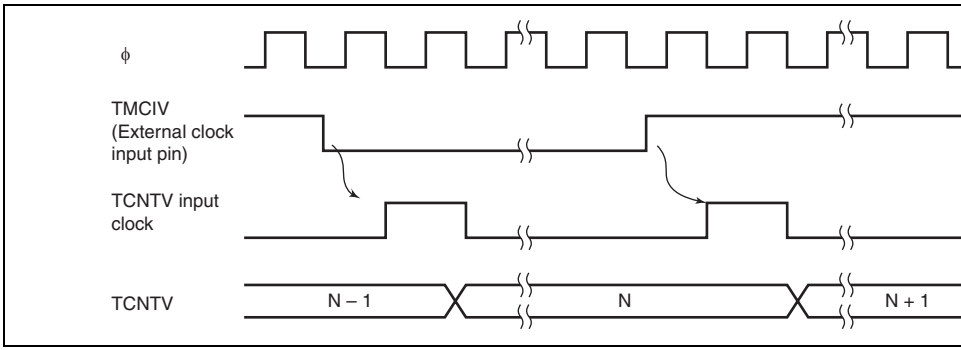


Figure 11.3 Increment Timing with External Clock

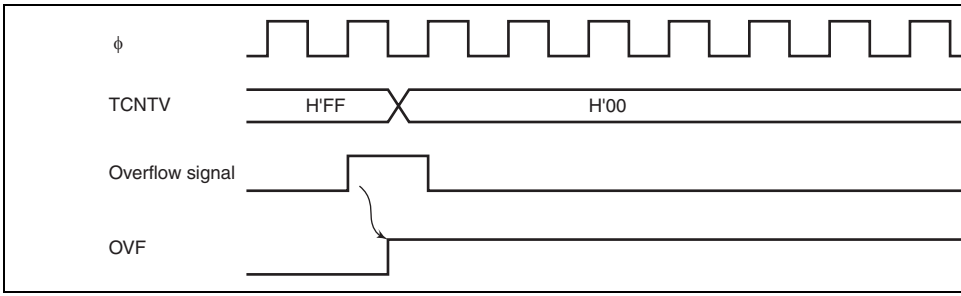


Figure 11.4 OVF Set Timing

Figure 11.5 CMFA and CMFB Set Timing

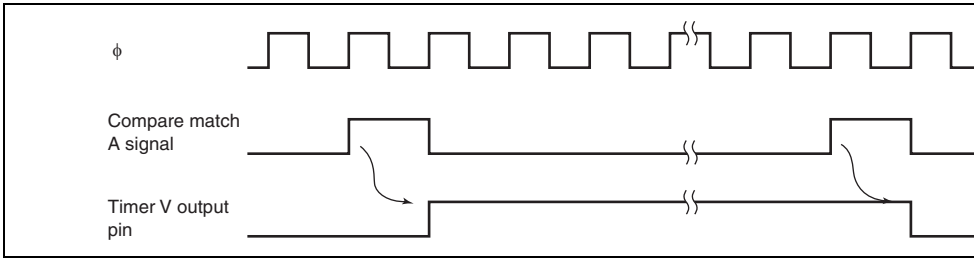


Figure 11.6 TMOV Output Timing

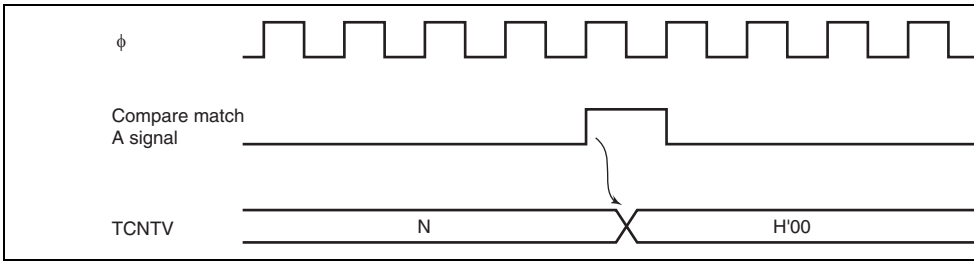


Figure 11.7 Clear Timing by Compare Match

11.5 Timer V Application Examples

11.5.1 Pulse Output with Arbitrary Duty Cycle

Figure 11.9 shows an example of output of pulses with an arbitrary duty cycle.

1. Set bits CCLR1 and CCLR0 in TCRV0 so that TCNTV will be cleared by compare m
   TCORA.
2. Set bits OS3 to OS0 in TCSRV so that the output will go to 1 at compare match with
   and to 0 at compare match with TCORB.
3. Set bits CKS2 to CKS0 in TCRV0 and bit ICKS0 in TCRV1 to select the desired clo
4. With these settings, a waveform is output without further software intervention, with
   determined by TCORA and a pulse width determined by TCORB.

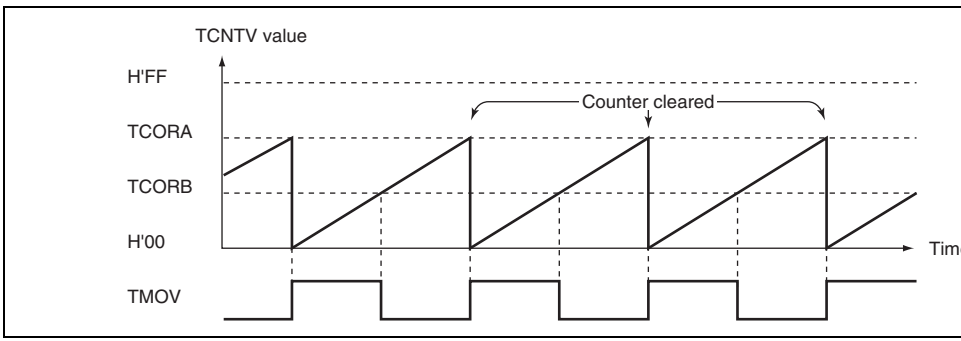


Figure 11.9 Pulse Output Example

4. Set bits CKS2 to CKS0 in TCRV0 and bit ICKS0 in TCRV1 to select the desired cloc
5. After these settings, a pulse waveform will be output without further software interve with a delay determined by TCORA from the TRGV input, and a pulse width determi (TCORB to TCORA).

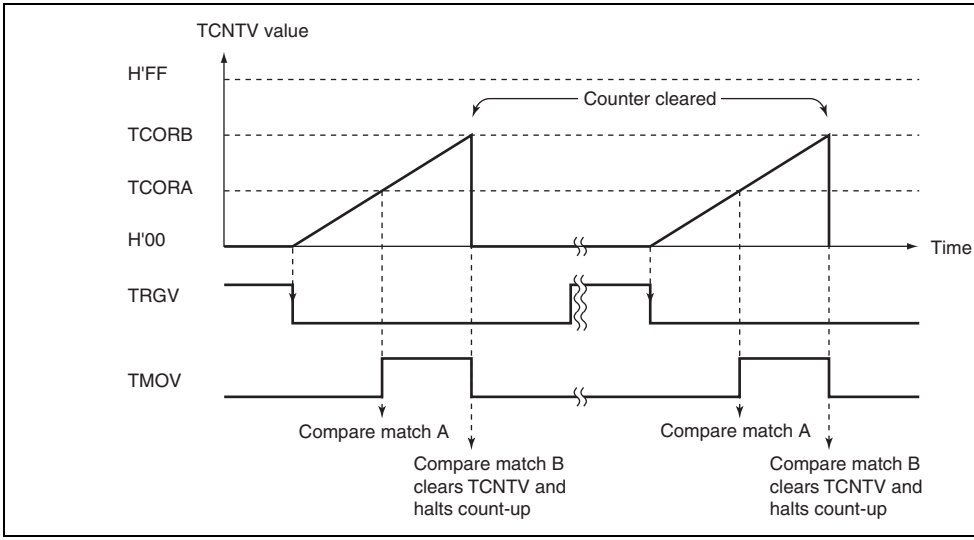


Figure 11.10 Example of Pulse Output Synchronized to TRGV Input

3. If compare matches A and B occur simultaneously, any conflict between the output for compare match A and compare match B is resolved by the following priority: to output > output 1 > output 0.
4. Depending on the timing, TCNTV may be incremented by a switch between differe clock sources. When TCNTV is internally clocked, an increment pulse is generated falling edge of an internal clock signal, that is divided system clock (ϕ). Therefore, in figure 11.13 the switch is from a high clock signal to a low clock signal, the swit seen as a falling edge, causing TCNTV to increment. TCNTV can also be incremen switch between internal and external clocks.

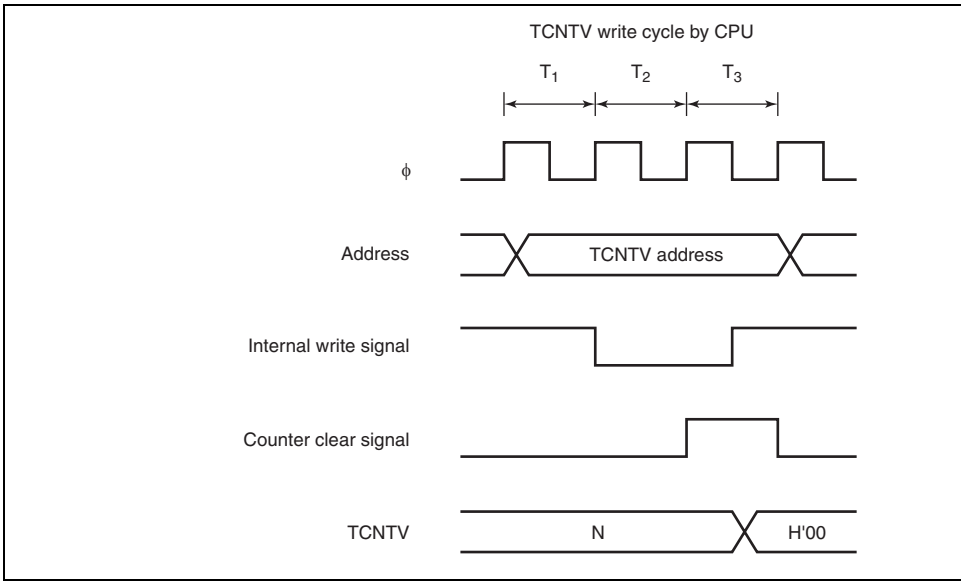


Figure 11.11 Contention between TCNTV Write and Clear

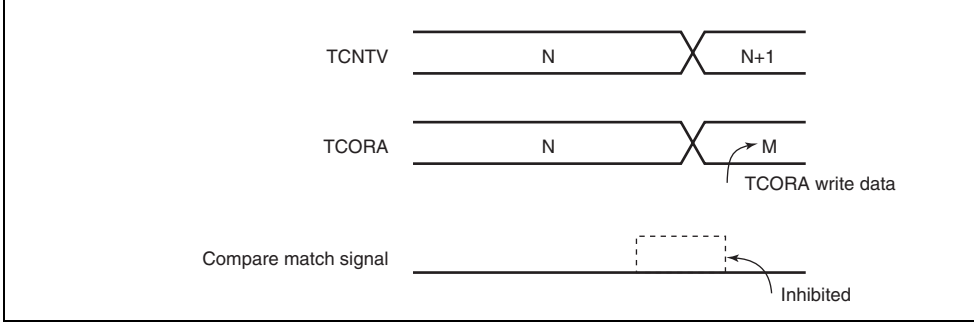


Figure 11.12 Contention between TCORA Write and Compare Match

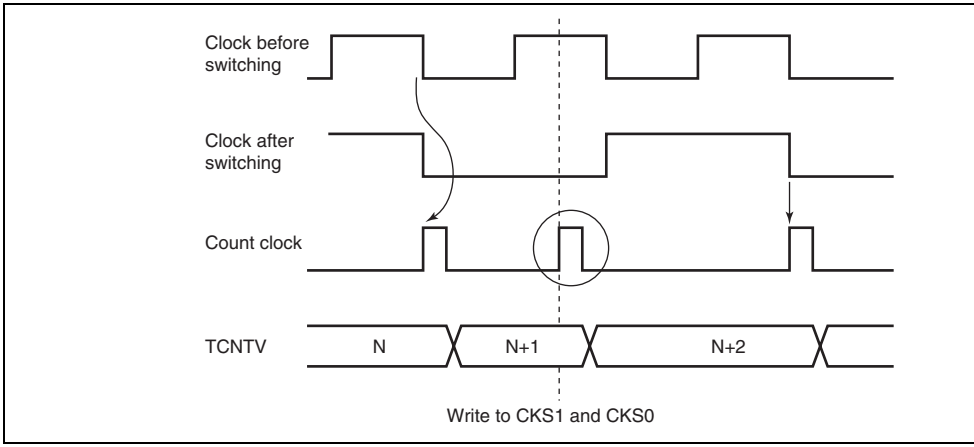
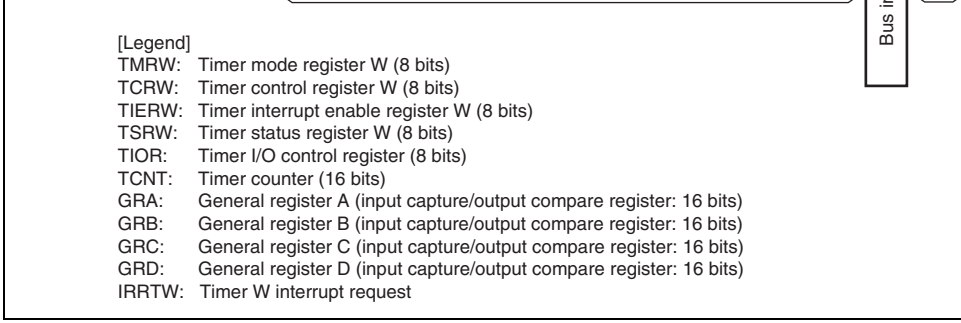


Figure 11.13 Internal Clock Switching and TCNTV Operation

- Capability to process up to four pulse outputs or four pulse inputs
- Four general registers:
  - Independently assignable output compare or input capture functions
  - Usable as two pairs of registers; one register of each pair operates as a buffer for compare or input capture register
- Four selectable operating modes:
  - Waveform output by compare match
    Selections of 0 output, 1 output, or toggle output
  - Input capture function
    Rising edge, falling edge, or both edges
  - Counter clearing function
    Counters can be cleared by compare match
  - PWM mode
    Up to three-phase PWM output can be provided with desired duty ratio.
- Any initial timer output value can be set
- Five interrupt sources
  Four compare match/input capture interrupts and an overflow interrupt.

capture registers)		GRA			GRA in buffer mode)	GRE mod
Counter clearing function		GRA compare match	GRA compare match	—	—	—
Initial output value setting function		—	Yes	Yes	Yes	Yes
Buffer function		—	Yes	Yes	—	—
Compare match output	0	—	Yes	Yes	Yes	Yes
	1	—	Yes	Yes	Yes	Yes
	Toggle	—	Yes	Yes	Yes	Yes
Input capture function		—	Yes	Yes	Yes	Yes
PWM mode		—	—	Yes	Yes	Yes
Interrupt sources		Overflow	Compare match/input capture	Compare match/input capture	Compare match/input capture	Com mat capt



[Legend]
TMRW: Timer mode register W (8 bits)
TCRW: Timer control register W (8 bits)
TIERW: Timer interrupt enable register W (8 bits)
TSRW: Timer status register W (8 bits)
TIOR: Timer I/O control register (8 bits)
TCNT: Timer counter (16 bits)
GRA: General register A (input capture/output compare register: 16 bits)
GRB: General register B (input capture/output compare register: 16 bits)
GRC: General register C (input capture/output compare register: 16 bits)
GRD: General register D (input capture/output compare register: 16 bits)
IRRTW: Timer W interrupt request

Figure 12.1 Timer W Block Diagram

compare B			input pin for GRB input captur PWM output pin in PWM mod
Input capture/output compare C	FTIOC	Input/output	Output pin for GRC output co input pin for GRC input captu PWM output pin in PWM mod
Input capture/output compare D	FTIOD	Input/output	Output pin for GRD output co input pin for GRD input captu PWM output pin in PWM mod

12.3 Register Descriptions

The timer W has the following registers.

- Timer mode register W (TMRW)
- Timer control register W (TCRW)
- Timer interrupt enable register W (TIERW)
- Timer status register W (TSRW)
- Timer I/O control register 0 (TIOR0)
- Timer I/O control register 1 (TIOR1)
- Timer counter (TCNT)
- General register A (GRA)
- General register B (GRB)
- General register C (GRC)
- General register D (GRD)

				This bit is always read as 1.
5	BUFEB	0	R/W	Buffer Operation B Selects the GRD function. 0: GRD operates as an input capture/output co register 1: GRD operates as the buffer register for GRB
4	BUFEA	0	R/W	Buffer Operation A Selects the GRC function. 0: GRC operates as an input capture/output co register 1: GRC operates as the buffer register for GRA
3	—	1	—	Reserved This bit is always read as 1.
2	PWMD	0	R/W	PWM Mode D Selects the output mode of the FTIOD pin. 0: FTIOD operates normally (output compare o 1: PWM output
1	PWMC	0	R/W	PWM Mode C Selects the output mode of the FTIOC pin. 0: FTIOC operates normally (output compare o 1: PWM output
0	PWMB	0	R/W	PWM Mode B Selects the output mode of the FTIOB pin. 0: FTIOB operates normally (output compare o 1: PWM output

5	CKS1	0	R/W	Select the TCNT clock source.
4	CKS0	0	R/W	000: Internal clock: counts on ϕ
				001: Internal clock: counts on $\phi/2$
				010: Internal clock: counts on $\phi/4$
				011: Internal clock: counts on $\phi/8$
				1XX: Counts on rising edges of the external ever
				When the internal clock source (ϕ) is selected, su sources are counted in subactive and subsleep m
3	TOD	0	R/W	Timer Output Level Setting D
				Sets the output value of the FTIOD pin until the f compare match D is generated.
				0: Output value is 0*
				1: Output value is 1*
2	TOC	0	R/W	Timer Output Level Setting C
				Sets the output value of the FTIOC pin until the f compare match C is generated.
				0: Output value is 0*
				1: Output value is 1*
1	TOB	0	R/W	Timer Output Level Setting B
				Sets the output value of the FTIOB pin until the f compare match B is generated.
				0: Output value is 0*
				1: Output value is 1*

12.3.3 Timer Interrupt Enable Register W (TIERW)

TIERW controls the timer W interrupt request.

Bit	Bit Name	Initial Value	R/W	Description
7	OVIE	0	R/W	Timer Overflow Interrupt Enable When this bit is set to 1, FOVI interrupt request flag in TSRW is enabled.
6 to 4	—	All 1	—	Reserved These bits are always read as 1.
3	IMIED	0	R/W	Input Capture/Compare Match Interrupt Enable When this bit is set to 1, IMID interrupt requeste IMFD flag in TSRW is enabled.
2	IMIEC	0	R/W	Input Capture/Compare Match Interrupt Enable When this bit is set to 1, IMIC interrupt requeste IMFC flag in TSRW is enabled.
1	IMIEB	0	R/W	Input Capture/Compare Match Interrupt Enable When this bit is set to 1, IMIB interrupt requeste IMFB flag in TSRW is enabled.
0	IMIEA	0	R/W	Input Capture/Compare Match Interrupt Enable When this bit is set to 1, IMIA interrupt requeste IMFA flag in TSRW is enabled.

				• Read OVF when OVF = 1, then write 0 in OV
6 to 4	—	All 1	—	Reserved These bits are always read as 1.
3	IMFD	0	R/W	Input Capture/Compare Match Flag D [Setting conditions] • TCNT = GRD when GRD functions as an out compare register • The TCNT value is transferred to GRD by an capture signal when GRD functions as an inp capture register [Clearing condition] • Read IMFD when IMFD = 1, then write 0 in IM
2	IMFC	0	R/W	Input Capture/Compare Match Flag C [Setting conditions] • TCNT = GRC when GRC functions as an out compare register • The TCNT value is transferred to GRC by an capture signal when GRC functions as an inp capture register [Clearing condition] • Read IMFC when IMFC = 1, then write 0 in IM

				• Read IMFB when IMFB = 1, then write 0 in
0	IMFA	0	R/W	Input Capture/Compare Match Flag A [Setting conditions] • TCNT = GRA when GRA functions as an ou compare register • The TCNT value is transferred to GRA by a capture signal when GRA functions as an in capture register [Clearing condition] • Read IMFA when IMFA = 1, then write 0 in

				0: GRB functions as an output compare register
				1: GRB functions as an input capture register
5	IOB1	0	R/W	I/O Control B1 and B0
4	IOB0	0	R/W	When IOB2 = 0,
				00: No output at compare match
				01: 0 output to the FTIOB pin at GRB compare m
				10: 1 output to the FTIOB pin at GRB compare m
				11: Output toggles to the FTIOB pin at GRB com match
				When IOB2 = 1,
				00: Input capture at rising edge at the FTIOB pin
				01: Input capture at falling edge at the FTIOB pir
				1X: Input capture at rising and falling edges of th pin
3	—	1	—	Reserved
				This bit is always read as 1.
2	IOA2	0	R/W	I/O Control A2
				Selects the GRA function.
				0: GRA functions as an output compare register
				1: GRA functions as an input capture register

00: Input capture at rising edge of the FTIOA pi

01: Input capture at falling edge of the FTIOA p

1X: Input capture at rising and falling edges of t pin

[Legend]

X: Don't care

12.3.6 Timer I/O Control Register 1 (TIOR1)

TIOR1 selects the functions of GRC and GRD, and specifies the functions of the FTIOC FTIOD pins.

Bit	Bit Name	Initial Value	R/W	Description
7	—	1	—	Reserved This bit is always read as 1.
6	IOD2	0	R/W	I/O Control D2 Selects the GRD function. 0: GRD functions as an output compare registe 1: GRD functions as an input capture register When GRB buffer operation has been selected BUFEB in TMRW, select the same function as

				00: Input capture at rising edge at the FTIOD pin
				01: Input capture at falling edge at the FTIOD pi
				1X: Input capture at rising and falling edges at th pin
3	—	1	—	Reserved
				This bit is always read as 1.
2	IOC2	0	R/W	I/O Control C2
				Selects the GRC function.
				0: GRC functions as an output compare register
				1: GRC functions as an input capture register
				When GRA buffer operation has been selected b BUFEA in TMRW, select the same function as G
1	IOC1	0	R/W	I/O Control C1 and C0
0	IOC0	0	R/W	When IOC2 = 0,
				00: No output at compare match
				01: 0 output to the FTIOC pin at GRC compare r
				10: 1 output to the FTIOC pin at GRC compare r
				11: Output toggles to the FTIOC pin at GRC com match
				When IOC2 = 1,
				00: Input capture to GRC at rising edge of the FT
				01: Input capture to GRC at falling edge of the F
				1X: Input capture to GRC at rising and falling edg the FTIOC pin

[Legend]

X: Don't care

Each general register is a 16-bit readable/writable register that can function as either an
compare register or an input-capture register. The function is selected by settings in TIO
TIOR1.

When a general register is used as an input-compare register, its value is constantly com
the TCNT value. When the two values match (a compare match), the corresponding flag
IMFB, IMFC, or IMFD) in TSRW is set to 1. An interrupt request is generated at this tir
IMIEA, IMIEB, IMIEC, or IMIED is set to 1. Compare match output can be selected in

When a general register is used as an input-capture register, an external input-capture sig
detected and the current TCNT value is stored in the general register. The corresponding
(IMFA, IMFB, IMFC, or IMFD) in TSRW is set to 1. If the corresponding interrupt-ena
(IMIEA, IMIEB, IMIEC, or IMIED) in TSRW is set to 1 at this time, an interrupt reques
generated. The edge of the input-capture signal is selected in TIOR.

GRC and GRD can be used as buffer registers of GRA and GRB, respectively, by settin
and BUFEB in TMRW.

For example, when GRA is set as an output-compare register and GRC is set as the buff
for GRA, the value in the buffer register GRC is sent to GRA whenever compare match
generated.

When GRA is set as an input-capture register and GRC is set as the buffer register for G
value in TCNT is transferred to GRA and the value in the buffer register GRC is transfe
GRA whenever an input capture is generated.

GRA to GRD must be written or read in 16-bit units; 8-bit access is not allowed. GRA to
initialized to H'FFFF by a reset.

When the count overflows from H'FFFF to H'0000, the OVF flag in TSRW is set to 1. If
in TIERW is set to 1, an interrupt request is generated. Figure 12.2 shows free-running co

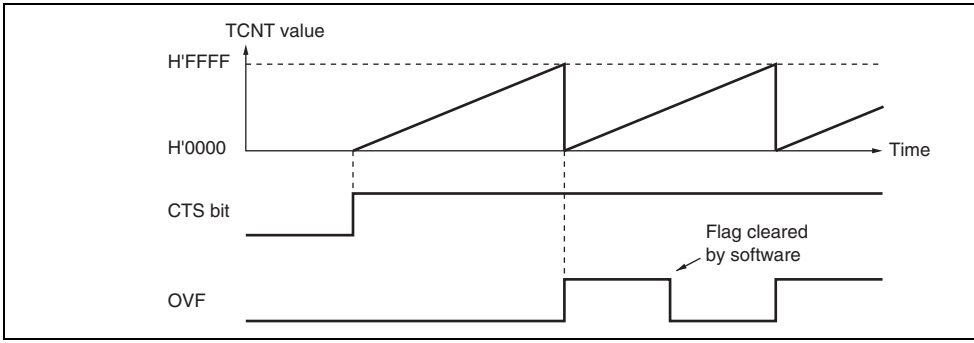


Figure 12.2 Free-Running Counter Operation

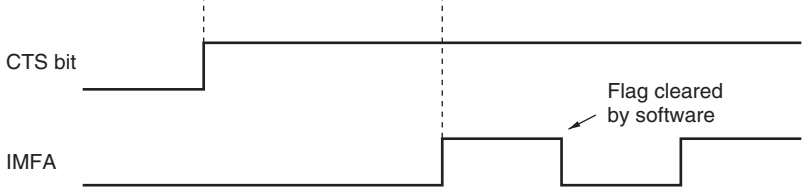


Figure 12.3 Periodic Counter Operation

By setting a general register as an output compare register, compare match A, B, C, or D
cause the output at the FTIOA, FTIOB, FTIOC, or FTIOD pin to output 0, output 1, or t
Figure 12.4 shows an example of 0 and 1 output when TCNT operates as a free-running
output is selected for compare match A, and 0 output is selected for compare match B. V
signal is already at the selected output level, the signal level does not change at compare

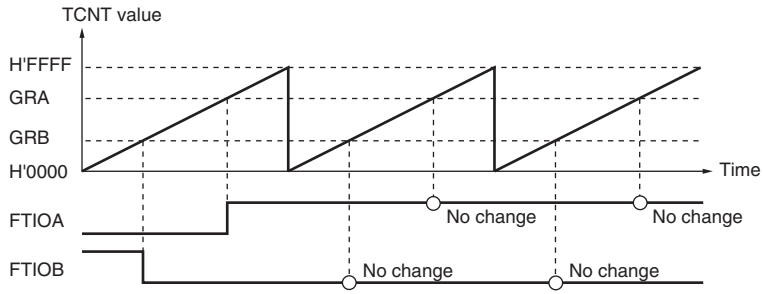


Figure 12.4 0 and 1 Output Example (TOA = 0, TOB = 1)

Figure 12.5 Toggle Output Example (TOA = 0, TOB = 1)

Figure 12.6 shows another example of toggle output when TCNT operates as a periodic c
cleared by compare match A. Toggle output is selected for both compare match A and B.

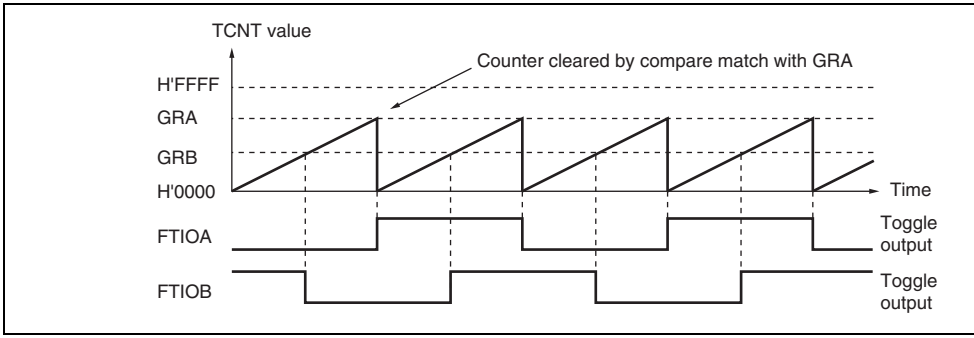


Figure 12.6 Toggle Output Example (TOA = 0, TOB = 1)

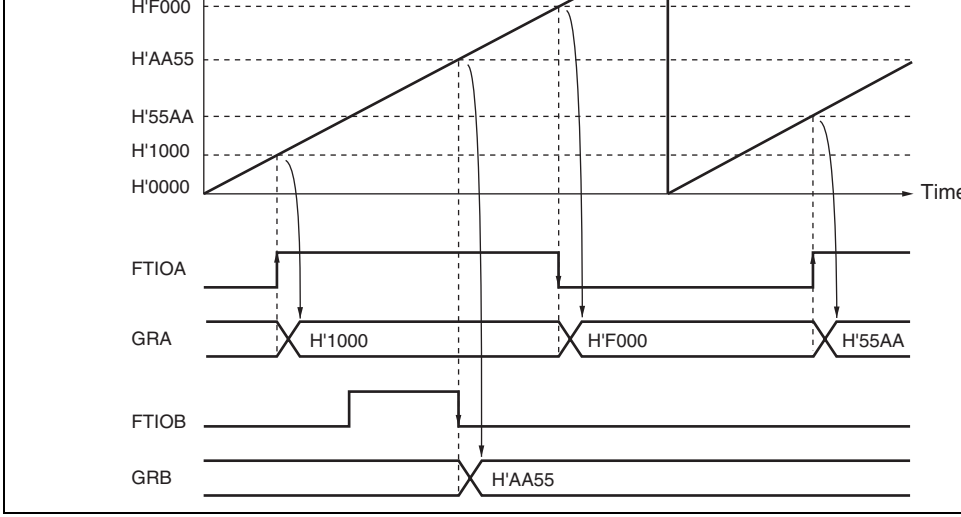


Figure 12.7 Input Capture Operating Example

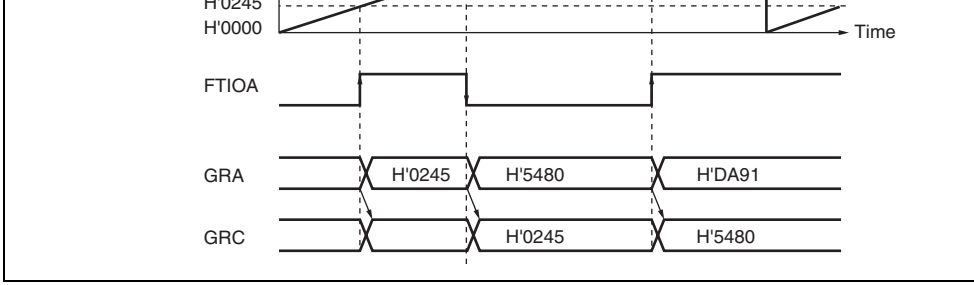


Figure 12.8 Buffer Operation Example (Input Capture)

12.4.2 PWM Operation

In PWM mode, PWM waveforms are generated by using GRA as the period register and GRC, and GRD as duty registers. PWM waveforms are output from the FTIOB, FTIOC, FTIOD pins. Up to three-phase PWM waveforms can be output. In PWM mode, a genera functions as an output compare register automatically. The output level of each pin depen corresponding timer output level set bit (TOB, TOC, and TOD) in TCRW. When TOB is FTIOB output goes to 1 at compare match A and to 0 at compare match B. When TOB is FTIOB output goes to 0 at compare match A and to 1 at compare match B. Thus the comp match output level settings in TIOR0 and TIOR1 are ignored for the output pin set to PW If the same value is set in the cycle register and the duty register, the output does not char a compare match occurs.

Figure 12.9 shows an example of operation in PWM mode. The output signals go to 1 an is cleared at compare match A, and the output signals go to 0 at compare match B, C, and (TOB, TOC, and TOD = 1: initial output values are set to 1).

Figure 12.9 PWM Mode Example (1)

Figure 12.10 shows another example of operation in PWM mode. The output signals go
TCNT is cleared at compare match A, and the output signals go to 1 at compare match B
D (TOB, TOC, and TOD = 0: initial output values are set to 1).

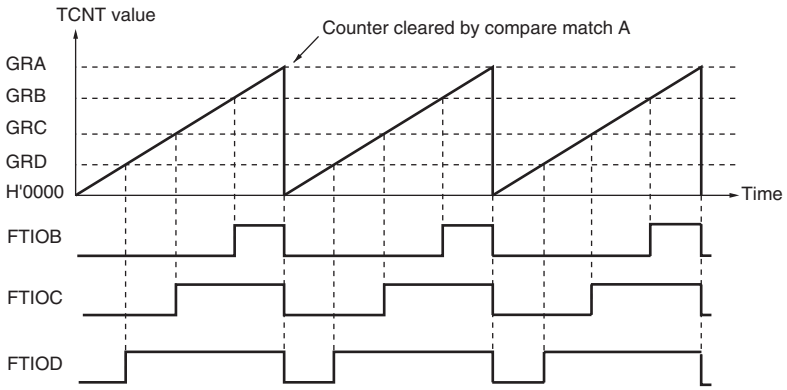


Figure 12.10 PWM Mode Example (2)

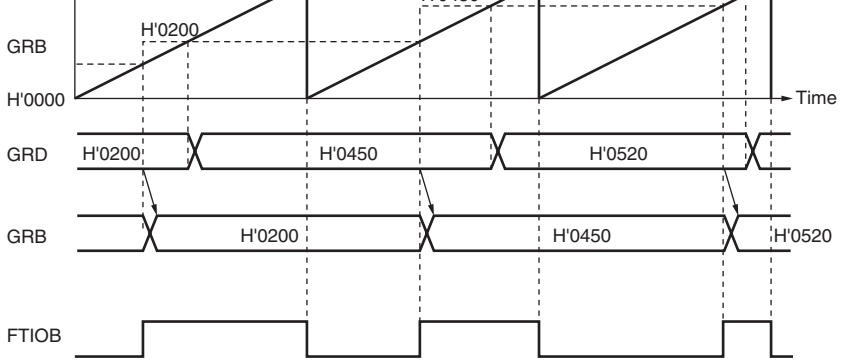


Figure 12.11 Buffer Operation Example (Output Compare)

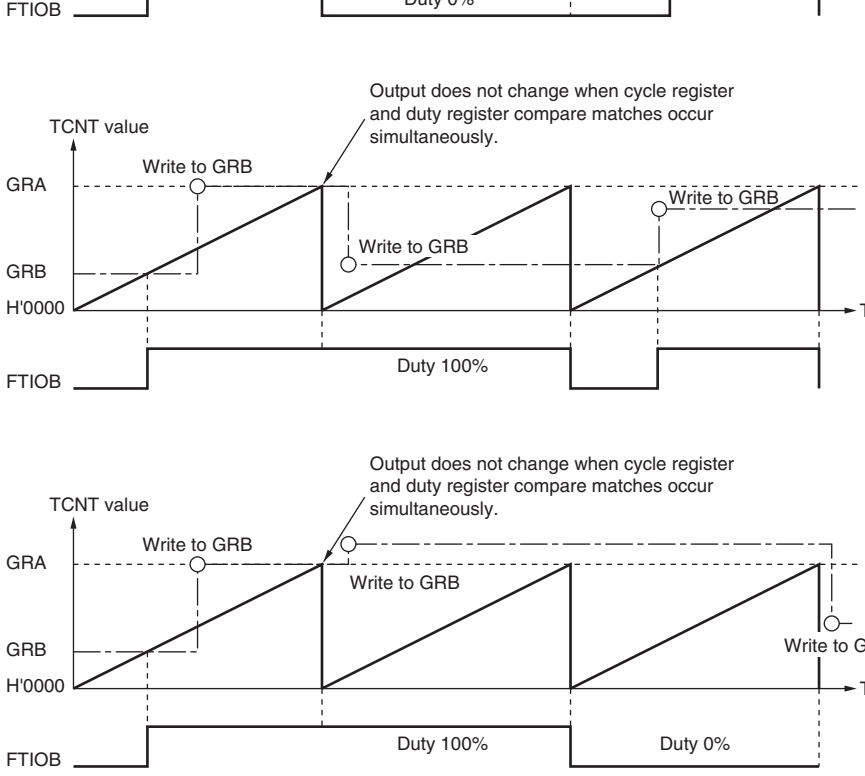


Figure 12.12 PWM Mode Example
(TOB, TOC, and TOD = 0: Initial Output Values are Set to 0)

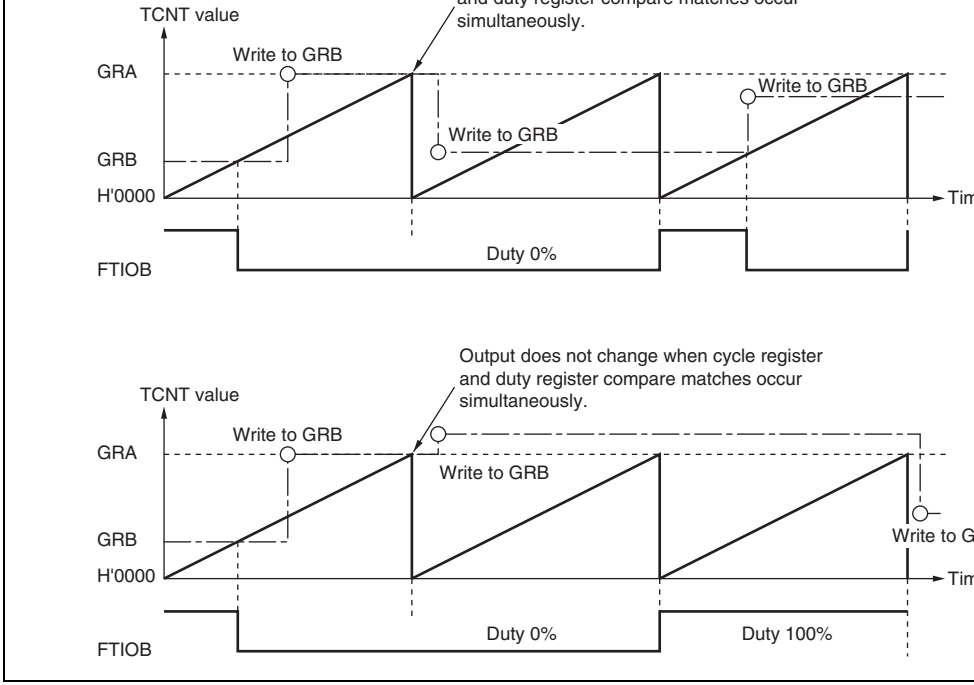


Figure 12.13 PWM Mode Example
(TOB, TOC, and TOD = 1: Initial Output Values are Set to 1)

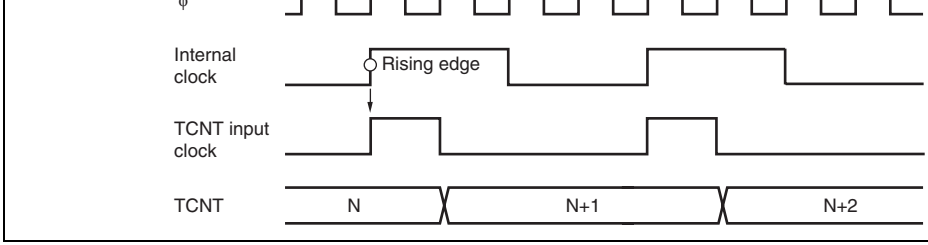


Figure 12.14 Count Timing for Internal Clock Source

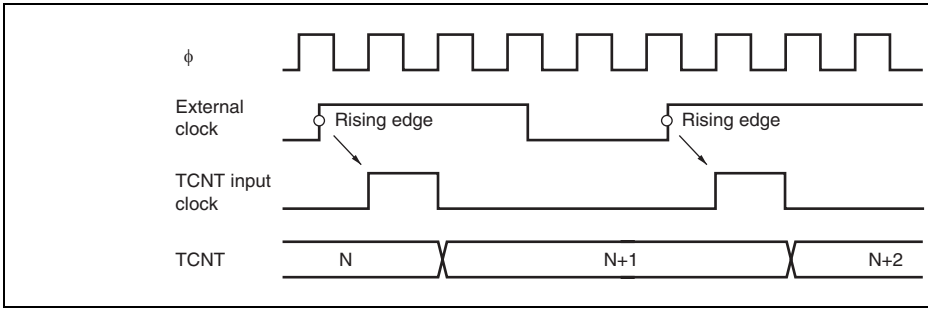


Figure 12.15 Count Timing for External Clock Source

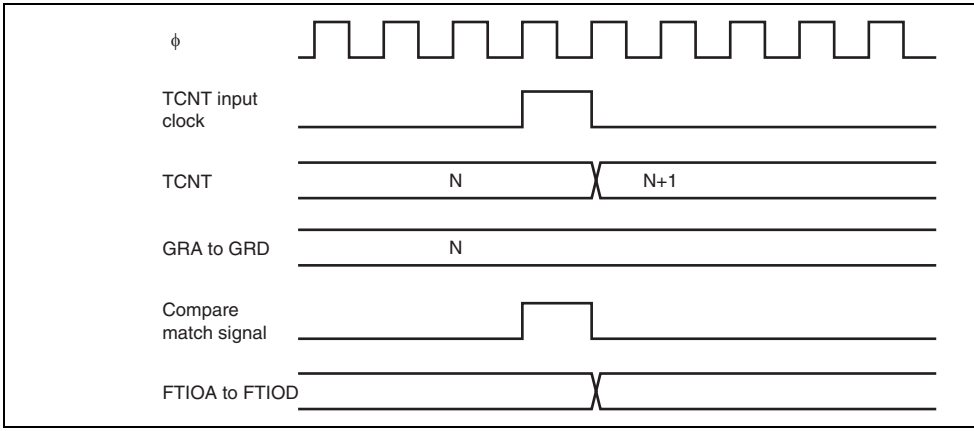


Figure 12.16 Output Compare Output Timing

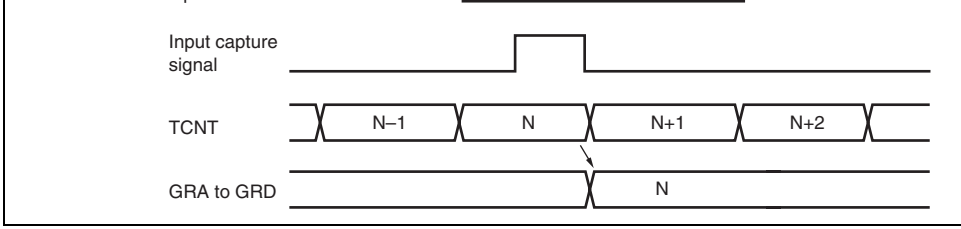


Figure 12.17 Input Capture Input Signal Timing

12.5.4 Timing of Counter Clearing by Compare Match

Figure 12.18 shows the timing when the counter is cleared by compare match A. When t
value is N, the counter counts from 0 to N, and its cycle is N + 1.

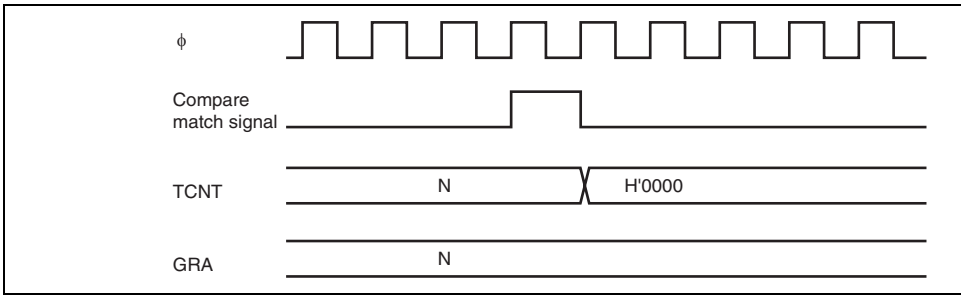


Figure 12.18 Timing of Counter Clearing by Compare Match

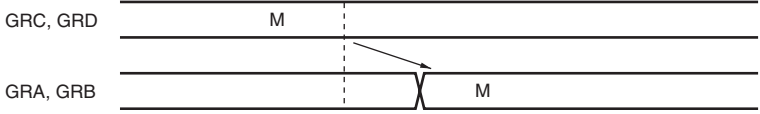


Figure 12.19 Buffer Operation Timing (Compare Match)

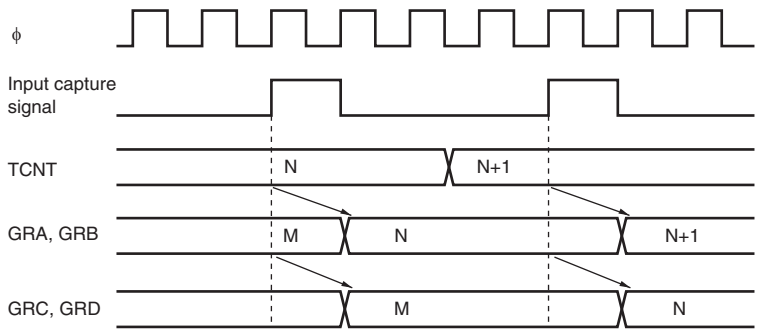


Figure 12.20 Buffer Operation Timing (Input Capture)

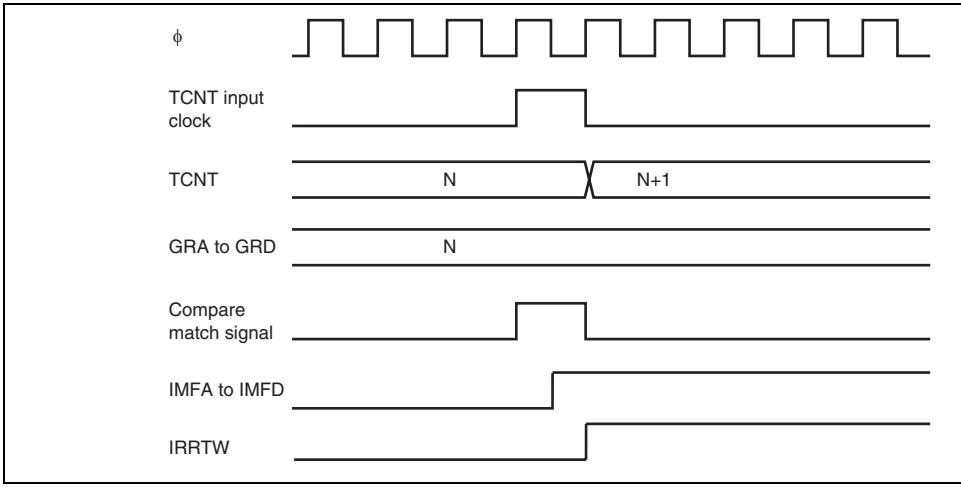


Figure 12.21 Timing of IMFA to IMFD Flag Setting at Compare Match

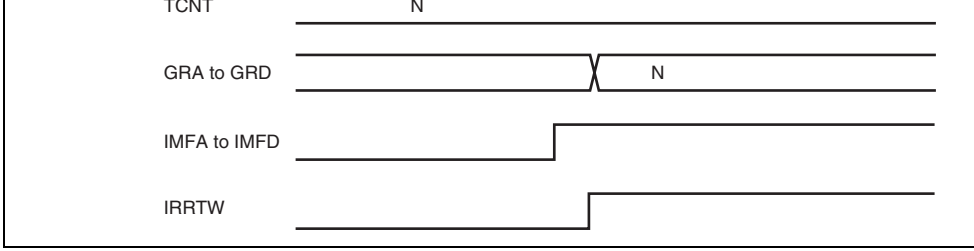


Figure 12.22 Timing of IMFA to IMFD Flag Setting at Input Capture

12.5.8 Timing of Status Flag Clearing

When the CPU reads a status flag while it is set to 1, then writes 0 in the status flag, the s
is cleared. Figure 12.23 shows the status flag clearing timing.

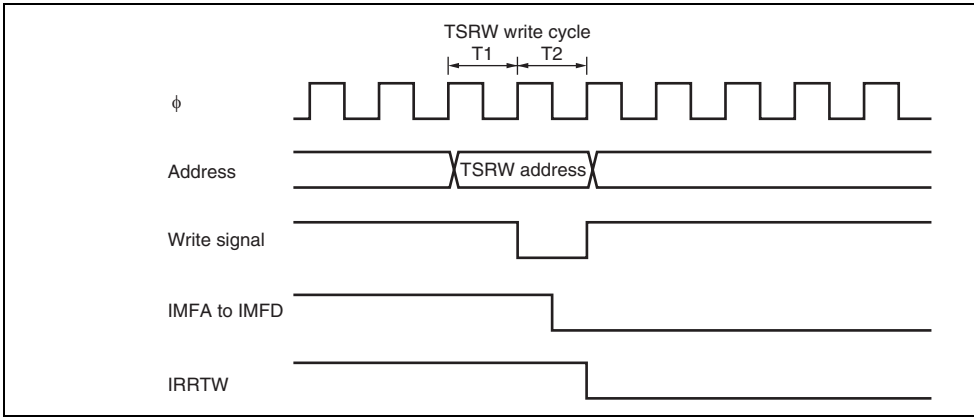


Figure 12.23 Timing of Status Flag Clearing by CPU

3. Depending on the timing, TCNT may be incremented by a switch between different clock sources. When TCNT is internally clocked, an increment pulse is generated fr rising edge of an internal clock signal, that is divided system clock (ϕ). Therefore, as figure 12.25 the switch is from a low clock signal to a high clock signal, the switcho as a rising edge, causing TCNT to increment.
4. If timer W enters module standby mode while an interrupt request is generated, the i request cannot be cleared. Before entering module standby mode, disable interrupt re
5. The TOA to TOD bits in TCRW decide the value of the FTIO pin, which is output u first compare match occurs. Once a compare match occurs and this compare match c values of FTIOA to FTIOD output, the values of the FTIOA to FTIOD pin output an values read from the TOA to TOD bits may differ. Moreover, when the writing to T the generation of the compare match A to D occur at the same timing, the writing to has the priority. Thus, output change due to the compare match is not reflected to the to FTIOD pins. Therefore, when bit manipulation instruction is used to write to TCR values of the FTIOA to FTIOD pin output may result in an unexpected result. When to be written to while compare match is operating, stop the counter once before acce TCRW, read the port 8 state to reflect the values of FTIOA to FTIOD output, to TOA and then restart the counter. Figure 12.26 shows an example when the compare matc bit manipulation instruction to TCRW occur at the same timing.

Figure 12.24 Contention between TCNT Write and Clear

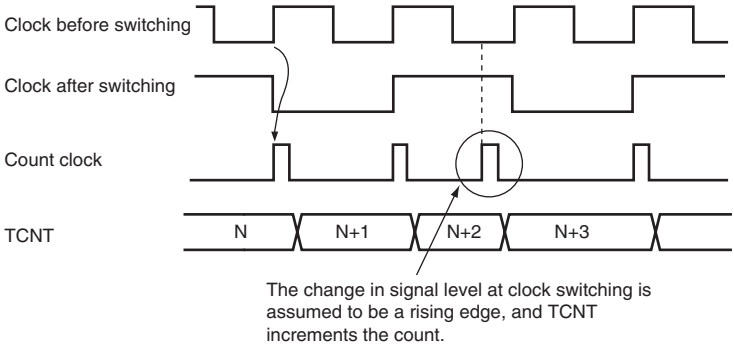


Figure 12.25 Internal Clock Switching and TCNT Operation

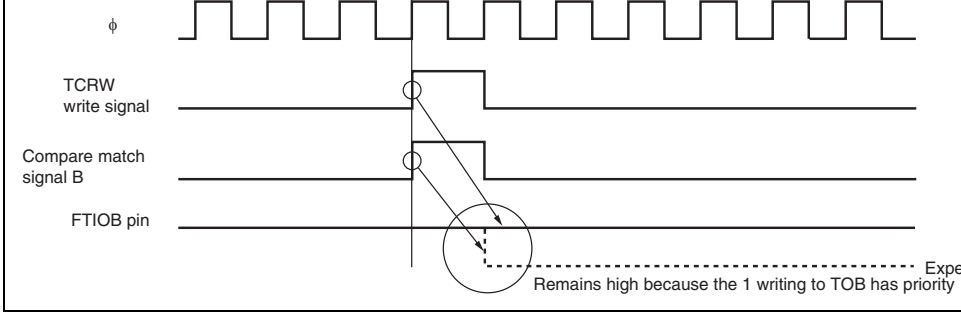


**Figure 12.26 When Compare Match and Bit Manipulation Instruction to TC
Occur at the Same Timing**

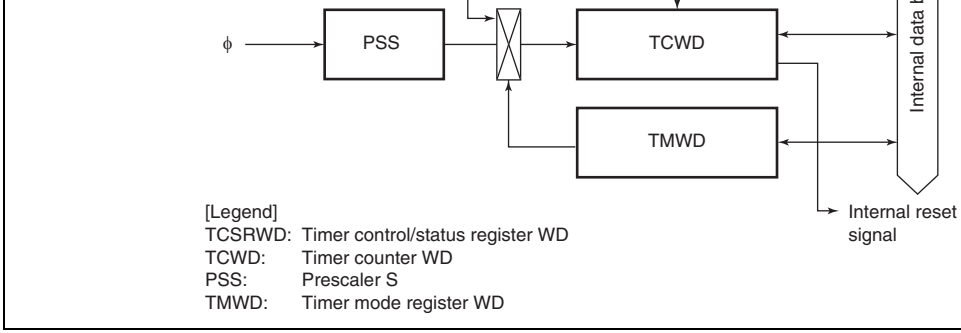


Figure 13.1 Block Diagram of Watchdog Timer

13.1 Features

- Selectable from nine counter input clocks.

  Eight clock sources (ϕ/64, ϕ/128, ϕ/256, ϕ/512, ϕ/1024, ϕ/2048, ϕ/4096, and ϕ/8192) WDT dedicated internal oscillator can be selected as the timer-counter clock. When dedicated internal oscillator is selected, it can operate as the watchdog timer in any c mode.
- Reset signal generated on counter overflow

  An overflow period of 1 to 256 times the selected clock can be set.
- The watchdog timer is enabled in the initial state.

  It starts operating after the reset state is canceled.

TCSRWD performs the TCSRWD and TCWD write control. TCSRWD also controls the
watchdog timer operation and indicates the operating state. TCSRWD must be rewritten
the MOV instruction. The bit manipulation instruction cannot be used to change the setti

Bit	Bit Name	Initial Value	R/W	Description
7	B6WI	1	R/W	Bit 6 Write Inhibit The TCWE bit can be written only when the write the B6WI bit is 0. This bit is always read as 1.
6	TCWE	0	R/W	Timer Counter WD Write Enable TCWD can be written when the TCWE bit is set When writing data to this bit, the value for bit 7 m
5	B4WI	1	R/W	Bit 4 Write Inhibit The TCSRWE bit can be written only when the w value of the B4WI bit is 0. This bit is always read
4	TCSRWE	0	R/W	Timer Control/Status Register WD Write Enable The WDON and WRST bits can be written when TCSRWE bit is set to 1. When writing data to this bit, the value for bit 5 m
3	B2WI	1	R/W	Bit 2 Write Inhibit This bit can be written to the WDON bit only whe write value of the B2WI bit is 0. This bit is always read as 1.

				[Clearing conditions] • When 0 is written to the WDON bit and 0 is the B2WI bit while the TCSRWE bit = 1
1	B0WI	1	R/W	Bit 0 Write Inhibit This bit can be written to the WRST bit only wh write value of the B0WI bit is 0. This bit is alway 1.
0	WRST*	0	R/W	Watchdog Timer Reset [Setting condition] • When TCWD overflows and an internal rese generated [Clearing conditions] • Reset by the $\overline{\text{RES}}$ pin • When 0 is written to the WRST bit and 0 is the B0WI bit while the TCSRWE bit = 1

Note: * The WRST bit cannot be modified to 1.

Bit	Bit Name	Initial Value	R/W	Description
7 to 4	—	All 1	—	Reserved These bits are always read as 1.
3	CKS3	1	R/W	Clock Select 3 to 0
2	CKS2	1	R/W	Select the clock to be input to TCWD.
1	CKS1	1	R/W	1000: Internal clock: counts on ϕ/64
0	CKS0	1	R/W	1001: Internal clock: counts on ϕ/128 1010: Internal clock: counts on ϕ/256 1011: Internal clock: counts on ϕ/512 1100: Internal clock: counts on ϕ/1024 1101: Internal clock: counts on ϕ/2048 1110: Internal clock: counts on ϕ/4096 1111: Internal clock: counts on ϕ8192 0XXX: WDT dedicated internal oscillator For the overflow periods of the WDT dedicated i oscillator, see section 20, Electrical Characteristi

[Legend]

X: Don't care

Figure 13.2 shows an example of watchdog timer operation.

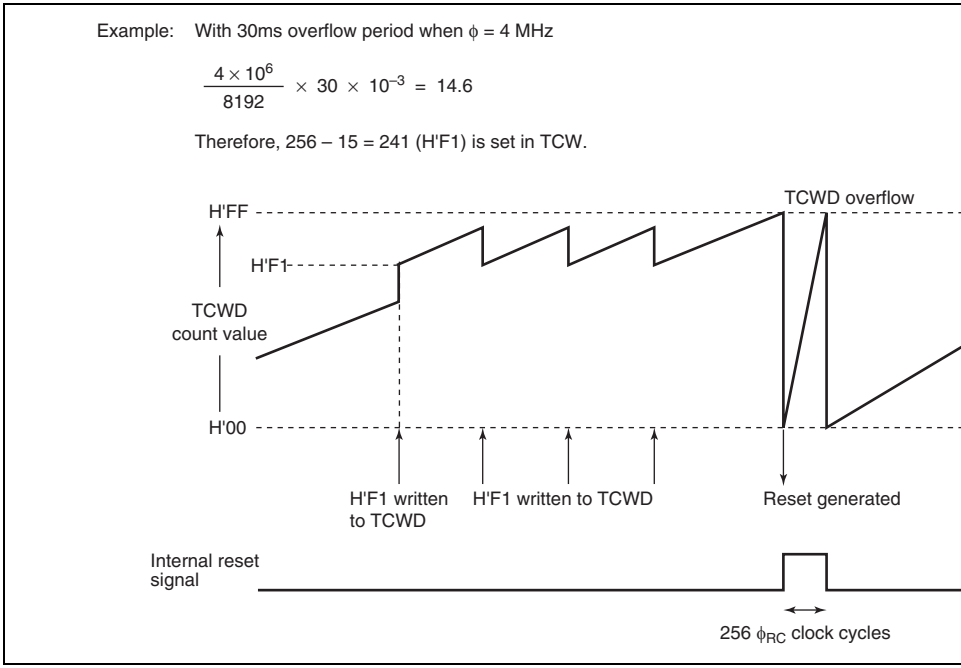


Figure 13.2 Watchdog Timer Operation Example

14.1 Features

- Choice of asynchronous or clocked synchronous serial communication mode
- Full-duplex communication capability

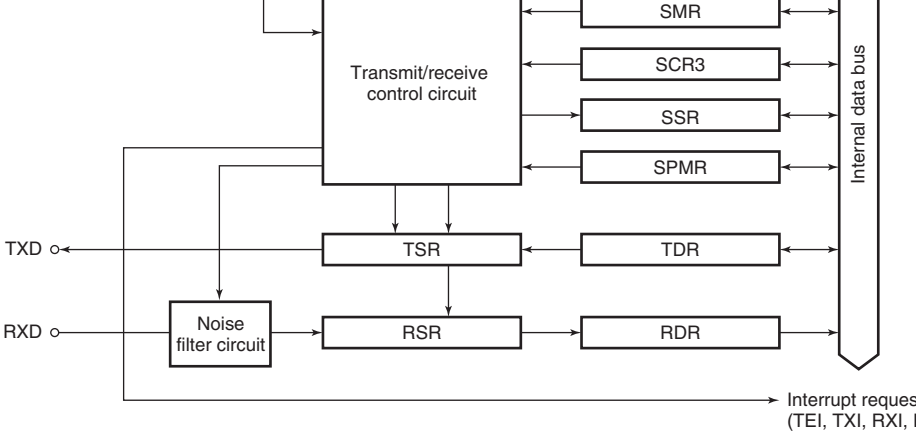
  The transmitter and receiver are mutually independent, enabling transmission and re
  be executed simultaneously.

  Double-buffering is used in both the transmitter and the receiver, enabling continuou
  transmission and continuous reception of serial data.
- On-chip baud rate generator allows any bit rate to be selected
- External clock or on-chip baud rate generator can be selected as a transfer clock sou
- Six interrupt sources

  Transmit-end, transmit-data-empty, receive-data-full, overrun error, framing error, a
  error.
- Internal noise filter circuit (available for asynchronous serial communication only)

Asynchronous mode

- Data length: 7 or 8 bits
- Stop bit length: 1 or 2 bits
- Parity: Even, odd, or none
- Receive error detection: Parity, overrun, and framing errors
- Break detection: Break can be detected by reading the RXD pin level directly in the
  framing error



[Legend]

RSR:	Receive shift register
RDR:	Receive data register
TSR:	Transmit shift register
TDR:	Transmit data register
SMR:	Serial mode register
SCR3:	Serial control register 3
SSR:	Serial status register
BRR:	Bit rate register
BRC:	Bit rate counter
SPMR:	Sampling mode register

Figure 14.1 Block Diagram of SCI3

14.3 Register Descriptions

SCI3 has the following registers for each channel.

- Receive shift register (RSR)
- Receive data register (RDR)
- Transmit shift register (TSR)
- Transmit data register (TDR)
- Serial mode register (SMR)
- Serial control register 3 (SCR3)
- Serial status register (SSR)
- Bit rate register (BRR)
- Sampling mode register (SPMR)

receive-enabled. As RSR and RDR function as a double buffer in this way, continuous re
operations are possible. After confirming that the RDRF bit in SSR is set to 1, read RDR
once. RDR cannot be written to by the CPU. RDR is initialized to H'00.

14.3.3 Transmit Shift Register (TSR)

TSR is a shift register that transmits serial data. To perform serial data transmission, SCI
transfers transmit data from TDR to TSR automatically, then sends the data that starts fro
LSB to the TXD pin. TSR cannot be directly accessed by the CPU.

14.3.4 Transmit Data Register (TDR)

TDR is an 8-bit register that stores data for transmission. When SCI3 detects that TSR is
transfers the transmit data written in TDR to TSR and starts transmission. The double-bu
structure of TDR and TSR enables continuous serial transmission. If the next transmit dat
already been written to TDR during transmission of one-frame data, SCI3 transfers the w
data to TSR to continue transmission. To achieve reliable serial transmission, write transm
to TDR only once after confirming that the TDRE bit in SSR is set to 1. TDR is initialize
H'FF.

6	CHR	0	R/W	Character Length (enabled only in asynchronou 0: Selects 8 bits as the data length. 1: Selects 7 bits as the data length.
5	PE	0	R/W	Parity Enable (enabled only in asynchronous m When this bit is set to 1, the parity bit is added t data before transmission, and the parity bit is ch reception.
4	PM	0	R/W	Parity Mode (enabled only when the PE bit is 1 asynchronous mode) 0: Selects even parity. 1: Selects odd parity.
3	STOP	0	R/W	Stop Bit Length (enabled only in asynchronous Selects the stop bit length in transmission. 0: 1 stop bit 1: 2 stop bits For reception, only the first stop bit is checked, of the value in the bit. If the second stop bit is 0 treated as the start bit of the next transmit chara
2	MP	0	R/W	Multiprocessor Mode When this bit is set to 1, the multiprocessor communication function is enabled. The PE bit bit settings are invalid in multiprocessor mode. synchronous mode, clear this bit to 0.

and the baud rate, see section 14.3.8, Bit Rate R
(BRR). n is the decimal representation of the val
BRR (see section 14.3.8, Bit Rate Register (BRR

14.3.6 Serial Control Register 3 (SCR3)

SCR3 is a register that enables or disables SCI3 transfer operations and interrupt requests
also used to select the transfer clock source. For details on interrupt requests, refer to sect
Interrupts.

Bit	Bit Name	Initial Value	R/W	Description
7	TIE	0	R/W	Transmit Interrupt Enable When this bit is set to 1, the TXI interrupt reques enabled.
6	RIE	0	R/W	Receive Interrupt Enable When this bit is set to 1, RXI and ERI interrupt re are enabled.
5	TE	0	R/W	Transmit Enable When this bit s set to 1, transmission is enabled.
4	RE	0	R/W	Receive Enable When this bit is set to 1, reception is enabled.

2	TEIE	0	R/W	Transmit End Interrupt Enable
				When this bit is set to 1, TEI interrupt request is
1	CKE1	0	R/W	Clock Enable 0 and 1
0	CKE0	0	R/W	Selects the clock source. • Asynchronous mode 00: On-chip baud rate generator 01: On-chip baud rate generator Outputs a clock of the same frequency as th from the SCK3 pin. 10: External clock Inputs a clock with a frequency 16 times the from the SCK3 pin. 11:Reserved • Clocked synchronous mode 00: On-chip clock (SCK3 pin functions as clock 01:Reserved 10: External clock (SCK3 pin functions as clock 11:Reserved

				• When the TE bit in SCR3 is 0 • When data is transferred from TDR to TSR [Clearing conditions] • When 0 is written to TDRE after reading TDR • When the transmit data is written to TDR
6	RDRF	0	R/W	Receive Data Register Full Indicates that the received data is stored in RDR [Setting condition] • When serial reception ends normally and rec is transferred from RSR to RDR [Clearing conditions] • When 0 is written to RDRF after reading RDR • When data is read from RDR
5	OER	0	R/W	Overrun Error [Setting condition] • When an overrun error occurs in reception [Clearing condition] • When 0 is written to OER after reading OER
4	FER	0	R/W	Framing Error [Setting condition] • When a framing error occurs in reception [Clearing condition] • When 0 is written to FER after reading FER =

				• When TDRE = 1 at transmission of the last frame serial transmit character [Clearing conditions] • When 0 is written to TDRE after reading TD • When the transmit data is written to TDR
1	MPBR	0	R	Multiprocessor Bit Receive MPBR stores the multiprocessor bit in the recei character data. When the RE bit in SCR3 is cle its state is retained.
0	MPBT	0	R/W	Multiprocessor Bit Transfer MPBT stores the multiprocessor bit to be added transmit character data.

[Asynchronous Mode]

$$N = \frac{\phi}{64 \times 2^{2n-1} \times B} \times 10^6 - 1$$

$$\text{Error (\%)} = \left\{ \frac{\phi \times 10^6}{(N + 1) \times B \times 64 \times 2^{2n-1}} - 1 \right\} \times 100$$

[Clocked Synchronous Mode]

$$N = \frac{\phi}{8 \times 2^{2n-1} \times B} \times 10^6 - 1$$

Legend B: Bit rate (bit/s)

N: BRR setting for baud rate generator ($0 \leq N \leq 255$)

ϕ: Operating frequency (MHz)

n: CSK1 and CSK0 settings in SMR ($0 \leq n \leq 3$)

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	–6.99	0	6	–2.48	0	7	0.00	0	9
19200	0	2	8.51	0	2	13.78	0	3	0.00	0	4
31250	0	1	0.00	0	1	4.86	0	1	22.88	0	2
38400	0	1	–18.62	0	1	–14.67	0	1	0.00	—	—

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	6	–6.99	0	7	0.00	0	7
31250	—	—	—	0	3	0.00	0	4	–1.70	0	4
38400	0	2	0.00	0	2	8.51	0	3	0.00	0	3

[Legend]

—: A setting is available but error occurs

	Operating Frequency ϕ (MHz)							
	6			6.144			7.3728	
Bit Rate (bit/s)	n	N	Error (%)	n	N	Error (%)	n	N
110	2	106	–0.44	2	108	0.08	2	130
150	2	77	0.16	2	79	0.00	2	95
300	1	155	0.16	1	159	0.00	1	191
600	1	77	0.16	1	79	0.00	1	95
1200	0	155	0.16	0	159	0.00	0	191
2400	0	77	0.16	0	79	0.00	0	95
4800	0	38	0.16	0	39	0.00	0	47
9600	0	19	–2.34	0	19	0.00	0	23
19200	0	9	–2.34	0	9	0.00	0	11
31250	0	5	0.00	0	5	2.40	0	6
38400	0	4	–2.34	0	4	0.00	0	5

4800	0	51	0.16	0	63	0.00	0	64	0
9600	0	25	0.16	0	31	0.00	0	32	-
19200	0	12	0.16	0	15	0.00	0	15	1
31250	0	7	0.00	0	9	–1.70	0	9	0
38400	0	6	-6.99	0	7	0.00	0	7	1

Table 14.3 Maximum Bit Rate for Each Frequency (Asynchronous Mode)

ϕ (MHz)	Maximum Bit Rate (bit/s)	n	N	ϕ (MHz)	Maximum Bit Rate (bit/s)	n
2	62500	0	0	5	156250	0
2.097152	65536	0	0	6	187500	0
2.4576	76800	0	0	6.144	192000	0
3	93750	0	0	7.3728	230400	0
3.6864	115200	0	0	8	250000	0
4	125000	0	0	9.8304	307200	0
4.9152	153600	0	0	10	312500	0

2.5k	0	199	1	99	1	199	1	
5k	0	99	0	199	1	99	1	
10k	0	49	0	99	0	199	0	
25k	0	19	0	39	0	79	0	
50k	0	9	0	19	0	39	0	
100k	0	4	0	9	0	19	0	
250k	0	1	0	3	0	7	0	
500k	0	0*	0	1	0	3	0	
1M			0	0*	0	1	—	
2M					0	0*	—	
2.5M							0	
4M								

[Legend]

Blank: No setting is available.

—: A setting is available but error occurs.

*: Continuous transfer is not possible.

				asynchronous mode. 0: Noise filter circuit is enabled 1: Noise filter circuit is disabled
1, 0	—	All 1	—	Reserved These bits are always read as 1.

- Noise Filter Circuit

  The RXD input signal is latched through the noise filter circuit. The noise filter circu comprises a series of three latch circuits and a match detection circuit. The RXD inp sampled by the basic clock with the 16 times the transfer clock frequency. If three la outputs match, its level is transferred to the next stage. If not, the circuit holds the pr value.

  That is, when the incoming signal holds the same level for three clock cycles, it is re the proper signal. If the levels of the signal is less than three clock cycles, the signal regarded as a noise.

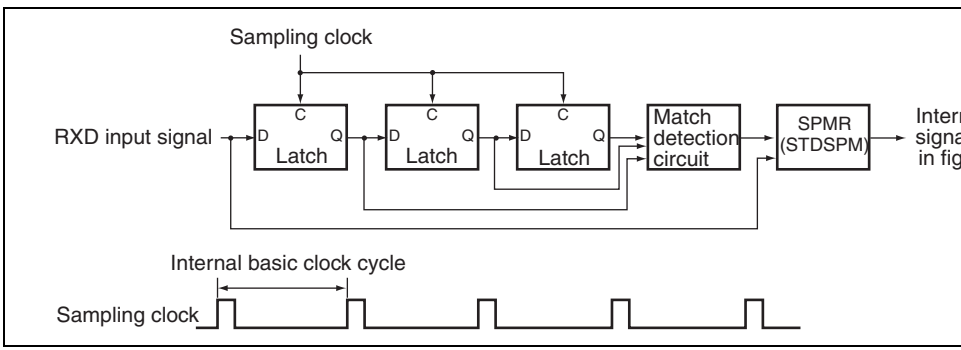


Figure 14.2 Block Diagram of Noise Filter Circuit

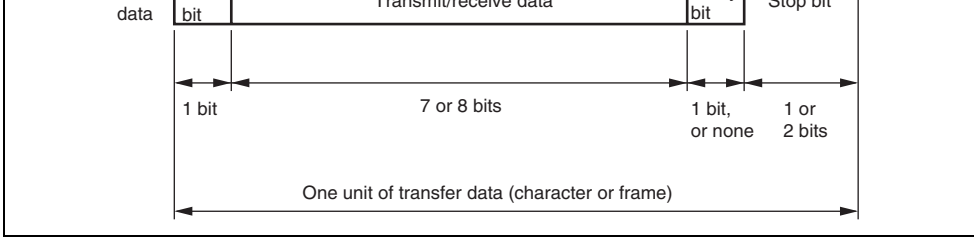


Figure 14.3 Data Format in Asynchronous Communication

14.4.1 Clock

Either an internal clock generated by the on-chip baud rate generator or an external clock
the SCK3 pin can be selected as the SCI3's serial clock, according to the setting of the C
SMR and the CKE0 and CKE1 bits in SCR3. When an external clock is input at the SCK
clock frequency should be 16 times the bit rate used.

When SCI3 is operated on an internal clock, the clock can be output from the SCK3 pin.
frequency of the clock output in this case is equal to the bit rate, and the phase is such tha
rising edge of the clock is in the middle of the transmit data, as shown in figure 14.4.

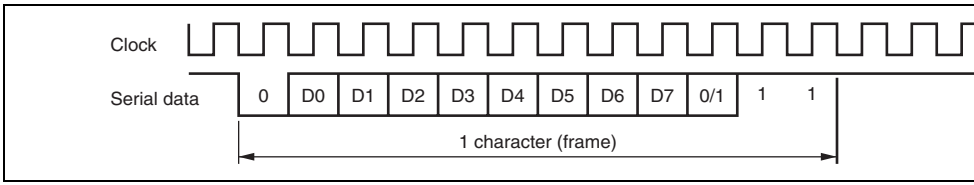
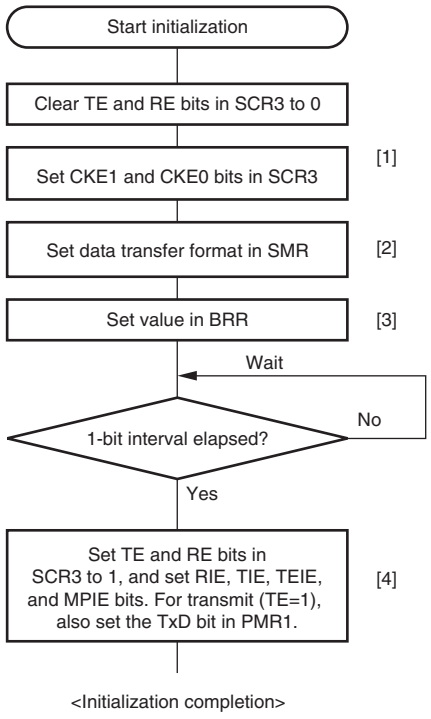


Figure 14.4 Relationship between Output Clock and Transfer Data Phase (Asynchronous Mode) (Example with 8-Bit Data, Parity, Two Stop Bits)



[1] Set the clock selection in SCR3. Be sure to clear bits RIE, TIE, TEIE, and MPIE, and bits TE and RE, to 0.

When the clock output is selected in asynchronous mode, clock is output immediately after CKE1 and CKE0 settings are made. When the clock output is selected at reception in clocked synchronous mode, clock is output immediately after CKE1, CKE0, and RE are set to 1.

[2] Set the data transfer format in SMR.

[3] Write a value corresponding to the bit rate to BRR. Not necessary if an external clock is used.

[4] Wait at least one bit interval, then set the TE bit or RE bit in SCR3 to 1. RE settings enable the RXD pin to be used. For transmission, set the TXD bit in PMR1 to 1 to enable the TXD output pin to be used. Also set the RIE, TIE, TEIE, and MPIE bits, depending on whether interrupts are required. In asynchronous mode, the bits are marked at transmission and idled at reception to wait for the start bit.

For transmission, set the TE bit to 1 and then output 1 for one frame to enable.

Figure 14.5 Sample SCI3 Initialization Flowchart

3. SCI3 checks the TDRE flag at the timing for sending the stop bit.
4. If the TDRE flag is 0, the data is transferred from TDR to TSR, the stop bit is sent, an serial transmission of the next frame is started.
5. If the TDRE flag is 1, the TEND flag in SSR is set to 1, the stop bit is sent, and then t state" is entered, in which 1 is output. If the TEIE bit in SCR3 is set to 1 at this time, a interrupt request is generated.
6. Figure 14.7 shows a sample flowchart for transmission in asynchronous mode.

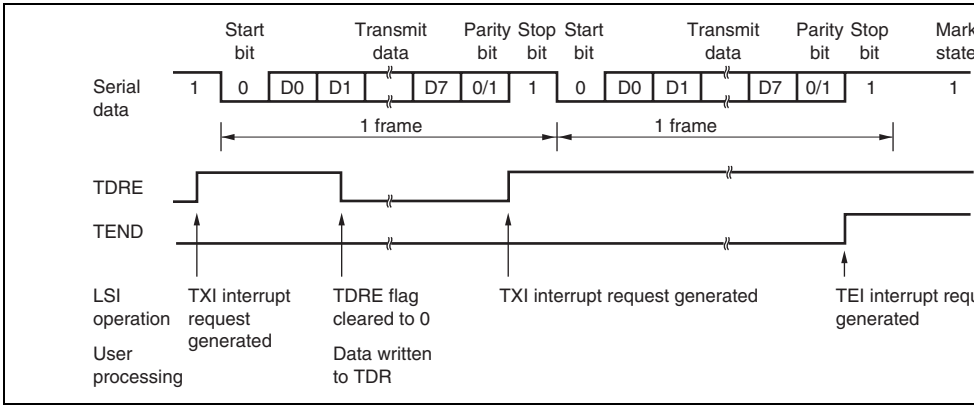
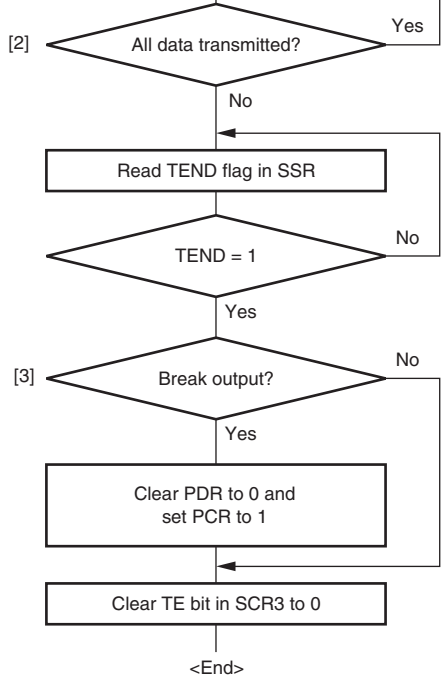


Figure 14.6 Example of SCI3 Transmission in Asynchronous Mode (8-Bit Data, Parity, One Stop Bit)



and PDR to 0, clear TxD in PMR
to 0, then clear the TE bit in SCR
to 0.

Figure 14.7 Sample Serial Transmission Data Flowchart (Asynchronous Mo

RDR. If the RIE bit in SCR3 is set to 1 at this time, an ERI interrupt request is genera

4. If a framing error is detected (when the stop bit is 0), the FER bit in SSR is set to 1 an data is transferred to RDR. If the RIE bit in SCR3 is set to 1 at this time, an ERI inter request is generated.
5. If reception is completed successfully, the RDRF bit in SSR is set to 1, and receive da transferred to RDR. If the RIE bit in SCR3 is set to 1 at this time, an RXI interrupt re generated. Continuous reception is possible because the RXI interrupt routine reads th data transferred to RDR before reception of the next receive data has been completed.

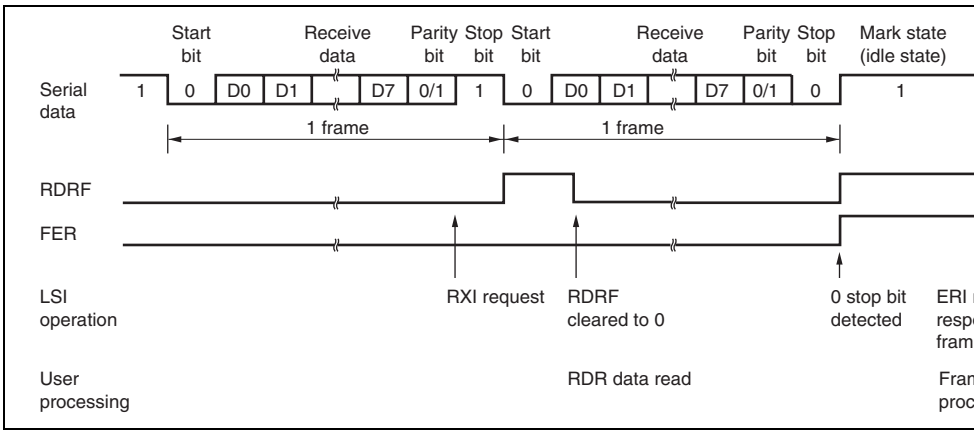
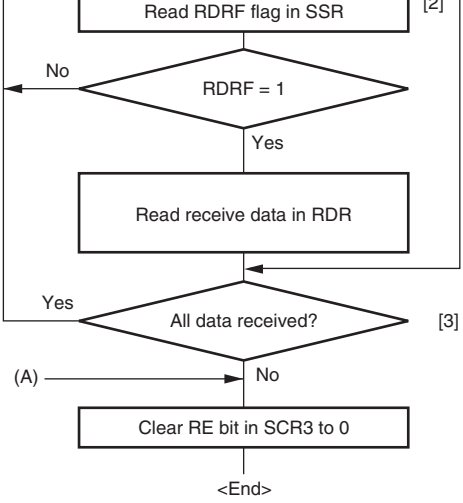


Figure 14.8 Example of SCI3 Reception in Asynchronous Mode (8-Bit Data, Parity, One Stop Bit)

0	0	1	0	Transferred to RDR	Framing error
0	0	0	1	Transferred to RDR	Parity error
1	1	1	0	Lost	Overrun error + frami
1	1	0	1	Lost	Overrun error + parity
0	0	1	1	Transferred to RDR	Framing error + parity
1	1	1	1	Lost	Overrun error + frami parity error

Note: * The RDRF flag retains the state it had before data reception.



the error. After performing the
appropriate error processing, en
that the OER, PER, and FER fla
all cleared to 0. Reception cann
resumed if any of these flags are
1. In the case of a framing error
break can be detected by readin
value of the input port correspon
the RxD pin.

Figure 14.9 Sample Serial Reception Data Flowchart (Asynchronous Mode

duplex communication through the use of a common clock. Both the transmitter and the
also have a double-buffered structure, so data can be read or written during transmission
reception, enabling continuous data transfer.

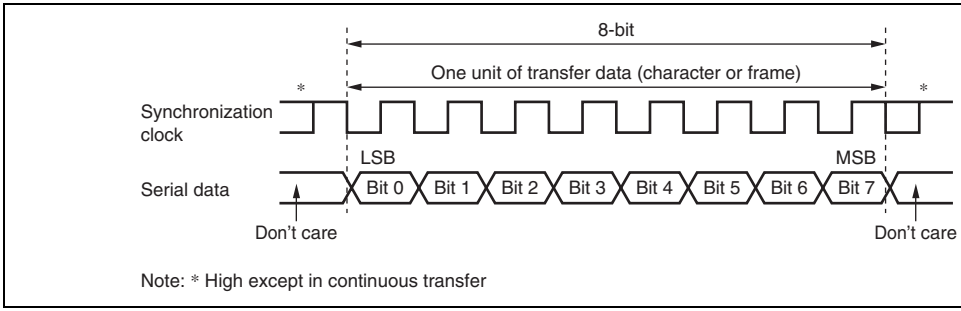


Figure 14.10 Data Format in Clocked Synchronous Communication

14.5.1 Clock

Either an internal clock generated by the on-chip baud rate generator or an external synchronization clock input at the SCK3 pin can be selected, according to the setting of bit in SMR and CKE0 and CKE1 bits in SCR3. When SCI3 is operated on an internal cl synchronization clock is output from the SCK3 pin. Eight synchronization clock pulses in the transfer of one character, and when no transfer is performed the clock is fixed hig

1. SCI3 monitors the TDRE flag in SSR, and if the flag is 0, SCI3 recognizes that data h written to TDR, and transfers the data from TDR to TSR.
2. SCI3 sets the TDRE flag to 1 and starts transmission. If the TIE bit in SCR3 is set to 1 time, a transmit data empty interrupt (TXI) is generated.
3. 8-bit data is sent from the TXD pin synchronized with the output clock when output c mode has been specified, and synchronized with the input clock when use of an extern has been specified. Serial data is transmitted sequentially from the LSB (bit 0), from t pin.
4. SCI3 checks the TDRE flag at the timing for sending the MSB (bit 7).
5. If the TDRE flag is cleared to 0, data is transferred from TDR to TSR, and serial trans of the next frame is started.
6. If the TDRE flag is set to 1, the TEND flag in SSR is set to 1, and the TDRE flag mai output state of the last bit. If the TEIE bit in SCR3 is set to 1 at this time, a TEI interru request is generated.
7. The SCK3 pin is fixed high at the end of transmission.

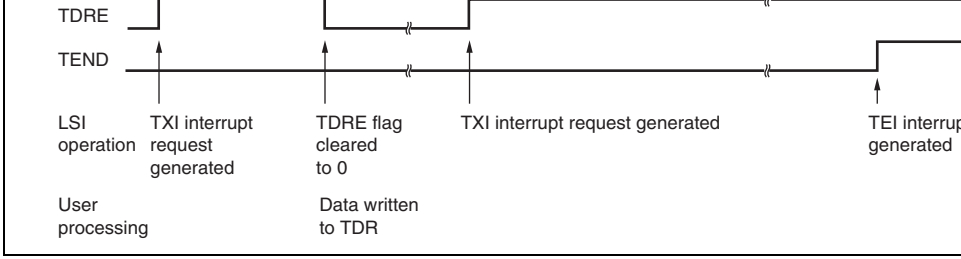


Figure 14.11 Example of SCI3 Transmission in Clocked Synchronous Mo

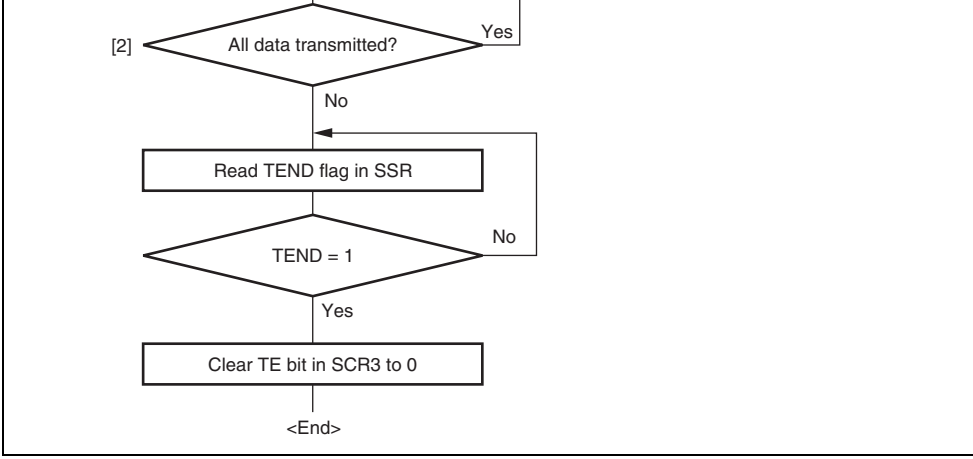


Figure 14.12 Sample Serial Transmission Flowchart (Clocked Synchronous M

time, an ERI interrupt request is generated, receive data is not transferred to RDR, a
RDRF flag remains to be set to 1.

4. If reception is completed successfully, the RDRF bit in SSR is set to 1, and receive
transferred to RDR. If the RIE bit in SCR3 is set to 1 at this time, an RXI interrupt
generated.

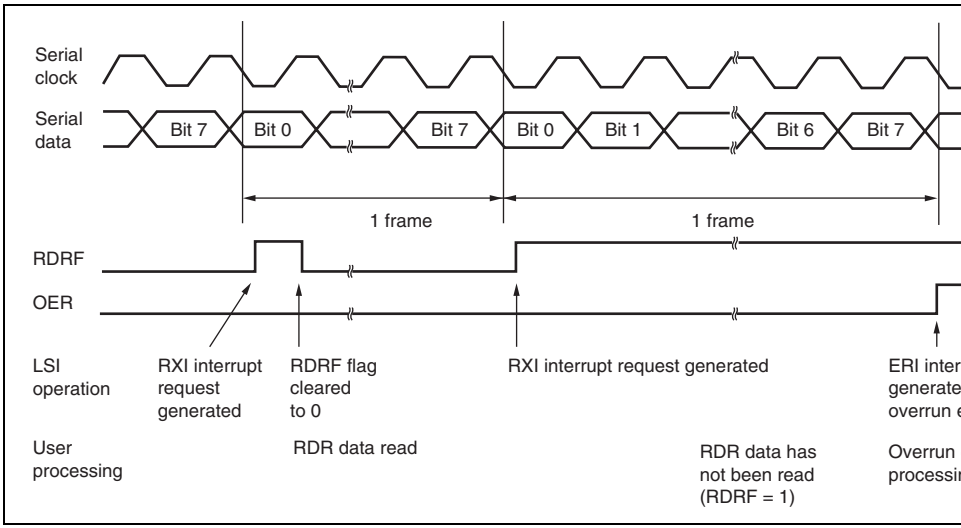
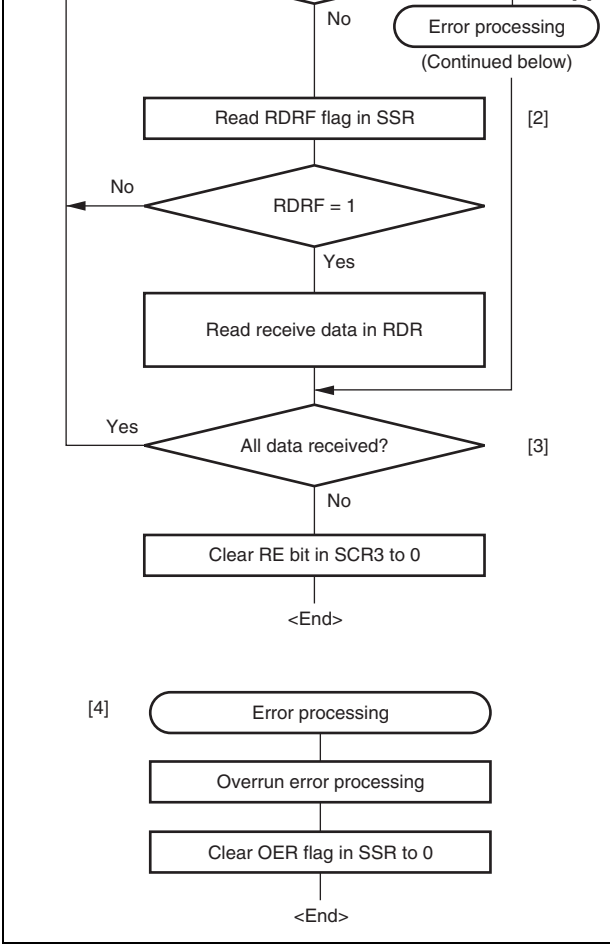


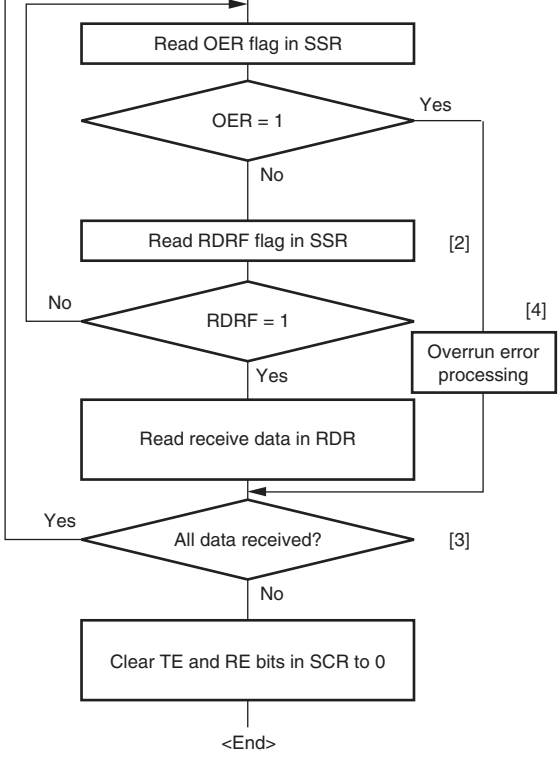
Figure 14.13 Example of SCI3 Reception in Clocked Synchronous Mode



[3] To continue serial reception, before th
MSB (bit 7) of the current frame is rec
reading the RDRF flag and reading R
should be finished. When data is read
RDR, the RDRF flag is automatically
cleared to 0.

[4] If an overrun error occurs, read the O
flag in SSR, and after performing the
appropriate error processing, clear th
flag to 0. Reception cannot be resum
the OER flag is set to 1.

Figure 14.14 Sample Serial Reception Flowchart (Clocked Synchronous Mod



reading the RDRF flag, reading R
Also, before the MSB (bit 7) of th
current frame is transmitted, read
from the TDRE flag to confirm th
writing is possible. Then write da
TDR.
When data is written to TDR, the
TDRE flag is automatically cleare
0. When data is read from RDR,
RDRF flag is automatically cleare
0.

[4] If an overrun error occurs, read t
OER flag in SSR, and after
performing the appropriate error
processing, clear the OER flag to
Transmission/reception cannot b
resumed if the OER flag is set to
For overrun error processing, see
figure 14.14.

Figure 14.15 Sample Flowchart of Simultaneous Serial Transmit and Receive Ope (Clocked Synchronous Mode)

cycle is a data transmission cycle. Figure 14.16 shows an example of inter-processor
communication using the multiprocessor format. The transmitting station first sends the
of the receiving station with which it wants to perform serial communication as data wit
multiprocessor bit added. It then sends transmit data as data with a 0 multiprocessor bit
When data with a 1 multiprocessor bit is received, the receiving station compares that da
own ID. The station whose ID matches then receives the data sent next. Stations whose
match continue to skip data until data with a 1 multiprocessor bit is again received.

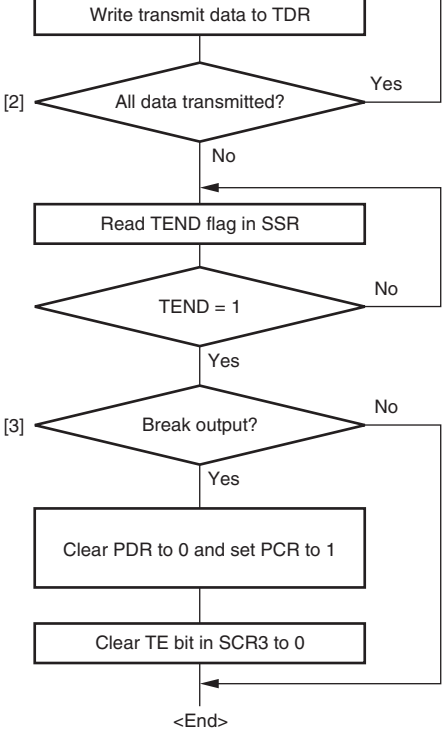
SCI3 uses the MPIE bit in SCR3 to implement this function. When the MPIE bit is set t
transfer of receive data from RSR to RDR, error flag detection, and setting the SSR statu
RDRF, FER, and OER, to 1, are inhibited until data with a 1 multiprocessor bit is receiv
reception of a receive character with a 1 multiprocessor bit, the MPBR bit in SSR is set
the MPIE bit is automatically cleared, thus normal reception is resumed. If the RIE bit i
set to 1 at this time, an RXI interrupt is generated.

When the multiprocessor format is selected, the parity bit setting is rendered invalid. Al
settings are the same as those in normal asynchronous mode. The clock used for multipr
communication is the same as that in normal asynchronous mode.

specification receiving station specified by ID

[Legend]
MPB: Multiprocessor bit

Figure 14.16 Example of Inter-Processor Communication Using Multiprocessor F (Transmission of Data H'AA to Receiving Station A)



[3] To output a break in serial transmission, set the port PCR to 1, clear PDR to 0, then clear the TE bit in SCR3 to 0.

Figure 14.17 Sample Multiprocessor Serial Transmission Flowchart

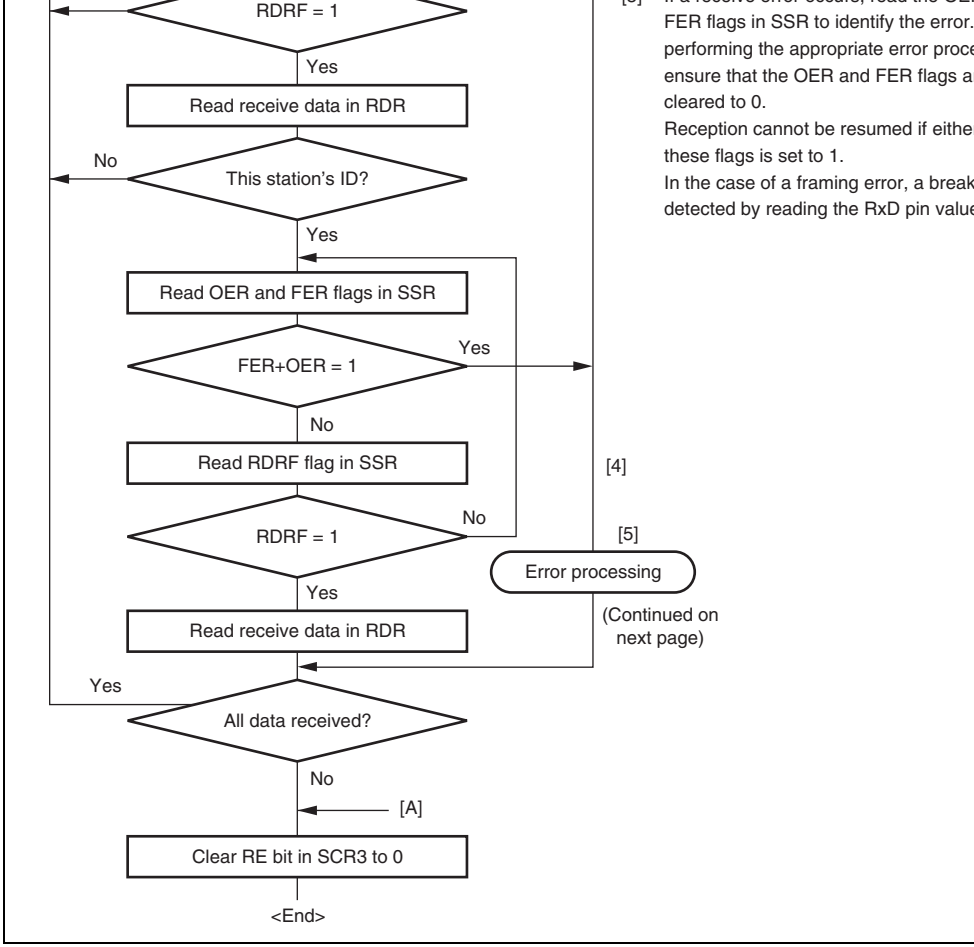


Figure 14.18 Sample Multiprocessor Serial Reception Flowchart (1)

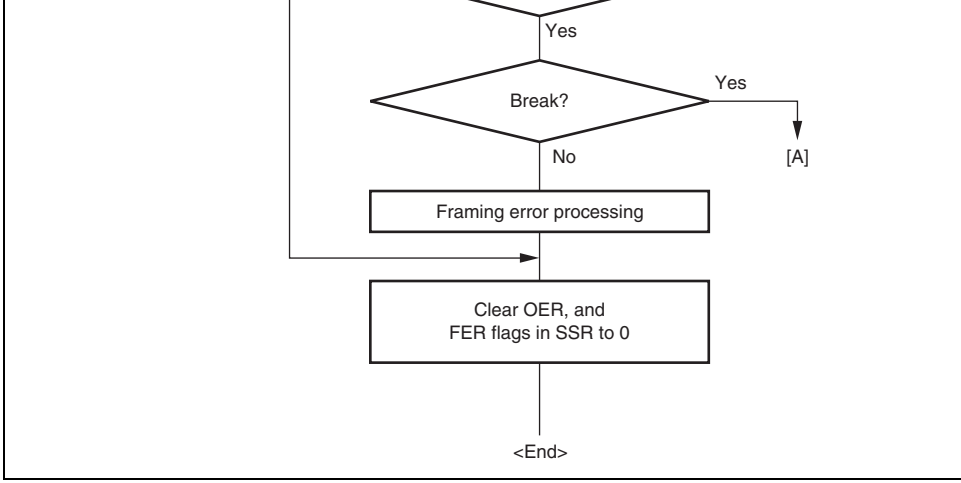


Figure 14.18 Sample Multiprocessor Serial Reception Flowchart (2)

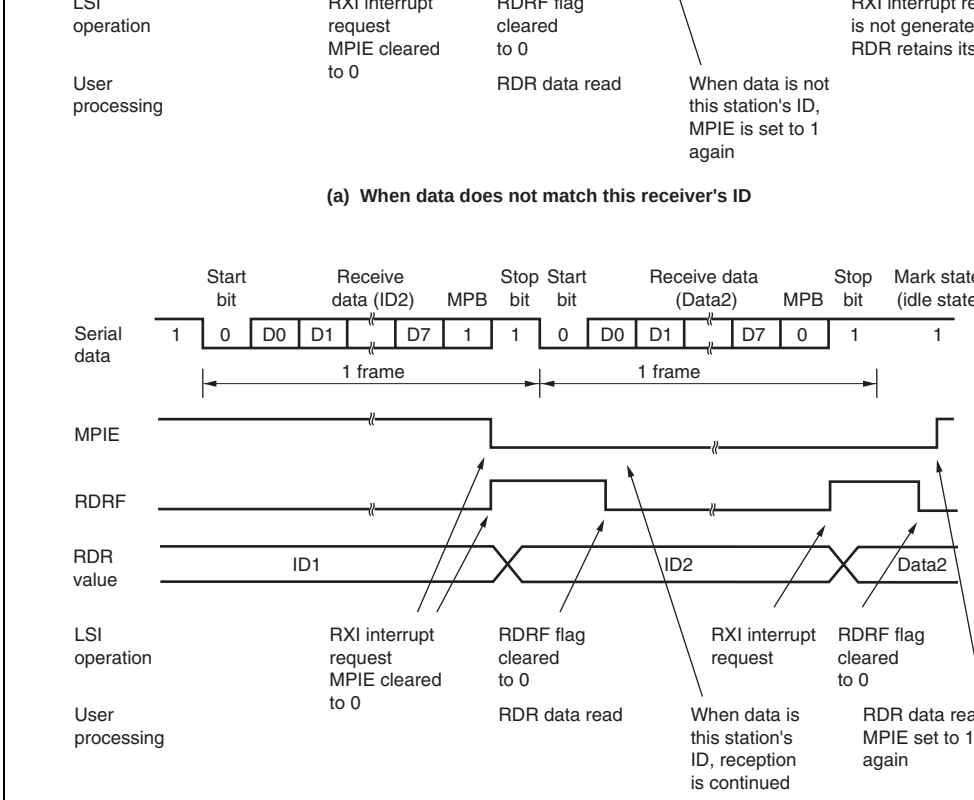


Figure 14.19 Example of SCI3 Reception Using Multiprocessor Format (Example with 8-Bit Data, Multiprocessor Bit, One Stop Bit)

Transmission End	TEI	Setting TEND in SSR
Receive Error	ERI	Setting OER, FER, and PER in SSR

The initial value of the TDRE flag in SSR is 1. Thus, when the TIE bit in SCR3 is set to
transferring the transmit data to TDR, a TXI interrupt request is generated even if the tra
is not ready. The initial value of the TEND flag in SSR is 1. Thus, when the TEIE bit in
set to 1 before transferring the transmit data to TDR, a TEI interrupt request is generated
the transmit data has not been sent. It is possible to make use of the most of these interru
requests efficiently by transferring the transmit data to TDR in the interrupt routine. To
the generation of these interrupt requests (TXI and TEI), set the enable bits (TIE and TE
correspond to these interrupt requests to 1, after transferring the transmit data to TDR.

When the TXD bit in PMR1 is 1, the TxD pin is used as an I/O port whose direction (inpu
output) and level are determined by PCR and PDR. This can be used to set the TxD pin to
state (high level) or send a break during serial data transmission. To maintain the commu
line at mark state until TE is set to 1, set PCR and PDR to 1 respectively, and also set the
to 1. At this time, the TxD pin becomes an I/O port, and 1 is output from the TxD pin. To
break during serial data transmission, first set PCR to 1 and clear PDR to 0, and then set t
bit to 1. Regardless of the current transmission state, the TxD pin becomes an I/O port, an
output from the TxD pin.

14.8.3 Receive Error Flags and Transmit Operations (Clocked Synchronous Mod

Transmission cannot be started when a receive error flag (OER, PER, or FER) is set to 1,
the TDRE flag is cleared to 0. Be sure to clear the receive error flags to 0 before starting
transmission. Note also that receive error flags cannot be cleared to 0 even if the RE bit is
to 0.

Legend N : Ratio of bit rate to clock (N = 16)
D : Clock duty (D = 0.5 to 1.0)
L : Frame length (L = 9 to 12)
F : Absolute value of clock rate deviation

Assuming values of F (absolute value of clock rate deviation) = 0 and D (clock duty) = 0
formula (1), the reception margin can be given by the formula.

$$M = \{0.5 - 1/(2 \times 16)\} \times 100\ [\%] = 46.875\%$$

However, this is only the computed value, and a margin of 20% to 30% should be allow
system design.

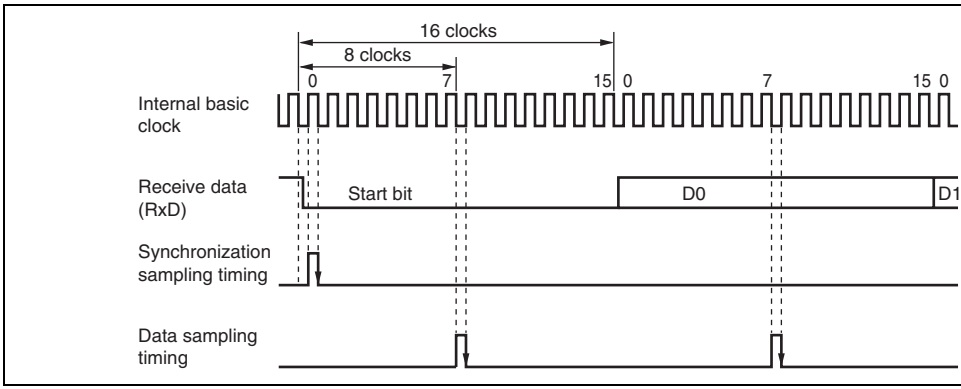


Figure 14.20 Receive Data Sampling Timing in Asynchronous Mode

15.1 Features

- Selection of I^2C format or clocked synchronous serial format
- Continuous transmission/reception

  Since the shift register, transmit data register, and receive data register are independe each other, the continuous transmission/reception can be performed.

I^2C bus format:

- Start and stop conditions generated automatically in master mode
- Selection of acknowledge output levels when receiving
- Automatic loading of acknowledge bit when transmitting
- Bit synchronization/wait function

  In master mode, the state of SCL is monitored per bit, and the timing is synchronized automatically.

  If transmission/reception is not yet possible, set the SCL to low until preparations ar completed.
- Six interrupt sources

  Transmit data empty (including slave-address match), transmit end, receive data full slave-address match), arbitration lost, NACK detection, and stop condition detection
- Direct bus drive

  Two pins, SCL and SDA pins, function as NMOS open-drain outputs when the bus d function is selected.

Clocked synchronous format:

- Four interrupt sources

  Transmit-data-empty, transmit-end, receive-data-full, and overrun error

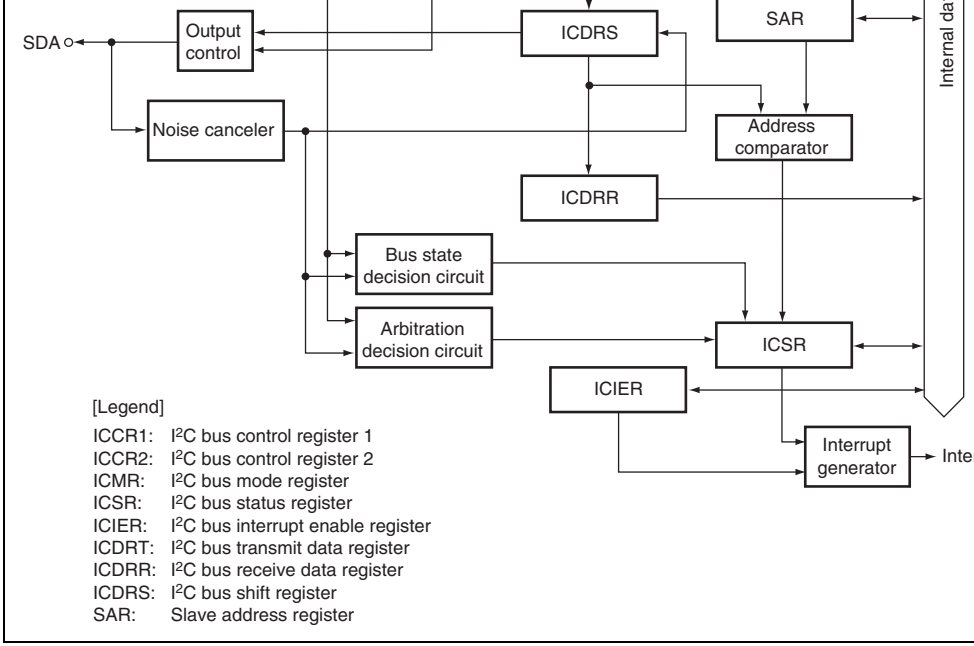


Figure 15.1 Block Diagram of I²C Bus Interface 2

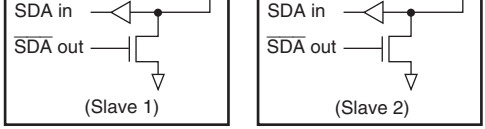


Figure 15.2 External Circuit Connections of I/O Pins

15.2 Input/Output Pins

Table 15.1 summarizes the input/output pins used by the I^2C bus interface 2.

Table 15.1 Pin Configuration

Name	Abbreviation	I/O	Function
Serial clock	SCL	I/O	I^2C serial clock input/output
Serial data	SDA	I/O	I^2C serial data input/output

- I²C bus transmit data register (ICDRT)
- I²C bus receive data register (ICDRR)
- I²C bus shift register (ICDRS)

15.3.1 I²C Bus Control Register 1 (ICCR1)

ICCR1 enables or disables the I²C bus interface 2, controls transmission or reception, and
master or slave mode, transmission or reception, and transfer clock frequency in master m

Bit	Bit Name	Initial Value	R/W	Description
7	ICE	0	R/W	I²C Bus Interface Enable 0: This module is halted. (SCL and SDA pins are port function.) 1: This bit is enabled for transfer operations. (SCL SDA pins are bus drive state.)
6	RCVD	0	R/W	Reception Disable This bit enables or disables the next operation wh is 0 and ICDRR is read. 0: Enables next reception 1: Disables next reception

				master mode with the clock synchronous serial f MST is cleared to 0 and slave receive mode is e Operating modes are described below according and TRS combination. When clocked synchrono format is selected and MST is 1, clock is output. 00: Slave receive mode 01: Slave transmit mode 10: Master receive mode 11: Master transmit mode
3 to 0	CKS3 to CKS0	All 0	R/W	Transfer Clock Select 3 to 0 These bits should be set according to the neces transfer rate (see table 15.2) in master mode. In mode, these bits are used reservation of the set transmit mode. The time is $10t_{cyc}$ when CKS3 = 0 $20t_{cyc}$ when CKS3 = 1.

		1	0	φ/112	44.6 kHz	71.4 kHz	89.3
			1	φ/128	39.1 kHz	62.5 kHz	78.1
1	0	0	0	φ/56	89.3 kHz	143 kHz	179 k
			1	φ/80	62.5 kHz	100 kHz	125 k
		1	0	φ/96	52.1 kHz	83.3 kHz	104 k
			1	φ/128	39.1 kHz	62.5 kHz	78.1
	1	0	0	φ/160	31.3 kHz	50.0 kHz	62.5
			1	φ/200	25.0 kHz	40.0 kHz	50.0
		1	0	φ/224	22.3 kHz	35.7 kHz	44.6
			1	φ/256	19.5 kHz	31.3 kHz	39.1

				SDA level changes from high to low under the cond SCL = high, assuming that the start condition has b issued. This bit is cleared to 0 when the SDA level from low to high under the condition of SCL = high, that the stop condition has been issued. Write 1 to 0 to SCP to issue a start condition. Follow this proc when also re-transmitting a start condition. Write 0 and 0 in SCP to issue a stop condition. To issue st conditions, use the MOV instruction.
6	SCP	1	R/W	Start/Stop Issue Condition Disable The SCP bit controls the issue of start/stop conditio master mode. To issue a start condition, write 1 in BBSY and 0 in retransmit start condition is issued in the same way a stop condition, write 0 in BBSY and 0 in SCP. Th always read as 1. If 1 is written, the data is not stor
5	SDAO	1	R/W	SDA Output Value Control This bit is used with SDAOP when modifying outpu SDA. This bit should not be manipulated during tra 0: When reading, SDA pin outputs low. When writing, SDA pin is changed to output low 1: When reading, SDA pin outputs high. When writing, SDA pin is changed to output Hi- high by external pull-up resistance).
4	SDAOP	1	R/W	SDAO Write Protect This bit controls change of output level of the SDA modifying the SDAO bit. To change the output leve SDAO and SDAOP to 0 or set SDAO to 1 and clea to 0 by the MOV instruction. This bit is always read

0	—	1	—	Reserved This bit is always read as 1.

15.3.3 I²C Bus Mode Register (ICMR)

ICMR selects whether the MSB or LSB is transferred first, performs master mode wait co and selects the transfer bit count.

Bit	Bit Name	Initial Value	R/W	Description
7	MLS	0	R/W	MSB-First/LSB-First Select 0: MSB-first 1: LSB-first Set this bit to 0 when the I²C bus format is used.
6	WAIT	0	R/W	Wait Insertion Bit In master mode with the I²C bus format, this bit sele whether to insert a wait after data transfer except th acknowledge bit. When WAIT is set to 1, after the fa clock for the final data bit, low period is extended for transfer clocks. If WAIT is cleared to 0, data and acknowledge bits are transferred consecutively with inserted. The setting of this bit is invalid in slave mode with th format or with the clocked synchronous serial forma
5, 4	—	All 1	—	Reserved These bits are always read as 1.

1	BC1	0	R/W	These bits specify the number of bits to be transfer
0	BC0	0	R/W	When read, the remaining number of transfer bits is indicated. With the I^2C bus format, the data is trans one addition acknowledge bit. Bit BC2 to BC0 setti be made during an interval between transfer frame BC2 to BC0 are set to a value other than 000, the s should be made while the SCL pin is low. The valu to 000 at the end of a data transfer, including the acknowledge bit. With the clock synchronous seria these bits should not be modified.

I^2C Bus Format	Clock Synchronous Serial Fo
000: 9 bits	000: 8 bits
001: 2 bits	001: 1 bits
010: 3 bits	010: 2 bits
011: 4 bits	011: 3 bits
100: 5 bits	100: 4 bits
101: 6 bits	101: 5 bits
110: 7 bits	110: 6 bits
111: 8 bits	111: 7 bits

				1: Transmit data empty interrupt request (TXI) is ena
6	TEIE	0	R/W	Transmit End Interrupt Enable This bit enables or disables the transmit end interrup the rising of the ninth clock while the TDRE bit in ICS TEI can be canceled by clearing the TEND bit or the to 0. 0: Transmit end interrupt request (TEI) is disabled. 1: Transmit end interrupt request (TEI) is enabled.
5	RIE	0	R/W	Receive Interrupt Enable This bit enables or disables the receive data full inte request (RXI) and the overrun error interrupt reques with the clocked synchronous format, when a receiv transferred from ICDRS to ICDRR and the RDRF bit is set to 1. RXI can be canceled by clearing the RDR bit to 0. 0: Receive data full interrupt request (RXI) and over interrupt request (ERI) with the clocked synchron format are disabled. 1: Receive data full interrupt request (RXI) and over interrupt request (ERI) with the clocked synchron format are enabled.

				0: Stop condition detection interrupt request (STPI) disabled.
				1: Stop condition detection interrupt request (STPI) enabled.
2	ACKE	0	R/W	Acknowledge Bit Judgement Select
				0: The value of the receive acknowledge bit is igno continuous transfer is performed.
				1: If the receive acknowledge bit is 1, continuous tr halted.
1	ACKBR	0	R	Receive Acknowledge
				In transmit mode, this bit stores the acknowledge d are returned by the receive device. This bit cannot modified.
				0: Receive acknowledge = 0
				1: Receive acknowledge = 1
0	ACKBT	0	R/W	Transmit Acknowledge
				In receive mode, this bit specifies the bit to be sent acknowledge timing.
				0: 0 is sent at the acknowledge timing.
				1: 1 is sent at the acknowledge timing.

				• When a start condition (including re-transfer) has issued • When transmit mode is entered from receive mo slave mode [Clearing conditions] • When 0 is written in TDRE after reading TDRE = • When data is written to ICDRT with an instructio
6	TEND	0	R/W	Transmit End [Setting conditions] • When the ninth clock of SCL rises with the I^2C bu while the TDRE flag is 1 • When the final bit of transmit frame is sent with t synchronous serial format [Clearing conditions] • When 0 is written in TEND after reading TEND = • When data is written to ICDRT with an instructio
5	RDRF	0	R/W	Receive Data Register Full [Setting condition] • When a receive data is transferred from ICDRS [Clearing conditions] • When 0 is written in RDRF after reading RDRF = • When ICDRR is read with an instruction

				frame transfer • In slave mode, when a stop condition is detecte after the general call address or the first byte sl address, next to detection of start condition, ac with the address set in SAR [Clearing condition] • When 0 is written in STOP after reading STOP
2	AL/OVE	0	R/W	Arbitration Lost Flag/Overrun Error Flag This flag indicates that arbitration was lost in maste with the I^2C bus format and that the final bit has be received while RDRF = 1 with the clocked synchro format. When two or more master devices attempt to seize at nearly the same time, if the I^2C bus interface det differing from the data it sent, it sets AL to 1 to indi the bus has been taken by another master. [Setting conditions] • If the internal SDA and SDA pin disagree at the SCL in master transmit mode • When the SDA pin outputs high in master mode start condition is detected • When the final bit is received with the clocked synchronous format while RDRF = 1 [Clearing condition] • When 0 is written in AL/OVE after reading AL/C

				[Clearing condition] • When 0 is written in AAS after reading AAS=1
0	ADZ	0	R/W	General Call Address Recognition Flag This bit is valid in I²C bus format slave receive mode [Setting condition] • When the general call address is detected in sla receive mode [Clearing condition] • When 0 is written in ADZ after reading ADZ=1

15.3.6 Slave Address Register (SAR)

SAR selects the communication format and sets the slave address. When the chip is in sla with the I²C bus format, if the upper 7 bits of SAR match the upper 7 bits of the first fram received after a start condition, the chip operates as the slave device.

Bit	Bit Name	Initial Value	R/W	Description
7 to 1	SVA6 to SVA0	All 0	R/W	Slave Address 6 to 0 These bits set a unique address in bits SVA6 to SVA differing form the addresses of other slave devices connected to the I²C bus.
0	FS	0	R/W	Format Select 0: I²C bus format is selected. 1: Clocked synchronous serial format is selected.

ICDRR is an 8-bit register that stores the receive data. When data of one byte is received transfers the receive data from ICDRS to ICDRR and the next data can be received. ICD receive-only register, therefore the CPU cannot write to this register. The initial value o is H'FF.

15.3.9 I^2C Bus Shift Register (ICDRS)

ICDRS is a register that is used to transfer/receive data. In transmission, data is transferr ICDRT to ICDRS and the data is sent from the SDA pin. In reception, data is transferred ICDRS to ICDRR after data of one byte is received. This register cannot be read directly CPU.

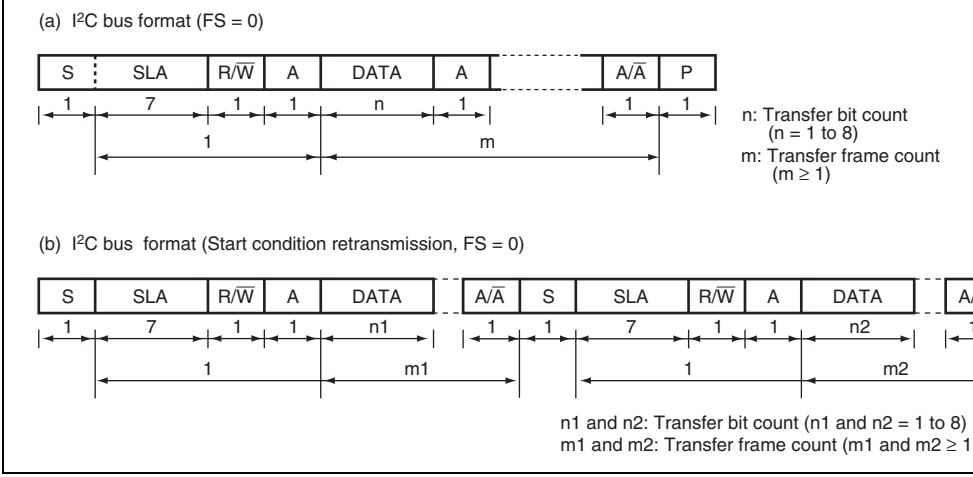


Figure 15.3 I²C Bus Formats

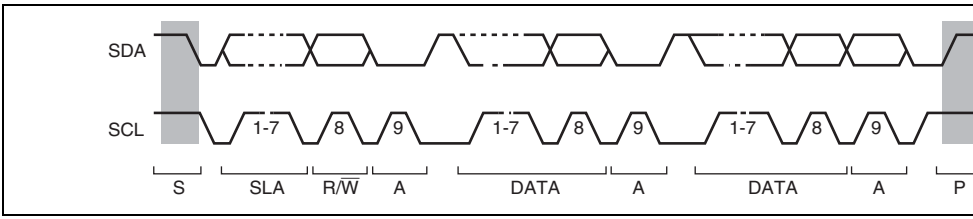


Figure 15.4 I²C Bus Timing

In master transmit mode, the master device outputs the transmit clock and transmit data, slave device returns an acknowledge signal. For master transmit mode operation timing, figures 15.5 and 15.6. The transmission procedure and operations in master transmit mo described below.

1. Set the ICE bit in ICCR1 to 1. Set the MLS and WAIT bits in ICMR and the CKS3 t bits in ICCR1 to 1. (Initial setting)
2. Read the BBSY flag in ICCR2 to confirm that the bus is free. Set the MST and TRS ICCR1 to select master transmit mode. Then, write 1 to BBSY and 0 to SCP using M instruction. (Start condition issued) This generates the start condition.
3. After confirming that TDRE in ICSR has been set, write the transmit data (the first b show the slave address and R/$\overline{W}$) to ICDRT. At this time, TDRE is automatically cle and data is transferred from ICDRT to ICDRS. TDRE is set again.
4. When transmission of one byte data is completed while TDRE is 1, TEND in ICSR i at the rise of the 9th transmit clock pulse. Read the ACKBR bit in ICIER, and confir slave device has been selected. Then, write second byte data to ICDRT. When ACK the slave device has not been acknowledged, so issue the stop condition. To issue the condition, write 0 to BBSY and SCP using MOV instruction. SCL is fixed low until transmit data is prepared or the stop condition is issued.
5. The transmit data after the second byte is written to ICDRT every time TDRE is set.
6. Write the number of bytes to be transmitted to ICDRT. Wait until TEND is set (the e byte data transmission) while TDRE is 1, or wait for NACK (NACKF in ICSR = 1) receive device while ACKE in ICIER is 1. Then, issue the stop condition to clear TE NACKF.
7. When the STOP bit in ICSR is set to 1, the operation returns to the slave receive mo

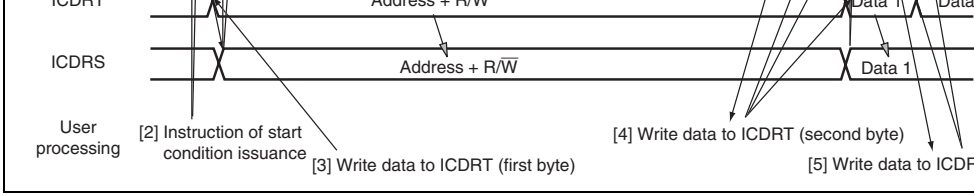


Figure 15.5 Master Transmit Mode Operation Timing (1)

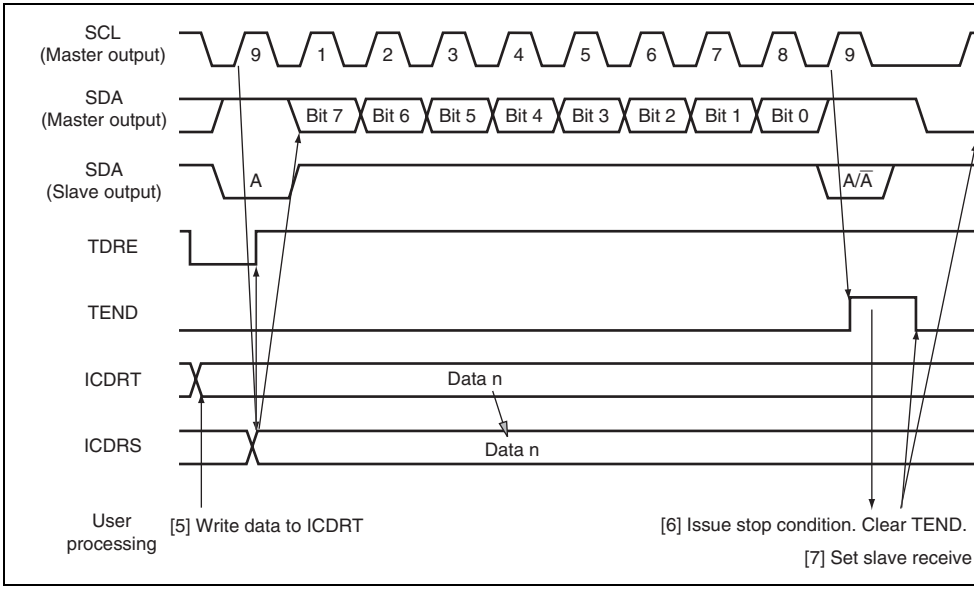


Figure 15.6 Master Transmit Mode Operation Timing (2)

level specified by ACKBT in ICIER to SDA, at the 9th receive clock pulse.

3. After the reception of first frame data is completed, the RDRF bit in ICST is set to 1 of 9th receive clock pulse. At this time, the receive data is read by reading ICDRR, a is cleared to 0.
4. The continuous reception is performed by reading ICDRR every time RDRF is set. I receive clock pulse falls after reading ICDRR by the other processing while RDRF i fixed low until ICDRR is read.
5. If next frame is the last receive data, set the RCVD bit in ICCR1 to 1 before reading This enables the issuance of the stop condition after the next reception.
6. When the RDRF bit is set to 1 at rise of the 9th receive clock pulse, issue the stage c
7. When the STOP bit in ICSR is set to 1, read ICDRR. Then clear the RCVD bit to 0.
8. The operation returns to the slave receive mode.

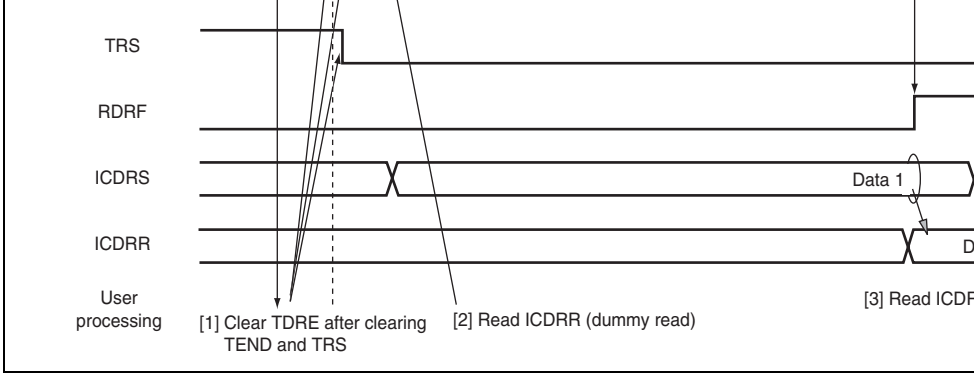


Figure 15.7 Master Receive Mode Operation Timing (1)

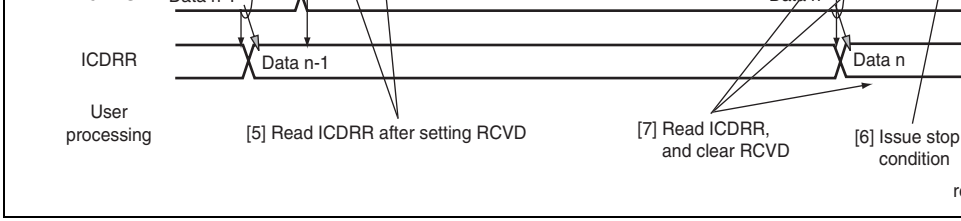


Figure 15.8 Master Receive Mode Operation Timing (2)

15.4.4 Slave Transmit Operation

In slave transmit mode, the slave device outputs the transmit data, while the master devi the receive clock and returns an acknowledge signal. For slave transmit mode operation refer to figures 15.9 and 15.10.

The transmission procedure and operations in slave transmit mode are described below.

1. Set the ICE bit in ICCR1 to 1. Set the MLS and WAIT bits in ICMR and the CKS3 t bits in ICCR1 to 1. (Initial setting) Set the MST and TRS bits in ICCR1 to select sla mode, and wait until the slave address matches.
2. When the slave address matches in the first frame following detection of the start co the slave device outputs the level specified by ACKBT in ICIER to SDA, at the rise clock pulse. At this time, if the 8th bit data (R/$\overline{W}$) is 1, the TRS and ICSR bits in IC set to 1, and the mode changes to slave transmit mode automatically. The continuou transmission is performed by writing transmit data to ICDRT every time TDRE is se
3. If TDRE is set after writing last transmit data to ICDRT, wait until TEND in ICSR i with TDRE = 1. When TEND is set, clear TEND.
4. Clear TRS for the end processing, and read ICDRR (dummy read). SCL is free.
5. Clear TDRE.

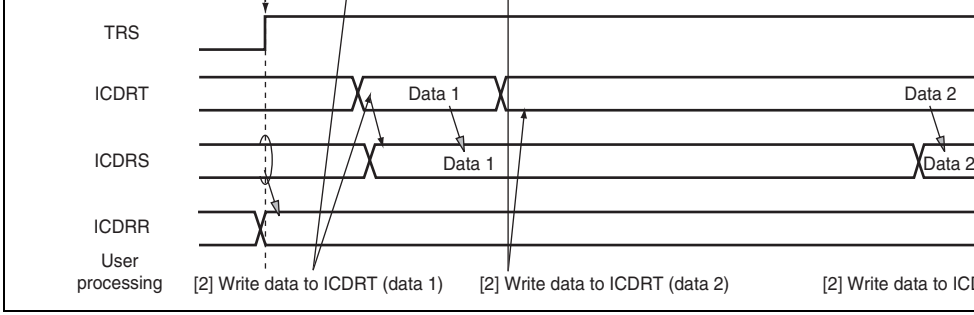


Figure 15.9 Slave Transmit Mode Operation Timing (1)

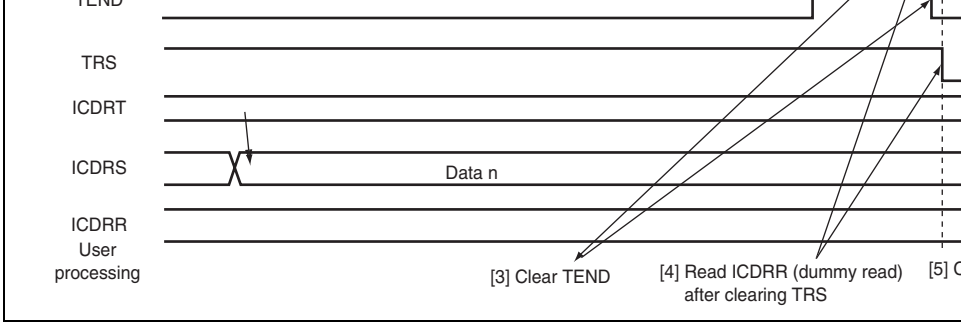


Figure 15.10 Slave Transmit Mode Operation Timing (2)

15.4.5 Slave Receive Operation

In slave receive mode, the master device outputs the transmit clock and transmit data, an slave device returns an acknowledge signal. For slave receive mode operation timing, re figures 15.11 and 15.12. The reception procedure and operations in slave receive mode a described below.

1. Set the ICE bit in ICCR1 to 1. Set the MLS and WAIT bits in ICMR and the CKS3 bits in ICCR1 to 1. (Initial setting) Set the MST and TRS bits in ICCR1 to select slav mode, and wait until the slave address matches.
2. When the slave address matches in the first frame following detection of the start co the slave device outputs the level specified by ACKBT in ICIER to SDA, at the rise clock pulse. At the same time, RDRF in ICSR is set to read ICDRR (dummy read). (read data show the slave address and R/$\overline{W}$, it is not used.)
3. Read ICDRR every time RDRF is set. If 8th receive clock pulse falls while RDRF is fixed low until ICDRR is read. The change of the acknowledge before reading ICDR returned to the master device, is reflected to the next transmit frame.

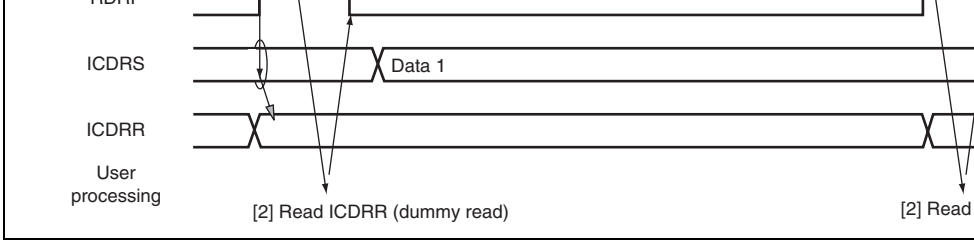


Figure 15.11 Slave Receive Mode Operation Timing (1)

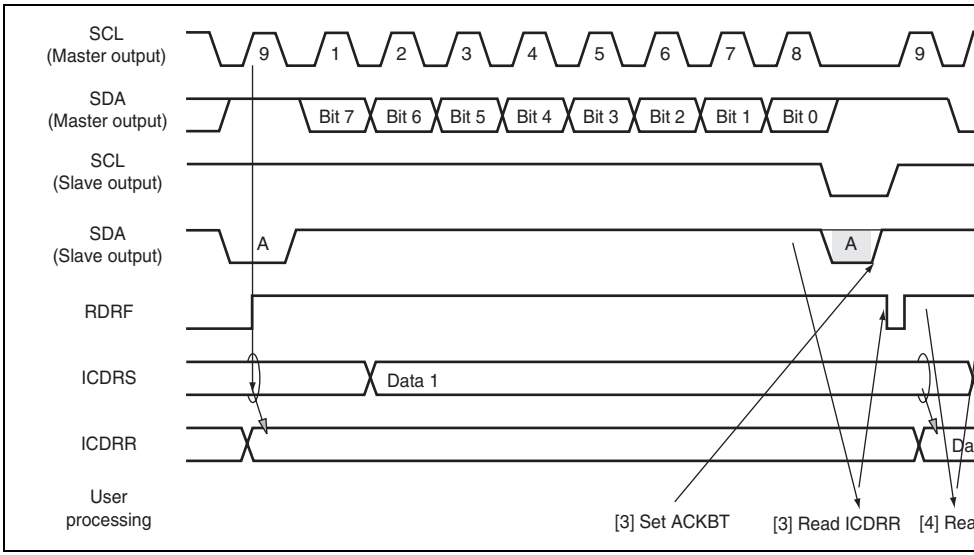


Figure 15.12 Slave Receive Mode Operation Timing (2)

of the SCL clock is guaranteed. The MLS bit in ICMR sets the order of data transfer, in MSB first or LSB first. The output level of SDA can be changed during the transfer wai SDAO bit in ICCR2.

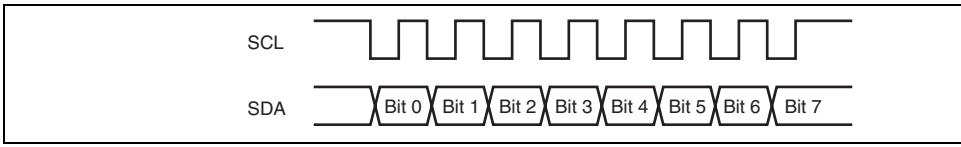


Figure 15.13 Clocked Synchronous Serial Transfer Format

(2) Transmit Operation:

In transmit mode, transmit data is output from SDA, in synchronization with the fall of t clock. The transfer clock is output when MST in ICCR1 is 1, and is input when MST is transmit mode operation timing, refer to figure 15.14. The transmission procedure and o in transmit mode are described below.

1. Set the ICE bit in ICCR1 to 1. Set the MST and CKS3 to CKS0 bits in ICCR1 to 1. (setting)
2. Set the TRS bit in ICCR1 to select the transmit mode. Then, TDRE in ICSR is set.
3. Confirm that TDRE has been set. Then, write the transmit data to ICDRT. The data i transferred from ICDRT to ICDRS, and TDRE is set automatically. The continuous transmission is performed by writing data to ICDRT every time TDRE is set. When from transmit mode to receive mode, clear TRS while TDRE is 1.

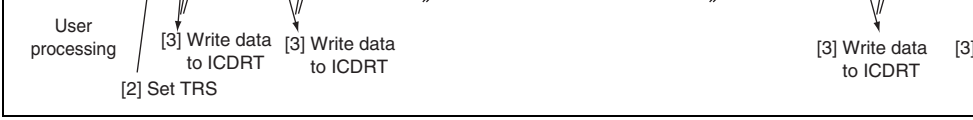


Figure 15.14 Transmit Mode Operation Timing

(3) Receive Operation:

In receive mode, data is latched at the rise of the transfer clock. The transfer clock is outp MST in ICCR1 is 1, and is input when MST is 0. For receive mode operation timing, refe figure 15.15. The reception procedure and operations in receive mode are described belov

1. Set the ICE bit in ICCR1 to 1. Set the MST and CKS3 to CKS0 bits in ICCR1 to 1. (I setting)
2. When the transfer clock is output, set MST to 1 to start outputting the receive clock.
3. When the receive operation is completed, data is transferred from ICDRS to ICDRR a RDRF in ICSR is set. When MST = 1, the next byte can be received, so the clock is continually output. The continuous reception is performed by reading ICDRR every ti RDRF is set. When the 8th clock is risen while RDRF is 1, the overrun is detected and AL/OVE in ICSR is set. At this time, the previous reception data is retained in ICDRI
4. To stop receiving when MST = 1, set RCVD in ICCR1 to 1, then read ICDRR. Then, fixed high after receiving the next byte data.

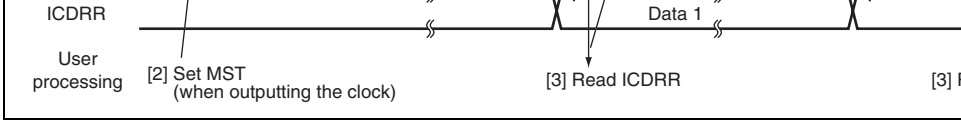


Figure 15.15 Receive Mode Operation Timing

15.4.7 Noise Canceler

The logic levels at the SCL and SDA pins are routed through the noise canceler before b latched internally. Figure 15.16 shows a block diagram of the noise canceler.

The noise canceler consists of two cascaded latches and a match detector. The SCL (or S input signal is sampled on the system clock, but is not passed forward to the next circuit outputs of both latches agree. If they do not agree, the previous value is held.

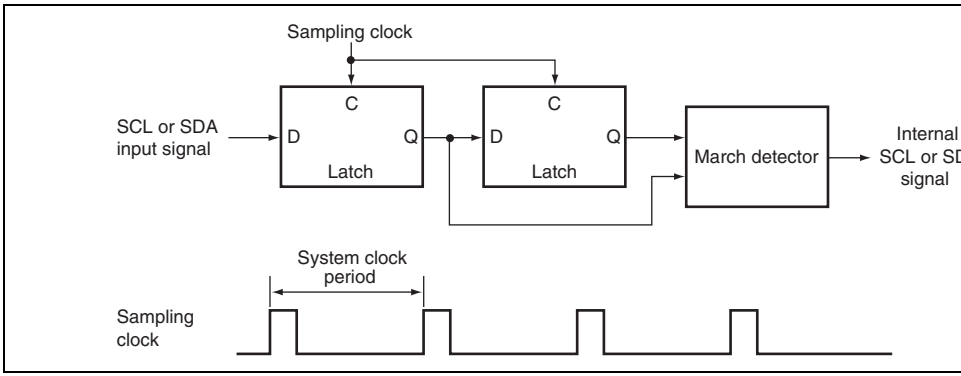


Figure 15.16 Block Diagram of Noise Canceler

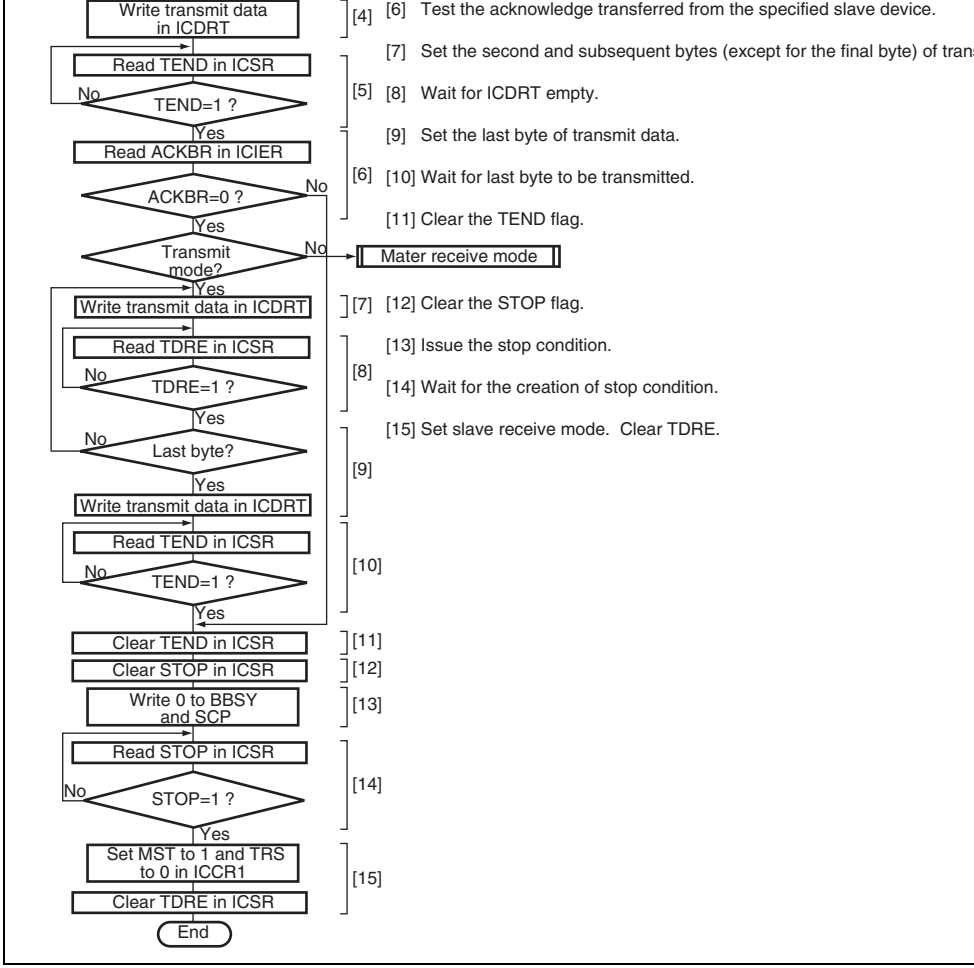
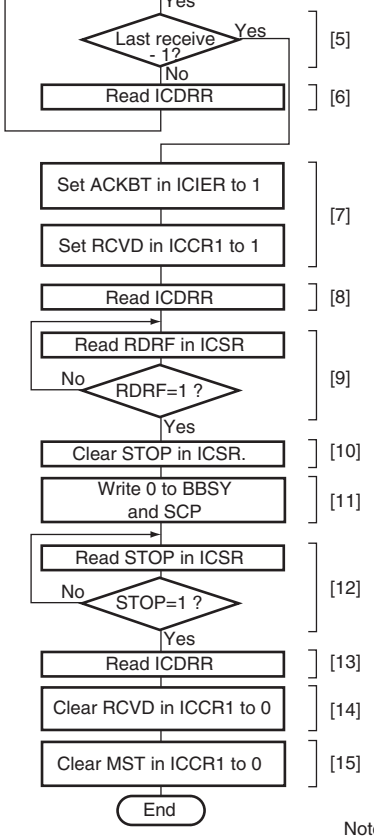


Figure 15.17 Sample Flowchart for Master Transmit Mode



[9] Wait for the last byte to be receive.

[10] Clear the STOP flag.

[11] Issue the stop condition.

[12] Wait for the creation of stop condition.

[13] Read the last byte of receive data.

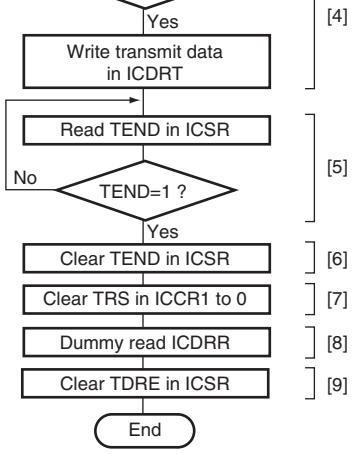
[14] Clear RCVD.

[15] Set slave receive mode.

Note: Do not activate an interrupt during the execution of steps [1] to [3].

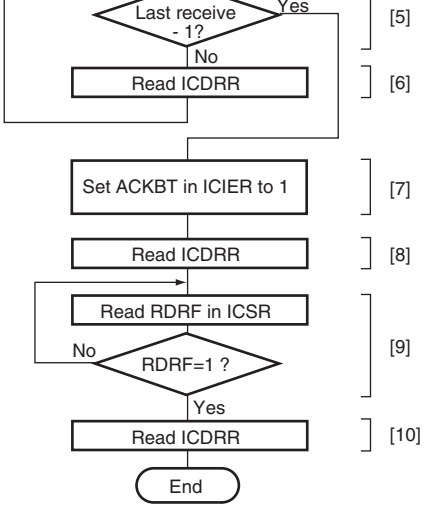
Supplementary explanation: When one byte is received, steps [2] to [6] are skipped after step [1 before jumping to step [7]. The step [8] is dammy-read in ICDRR.

Figure 15.18 Sample Flowchart for Master Receive Mode



[9] Clear the TDRE flag.

Figure 15.19 Sample Flowchart for Slave Transmit Mode



[9] Wait the last byte to be received.

[10] Read for the last byte of receive data.

Supplementary explanation: When one byte is received, steps [2] to [6] are skipped after step [1
before jumping to step [7]. The step [8] is dammy-read in ICDRR.

Figure 15.20 Sample Flowchart for Slave Receive Mode

Transmit Data Empty	TXI	(TDRE=1) • (TIE=1)	○	○
Transmit End	TEI	(TEND=1) • (TEIE=1)	○	○
Receive Data Full	RXI	(RDRF=1) • (RIE=1)	○	○
STOP Recognition	STPI	(STOP=1) • (STIE=1)	○	×
NACK Receive	NAKI	{(NACKF=1)+(AL=1)} • (NAKIE=1)	○	×
Arbitration Lost/Overrun Error			○	○

When interrupt conditions described in table 15.3 are 1 and the I bit in CCR is 0, the CPU executes an interrupt exception processing. Interrupt sources should be cleared in the exc processing. TDRE and TEND are automatically cleared to 0 by writing the transmit data ICDRT. RDRF are automatically cleared to 0 by reading ICDRR. TDRE is set to 1 again same time when transmit data is written to ICDRT. When TDRE is cleared to 0, then an e data of one byte may be transmitted.

Figure 15.21 shows the timing of the bit synchronous circuit and table 15.4 shows the ti
SCL output changes from low to Hi-Z then SCL is monitored.

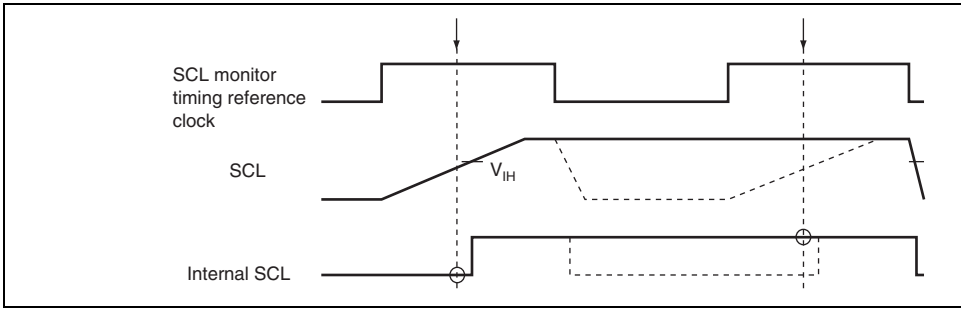


Figure 15.21 Timing of Bit Synchronous Circuit

Table 15.4 Time for Monitoring SCL

CKS3	CKS2	Time for Monitoring SCL
0	0	7.5 tcyc
	1	19.5 tcyc
1	0	17.5 tcyc
	1	41.5 tcyc

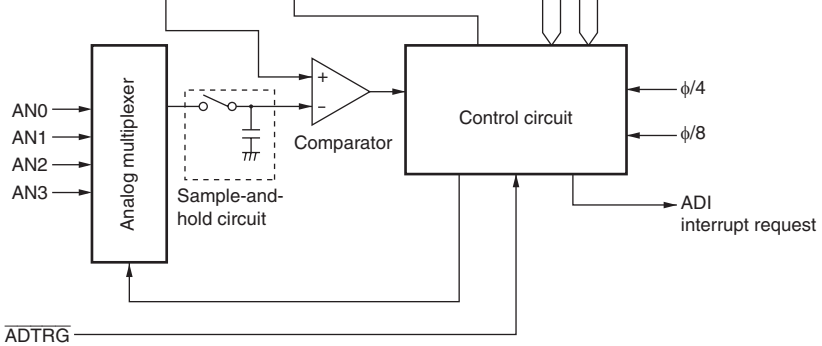
Circuit, by the load of the SCL bus (load capacitance or pull-up resistance)

2. When the bit synchronous circuit is activated by extending the low period of eighth an clocks, that is driven by the slave device

15.7.2 WAIT Setting in I²C Bus Mode Register (ICMR)

If the WAIT bit is set to 1, and the SCL signal is driven low for two or more transfer cloc slave device at the eighth and ninth clocks, the high period of ninth clock may be shorten avoid this, set the WAIT bit in ICMR to 0.

- Conversion time: At least 7 µs per channel (at 10 MHz operation)
- Two operating modes
  - Single mode: Single-channel A/D conversion
  - Scan mode: Continuous A/D conversion on 1 to 4 channels
- Four data registers
  - Conversion results are held in a 16-bit data register for each channel
- Sample and hold function
- Two conversion start methods
  - Software
  - External trigger signal
- Interrupt request
  - An A/D conversion end interrupt request (ADI) can be generated



[Legend]
ADCR: A/D control register
ADCSR: A/D control/status register
ADDRA: A/D data register A
ADDRB: A/D data register B
ADDRC: A/D data register C
ADDRD: A/D data register D

Figure 16.1 Block Diagram of A/D Converter

Analog input pin 2	AN2	Input	
Analog input pin 3	AN3	Input	
A/D external trigger input pin	$\overline{\text{ADTRG}}$	Input	External trigger input pin for sta conversion

16.3 Register Description

The A/D converter has the following registers.

- A/D data register A (ADDRA)
- A/D data register B (ADDRB)
- A/D data register C (ADDRC)
- A/D data register D (ADDRD)
- A/D control/status register (ADCSR)
- A/D control register (ADCR)

16.3.1 A/D Data Registers A to D (ADDRA to ADDRD)

There are four 16-bit read-only ADDR registers; ADDRA to ADDRD, used to store the A/D conversion. The ADDR registers, which store a conversion result for each channel, in table 16.2.

The converted 10-bit data is stored in bits 6 to 15. The lower 6 bits are always read as 0.

The data bus between the CPU and the A/D converter is 8 bits wide. The upper byte can directly from the CPU, however the lower byte should be read via a temporary register. temporary register contents are transferred from the ADDR when the upper byte data is

16.3.2 A/D Control/Status Register (ADCSR)

ADCSR consists of the control bits and conversion end status bits of the A/D converter.

Bit	Bit Name	Initial Value	R/W	Description
7	ADF	0	R/W	A/D End Flag [Setting conditions] • When A/D conversion ends in single mode • When A/D conversion ends on all the channe selected in scan mode [Clearing condition] • When 0 is written after reading ADF = 1
6	ADIE	0	R/W	A/D Interrupt Enable A/D conversion end interrupt (ADI) request enab ADF when 1 is set
5	ADST	0	R/W	A/D Start Setting this bit to 1 starts A/D conversion. In sing this bit is cleared to 0 automatically when conver the specified channel is complete. In scan mode, conversion continues sequentially on the specifie channels until this bit is cleared to 0 by software, or a transition to standby mode.

				1: Conversion time = 70 states (max.)	
				Clear the ADST bit to 0 before switching the co time.	
2	CH2	0	R/W	Channel Select 0 to 2	
1	CH1	0	R/W	Select analog input channels.	
0	CH0	0	R/W	When SCAN = 0	When SCAN = 1
				000: AN0	000: AN0
				001: AN1	001: AN0 to AN1
				010: AN2	010: AN0 to AN2
				011: AN3	011: AN0 to AN3
				Note: When executing the A/D conversion throu or AN2, do not set the VDDII bit in LVDCI is set, the A/D conversion accuracy is not guaranteed.	

				the external trigger pin (ADTRG) conforms to the bit in the interrupt edge select register 2 (IEGR2)
6 to 4	—	All 1	—	Reserved These bits are always read as 1.
3, 2	—	All 0	R/W	Reserved Although these bits are readable/writable, they s be set to 1.
1	—	1	R/W	Reserved This bit is always read as 1.
0	—	0	R/W	Reserved Although this bit is readable/writable, it should no to 1.

channel as follows:

1. A/D conversion is started from the first channel when the ADST bit in ADCSR is s
   according to software or external trigger input.
2. When A/D conversion is completed, the result is transferred to the corresponding A
   register to the channel.
3. On completion of conversion, the ADF bit in ADCSR is set to 1. If the ADIE bit is
   this time, an ADI interrupt request is generated.
4. The ADST bit remains set to 1 during A/D conversion. When A/D conversion ends
   ADST bit is automatically cleared to 0 and the A/D converter enters the wait state.

16.4.2 Scan Mode

In scan mode, A/D conversion is performed sequentially for the analog input on the spec
channels (four channels maximum) as follows:

1. When the ADST bit is set to 1 by software, or external trigger input, A/D conversion
   the first channel in the group.
2. When A/D conversion for each channel is completed, the result is sequentially transf
   the A/D data register corresponding to each channel.
3. When conversion of all the selected channels is completed, the ADF flag in ADCSR
   If the ADIE bit is set to 1 at this time, an ADI interrupt is requested. Conversion of t
   channel in the group starts again.
4. The ADST bit is not automatically cleared to 0. Steps [2] to [3] are repeated as long
   ADST bit remains set to 1. When the ADST bit is cleared to 0, A/D conversion stops

In scan mode, the values given in table 16.3 apply to the first conversion time. In the secc
subsequent conversions, the conversion time is 128 states (fixed) when CKS = 0 and 66 s
(fixed) when CKS = 1.

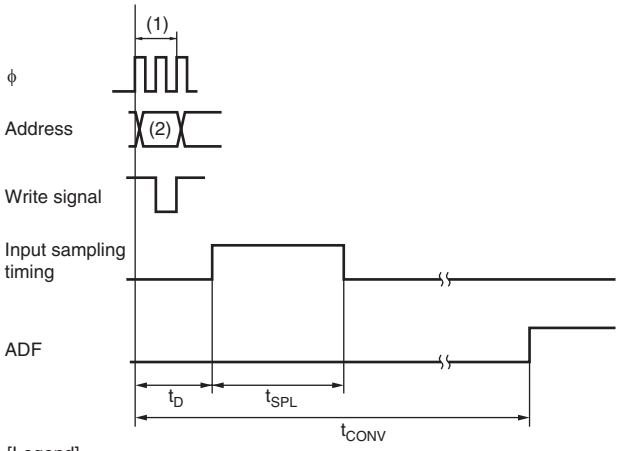


Figure 16.2 A/D Conversion Timing

16.4.4 External Trigger Input Timing

A/D conversion can also be started by an external trigger input. When the TRGE bit is s
ADCR, external trigger input is enabled at the $\overline{\text{ADTRG}}$ pin. A falling edge at the $\overline{\text{ADTR}}$
pin sets the ADST bit to 1 in ADCSR, starting A/D conversion. Other operations, in bot
and scan modes, are the same as when the bit ADST has been set to 1 by software. Figu
shows the timing.

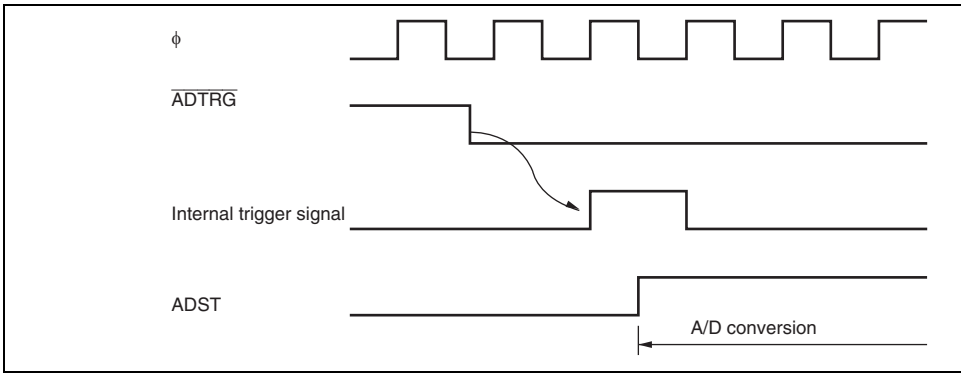


Figure 16.3 External Trigger Input Timing

when the digital output changes from the minimum voltage value 0000000000 to 000
(see figure 16.5).

- Full-scale error

  The deviation of the analog input voltage value from the ideal A/D conversion charac
  when the digital output changes from 1111111110 to 1111111111 (see figure 16.5).

- Nonlinearity error

  The error with respect to the ideal A/D conversion characteristics between zero voltag
  full-scale voltage. Does not include offset error, full-scale error, or quantization error.

- Absolute accuracy

  The deviation between the digital value and the analog input value. Includes offset err
  scale error, quantization error, and nonlinearity error.

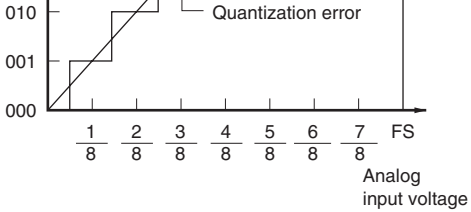


Figure 16.4 A/D Conversion Accuracy Definitions (1)

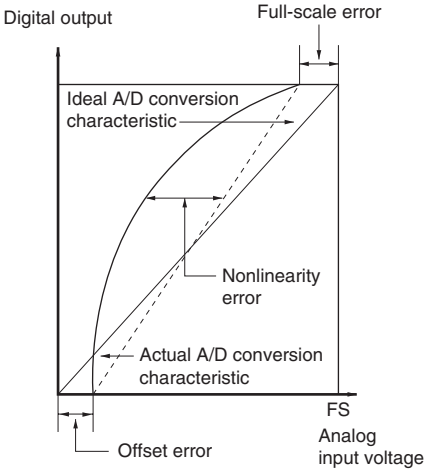


Figure 16.5 A/D Conversion Accuracy Definitions (2)

input resistance of 10 kΩ, and the signal source impedance is ignored. However, as a low filter effect is obtained in this case, it may not be possible to follow an analog signal with differential coefficient (e.g., 5 mV/µs or greater) (see figure 16.6). When converting a hig analog signal or converting in scan mode, a low-impedance buffer should be inserted.

16.6.2 Influences on Absolute Accuracy

Adding capacitance results in coupling with GND, and therefore noise in GND may adve affect absolute accuracy. Be sure to make the connection to an electrically stable GND.

Care is also required to ensure that filter circuits do not interfere with digital signals or ac antennas on the mounting board.

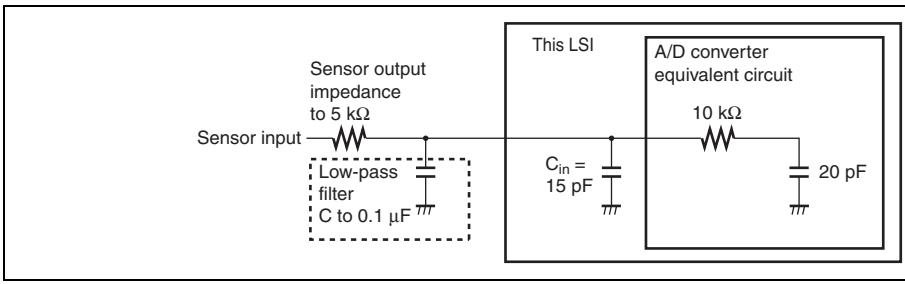


Figure 16.6 Analog Input Circuit Example

detection) and LVDR (reset by low voltage detection) circuits.

This circuit is used to prevent abnormal operation (program runaway) from occurring du
power supply voltage fall and to recreate the state before the power supply voltage fall w
power supply voltage rises again.

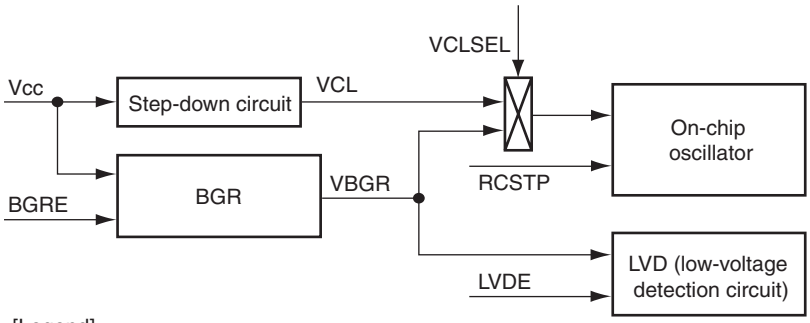
Even if the power supply voltage falls, the unstable state when the power supply voltage
below the guaranteed operating voltage can be removed by entering standby mode when
exceeding the guaranteed operating voltage and during normal operation. Thus, system s
can be improved. If the power supply voltage falls more, the reset state is automatically
the power supply voltage rises again, the reset state is held for a specified period, then ac
is automatically entered.

Figure 17.2 is a block diagram of the power-on reset circuit and the low-voltage detectio

voltage falls below a given value.

LVDI: Monitors the power-supply voltage, and generates an interrupt when the voltag
below or rises above respective given values.

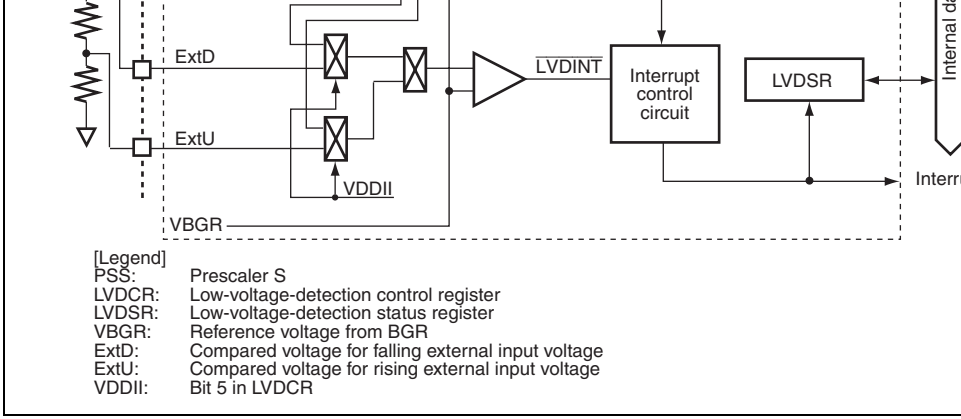
Two detection levels for reset generation voltage are available: when only the LVDR
used, or when the LVDI and LVDR circuits are both used.



[Legend]

Vcc:	Power supply
BGRE:	BGR circuit enable signal
VCL:	Internal power supply generated from Vcc by the step-down circuit
VBGR:	Reference voltage from BGR
VCLSEL:	Select signal for the source of the on-chip oscillator power supply
RCSTP:	On-chip oscillator stop signal
LVDE:	LVD enable signal

Figure 17.1 Block Diagram around BGR



[Legend]
PSS: Prescaler S
LVDCR: Low-voltage-detection control register
LVDSR: Low-voltage-detection status register
VBGR: Reference voltage from BGR
ExtD: Compared voltage for falling external input voltage
ExtU: Compared voltage for rising external input voltage
VDDII: Bit 5 in LVDCR

Figure 17.2 Block Diagram of Power-On Reset Circuit and Low-Voltage Detectio

compared voltage of the LVDI circuit, sets the detection levels for the LVDR circuit, ena
disables the LVDR circuit, and enables or disables generation of an interrupt when the po
supply voltage rises above or falls below the respective levels.

Table 17.1 shows the relationship between the LVDCR settings and functions to be selec
LVDCR should be set according to table 17.1.

Bit	Bit Name	Initial Value	R/W	Description
7	LVDE	1*	R/W	LVD Enable 0: Low-voltage detection circuit is not used (stan mode) 1: Low-voltage detection circuit is used
6	BGRE	1*	R/W	BGR Enable 0: BGR circuit is not used (standby mode) 1: BGR circuit is used
5	VDDII	1*	R/W	LVDR External Compared Voltage Input Inhibit 0: Use external voltage as LVDI compared volta 1: Use internal voltage as LVDI compared voltag
4	—	1	—	Reserved This bit is always read as 1 and cannot be modif

				0: Disables an LVDR
				1: Enables an LVDR
1	LVDDE	0	R/W	Voltage-Fall-Interrupt Enable
				0: Interrupt on the power-supply voltage falling
				1: Interrupt on the power-supply voltage falling
0	LVDUE	0	R/W	Voltage-Rise-Interrupt Enable
				0: Interrupt on the power-supply voltage rising c
				1: Interrupt on the power-supply voltage rising e

Note: * Not initialized by an LVDR but initialized by a power-on reset or a watchdog t

Table 17.1 LVDCR Settings and Select Functions

LVDCR Settings							Select Functions		
LVDE	**BGRE**	**VDDII**	**LVDSEL**	**LVDRE**	**LVDDE**	**LVDUE**	**Power-On Reset**	**LVDR**	**Low-Voltage-Detection Fall Interrupt**
0	*[1]	*[2]	*[2]	*[2]	*[2]	*[2]	√	—	—
1	1	*[1]	1	1	0	0	√	√	—
1	1	*[1]	0	0	1	0	√	—	√
1	1	*[1]	0	0	1	1	√	—	√
1	1	*[1]	0	1	1	1	√	√	√

Notes: 1. Set these bits if necessary.
2. Settings are ignored.

				• When the power-supply voltage falls below V (Typ. = 3.7 V) [Clearing condition] • When writing 0 to this bit after reading it as 1
0	LVDUF	0*	R/W	LVD Power-Supply Voltage Rise Flag [Setting condition] • When the power supply voltage falls below V while the LVDUE bit in LVDCR is set to 1 and rises above Vint (U) (Typ. = 4.0 V) before fall below Vreset1 (Typ. = 2.3 V) [Clearing condition] • When writing 0 to this bit after reading it as 1

Note: * Initialized by an LVDR.

noise filter circuit which removes noise with less than 400 ns (Typ.) is included to preve
incorrect operation of this LSI caused by noise on the $\overline{\text{RES}}$ signal.

To achieve stable operation of this LSI, the power supply needs to rise to its full level ar
within the specified time. The maximum time required for the power supply to rise and s
(t_{PWON}) is determined by the oscillation frequency (f_{OSC}) and capacitance which is connect
pin ($C_{\overline{RES}}$). Where t_{PWON} is assumed to be the time required to reach 90 % of the full level
power supply, the power supply circuit should be designed to satisfy the following form

$$t_{PWON}\ (\text{ms}) \leq 90 \times C_{\overline{RES}}\ (\mu\text{F}) + 162/f_{OSC}\ (\text{MHz})$$

($t_{PWON} \leq 3000$ ms, $C_{\overline{RES}} \geq 0.22$ µF, and $f_{OSC} = 10$ in 2-MHz to 10-MHz operatio

Note that the power supply voltage (Vcc) must fall below Vpor = 100 mV to remove cha
$\overline{\text{RES}}$ pin. After that, it can be risen. To remove charge on the $\overline{\text{RES}}$ pin, it is recommende
diode should be placed to Vcc. If the power supply voltage (Vcc) rises from the point ab
a power-on reset may not occur.

Figure 17.3 Operational Timing of Power-On Reset Circuit

17.3.2 Low-Voltage Detection Circuit

(1) LVDR (Reset by Low Voltage Detection) Circuit

Figure 17.4 shows the timing of the operation of the LVDR circuit. The LVDR circuit is
after a power-on reset is released. To cancel the LVDR circuit, first the LVDRE bit in LV
should be cleared to 0 and then the LVDE bit in LVDCR and, if necessary, the BGRE bit
be cleared to 0. The LVDE and the BGRE bits must not be cleared to 0 simultaneously w
LVDRE bit because incorrect operation may occur. To restart the LVDR circuit, set the L
and the BGRE bit to 1, wait for 50 µs (t_{LVDON}) given by a software timer until the reference
and the low-voltage-detection power supply have settled, then set the LVDRE bit to 1. Af
the output settings of ports must be made.

When the power-supply voltage falls below the Vreset voltage (2.3 V or 3.6 V (Typ.)), th
circuit clears the $\overline{\text{LVDRES}}$ signal to 0, and resets prescaler S. The low-voltage detection r
remains in place until a power-on reset is generated. When the power-supply voltage rises
the Vreset voltage again, prescaler S starts counting. It counts 131,072 clock (ϕ) cycles, a
releases the internal reset signal. In this case, the LVDE, BGRE, VDDII, LVDSEL, and I
bits in LVDCR are not initialized.

Note that if the power supply voltage (Vcc) falls below $V_{LVDRmin}$ = 1.0 V and then rises fro
point, the low-voltage detection reset may not occur.

If the power supply voltage (Vcc) falls below Vpor = 100 mV, a power-on reset occurs.

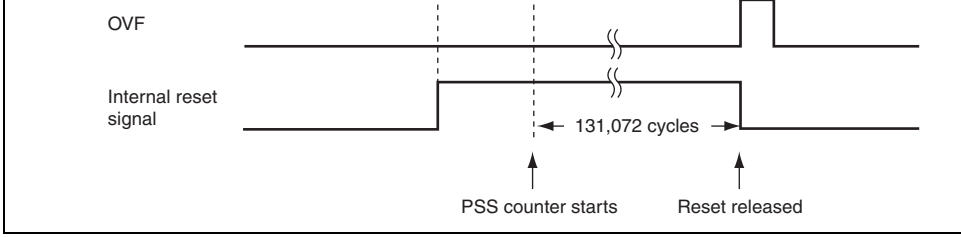


Figure 17.4 Operating Timing of LVDR Circuit

(2) Low Voltage Detection Interrupt (LVDI) Circuit (When Internally Generated Voltage is used for Detection)

Figure 17.5 shows the timing of the operation of the LVDI circuit.

The LVDI circuit is enabled after a power-on reset, however, the interrupt request is disa
enable the LVDI, the LVDDF bit and LVDUF bit in LVDSR must be cleared to 0 and th
LVDDE bit or LVDUE bit in LVDCR must be set to 1. After that, the output settings of
must be made.

To cancel the LVDI, follow the procedures written in section 17.3.2 (4), Operating Proc
Enabling/Disabling LVDR and LVDI Circuits.

To restart the LVDI circuit after standby mode, set the LVDE bit to 1, write 1 to VDDII
necessary), and wait for 50 µs (t_{LVDON}) given by a software timer until the reference volta
low-voltage detection power supply have settled. Then, clear the LVDDF and LVDUF b
and set the LVDDE or the LVDUE bit to 1. After that, the output settings of ports must

When the power-supply voltage falls below Vint (D) (Typ. = 3.7 V) voltage, the LVDI c
clears the $\overline{\text{LVDINT}}$ signal to 0 and sets the LVDDF bit to 1. If the LVDDE bit is 1 at thi
IRQ0 interrupt request is generated. In this case, the necessary data must be saved in the

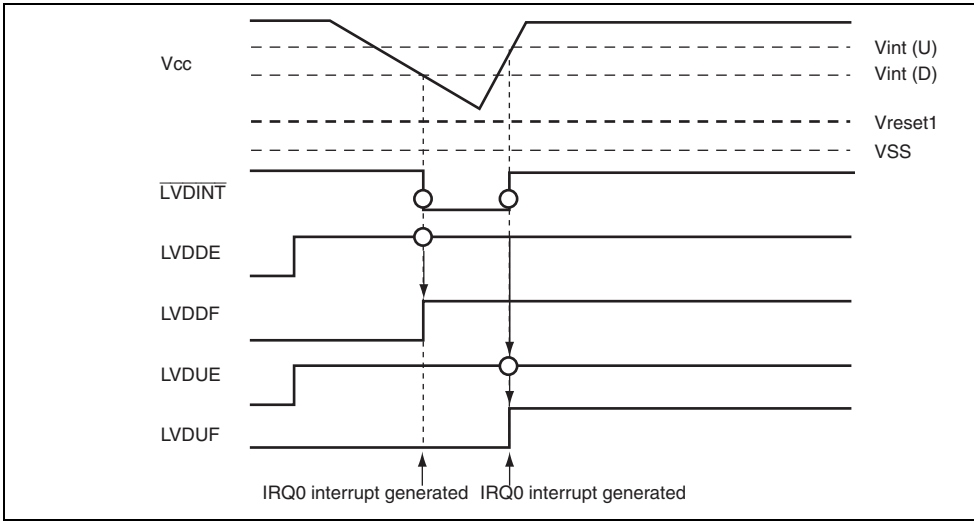


Figure 17.5 Operational Timing of LVDI Circuit

To cancel the LVDI, follow the procedures written in section 17.3.2 (4), Operating Proc
Enabling/Disabling LVDR and LVDI Circuits.

When the external comparison voltage of ExtD pin falls below the Vexd (D) (Typ. = 1.1
voltage, the LVDI clears the $\overline{\text{LVDINT}}$ signal to 0 and sets the LVDDF bit in LVDSR to
LVDDE bit is 1 at this time, an IRQ0 interrupt request is generated. In this case, the nec
data must be saved in the external EEPROM, and a transition to standby mode or subsle
must be made. Until this processing is completed, the power supply voltage must be hig
the lower limit of the guaranteed operating voltage.

When the power-supply voltage does not fall below the Vreset1 (Typ. = 2.3 V) voltage a
input voltage of the ExtU pin rises above Vexd (Typ. = 1.15 V) voltage, the LVDI circu
$\overline{\text{LVDINT}}$ signal to 1. If the LVDUE bit is 1 at this time, the LVDUF bit in LVDSR is se
an IRQ0 interrupt request is generated.

If the power supply voltage falls below the Vreset1 (Typ. = 2.3 V) voltage, this LSI ente
voltage detection reset operation. When the voltages input on the ExtU and ExtD pins ar
the compared voltage, ensure to use the LVDR (reset detection voltage: Typ. = 2.3 V) ci

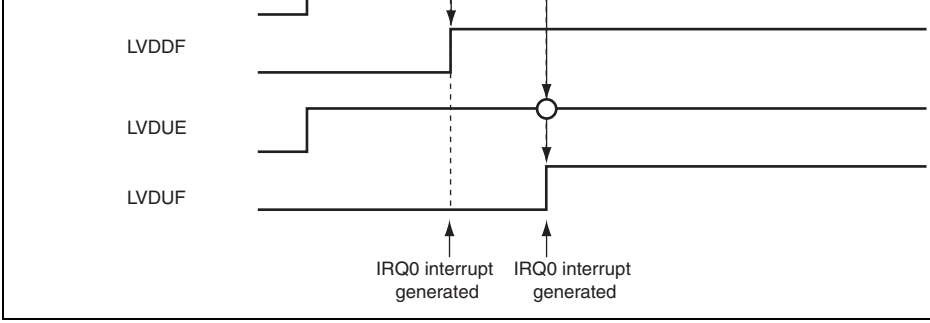


Figure 17.6 Operational Timing of LVDI Circuit (When Compared Voltage is I through ExtU and ExtD Pins)

When the voltages input on the ExtU and ExtD pins are used as the compared volta
the LVDDII bit to 0.

3. Wait for 50 µs (t_{LVDON}) given by a software timer until the reference voltage and the
voltage-detection power supply have settled. Then, clear the LVDDF and LVDUF b
LVDSR to 0 and set the LVDRE, LVDDE, and LVDUE bits in LVDCR to 1, if nec

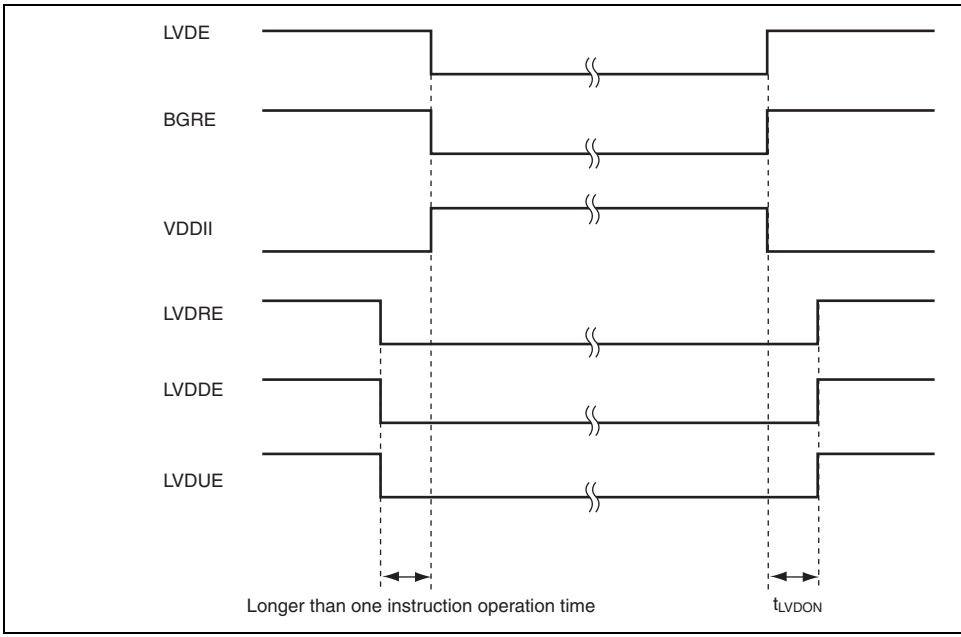


Figure 17.7 Timing for Enabling/Disabling of Low-Voltage Detection Circ

18.1 When Using Internal Power Supply Step-Down Circuit

Connect the external power supply to the V_{cc} pin, and connect a capacitance of approxin
μF between V_{cl} and V_{ss}, as shown in figure 18.1. The internal step-down circuit is made
simply by adding this external circuit. In the external circuit interface, the external powe
voltage connected to V_{cc} and the GND potential connected to V_{ss} are the reference levels
example, for port input/output levels, the V_{cc} level is the reference for the high level, and
level is that for the low level. The A/D converter analog power supply is not affected by
internal step-down circuit.

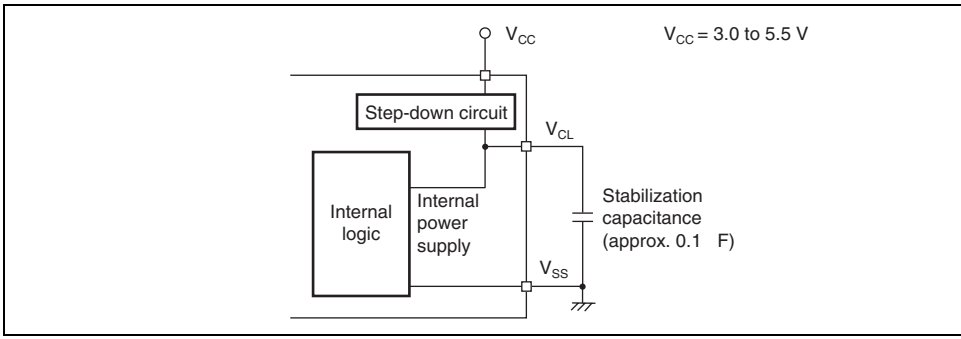


Figure 18.1 Power Supply Connection when Internal Step-Down Circuit is U

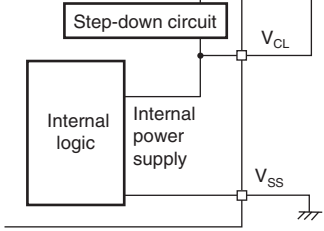


Figure 18.2 Power Supply Connection when Internal Step-Down Circuit is Not

- The number of access states is indicated.

2. Register bits
- Bit configurations of the registers are described in the same order as the register addr
- Reserved bits are indicated by — in the bit name column.
- When registers consist of 16 bits, bits are described from the MSB side.

3. Register states in each operating mode
- Register states are described in the same order as the register addresses.
- The register states described here are for the basic operating modes. If there is a spec
  for an on-chip peripheral module, refer to the section on that on-chip peripheral mod

Low-voltage-detection status register	LVDSR	8	H'F731	Low-voltage detection circuit	8
Clock control status register	CKCSR	8	H'F734	Clock oscillator	8
RC control register	RCCR	8	H'F735	On-chip oscillator	8
RC trimming data protect register	RCTRMDPR	8	H'F736	On-chip oscillator	8
RC trimming data register	RCTRMDR	8	H'F737	On-chip oscillator	8
I^2C bus control register 1	ICCR1	8	H'F748	IIC2	8
I^2C bus control register 2	ICCR2	8	H'F749	IIC2	8
I^2C bus mode register	ICMR	8	H'F74A	IIC2	8
I^2C bus interrupt enable register	ICIER	8	H'F74B	IIC2	8
I^2C bus status register	ICSR	8	H'F74C	IIC2	8
Slave address register	SAR	8	H'F74D	IIC2	8
I^2C bus transmit data register	ICDRT	8	H'F74E	IIC2	8
I^2C bus receive data register	ICDRR	8	H'F74F	IIC2	8
Timer mode register B1	TMB1	8	H'F760	Timer B1	8
Timer counter B1/Timer load register B1	TCB1(R)/ TLB1 (W)	8	H'F761	Timer B1	8
Timer mode register W	TMRW	8	H'FF80	Timer W	8
Timer control register W	TCRW	8	H'FF81	Timer W	8
Timer interrupt enable register W	TIERW	8	H'FF82	Timer W	8
Timer status register W	TSRW	8	H'FF83	Timer W	8

Flash memory control register 1	FLMCR1	8	H'FF90	ROM	8
Flash memory control register 2	FLMCR2	8	H'FF91	ROM	8
Erase block register 1	EBR1	8	H'FF93	ROM	8
Flash memory enable register	FENR	8	H'FF9B	ROM	8
Timer control register V0	TCRV0	8	H'FFA0	Timer V	8
Timer control/status register V	TCSRV	8	H'FFA1	Timer V	8
Timer constant register A	TCORA	8	H'FFA2	Timer V	8
Timer constant register B	TCORB	8	H'FFA3	Timer V	8
Timer counter V	TCNTV	8	H'FFA4	Timer V	8
Timer control register V1	TCRV1	8	H'FFA5	Timer V	8
Serial mode register	SMR	8	H'FFA8	SCI3	8
Bit rate register	BRR	8	H'FFA9	SCI3	8
Serial control register 3	SCR3	8	H'FFAA	SCI3	8
Transmit data register	TDR	8	H'FFAB	SCI3	8
Serial status register	SSR	8	H'FFAC	SCI3	8
Receive data register	RDR	8	H'FFAD	SCI3	8
Sampling mode register	SPMR	8	H'FFAE	SCI3	8
A/D data register A	ADDRA	16	H'FFB0	A/D converter	8
A/D data register B	ADDRB	16	H'FFB2	A/D converter	8
A/D data register C	ADDRC	16	H'FFB4	A/D converter	8
A/D data register D	ADDRD	16	H'FFB6	A/D converter	8
A/D control/status register	ADCSR	8	H'FFB8	A/D converter	8
A/D control register	ADCR	8	H'FFB9	A/D converter	8
Timer control/status register WD	TCSRWD	8	H'FFC0	WDT*2	8

Break data register L	BDRL	8	H'FFCD	Address break	8
Port pull-up control register 1	PUCR1	8	H'FFD0	I/O port	8
Port pull-up control register 5	PUCR5	8	H'FFD1	I/O port	8
Port data register 1	PDR1	8	H'FFD4	I/O port	8
Port data register 2	PDR2	8	H'FFD5	I/O port	8
Port data register 5	PDR5	8	H'FFD8	I/O port	8
Port data register 7	PDR7	8	H'FFDA	I/O port	8
Port data register 8	PDR8	8	H'FFDB	I/O port	8
Port data register B	PDRB	8	H'FFDD	I/O port	8
Port data register C	PDRC	8	H'FFDE	I/O port	8
Port mode register 1	PMR1	8	H'FFE0	I/O port	8
Port mode register 5	PMR5	8	H'FFE1	I/O port	8
Port control register 1	PCR1	8	H'FFE4	I/O port	8
Port control register 2	PCR2	8	H'FFE5	I/O port	8
Port control register 5	PCR5	8	H'FFE8	I/O port	8
Port control register 7	PCR7	8	H'FFEA	I/O port	8
Port control register 8	PCR8	8	H'FFEB	I/O port	8
Port control register C	PCRC	8	H'FFEE	I/O port	8
System control register 1	SYSCR1	8	H'FFF0	Power-down	8
System control register 2	SYSCR2	8	H'FFF1	Power-down	8
Interrupt edge select register 1	IEGR1	8	H'FFF2	Interrupts	8
Interrupt edge select register 2	IEGR2	8	H'FFF3	Interrupts	8
Interrupt enable register 1	IENR1	8	H'FFF4	Interrupts	8
Interrupt enable register 2	IENR2	8	H'FFF5	Interrupts	8

2. WDT: Watchdog timer

RCCR	RCSTP	FSEL	VCLSEL	—	—	—	RCPSC1	RCPSC0	C
RCTRMDPR	WRI	PRWE	LOCKDW	TRMDRWE	—	—	—	—	
RCTRMDR	TRMD7	TRMD6	TRMD5	TRMD4	TRMD3	TRMD2	TRMD1	TRMD0	
ICCR1	ICE	RCVD	MST	TRS	CKS3	CKS2	CKS1	CKS0	I
ICCR2	BBSY	SCP	SDAO	SDAOP	SCLO	—	IICRST	—	
ICMR	MLS	WAIT	—	—	BCWP	BC2	BC1	BC0	
ICIER	TIE	TEIE	RIE	NAKIE	STIE	ACKE	ACKBR	ACKBT	
ICSR	TDRE	TEND	RDRF	NACKF	STOP	AL/OVE	AAS	ADZ	
SAR	SVA6	SVA5	SVA4	SVA3	SVA2	SVA1	SVA0	FS	
ICDRT	ICDRT7	ICDRT6	ICDRT5	ICDRT4	ICDRT3	ICDRT2	ICDRT1	ICDRT0	
ICDRR	ICDRR7	ICDRR6	ICDRR5	ICDRR4	ICDRR3	ICDRR2	ICDRR1	ICDRR0	
TMB1	TMB17	—	—	—	—	TMB12	TMB11	TMB10	T
TCB1 (R)/ TLB1 (W)	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0	
TMRW	CTS	—	BUFEB	BUFEA	—	PWMD	PWMC	PWMB	T
TCRW	CCLR	CKS2	CKS1	CKS0	TOD	TOC	TOB	TOA	
TIERW	OVIE	—	—	—	IMIED	IMIEC	IMIEB	IMIEA	
TSRW	OVF	—	—	—	IMFD	IMFC	IMFB	IMFA	
TIOR0	—	IOB2	IOB1	IOB0	—	IOA2	IOA1	IOA0	
TIOR1	—	IOD2	IOD1	IOD0	—	IOC2	IOC1	IOC0	
TCNT	TCNT15	TCNT14	TCNT13	TCNT12	TCNT11	TCNT10	TCNT9	TCNT8	
	TCNT7	TCNT6	TCNT5	TCNT4	TCNT3	TCNT2	TCNT1	TCNT0	
GRA	GRA15	GRA14	GRA13	GRA12	GRA11	GRA10	GRA9	GRA8	
	GRA7	GRA6	GRA5	GRA4	GRA3	GRA2	GRA1	GRA0	
GRB	GRB15	GRB14	GRB13	GRB12	GRB11	GRB10	GRB9	GRB8	
	GRB7	GRB6	GRB5	GRB4	GRB3	GRB2	GRB1	GRB0	

TCSRV	CMFB	CMFA	OVF	—	OS3	OS2	OS1	OS0	
TCORA	TCORA7	TCORA6	TCORA5	TCORA4	TCORA3	TCORA2	TCORA1	TCORA0	
TCORB	TCORB7	TCORB6	TCORB5	TCORB4	TCORB3	TCORB2	TCORB1	TCORB0	
TCNTV	TCNTV7	TCNTV6	TCNTV5	TCNTV4	TCNTV3	TCNTV2	TCNTV1	TCNTV0	
TCRV1	—	—	—	TVEG1	TVEG0	TRGE	—	ICKS0	
SMR	COM	CHR	PE	PM	STOP	MP	CKS1	CKS0	S
BRR	BRR7	BRR6	BRR5	BRR4	BRR3	BRR2	BRR1	BRR0	
SCR3	TIE	RIE	TE	RE	MPIE	TEIE	CKE1	CKE0	
TDR	TDR7	TDR6	TDR5	TDR4	TDR3	TDR2	TDR1	TDR0	
SSR	TDRE	RDRF	OER	FER	PER	TEND	MPBR	MPBT	
RDR	RDR7	RDR6	RDR5	RDR4	RDR3	RDR2	RDR1	RDR0	
SPMR	—	—	—	—	—	STDSPM	—	—	
ADDRA	AD9	AD8	AD7	AD6	AD5	AD4	AD3	AD2	A
	AD1	AD0	—	—	—	—	—	—	
ADDRB	AD9	AD8	AD7	AD6	AD5	AD4	AD3	AD2	
	AD1	AD0	—	—	—	—	—	—	
ADDRC	AD9	AD8	AD7	AD6	AD5	AD4	AD3	AD2	
	AD1	AD0	—	—	—	—	—	—	
ADDRD	AD9	AD8	AD7	AD6	AD5	AD4	AD3	AD2	
	AD1	AD0	—	—	—	—	—	—	
ADCSR	ADF	ADIE	ADST	SCAN	CKS	CH2	CH1	CH0	
ADCR	TRGE	—	—	—	—	—	—	—	
TCSRWD	B6WI	TCWE	B4WI	TCSRWE	B2WI	WDON	B0WI	WRST	W
TCWD	TCWD7	TCWD6	TCWD5	TCWD4	TCWD3	TCWD2	TCWD1	TCWD0	
TMWD	—	—	—	—	CKS3	CKS2	CKS1	CKS0	

PDR1	P17	—	—	P14	—	—	—	—	
PDR2	—	—	—	—	—	P22	P21	P20	
PDR5	P57	P56	P55	—	—	—	—	—	
PDR7	—	P76	P75	P74	—	—	—	—	
PDR8	—	—	—	P84	P83	P82	P81	P80	
PDRB	—	—	—	—	PB3	PB2	PB1	PB0	
PDRC	—	—	—	—	—	—	PC1	PC0	
PMR1	IRQ3	—	—	IRQ0	—	—	TXD	—	
PMR5	—	—	WKP5	—	—	—	—	—	
PCR1	PCR17	—	—	PCR14	—	—	—	—	
PCR2	—	—	—	—	—	PCR22	PCR21	PCR20	
PCR5	PCR57	PCR56	PCR55	—	—	—	—	—	
PCR7	—	PCR76	PCR75	PCR74	—	—	—	—	
PCR8	—	—	—	PCR84	PCR83	PCR82	PCR81	PCR80	
PCRC	—	—	—	—	—	—	PCRC1	PCRC0	
SYSCR1	SSBY	STS2	STS1	STS0	—	—	—	—	Pow
SYSCR2	SMSEL	—	DTON	MA2	MA1	MA0	—	—	
IEGR1	—	—	—	—	IEG3	—	—	IEG0	Inte
IEGR2	—	—	WPEG5	—	—	—	—	—	
IENR1	IENDT	—	IENWP	—	IEN3	—	—	IEN0	
IENR2	—	—	IENTB1	—	—	—	—	—	
IRR1	IRRDT	—	—	—	IRRI3	—	—	IRRI0	
IRR2	—	—	IRRTB1	—	—	—	—	—	
IWPR	—	—	IWPF5	—	—	—	—	—	
MSTCR1	—	MSTIIC	MSTS3	MSTAD	MSTWD	MSTTW	MSTTV	—	Pow
MSTCR2	—	—	—	MSTTB1	—	—	—	—	

Note: * WDT:Watchdog timer

ICCR1	Initialized	—	—	—	—	IIC2
ICCR2	Initialized	—	—	—	—	
ICMR	Initialized	—	—	—	—	
ICIER	Initialized	—	—	—	—	
ICSR	Initialized	—	—	—	—	
SAR	Initialized	—	—	—	—	
ICDRT	Initialized	—	—	—	—	
ICDRR	Initialized	—	—	—	—	
TMB1	Initialized	—	—	—	—	Timer B1
TCB1/TLB1	Initialized	—	—	—	—	
TMRW	Initialized	—	—	—	—	Timer W
TCRW	Initialized	—	—	—	—	
TIERW	Initialized	—	—	—	—	
TSRW	Initialized	—	—	—	—	
TIOR0	Initialized	—	—	—	—	
TIOR1	Initialized	—	—	—	—	
TCNT	Initialized	—	—	—	—	
GRA	Initialized	—	—	—	—	
GRB	Initialized	—	—	—	—	
GRC	Initialized	—	—	—	—	
GRD	Initialized	—	—	—	—	
FLMCR1	Initialized	—	—	Initialized	Initialized	ROM
FLMCR2	Initialized	—	—	Initialized	Initialized	
EBR1	Initialized	—	—	Initialized	Initialized	
FENR	Initialized	—	—	Initialized	Initialized	
TCRV0	Initialized	—	—	Initialized	Initialized	Timer V

SSR	Initialized	—	—	Initialized	Initialized	
RDR	Initialized	—	—	Initialized	Initialized	
SPMR	Initialized	—	—	Initialized	Initialized	
ADDRA	Initialized	—	—	Initialized	Initialized	A/D converter
ADDRB	Initialized	—	—	Initialized	Initialized	
ADDRC	Initialized	—	—	Initialized	Initialized	
ADDRD	Initialized	—	—	Initialized	Initialized	
ADCSR	Initialized	—	—	Initialized	Initialized	
ADCR	Initialized	—	—	Initialized	Initialized	
TCSRWD	Initialized	—	—	—	—	WDT*
TCWD	Initialized	—	—	—	—	
TMWD	Initialized	—	—	—	—	
ABRKCR	Initialized	—	—	—	—	Address Break
ABRKSR	Initialized	—	—	—	—	
BARH	Initialized	—	—	—	—	
BARL	Initialized	—	—	—	—	
BDRH	Initialized	—	—	—	—	
BDRL	Initialized	—	—	—	—	
PUCR1	Initialized	—	—	—	—	I/O port
PUCR5	Initialized	—	—	—	—	
PDR1	Initialized	—	—	—	—	
PDR2	Initialized	—	—	—	—	
PDR5	Initialized	—	—	—	—	
PDR7	Initialized	—	—	—	—	
PDR8	Initialized	—	—	—	—	
PDRB	Initialized	—	—	—	—	

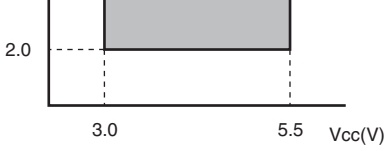
SYSCR1	Initialized	—	—	—	—	Power-down
SYSCR2	Initialized	—	—	—	—	
IEGR1	Initialized	—	—	—	—	Interrupts
IEGR2	Initialized	—	—	—	—	
IENR1	Initialized	—	—	—	—	
IENR2	Initialized	—	—	—	—	
IRR1	Initialized	—	—	—	—	
IRR2	Initialized	—	—	—	—	
IWPR	Initialized	—	—	—	—	
MSTCR1	Initialized	—	—	—	—	Power-down
MSTCR2	Initialized	—	—	—	—	

Note: — is not initialized

* WDT: Watchdog timer

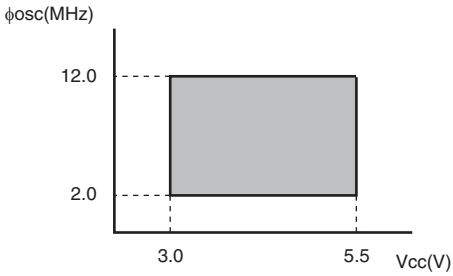
	Port B		-0.3 to AV_{CC} $+0.3$	V
Operating temperature		T_{opr}	–20 to +75	°C
Storage temperature		T_{stg}	–55 to +125	°C

Note: * Permanent damage may result if maximum ratings are exceeded. Normal ope should be under the conditions specified in Electrical Characteristics. Exceed values can result in incorrect operation and reduced reliability.



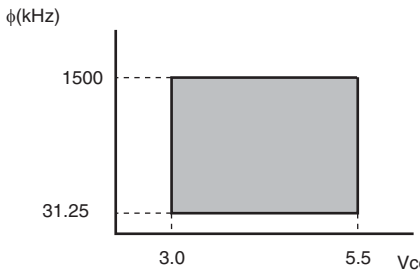
AVcc = 3.0 to 5.5 V
- Active mode
- Sleep mode

2. Power supply voltage and operating frequency range



AVcc = 3.0 to 5.5 V
- Active mode
- Sleep mode

(When MA2 = 0 in SYSCR2)



AVcc = 3.0 to 5.5 V
- Active mode
- Sleep mode

(When MA2 = 1 in SYSCR2

			Vcc = 3.0 to 5.5 V • Active mode • Sleep mode	

		FTIOA to FTIOD, SCK3, TRGV					
		RXD, SCL, SDA, P17, P14, P22 to P20, P57 to P55, P76 to P74, P84 to P80, PC1, PC0	V_{cc} = 4.0 V to 5.5 V	$V_{cc} \times 0.7$	—	$V_{cc} + 0.3$	V
				$V_{cc} \times 0.8$	—	$V_{cc} + 0.3$	V
		PB3 to PB0	AV_{cc} = 4.0 V to 5.5 V	$AV_{cc} \times 0.7$	—	$AV_{cc} + 0.3$	V
			AV_{cc} = 3.0 V to 5.5 V	$AV_{cc} \times 0.8$	—	$AV_{cc} + 0.3$	V
		OSC1	V_{cc} = 4.0 V to 5.5 V	$V_{cc} - 0.5$	—	$V_{cc} + 0.3$	V
				$V_{cc} - 0.3$		$V_{cc} + 0.3$	V
Input low voltage	V_{IL}	$\overline{RES}$, $\overline{NMI}$, $\overline{WKP5}$, $\overline{IRQ0}$, $\overline{IRQ3}$, $\overline{ADTRG}$, TMRIV, TMCIV, FTCI, FTIOA to FTIOD, SCK3, TRGV	V_{cc} = 4.0 V to 5.5 V	–0.3	—	$V_{cc} \times 0.2$	V
				–0.3	—	$V_{cc} \times 0.1$	V
		RXD, SCL, SDA, P17, P14, P22 to P20, P57 to P55, P76 to P74, P84 to P80, PC1, PC0	V_{cc} = 4.0 V to 5.5 V	–0.3	—	$V_{cc} \times 0.3$	V
				–0.3	—	$V_{cc} \times 0.2$	V
		PB3 to PB0	AV_{cc} = 4.0 V to 5.5 V	–0.3	—	$AV_{cc} \times 0.3$	V
			AV_{cc} = 3.0 V to 5.5 V	–0.3	—	$AV_{cc} \times 0.2$	
		OSC1	V_{cc} = 4.0 V to 5.5 V	–0.3	—	0.5	V
				–0.3	—	0.3	V

			$-I_{OH}$ = 0.1 mA				
Output low voltage	V_{OL}	P17, P14, P22 to P20, P57 to P55, P76 to P74, PC1, PC0	V_{CC} = 4.0 V to 5.5 V I_{OL} = 1.6 mA	—	—	0.6	V
			I_{OL} = 0.4 mA	—	—	0.4	V
		P84 to P80	V_{CC} = 4.0 V to 5.5 V I_{OL} = 20.0 mA	—	—	1.5	V
			V_{CC} = 4.0 V to 5.5 V I_{OL} = 10.0 mA	—	—	1.0	V
			V_{CC} = 4.0 V to 5.5 V I_{OL} = 1.6 mA	—	—	0.4	V
			I_{OL} = 0.4 mA	—	—	0.4	V
		SCL, SDA	V_{CC} = 4.0 V to 5.5 V I_{OL} = 6.0 mA	—	—	0.6	V
			I_{OL} = 3.0 mA	—	—	0.4	V
Input/ output leakage current	$\lvert I_{IL} \rvert$	OSC1, $\overline{\text{NMI}}$, $\overline{\text{WKP5}}$, $\overline{\text{IRQ0}}$, $\overline{\text{IRQ3}}$, $\overline{\text{ADTRG}}$, TRGV, TMRIV, TMCIV, FTCI, FTIOA to FTIOD, RXD, SCK3, SCL, SDA	V_{IN} = 0.5 V to (V_{CC} – 0.5 V)	—	—	1.0	µA
		P17, P14, P22 to P20, P57 to P55, P76 to P74, P84 to P80, PC1, PC0	V_{IN} = 0.5 V to (V_{CC} – 0.5 V)	—	—	1.0	µA
		PB3 to PB0	V_{IN} = 0.5 V to (AV_{CC} – 0.5 V)	—	—	1.0	µA

consumption								
			Active mode 1 V_{CC} = 3.0 V, f_{OSC} = 12 MHz	—	9.6	—	mA	P v
	I_{OPE2}	V_{CC}	Active mode 2 V_{CC} = 5.0 V, f_{OSC} = 12 MHz	—	2.0	2.5	mA	*
			Active mode 2 V_{CC} = 3.0 V, f_{OSC} = 12 MHz	—	1.5	—	mA	P v
Sleep mode current consumption	I_{SLEEP1}	V_{CC}	Sleep mode 1 V_{CC} = 5.0 V, f_{OSC} = 12 MHz	—	7.2	12.0	mA	*
			Sleep mode 1 V_{CC} = 3.0 V, f_{OSC} = 12 MHz	—	6.0	—	mA	P v
	I_{SLEEP2}	V_{CC}	Sleep mode 2 V_{CC} = 5.0 V, f_{OSC} = 12 MHz	—	1.8	2.2	mA	*
			Sleep mode 2 V_{CC} = 3.0 V, f_{OSC} = 12 MHz	—	1.4	—	mA	P v
Subsleep mode current consumption	I_{SUBSP}	V_{CC}	V_{CC} = 5.0 V LVDE = 0, BGRE = 0	—	—	5.0	µA	*
Standby mode current consumption	I_{STBY}	V_{CC}	LVDE = 0, BGRE = 0	—	—	5.0	µA	*

				resonator, and oi
Sleep mode 1	V_{CC}	Only timers operate	V_{CC}	oscillator
Sleep mode 2		Only timers operate (ϕ/64)		
Subsleep mode Standby mode	V_{CC}	CPU and timers both stop	V_{CC}	—

	SDA					
	P84 to P80		—	—	10.0	m
	SCL, SDA		—	—	6.0	m
Allowable output low current (total)	$\sum I_{OL}$	Output pins except P84 to P80, SCL, and SDA	V_{CC} = 4.0 V to 5.5 V	—	—	40.0	m
		P84 to P80, SCL, and SDA		—	—	80.0	m
		Output pins except P84 to P80, SCL, and SDA		—	—	20.0	m
		P84 to P80, SCL, and SDA		—	—	40.0	m
Allowable output high current (per pin)	$\lvert -I_{OH} \rvert$	Output pins except P56, P57	V_{CC} = 4.0 V to 5.5 V	—	—	4.0	m
				—	—	0.2	m
		P56, P57	V_{CC} = 4.0 V to 5.5 V	—	—	2.0	m
				—	—	0.2	m
Allowable output high current (total)	$\lvert -\sum I_{OH} \rvert$	All output pins	V_{CC} = 4.0 V to 5.5 V	—	—	40.0	m
				—	—	8.0	m

time			—	—	32.0	µs
Instruction cycle time			2	—	—	t_{cyc}
Oscillation stabilization time (crystal resonator)	t_{rc}	OSC1, OSC2	—	—	10.0	ms
Oscillation stabilization time (ceramic resonator)	t_{rc}	OSC1, OSC2	—	—	5.0	ms
External clock high width	t_{CPH}	OSC1	35.0	—	—	ns
External clock low width	t_{CPL}	OSC1	35.0	—	—	ns
External clock rise time	t_{CPr}	OSC1	—	—	15.0	ns
External clock fall time	t_{CPf}	OSC1	—	—	15.0	ns
$\overline{RES}$ pin low width*[4]	t_{REL}	$\overline{RES}$	2500	—	—	ns
$\overline{NMI}$ pin high width	t_{IHNMI}	$\overline{NMI}$	1500	—	—	ns
$\overline{NMI}$ pin low width	t_{ILNMI}	$\overline{NMI}$	1500	—	—	ns
Input pin high width	t_{IH}	$\overline{IRQ0}$, $\overline{IRQ3}$, $\overline{WKP5}$, TMCIV, TMRIV, TRGV, $\overline{ADTRG}$, FTCI, FTIOA to FTIOD	2	—	—	t_{cyc}
Input pin low width	t_{IL}	$\overline{IRQ0}$, $\overline{IRQ3}$, $\overline{WKP5}$, TMCIV, TMRIV, TRGV, $\overline{ADTRG}$, FTCI, FTIOA to FTIOD	2	—	—	t_{cyc}

Notes: 1. Determined by MA2 to MA0 in system control register 2 (SYSCR2).
2. For the oscillation frequency of the masked ROM version, refer to the electrica characteristics specified separately.
3. The values are for reference.
4. Except when power-on reset circuit is used.

SCL and SDA input fall time	t_{Sf}				—	—	300	ns
SCL and SDA input spike pulse removal time	t_{SP}				—	—	$1t_{cyc}$	ns
SDA input bus-free time	t_{BUF}				$5t_{cyc}$	—	—	ns
Start condition input hold time	t_{STAH}				$3t_{cyc}$	—	—	ns
Retransmission start condition input setup time	t_{STAS}				$3t_{cyc}$	—	—	ns
Setup time for stop condition input	t_{STOS}				$3t_{cyc}$	—	—	ns
Data input setup time	t_{SDAS}				$1t_{cyc}$ + 20	—	—	ns
Data input hold time	t_{SDAH}				0	—	—	ns
Capacitive load of SCL and SDA	c_b				0	—	400	pF
SCL and SDA output fall time	t_{Sf}	V_{CC} = 4.0 to 5.5 V	—		—	—	250	ns
			—		—	—	300	ns

Transmit data delay time (clocked synchronous)	t_{TXD}	TXD	—	—	1	t_{cyc}	F 2
Receive data setup time (clocked synchronous)	t_{RXS}	RXD	83.3	—	—	ns	
Receive data hold time (clocked synchronous)	t_{RXH}	RXD	83.3	—	—	ns	

Analog power supply current	AI_{OPE}	AV_{CC}	AV_{CC} = 5.0 V f_{OSC} = 12 MHz	—	—	2.0	mA
	AI_{STOP1}	AV_{CC}		—	50	—	µA
	AI_{STOP2}	AV_{CC}		—	—	5.0	µA
Analog input capacitance	C_{AIN}	AN3 to AN0		—	—	30.0	pF
Allowable signal source impedance	R_{AIN}	AN3 to AN0		—	—	5.0	kΩ
Resolution (data length)				10	10	10	bit
Conversion time (single mode)			AV_{CC} = 3.0 V to 5.5 V	134	—	—	t_{cyc}
Nonlinearity error				—	—	±7.5	LSB
Offset error				—	—	±7.5	LSB
Full-scale error				—	—	±7.5	LSB
Quantization error				—	—	±0.5	LSB
Absolute accuracy				—	—	±8.0	LSB
Conversion time (single mode)			AV_{CC} = 4.0 V to 5.5 V	70	—	—	t_{cyc}
Nonlinearity error				—	—	±7.5	LSB
Offset error				—	—	±7.5	LSB
Full-scale error				—	—	±7.5	LSB
Quantization error				—	—	±0.5	LSB
Absolute accuracy				—	—	±8.0	LSB

2. AI_{STOP1} is the current in active and sleep modes while the A/D converter is idle.
3. AI_{STOP2} is the current at reset and in standby and subsleep modes while the A/D converter is idle.

20.2.5 Watchdog Timer Characteristics

Table 20.7 Watchdog Timer Characteristics

V_{CC} = 3.0 V to 5.5 V, V_{SS} = 0.0 V, T_a = –20°C to +75°C, unless otherwise specified.

Item	Symbol	Applicable Pins	Test Condition	Values			Unit	N
				Min.	Typ.	Max.		
Internal oscillator overflow time	t_{OVF}			0.2	0.4	—	s	*

Note: * Shows the time to count from 0 to 255, at which point an internal reset is gene when the internal oscillator is selected.

Reset detection voltage 1*[1]	Vreset1	LVDSEL = 0	2.0	2.3	2.7
Reset detection voltage 2*[2]	Vreset2	LVDSEL = 1	3.0	3.6	4.2
Lower-limit voltage of LVDR operation*[3]	$V_{LVDRmin}$		1.0	—	—
LVD stabilization time	t_{LVDON}		50	—	—
Current consumption in standby mode	I_{STBY}	LVDE = 1, BGRE = 1 Vcc = 5.0 V	—	—	350

Notes: 1. This voltage should be used when the falling and rising voltage detection func used.
2. Select the low-voltage reset 2 when only the low-voltage detection reset is us
3. When the power-supply voltage (Vcc) falls below $V_{LVDRmin}$ = 1.0 V and then rise may not occur. Therefore sufficient evaluation is required.

20.2.7 LVDI External Voltage Detection Circuit Characteristics

Table 20.9 LVDI External Voltage Detection Circuit Characteristics

Vcc = 4.5 to 5.5 V, AVcc = 3.0 to 5.5 V, V_{SS} = 0.0 V, T_a = –20 to +75°C

Item	Symbol	Test Condition	Values Min.	Typ.	Max.
ExtD/ExtU input detection voltage	Vexd		0.85	1.15	1.45
ExtD/ExtU input voltage range	VextD/U	VextD > VextU	–0.3	—	Lower voltage, either AVcc + 0 or Vcc + 0.3

charge of the RES pin is removed completely. In order to remove charge of the
pin, it is recommended that the diode be placed in the Vcc side. If the power-s
voltage (Vcc) rises from the point over 100 mV, a power-on reset may not occu

Programming	Wait time after SWE bit setting*[1]	x		1	—	—
	Wait time after PSU bit setting*[1]	y		50	—	—
	Wait time after P bit setting*[1]*[4]	z1	1 ≤ n ≤ 6	28	30	32
		z2	7 ≤ n ≤ 1000	198	200	202
		z3	Additional-programming	8	10	12
	Wait time after P bit clear*[1]	α		5	—	—
	Wait time after PSU bit clear*[1]	β		5	—	—
	Wait time after PV bit setting*[1]	γ		4	—	—
	Wait time after dummy write*[1]	ε		2	—	—
	Wait time after PV bit clear*[1]	η		2	—	—
	Wait time after SWE bit clear*[1]	θ		100	—	—
	Maximum programming count*[1]*[4]*[5]	N		—	—	1000

bit clear*[1]					
Wait time after EV bit setting*[1]	γ	20	—	—	μ
Wait time after dummy write*[1]	ε	2	—	—	μ
Wait time after EV bit clear*[1]	η	4	—	—	μ
Wait time after SWE bit clear*[1]	θ	100	—	—	μ
Maximum erase count*[1]*[6]*[7]	N	—	—	120	

Notes: 1. Make the time settings in accordance with the program/erase algorithms.

2. The programming time for 64 bytes. (Indicates the total time for which the P bit memory control register 1 (FLMCR1) is set. The program-verify time is not incl

3. The time required to erase one block. (Indicates the time for which the E bit in memory control register 1 (FLMCR1) is set. The erase-verify time is not include

4. Programming time maximum value (t_P (max.)) = wait time after P bit setting (z) maximum programming count (N)

5. Set the maximum programming count (N) according to the actual set values of and z3, so that it does not exceed the programming time maximum value (t_P (m The wait time after P bit setting (z1, z2) should be changed as follows accordin value of the programming count (n).

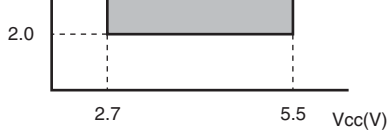
   Programming count (n)

   $1 \leq n \leq 6$ z1 = 30 μs

   $7 \leq n \leq 1000$ z2 = 200 μs

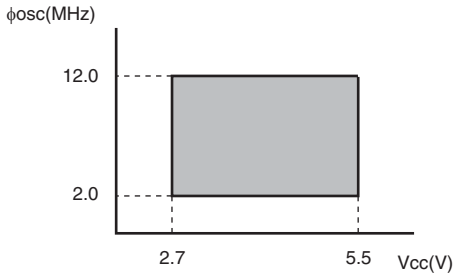
6. Erase time maximum value (t_E (max.)) = wait time after E bit setting (z) × maxi erase count (N)

7. Set the maximum erase count (N) according to the actual set value of (z), so th does not exceed the erase time maximum value (t_E (max.)).



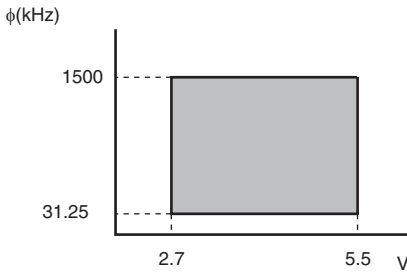
AVcc = 2.7 to 5.5 V
- Active mode
- Sleep mode

2. Power supply voltage and operating frequency range



AVcc = 2.7 to 5.5 V
- Active mode
- Sleep mode

(When MA2 = 0 in SYSCR2)



AVcc = 2.7 to 5.5 V
- Active mode
- Sleep mode

(When MA2 = 1 in SYSC

Vcc = 2.7 to 5.5 V
- Active mode
- Sleep mode

		FTIOA to FTIOD, SCK3, TRGV					
		RXD, SCL, SDA, P17, P14, P22 to P20, P57 to P55, P76 to P74, P84 to P80, PC1, PC0	V_{cc} = 4.0 V to 5.5 V	$V_{cc} \times 0.7$	—	V_{cc} + 0.3	V
				$V_{cc} \times 0.8$	—	V_{cc} + 0.3	V
		PB3 to PB0	AV_{cc} = 4.0 V to 5.5 V	$AV_{cc} \times 0.7$	—	AV_{cc} + 0.3	V
			AV_{cc} = 2.7 V to 5.5 V	$AV_{cc} \times 0.8$	—	AV_{cc} + 0.3	V
		OSC1	V_{cc} = 4.0 V to 5.5 V	V_{cc} – 0.5	—	V_{cc} + 0.3	V
				V_{cc} – 0.3		V_{cc} + 0.3	V
Input low voltage	V_{IL}	$\overline{RES}$, $\overline{NMI}$, $\overline{WKP5}$, $\overline{IRQ0}$, $\overline{IRQ3}$, $\overline{ADTRG}$, TMRIV, TMCIV, FTCI, FTIOA to FTIOD, SCK3, TRGV	V_{cc} = 4.0 V to 5.5 V	–0.3	—	$V_{cc} \times 0.2$	V
				–0.3	—	$V_{cc} \times 0.1$	V
		RXD, SCL, SDA, P17, P14, P22 to P20, P57 to P55, P76 to P74, P84 to P80, PC1, PC0	V_{cc} = 4.0 V to 5.5 V	–0.3	—	$V_{cc} \times 0.3$	V
				–0.3	—	$V_{cc} \times 0.2$	V
		PB3 to PB0	AV_{cc} = 4.0 V to 5.5 V	–0.3	—	$AV_{cc} \times 0.3$	V
			AV_{cc} = 2.7 V to 5.5 V	–0.3	—	$AV_{cc} \times 0.2$	
		OSC1	V_{cc} = 4.0 V to 5.5 V	–0.3	—	0.5	V
				–0.3	—	0.3	V

			$-I_{OH}$ = 0.1 mA				
Output low voltage	V_{OL}	P17, P14, P22 to P20, P57 to P55, P76 to P74, PC1, PC0	V_{CC} = 4.0 V to 5.5 V I_{OL} = 1.6 mA	—	—	0.6	V
			I_{OL} = 0.4 mA	—	—	0.4	V
		P84 to P80	V_{CC} = 4.0 V to 5.5 V I_{OL} = 20.0 mA	—	—	1.5	V
			V_{CC} = 4.0 V to 5.5 V I_{OL} = 10.0 mA	—	—	1.0	V
			V_{CC} = 4.0 V to 5.5 V I_{OL} = 1.6 mA	—	—	0.4	V
			I_{OL} = 0.4 mA	—	—	0.4	V
		SCL, SDA	V_{CC} = 4.0 V to 5.5 V I_{OL} = 6.0 mA	—	—	0.6	V
			I_{OL} = 3.0 mA	—	—	0.4	V
Input/ output leakage current	$\lvert I_{IL} \rvert$	OSC1, $\overline{\text{NMI}}$, $\overline{\text{WKP5}}$, $\overline{\text{IRQ0}}$, $\overline{\text{IRQ3}}$, $\overline{\text{ADTRG}}$, TRGV, TMRIV, TMCIV, FTCI, FTIOA to FTIOD, RXD, SCK3, SCL, SDA	V_{IN} = 0.5 V to (V_{CC} – 0.5 V)	—	—	1.0	µA
		P17, P14, P22 to P20, P57 to P55, P76 to P74, P84 to P80, PC1, PC0	V_{IN} = 0.5 V to (V_{CC} – 0.5 V)	—	—	1.0	µA
		PB3 to PB0	V_{IN} = 0.5 V to (AV_{CC} – 0.5 V)	—	—	1.0	µA

consumption								
			Active mode 1 V_{CC} = 2.7 V, f_{OSC} = 12 MHz	—	9.6	—	mA	R v
	I_{OPE2}	V_{CC}	Active mode 2 V_{CC} = 5.0 V, f_{OSC} = 12 MHz	—	2.0	2.5	mA	*
			Active mode 2 V_{CC} = 2.7 V, f_{OSC} = 12 MHz	—	1.5	—	mA	R v
Sleep mode current consumption	I_{SLEEP1}	V_{CC}	Sleep mode 1 V_{CC} = 5.0 V, f_{OSC} = 12 MHz	—	7.2	12.0	mA	*
			Sleep mode 1 V_{CC} = 2.7 V, f_{OSC} = 12 MHz	—	6.0	—	mA	R v
	I_{SLEEP2}	V_{CC}	Sleep mode 2 V_{CC} = 5.0 V, f_{OSC} = 12 MHz	—	1.8	2.2	mA	*
			Sleep mode 2 V_{CC} = 2.7 V, f_{OSC} = 12 MHz	—	1.4	—	mA	R v
Subsleep mode current consumption	I_{SUBSP}	V_{CC}	V_{CC} = 5.0 V LVDE = 0, BGRE = 0	—	—	5.0	µA	*
Standby mode current consumption	I_{STBY}	V_{CC}	LVDE = 0, BGRE = 0	—	—	5.0	µA	*

				resonator, and oscillator
Sleep mode 1	V_{CC}	Only timers operate	V_{CC}	
Sleep mode 2		Only timers operate (ϕ/64)		
Subsleep mode Standby mode	V_{CC}	CPU and timers both stop	V_{CC}	—

	and SDA				
	P84 to P80		—	—	10.0
	SCL, SDA		—	—	6.0
Allowable output low current (total) ΣI_{OL}	Output pins except P84 to P80, SCL, and SDA	V_{CC} = 4.0 V to 5.5 V	—	—	40.0
	P84 to P80, SCL, and SDA		—	—	80.0
	Output pins except P84 to P80, SCL, and SDA		—	—	20.0
	P84 to P80, SCL, and SDA		—	—	40.0
Allowable output high current (per pin) $\lvert -I_{OH} \rvert$	Output pins except P56, P57	V_{CC} = 4.0 V to 5.5 V	—	—	4.0
			—	—	0.2
	P56, P57	V_{CC} = 4.0 V to 5.5 V	—	—	2.0
			—	—	0.2
Allowable output high current (total) $\lvert -\Sigma I_{OH} \rvert$	All output pins	V_{CC} = 4.0 V to 5.5 V	—	—	40.0
			—	—	8.0

cycle time			—	—	32.0	µs	
Instruction cycle time			2	—	—	t_{cyc}	
Oscillation stabilization time (crystal resonator)	t_{rc}	OSC1, OSC2	—	—	10.0	ms	
Oscillation stabilization time (ceramic resonator)	t_{rc}	OSC1, OSC2	—	—	5.0	ms	
External clock high width	t_{CPH}	OSC1	35.0	—	—	ns	
External clock low width	t_{CPL}	OSC1	35.0	—	—	ns	
External clock rise time	t_{CPr}	OSC1	—	—	15.0	ns	
External clock fall time	t_{CPf}	OSC1	—	—	15.0	ns	
$\overline{RES}$ pin low width*	t_{REL}	$\overline{RES}$	2500	—	—	ns	
$\overline{NMI}$ pin high width	t_{IHNMI}	$\overline{NMI}$	1500	—	—	ns	
$\overline{NMI}$ pin low width	t_{ILNMI}	$\overline{NMI}$	1500	—	—	ns	
Input pin high width	t_{IH}	$\overline{IRQ0}$, $\overline{IRQ3}$, $\overline{WKP5}$,TMCIV, TMRIV, TRGV, $\overline{ADTRG}$, FTCI, FTIOA to FTIOD	2	—	—	t_{cyc}	

Note: * Except when power-on reset circuit is used.

				5.5 V FSEL = 1, VCLSEL = 0	

Notes: * Determined by MA2 to MA0 in system control register 2 (SYSCR2).

SCL and SDA input fall time	t_{Sf}		—	—	300	ns
SCL and SDA input spike pulse removal time	t_{SP}		—	—	$1t_{cyc}$	ns
SDA input bus-free time	t_{BUF}		$5t_{cyc}$	—	—	ns
Start condition input hold time	t_{STAH}		$3t_{cyc}$	—	—	ns
Retransmission start condition input setup time	t_{STAS}		$3t_{cyc}$	—	—	ns
Setup time for stop condition input	t_{STOS}		$3t_{cyc}$	—	—	ns
Data input setup time	t_{SDAS}		$1t_{cyc}$ + 20	—	—	ns
Data input hold time	t_{SDAH}		0	—	—	ns
Capacitive load of SCL and SDA	c_b		0	—	400	pF
SCL and SDA output fall time	t_{Sf}	V_{CC} = 4.0 to 5.5 V	—	—	250	ns
		—	—	—	300	ns

Transmit data delay time (clocked synchronous)	t_{TXD}	TXD	—	—	1	t_{cyc}
Receive data setup time (clocked synchronous)s	t_{RXS}	RXD	83.3	—	—	ns
Receive data hold time (clocked synchronous)	t_{RXH}	RXD	83.3	—	—	ns

Analog power supply current	AI_{OPE}	AV_{CC}	AV_{CC} = 5.0 V f_{OSC} = 12 MHz	—	—	2.0	mA	
	AI_{STOP1}	AV_{CC}		—	50	—	µA	*
	AI_{STOP2}	AV_{CC}		—	—	5.0	µA	*
Analog input capacitance	C_{AIN}	AN3 to AN0		—	—	30.0	pF	
Allowable signal source impedance	R_{AIN}	AN3 to AN0		—	—	5.0	kΩ	
Resolution (data length)				10	10	10	bit	
Conversion time (single mode)			AV_{CC} = 2.7 V to 5.5 V	134	—	—	t_{cyc}	
Nonlinearity error				—	—	±7.5	LSB	
Offset error				—	—	±7.5	LSB	
Full-scale error				—	—	±7.5	LSB	
Quantization error				—	—	±0.5	LSB	
Absolute accuracy				—	—	±8.0	LSB	
Conversion time (single mode)			AV_{CC} = 4.0 V to 5.5 V	70	—	—	t_{cyc}	
Nonlinearity error				—	—	±7.5	LSB	
Offset error				—	—	±7.5	LSB	
Full-scale error				—	—	±7.5	LSB	
Quantization error				—	—	±0.5	LSB	
Absolute accuracy				—	—	±8.0	LSB	

2. AI_{STOP1} is the current in active and sleep modes while the A/D converter is idle
3. AI_{STOP2} is the current at reset and in standby and subsleep modes while the A/ converter is idle.

20.3.5 Watchdog Timer Characteristics

Table 20.17 Watchdog Timer Characteristics

V_{CC} = 2.7 V to 5.5 V, V_{SS} = 0.0 V, T_a = –20°C to +75°C, unless otherwise specified.

Item	Symbol	Applicable Pins	Test Condition	Values			Unit
				Min.	Typ.	Max.	
Internal oscillator overflow time	t_{OVF}			0.2	0.4	—	s

Note: * Shows the time to count from 0 to 255, at which point an internal reset is gen when the internal oscillator is selected.

voltage					
Reset detection voltage 1*[1]	Vreset1	LVDSEL = 0	2.0	2.3	2.7
Reset detection voltage 2*[2]	Vreset2	LVDSEL = 1	3.0	3.6	4.2
Lower-limit voltage of LVDR operation*[3]	$V_{LVDRmin}$		1.0	—	—
LVD stabilization time	t_{LVDON}		50	—	—
Current consumption in standby mode	I_{STBY}	LVDE = 1, BGRE = 1 Vcc = 5.0 V	—	—	350

Notes: 1. This voltage should be used when the falling and rising voltage detection funct used.
2. Select the low-voltage reset 2 when only the low-voltage detection reset is use
3. When the power-supply voltage (Vcc) falls below $V_{LVDRmin}$ = 1.0 V and then rises may not occur. Therefore sufficient evaluation is required.

20.3.7 LVDI External Voltage Detection Circuit Characteristics

Table 20.19 LVDI External Voltage Detection Circuit Characteristics

Vcc = 4.5 to 5.5 V, AVcc = 2.7 to 5.5 V, V_{SS} = 0.0 V, T_a = –20 to +75°C

Item	Symbol	Test Condition	Values		
			Min.	Typ.	Max.
ExtD/ExtU input detection voltage	Vexd		0.85	1.15	1.45
ExtD/ExtU input voltage range	VextD/U	VextD > VextU	–0.3	—	Lower voltage, either AVcc + 0.3 or Vcc + 0.3

charge of the RES pin is removed completely. In order to remove charge of th
pin, it is recommended that the diode be placed in the Vcc side. If the power-
voltage (Vcc) rises from the point over 100 mV, a power-on reset may not oc

Figure 20.1 System Clock Input Timing

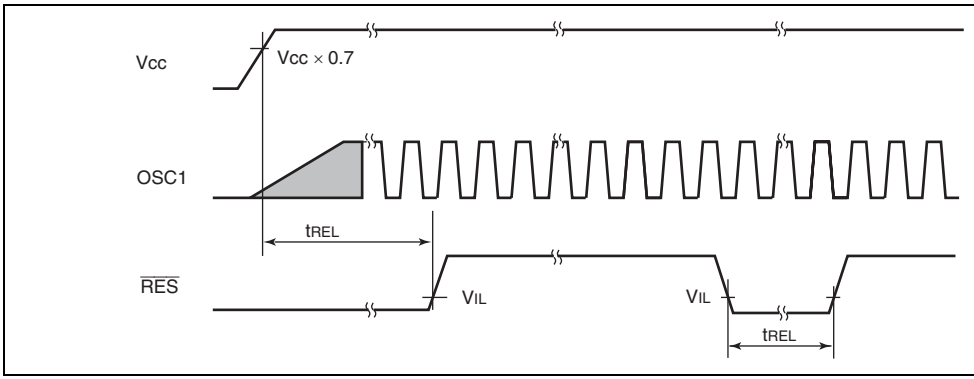


Figure 20.2 $\overline{\text{RES}}$ Low Width Timing

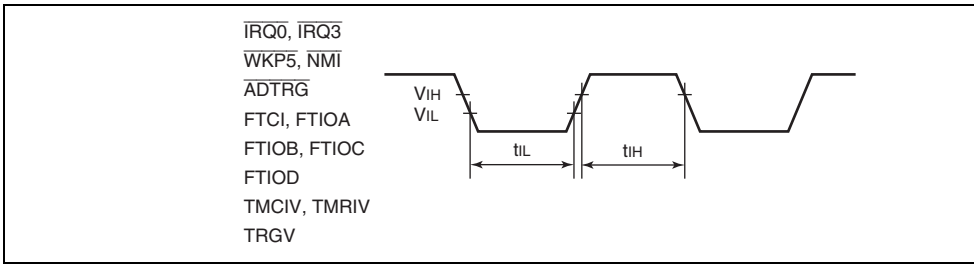


Figure 20.3 Input Timing

Note: * S, P, and Sr represent the following:
S: Start condition
P: Stop comdition
Sr: Retransmission start condition

Figure 20.4 I^2C Bus Interface Input/Output Timing

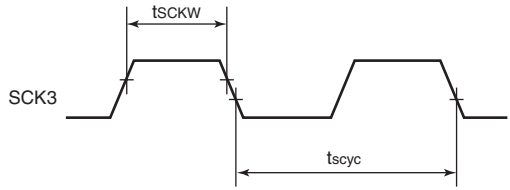
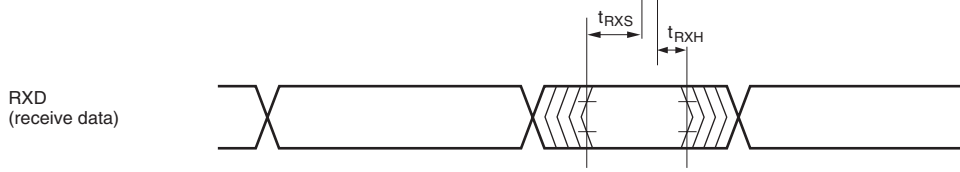


Figure 20.5 SCK3 Input Clock Timing



Note: * Output timing reference levels
Output high: V_{OH} = 2.0 V
Output low: V_{OL} = 0.8 V
Load conditions are shown in figure 20.7.

Figure 20.6 SCI3 Input/Output Timing in Clocked Synchronous Mode

20.5 Output Load Condition

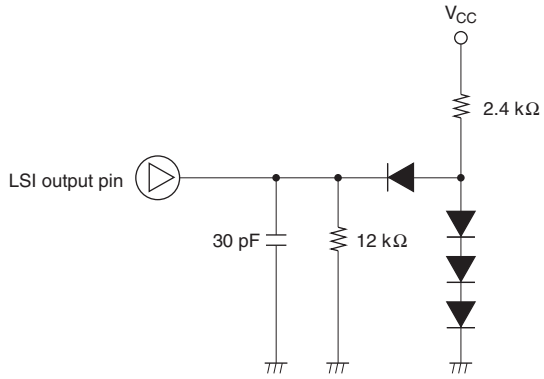


Figure 20.7 Output Load Circuit

ERd	General destination register (address register or 32-bit register)
ERs	General source register (address register or 32-bit register)
ERn	General register (32-bit register)
(EAd)	Destination operand
(EAs)	Source operand
PC	Program counter
SP	Stack pointer
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
disp	Displacement
→	Transfer from the operand on the left to the operand on the right, or transi the state on the left to the state on the right
+	Addition of the operands on both sides
–	Subtraction of the operand on the right from the operand on the left
×	Multiplication of the operands on both sides
÷	Division of the operand on the left by the operand on the right
∧	Logical AND of the operands on both sides
∨	Logical OR of the operands on both sides
⊕	Logical exclusive OR of the operands on both sides
¬	NOT (logical complement)

1	Set to 1
—	Not affected by execution of the instruction
Δ	Varies depending on conditions, described in notes

	MOV.B @ERs, Rd	B			2						@ERs → Rd8	—	—	↕	↕	0
	MOV.B @(d:16, ERs), Rd	B				4					@(d:16, ERs) → Rd8	—	—	↕	↕	0
	MOV.B @(d:24, ERs), Rd	B				8					@(d:24, ERs) → Rd8	—	—	↕	↕	0
	MOV.B @ERs+, Rd	B					2				@ERs → Rd8 ERs32+1 → ERs32	—	—	↕	↕	0
	MOV.B @aa:8, Rd	B						2			@aa:8 → Rd8	—	—	↕	↕	0
	MOV.B @aa:16, Rd	B						4			@aa:16 → Rd8	—	—	↕	↕	0
	MOV.B @aa:24, Rd	B						6			@aa:24 → Rd8	—	—	↕	↕	0
	MOV.B Rs, @ERd	B			2						Rs8 → @ERd	—	—	↕	↕	0
	MOV.B Rs, @(d:16, ERd)	B				4					Rs8 → @(d:16, ERd)	—	—	↕	↕	0
	MOV.B Rs, @(d:24, ERd)	B				8					Rs8 → @(d:24, ERd)	—	—	↕	↕	0
	MOV.B Rs, @–ERd	B					2				ERd32–1 → ERd32 Rs8 → @ERd	—	—	↕	↕	0
	MOV.B Rs, @aa:8	B						2			Rs8 → @aa:8	—	—	↕	↕	0
	MOV.B Rs, @aa:16	B						4			Rs8 → @aa:16	—	—	↕	↕	0
	MOV.B Rs, @aa:24	B						6			Rs8 → @aa:24	—	—	↕	↕	0
	MOV.W #xx:16, Rd	W	4								#xx:16 → Rd16	—	—	↕	↕	0
	MOV.W Rs, Rd	W		2							Rs16 → Rd16	—	—	↕	↕	0
	MOV.W @ERs, Rd	W			2						@ERs → Rd16	—	—	↕	↕	0
	MOV.W @(d:16, ERs), Rd	W				4					@(d:16, ERs) → Rd16	—	—	↕	↕	0
	MOV.W @(d:24, ERs), Rd	W				8					@(d:24, ERs) → Rd16	—	—	↕	↕	0
	MOV.W @ERs+, Rd	W					2				@ERs → Rd16 ERs32+2 → @ERd32	—	—	↕	↕	0
	MOV.W @aa:16, Rd	W						4			@aa:16 → Rd16	—	—	↕	↕	0
	MOV.W @aa:24, Rd	W						6			@aa:24 → Rd16	—	—	↕	↕	0
	MOV.W Rs, @ERd	W			2						Rs16 → @ERd	—	—	↕	↕	0
	MOV.W Rs, @(d:16, ERd)	W				4					Rs16 → @(d:16, ERd)	—	—	↕	↕	0
	MOV.W Rs, @(d:24, ERd)	W				8					Rs16 → @(d:24, ERd)	—	—	↕	↕	0

	MOV.L ERs, ERd	L			2							MOV.L ERs, ERd	L			2		
	MOV.L @ERs, ERd	L			4							@ERs → ERd32	—	—	↕	↕	0	
	MOV.L @(d:16, ERs), ERd	L				6						@(d:16, ERs) → ERd32	—	—	↕	↕	0	
	MOV.L @(d:24, ERs), ERd	L				10						@(d:24, ERs) → ERd32	—	—	↕	↕	0	
	MOV.L @ERs+, ERd	L					4					@ERs → ERd32 ERs32+4 → ERs32	—	—	↕	↕	0	
	MOV.L @aa:16, ERd	L						6				@aa:16 → ERd32	—	—	↕	↕	0	
	MOV.L @aa:24, ERd	L						8				@aa:24 → ERd32	—	—	↕	↕	0	
	MOV.L ERs, @ERd	L			4							ERs32 → @ERd	—	—	↕	↕	0	
	MOV.L ERs, @(d:16, ERd)	L				6						ERs32 → @(d:16, ERd)	—	—	↕	↕	0	
	MOV.L ERs, @(d:24, ERd)	L				10						ERs32 → @(d:24, ERd)	—	—	↕	↕	0	
	MOV.L ERs, @–ERd	L					4					ERd32–4 → ERd32 ERs32 → @ERd	—	—	↕	↕	0	
	MOV.L ERs, @aa:16	L						6				ERs32 → @aa:16	—	—	↕	↕	0	
	MOV.L ERs, @aa:24	L						8				ERs32 → @aa:24	—	—	↕	↕	0	
POP	POP.W Rn	W									2	@SP → Rn16 SP+2 → SP	—	—	↕	↕	0	
	POP.L ERn	L									4	@SP → ERn32 SP+4 → SP	—	—	↕	↕	0	
PUSH	PUSH.W Rn	W									2	SP–2 → SP Rn16 → @SP	—	—	↕	↕	0	
	PUSH.L ERn	L									4	SP–4 → SP ERn32 → @SP	—	—	↕	↕	0	
MOVFPE	MOVFPE @aa:16, Rd	B						4				Cannot be used in this LSI	Cannot be used in this LSI					
MOVTPE	MOVTPE Rs, @aa:16	B						4				Cannot be used in this LSI	Cannot be used in this LSI					

	ADD.L #xx:32, ERd	L	6									ERd32+#xx:32 → ERd32	—	(2)	↕	↕	↕
	ADD.L ERs, ERd	L		2								ERd32+ERs32 → ERd32	—	(2)	↕	↕	↕
ADDX	ADDX.B #xx:8, Rd	B	2									Rd8+#xx:8 +C → Rd8	—	↕	↕	(3)	↕
	ADDX.B Rs, Rd	B		2								Rd8+Rs8 +C → Rd8	—	↕	↕	(3)	↕
ADDS	ADDS.L #1, ERd	L		2								ERd32+1 → ERd32	—	—	—	—	—
	ADDS.L #2, ERd	L		2								ERd32+2 → ERd32	—	—	—	—	—
	ADDS.L #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	—	(1)	↕	↕	↕
	SUB.W Rs, Rd	W		2								Rd16–Rs16 → Rd16	—	(1)	↕	↕	↕
	SUB.L #xx:32, ERd	L	6									ERd32–#xx:32 → ERd32	—	(2)	↕	↕	↕
	SUB.L ERs, ERd	L		2								ERd32–ERs32 → ERd32	—	(2)	↕	↕	↕
SUBX	SUBX.B #xx:8, Rd	B	2									Rd8–#xx:8–C → Rd8	—	↕	↕	(3)	↕
	SUBX.B Rs, Rd	B		2								Rd8–Rs8–C → Rd8	—	↕	↕	(3)	↕
SUBS	SUBS.L #1, ERd	L		2								ERd32–1 → ERd32	—	—	—	—	—
	SUBS.L #2, ERd	L		2								ERd32–2 → ERd32	—	—	—	—	—
	SUBS.L #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	—	—	↕	↕	↕

	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	—	(1)	↕	↕	↕	
	CMP.W Rs, Rd	W		2								Rd16–Rs16	—	(1)	↕	↕	↕	
	CMP.L #xx:32, ERd	L	6									ERd32–#xx:32	—	(2)	↕	↕	↕	
	CMP.L ERs, ERd	L		2								ERd32–ERs32	—	(2)	↕	↕	↕	

			of ERd32)																
EXTS	EXTS.W Rd	W	(<bit 7> of Rd16) → (<bits 15 to 8> of Rd16)		2								—	—	↕	↕	0
	EXTS.L ERd	L	(<bit 15> of ERd32) → (<bits 31 to 16> of ERd32)		2								—	—	↕	↕	0

	AND.L #xx:32, ERd	L	6									ERd32∧#xx:32 → ERd32	—	—	↕	↕	0	-
	AND.L ERs, ERd	L		4								ERd32∧ERs32 → ERd32	—	—	↕	↕	0	-
OR	OR.B #xx:8, Rd	B	2									Rd8∨#xx:8 → Rd8	—	—	↕	↕	0	-
	OR.B Rs, Rd	B		2								Rd8∨Rs8 → Rd8	—	—	↕	↕	0	-
	OR.W #xx:16, Rd	W	4									Rd16∨#xx:16 → Rd16	—	—	↕	↕	0	-
	OR.W Rs, Rd	W		2								Rd16∨Rs16 → Rd16	—	—	↕	↕	0	-
	OR.L #xx:32, ERd	L	6									ERd32∨#xx:32 → ERd32	—	—	↕	↕	0	-
	OR.L ERs, ERd	L		4								ERd32∨ERs32 → ERd32	—	—	↕	↕	0	-
XOR	XOR.B #xx:8, Rd	B	2									Rd8⊕#xx:8 → Rd8	—	—	↕	↕	0	-
	XOR.B Rs, Rd	B		2								Rd8⊕Rs8 → Rd8	—	—	↕	↕	0	-
	XOR.W #xx:16, Rd	W	4									Rd16⊕#xx:16 → Rd16	—	—	↕	↕	0	-
	XOR.W Rs, Rd	W		2								Rd16⊕Rs16 → Rd16	—	—	↕	↕	0	-
	XOR.L #xx:32, ERd	L	6									ERd32⊕#xx:32 → ERd32	—	—	↕	↕	0	-
	XOR.L ERs, ERd	L		4								ERd32⊕ERs32 → ERd32	—	—	↕	↕	0	-
NOT	NOT.B Rd	B		2								¬ Rd8 → Rd8	—	—	↕	↕	0	-
	NOT.W Rd	W		2								¬ Rd16 → Rd16	—	—	↕	↕	0	-
	NOT.L ERd	L		2								¬ Rd32 → Rd32	—	—	↕	↕	0	-

	SHAR.W Rd	W		2								—	—	↕	↕	0
	SHAR.L ERd	L		2								—	—	↕	↕	0
SHLL	SHLL.B Rd	B		2								—	—	↕	↕	0
	SHLL.W Rd	W		2								—	—	↕	↕	0
	SHLL.L ERd	L		2								—	—	↕	↕	0
SHLR	SHLR.B Rd	B		2								—	—	↕	↕	0
	SHLR.W Rd	W		2								—	—	↕	↕	0
	SHLR.L ERd	L		2								—	—	↕	↕	0
ROTXL	ROTXL.B Rd	B		2								—	—	↕	↕	0
	ROTXL.W Rd	W		2								—	—	↕	↕	0
	ROTXL.L ERd	L		2								—	—	↕	↕	0
ROTXR	ROTXR.B Rd	B		2								—	—	↕	↕	0
	ROTXR.W Rd	W		2								—	—	↕	↕	0
	ROTXR.L ERd	L		2								—	—	↕	↕	0
ROTL	ROTL.B Rd	B		2								—	—	↕	↕	0
	ROTL.W Rd	W		2								—	—	↕	↕	0
	ROTL.L ERd	L		2								—	—	↕	↕	0
ROTR	ROTR.B Rd	B		2								—	—	↕	↕	0
	ROTR.W Rd	W		2								—	—	↕	↕	0
	ROTR.L ERd	L		2								—	—	↕	↕	0

	BSET Rn, @ERd	B			4							(Rn8 of @ERd) ← 1	—	—	—	—	—	
	BSET Rn, @aa:8	B						4				(Rn8 of @aa:8) ← 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 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	—	—	—	—	—	
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 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)	—	—	—	—	—	
BTST	BTST #xx:3, Rd	B		2								¬ (#xx:3 of Rd8) → Z	—	—	—	↕	—	
	BTST #xx:3, @ERd	B			4							¬ (#xx:3 of @ERd) → Z	—	—	—	↕	—	
	BTST #xx:3, @aa:8	B						4				¬ (#xx:3 of @aa:8) → 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	—	—	—	↕	—	
BLD	BLD #xx:3, Rd	B		2								(#xx:3 of Rd8) → C	—	—	—	—	—	

BST	BST #xx:3, @ERd	B			4						C → (#xx:3 of @ERd24)	—	—	—	—	—
BIST	BST #xx:3, @aa:8	B						4			C → (#xx:3 of @aa:8)	—	—	—	—	—
	BIST #xx:3, Rd	B		2							¬ C → (#xx:3 of Rd8)	—	—	—	—	—
	BIST #xx:3, @ERd	B			4						¬ C → (#xx:3 of @ERd24)	—	—	—	—	—
	BIST #xx:3, @aa:8	B						4			¬ C → (#xx:3 of @aa:8)	—	—	—	—	—
BAND	BAND #xx:3, Rd	B		2							C∧(#xx:3 of Rd8) → C	—	—	—	—	—
	BAND #xx:3, @ERd	B			4						C∧(#xx:3 of @ERd24) → C	—	—	—	—	—
BIAND	BAND #xx:3, @aa:8	B						4			C∧(#xx:3 of @aa:8) → C	—	—	—	—	—
	BIAND #xx:3, Rd	B		2							C∧ ¬ (#xx:3 of Rd8) → C	—	—	—	—	—
	BIAND #xx:3, @ERd	B			4						C∧ ¬ (#xx:3 of @ERd24) → C	—	—	—	—	—
	BIAND #xx:3, @aa:8	B						4			C∧ ¬ (#xx:3 of @aa:8) → C	—	—	—	—	—
BOR	BOR #xx:3, Rd	B		2							C∕(#xx:3 of Rd8) → C	—	—	—	—	—
	BOR #xx:3, @ERd	B			4						C∕(#xx:3 of @ERd24) → C	—	—	—	—	—
	BOR #xx:3, @aa:8	B						4			C∕(#xx:3 of @aa:8) → C	—	—	—	—	—
BIOR	BIOR #xx:3, Rd	B		2							C∕ ¬ (#xx:3 of Rd8) → C	—	—	—	—	—
	BIOR #xx:3, @ERd	B			4						C∕ ¬ (#xx:3 of @ERd24) → C	—	—	—	—	—
	BIOR #xx:3, @aa:8	B						4			C∕ ¬ (#xx:3 of @aa:8) → C	—	—	—	—	—
BXOR	BXOR #xx:3, Rd	B		2							C⊕(#xx:3 of Rd8) → C	—	—	—	—	—
	BXOR #xx:3, @ERd	B			4						C⊕(#xx:3 of @ERd24) → C	—	—	—	—	—
	BXOR #xx:3, @aa:8	B						4			C⊕(#xx:3 of @aa:8) → C	—	—	—	—	—
BIXOR	BIXOR #xx:3, Rd	B		2							C⊕ ¬ (#xx:3 of Rd8) → C	—	—	—	—	—
	BIXOR #xx:3, @ERd	B			4						C⊕ ¬ (#xx:3 of @ERd24) → C	—	—	—	—	—
	BIXOR #xx:3, @aa:8	B						4			C⊕ ¬ (#xx:3 of @aa:8) → C	—	—	—	—	—

	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				—	—	—	—	—			

													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+	—	—	—	—	—

												PC ← @SP+						
SLEEP	SLEEP	—										Transition to power-down state	—	—	—	—	—	
LDC	LDC #xx:8, CCR	B	2									#xx:8 → CCR	↕	↕	↕	↕	↕	
	LDC Rs, CCR	B		2								Rs8 → CCR	↕	↕	↕	↕	↕	
	LDC @ERs, CCR	W			4							@ERs → CCR	↕	↕	↕	↕	↕	
	LDC @(d:16, ERs), CCR	W				6						@(d:16, ERs) → CCR	↕	↕	↕	↕	↕	
	LDC @(d:24, ERs), CCR	W				10						@(d:24, ERs) → CCR	↕	↕	↕	↕	↕	
	LDC @ERs+, CCR	W					4					@ERs → CCR ERs32+2 → ERs32	↕	↕	↕	↕	↕	
	LDC @aa:16, CCR	W						6				@aa:16 → CCR	↕	↕	↕	↕	↕	
	LDC @aa:24, CCR	W						8				@aa:24 → CCR	↕	↕	↕	↕	↕	
STC	STC CCR, Rd	B		2								CCR → Rd8	—	—	—	—	—	
	STC CCR, @ERd	W			4							CCR → @ERd	—	—	—	—	—	
	STC CCR, @(d:16, ERd)	W				6						CCR → @(d:16, ERd)	—	—	—	—	—	
	STC CCR, @(d:24, ERd)	W				10						CCR → @(d:24, ERd)	—	—	—	—	—	
	STC CCR, @–ERd	W					4					ERd32–2 → ERd32 CCR → @ERd	—	—	—	—	—	
	STC CCR, @aa:16	W						6				CCR → @aa:16	—	—	—	—	—	
	STC CCR, @aa:24	W						8				CCR → @aa:24	—	—	—	—	—	
ANDC	ANDC #xx:8, CCR	B	2									CCR∧#xx:8 → CCR	↕	↕	↕	↕	↕	
ORC	ORC #xx:8, CCR	B	2									CCR∨#xx:8 → CCR	↕	↕	↕	↕	↕	
XORC	XORC #xx:8, CCR	B	2									CCR⊕#xx:8 → CCR	↕	↕	↕	↕	↕	
NOP	NOP	—									2	PC ← PC+2	—	—	—	—	—	

												R4L–1 → R4L									
												until R4L=0									
												else next									
	EEPMOV. W	—									4	if R4 ≠ 0 then	—	—	—	—	—				
												repeat @R5 → @R6									
												R5+1 → R5									
												R6+1 → R6									
												R4–1 → R4									
												until R4=0									
												else next									

Notes: 1. The number of states in cases where the instruction code and its operands ar
in on-chip memory is shown here. For other cases see appendix A.3, Numbe
Execution States.

2. n is the value set in register R4L or R4.
   (1) Set to 1 when a carry or borrow occurs at bit 11; otherwise cleared to 0.
   (2) Set to 1 when a carry or borrow occurs at bit 27; otherwise cleared to 0.
   (3) Retains its previous value when the result is zero; otherwise cleared to 0.
   (4) Set to 1 when the adjustment produces a carry; otherwise retains its prev
   (5) The number of states required for execution of an instruction that transfer
   synchronization with the E clock is variable.
   (6) Set to 1 when the divisor is negative; otherwise cleared to 0.
   (7) Set to 1 when the divisor is zero; otherwise cleared to 0.
   (8) Set to 1 when the quotient is negative; otherwise cleared to 0.

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	E	F
0	NOP	Table A-2 (2)	STC	LDC	ORC	XORC	ANDC	LDC	ADD		Table A-2 (2)	Table A-2 (2)				
1	Table A-2 (2)	Table A-2 (2)	Table A-2 (2)	Table A-2 (2)	OR.B	XOR.B	AND.B	Table A-2 (2)	SUB		Table A-2 (2)	Table A-2 (2)				
2	MOV.B															
3																
4	BRA	BRN	BHI	BLS	BCC	BCS	BNE	BEQ	BVC	BVS	BPL	BMI	BGE	BLT	BGT	BLE
5	MULXU	DIVXU	MULXU	DIVXU	RTS	BSR	RTE	TRAPA (2)	JMP				BSR	JSR	RTS	
6	BSET	BNOT	BCLR	BTST	OR / BIOR	XOR / BIXOR	AND / BIAND	BST / BIST	MOV							
7								BLD / BILD	MOV	Table A-2 (2)	Table A-2 (2)	EEPMOV				
8	ADD															
9	ADDX															
A	CMP															
B	SUBX															
C	OR															
D	XOR															
E	AND															

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
01	MOV				LDC/STC				SLEEP			
0A	INC											
0B	ADDS					INC		INC	ADDS			
0F	DAA											
10	SHLL			SHLL					SHAL			SHAL
11	SHLR			SHLR					SHAR			SHAR
12	ROTXL			ROTXL					ROTL			ROTL
13	ROTXR			ROTXR					ROTR			ROTR
17	NOT			NOT		EXTU		EXTU	NEG			NEG
1A	DEC											
1B	SUBS					DEC		DEC	SUB			
1F	DAS											
58	BRA	BRN	BHI	BLS	BCC	BCS	BNE	BEQ	BVC	BVS	BPL	BMI
79	MOV	ADD	CMP	SUB	OR	XOR	AND					

Instruction code:

1st byte		2nd byte		3rd byte		4th byte	
AH	AL	BH	BL	CH	CL	DH	DL

Instruction when m
Instruction when m

CL / AH / ALBH / BLCH	0	1	2	3	4	5	6	7	8	9	A	B
01406										LDC / STC		LDC / ST
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 designation field.
2. aa is the absolute address field.

BSET #0, @FF00

From table A.4:

I = L = 2, J = K = M = N= 0

From table A.3:

$S_I = 2$, $S_L = 2$

Number of states required for execution = $2 \times 2 + 2 \times 2 = 8$

When instruction is fetched from on-chip ROM, branch address is read from on-chip RO
on-chip RAM is used for stack area.

JSR @@ 30

From table A.4:

I = 2, J = K = 1, L = M = N = 0

From table A.3:

$S_I = S_J = S_K = 2$

Number of states required for execution = $2 \times 2 + 1 \times 2 + 1 \times 2 = 8$

Note: * Depends on which on-chip peripheral module is accessed. See section 19.1, F
Addresses (Address Order).

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		
BAND	BAND #xx:3, Rd	1		
	BAND #xx:3, @ERd	2		1
	BAND #xx:3, @aa:8	2		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		
	BMI d:8	2		
	BGE d:8	2		

	BCS d:16(BLO d:16)	2		2
	BNE d:16	2		2
	BEQ d:16	2		2
	BVC d:16	2		2
	BVS d:16	2		2
	BPL d:16	2		2
	BMI d:16	2		2
	BGE d:16	2		2
	BLT d:16	2		2
	BGT d:16	2		2
	BLE d:16	2		2
BCLR	BCLR #xx:3, Rd	1		
	BCLR #xx:3, @ERd	2	2	
	BCLR #xx:3, @aa:8	2	2	
	BCLR Rn, Rd	1		
	BCLR Rn, @ERd	2	2	
	BCLR Rn, @aa:8	2	2	
BIAND	BIAND #xx:3, Rd	1		
	BIAND #xx:3, @ERd	2	1	
	BIAND #xx:3, @aa:8	2	1	
BILD	BILD #xx:3, Rd	1		
	BILD #xx:3, @ERd	2	1	
	BILD #xx:3, @aa:8	2	1	

	BIXOR #xx:3, @aa:8	2		1
BLD	BLD #xx:3, Rd	1		
	BLD #xx:3, @ERd	2		1
	BLD #xx:3, @aa:8	2		1
BNOT	BNOT #xx:3, Rd	1		
	BNOT #xx:3, @ERd	2		2
	BNOT #xx:3, @aa:8	2		2
	BNOT Rn, Rd	1		
	BNOT Rn, @ERd	2		2
	BNOT Rn, @aa:8	2		2
BOR	BOR #xx:3, Rd	1		
	BOR #xx:3, @ERd	2		1
	BOR #xx:3, @aa:8	2		1
BSET	BSET #xx:3, Rd	1		
	BSET #xx:3, @ERd	2		2
	BSET #xx:3, @aa:8	2		2
	BSET Rn, Rd	1		
	BSET Rn, @ERd	2		2
	BSET Rn, @aa:8	2		2
BSR	BSR d:8	2	1	
	BSR d:16	2	1	
BST	BST #xx:3, Rd	1		
	BST #xx:3, @ERd	2		2
	BST #xx:3, @aa:8	2		2

	BXOR #xx:3, @aa:8	2		1	
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			
DUVXS	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*1	
	EEPMOV.W	2		2n+2*1	
EXTS	EXTS.W Rd	1			
	EXTS.L ERd	1			
EXTU	EXTU.W Rd	1			
	EXTU.L ERd	1			

	JSR @@aa:8	2	1	1		
LDC	LDC #xx:8, CCR	1				
	LDC Rs, CCR	1				
	LDC@ERs, CCR	2				1
	LDC@(d:16, ERs), CCR	3				1
	LDC@(d:24,ERs), CCR	5				1
	LDC@ERs+, CCR	2				1
	LDC@aa:16, CCR	3				1
	LDC@aa:24, CCR	4				1
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:24, 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 @aa:24, Rd	3			1	
	MOV.B Rs, @Erd	1			1	
	MOV.B Rs, @(d:16, ERd)	2			1	
	MOV.B Rs, @(d:24, ERd)	4			1	
	MOV.B Rs, @-ERd	1			1	
	MOV.B Rs, @aa:8	1			1	

	MOV.W @aa:16, Rd	2		1	
	MOV.W @aa:24, Rd	3		1	
	MOV.W Rs, @ERd	1		1	
	MOV.W Rs, @(d:16,ERd)	2		1	
	MOV.W Rs, @(d:24,ERd)	4		1	
MOV	MOV.W Rs, @-ERd	1		1	
	MOV.W Rs, @aa:16	2		1	
	MOV.W Rs, @aa:24	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:24,ERs), ERd	5		2	
	MOV.L @ERs+, ERd	2		2	
	MOV.L @aa:16, ERd	3		2	
	MOV.L @aa:24, ERd	4		2	
	MOV.L ERs,@ERd	2		2	
	MOV.L ERs, @(d:16,ERd)	3		2	
	MOV.L ERs, @(d:24,ERd)	5		2	
	MOV.L ERs, @-ERd	2		2	
	MOV.L ERs, @aa:16	3		2	
	MOV.L ERs, @aa:24	4		2	
MOVFPE	MOVFPE @aa:16, Rd*[2]	2	1		
MOVTPE	MOVTPE Rs,@aa:16*[2]	2	1		

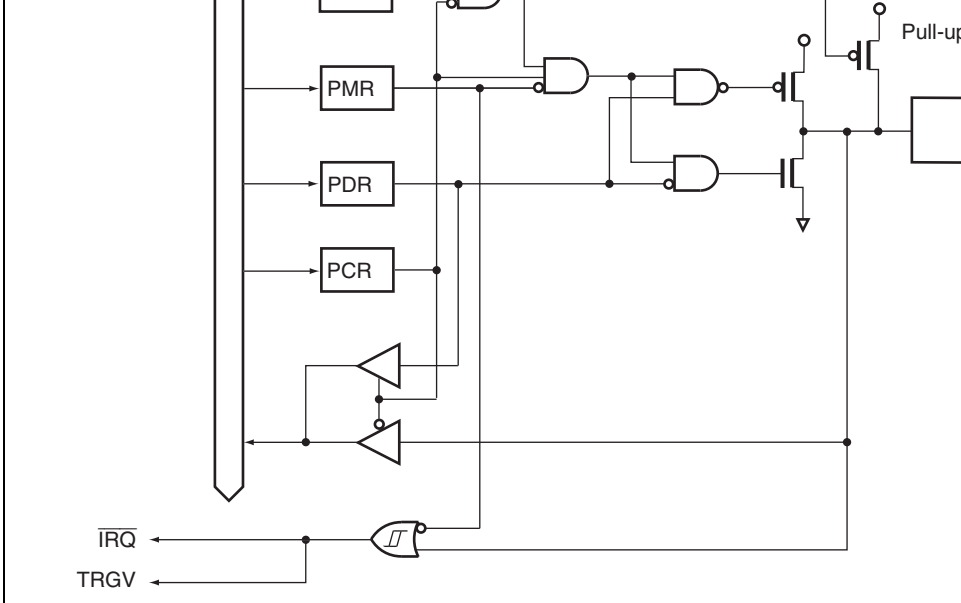
NOT	NOT.B Rd	1		
	NOT.W Rd	1		
	NOT.L ERd	1		
OR	OR.B #xx:8, Rd	1		
	OR.B Rs, Rd	1		
	OR.W #xx:16, Rd	2		
	OR.W Rs, Rd	1		
	OR.L #xx:32, ERd	3		
	OR.L ERs, ERd	2		
ORC	ORC #xx:8, CCR	1		
POP	POP.W Rn	1		1
	POP.L ERn	2		2
PUSH	PUSH.W Rn	1		1
	PUSH.L ERn	2		2
ROTL	ROTL.B Rd	1		
	ROTL.W Rd	1		
	ROTL.L ERd	1		
ROTR	ROTR.B Rd	1		
	ROTR.W Rd	1		
	ROTR.L ERd	1		
ROTXL	ROTXL.B Rd	1		
	ROTXL.W Rd	1		
	ROTXL.L ERd	1		

SHAR	SHAR.B Rd	1			
	SHAR.W Rd	1			
	SHAR.L ERd	1			
SHLL	SHLL.B Rd	1			
	SHLL.W Rd	1			
	SHLL.L ERd	1			
SHLR	SHLR.B Rd	1			
	SHLR.W Rd	1			
	SHLR.L ERd	1			
SLEEP	SLEEP	1			
STC	STC CCR, Rd	1			
	STC CCR, @ERd	2		1	
	STC CCR, @(d:16,ERd)	3		1	
	STC CCR, @(d:24,ERd)	5		1	
	STC CCR,@-ERd	2		1	
	STC CCR, @aa:16	3		1	
	STC CCR, @aa:24	4		1	
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			
SUBS	SUBS #1/2/4, ERd	1			

	XOR.L ERs, ERd	2
XORC	XORC #xx:8, CCR	1

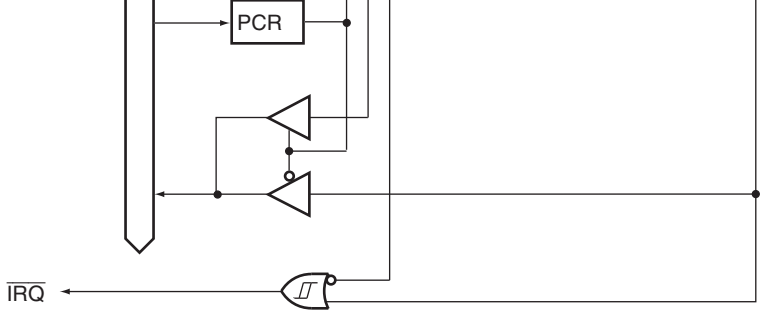
Notes: 1. n: Specified value in R4L and R4. The source and destination operands are a
n+1 times respectively.
2. Cannot be used in this LSI.

instructions	MOVFPE, MOVTPE	—	—	—	—	—	—	—	—	—	—	—	—
Arithmetic operations	ADD, CMP	BWL	BWL	—	—	—	—	—	—	—	—	—	—
	SUB	WL	BWL	—	—	—	—	—	—	—	—	—	—
	ADDX, SUBX	B	B	—	—	—	—	—	—	—	—	—	—
	ADDS, SUBS	—	L	—	—	—	—	—	—	—	—	—	—
	INC, DEC	—	BWL	—	—	—	—	—	—	—	—	—	—
	DAA, DAS	—	B	—	—	—	—	—	—	—	—	—	—
	MULXU, MULXS, DIVXU, DIVXS	—	BW	—	—	—	—	—	—	—	—	—	—
	NEG	—	BWL	—	—	—	—	—	—	—	—	—	—
	EXTU, EXTS	—	WL	—	—	—	—	—	—	—	—	—	—
Logical operations	AND, OR, XOR	—	BWL	—	—	—	—	—	—	—	—	—	—
	NOT	—	BWL	—	—	—	—	—	—	—	—	—	—
Shift operations		—	BWL	—	—	—	—	—	—	—	—	—	—
Bit manipulations		—	B	B	—	—	—	B	—	—	—	—	—
Branching instructions	BCC, BSR	—	—	—	—	—	—	—	—	—	—	—	—
	JMP, JSR	—	—	◯	—	—	—	—	—	—	◯	◯	—
	RTS	—	—	—	—	—	—	—	—	◯	—	—	◯
System control instructions	TRAPA	—	—	—	—	—	—	—	—	—	—	—	—
	RTE	—	—	—	—	—	—	—	—	—	—	—	—
	SLEEP	—	—	—	—	—	—	—	—	—	—	—	—
	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 instructions		—	—	—	—	—	—	—	—	—	—	—	—



[Legend]
PUCR: Port pull-up control register
PMR: Port mode register
PDR: Port data register
PCR: Port control register

Figure B.1 Port 1 Block Diagram (P17)



[Legend]
PUCR: Port pull-up control register
PMR: Port mode register
PDR: Port data register
PCR: Port control register

Figure B.2 Port 1 Block Diagram (P14)

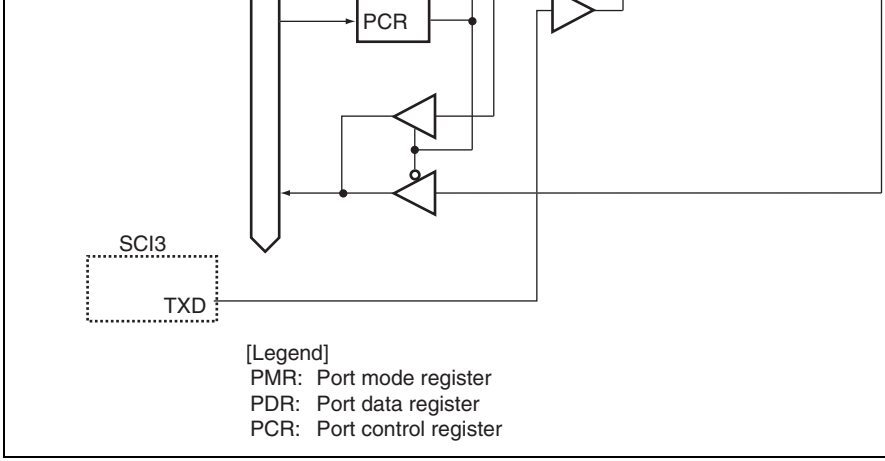


Figure B.3 Port 2 Block Diagram (P22)

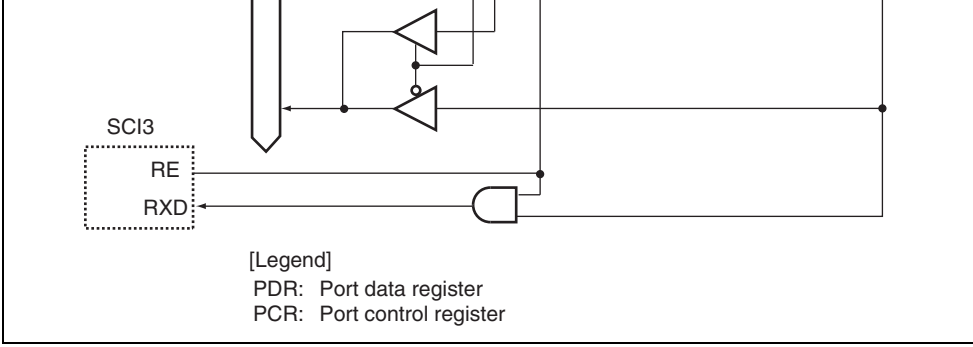
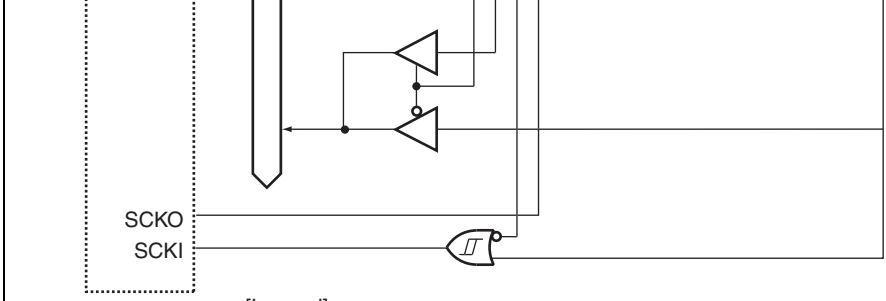
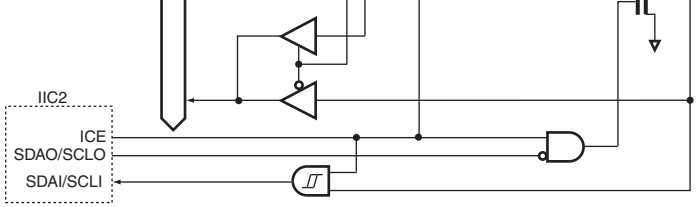


Figure B.4 Port 2 Block Diagram (P21)



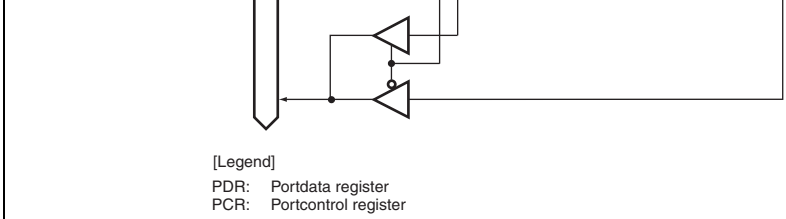
[Legend]
PDR: Port data register
PCR: Port control register

Figure B.5 Port 2 Block Diagram (P20)



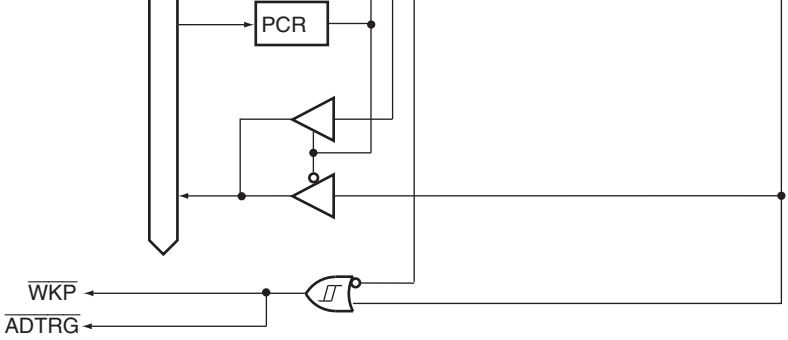
[Legend]
PDR: Portdata register
PCR: Portcontrol register

Figure B.6 (1) Port 5 Block Diagram (P57, P56) (for H8/36912 Group)



[Legend]
PDR: Portdata register
PCR: Portcontrol register

Figure B.6 (2) Port 5 Block Diagram (P57, P56) (for H8/36902 Group)



[Legend]
PUCR: Port pull-up control register
PMR: Port mode register
PDR: Port data register
PCR: Port control register

Figure B.7 Port 5 Block Diagram (P55)

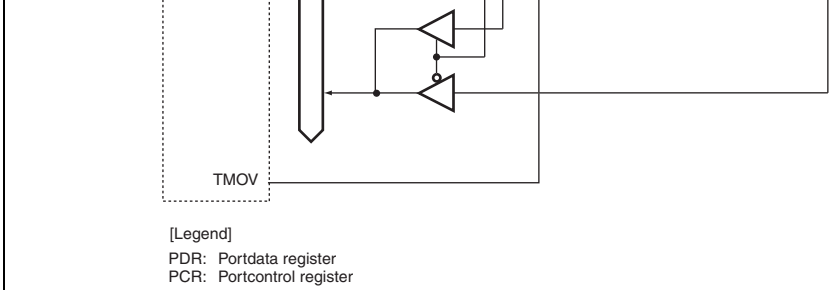
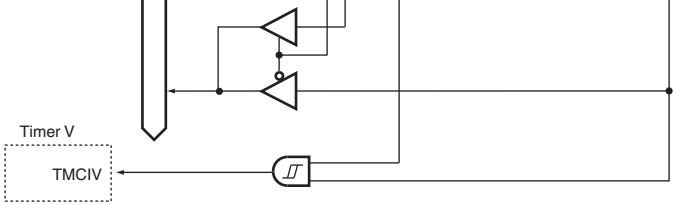
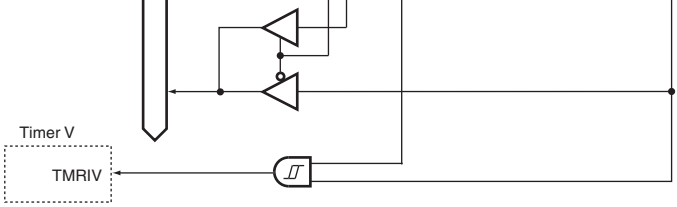


Figure B.8 Port 5 Block Diagram (P76)



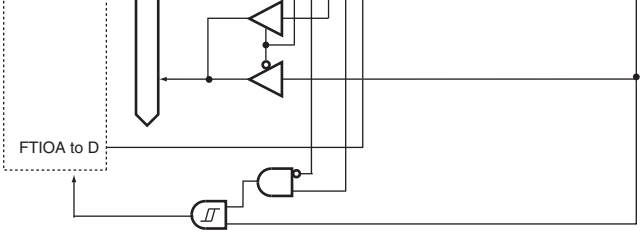
[Legend]
PDR: Portdata register
PCR: Portcontrol register

Figure B.9 Port 7 Block Diagram (P75)



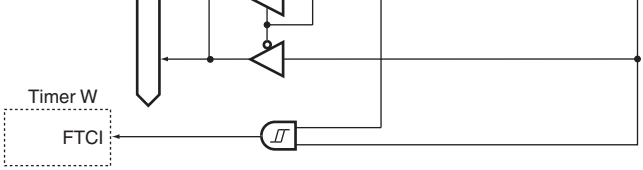
[Legend]
PDR: Portdata register
PCR: Portcontrol register

Figure B.10 Port 7 Block Diagram (P74)



[Legend]
PDR: Portdata register
PCR: Portcontrol register

Figure B.11 Port 8 Block Diagram (P84 to P81)



[Legend]
PDR: Portdata register
PCR: Portcontrol register

Figure B.12 Port 8 Block Diagram (P80)

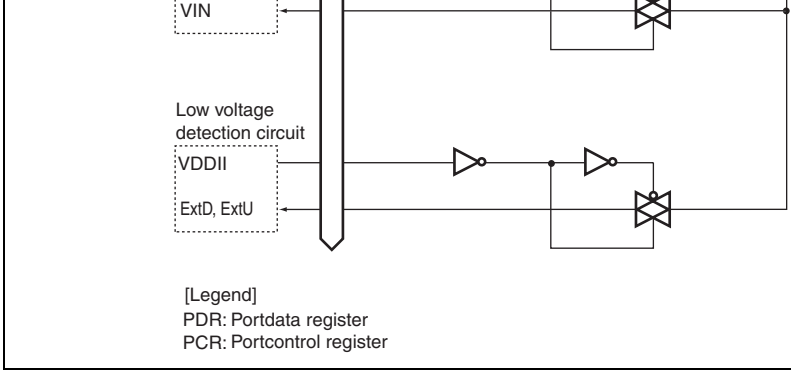


Figure B.13 Port B Block Diagram (PB3, PB2)

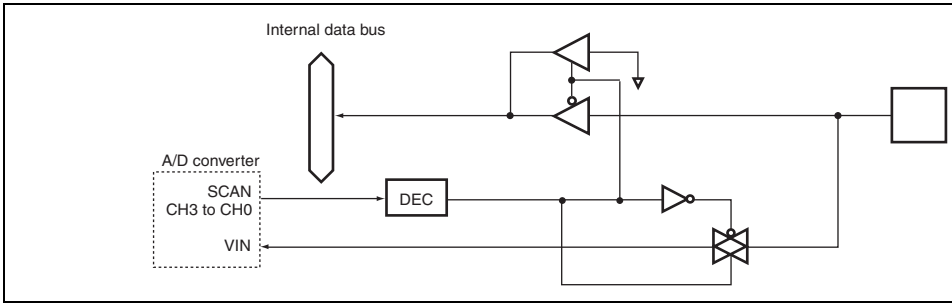
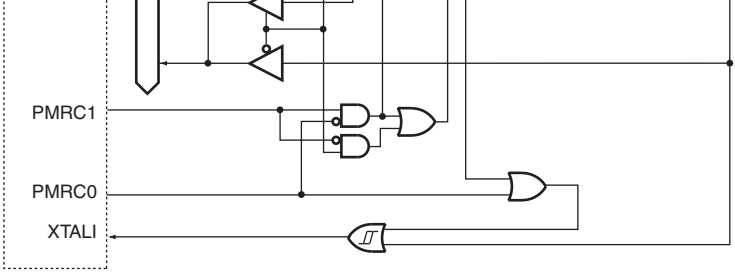


Figure B.14 Port B Block Diagram (PB1, PB0)



[Legend]
PDR: Portdata register
PCR: Portcontrol register

Figure B.15 Port C Block Diagram (PC1)

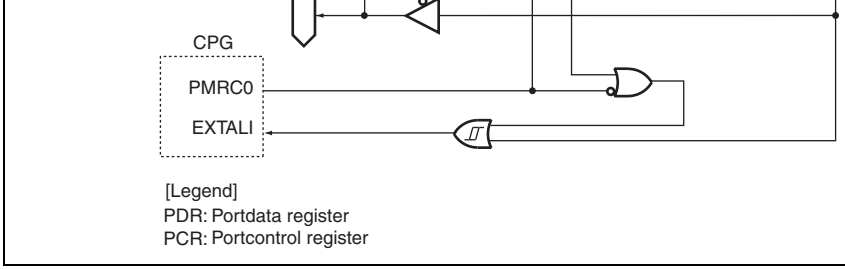


Figure B.16 Port C Block Diagram (PC0)

B.2 Port States in Each Operating State

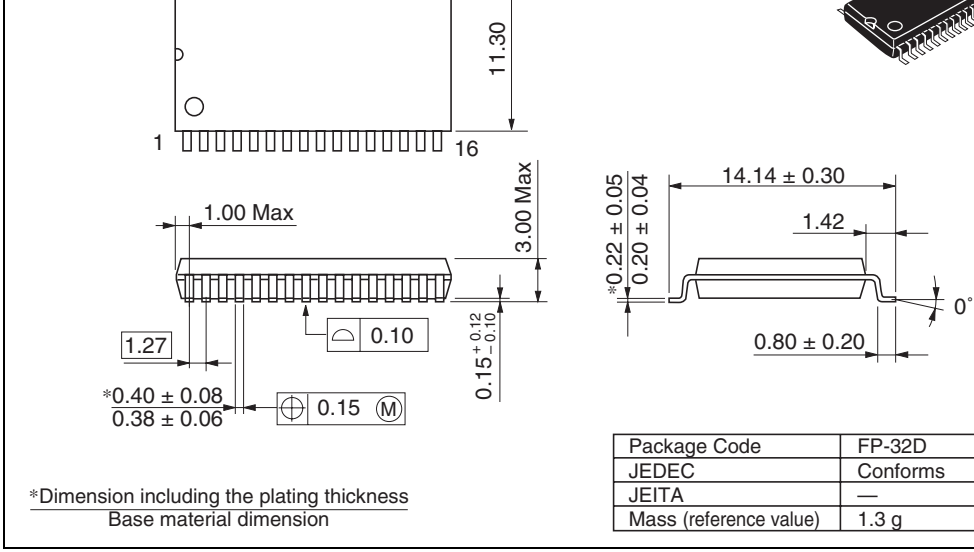
Port	Reset	Active	Sleep	Subsleep	Standby
P17, P14	High impedance	Functioning	Retained	Retained	High imped
P22 to P20	High impedance	Functioning	Retained	Retained	High imped
P57 to P55	High impedance	Functioning	Retained	Retained	High imped
P76 to P74	High impedance	Functioning	Retained	Retained	High imped
P84 to P80	High impedance	Functioning	Retained	Retained	High imped
PB3 to PB0	High impedance	High impedance	High impedance	Retained	High imped
PC1, PC0	High impedance	Functioning	Retained	Retained	High imped

Note: * High level output when the pull-up MOS is in on state.

	version		HD64336911G (***) TP	SOP-32 (FP
H8/36902	Flash memory version	HD64F36902G	HD64F36902GFH	LQFP-32 (F
			HD64F36902GTP	SOP-32 (FP
			HD64F36902GP	SDIP-32 (3
	Masked ROM version	HD64336902G	HD64336902G (***) FH	LQFP-32 (F
			HD64336902G (***) TP	SOP-32 (FP
H8/36901	Masked ROM version	HD64336901G	HD64336901G (***) FH	LQFP-32 (F
			HD64336901G (***) TP	SOP-32 (FP
H8/36900	Masked ROM version	HD64336900G	HD64336900G (***) FH	LQFP-32 (F
			HD64336900G (***) TP	SOP-32 (FP

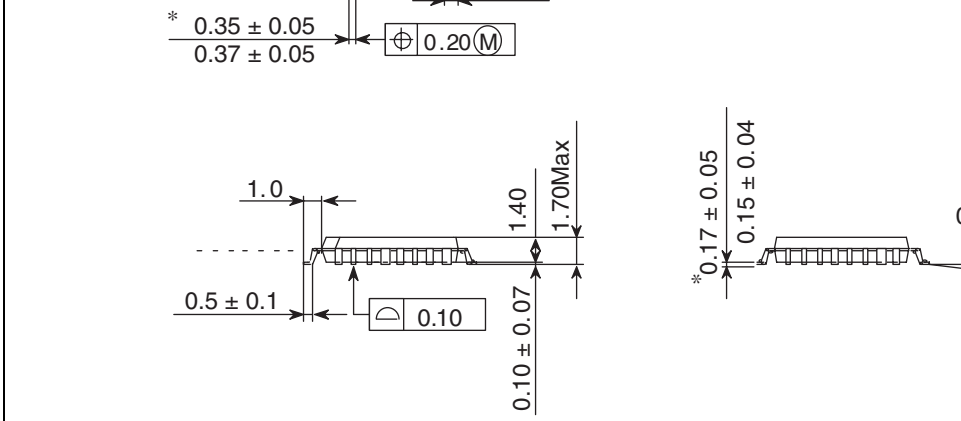
[Legend]

(***): ROM code



Package Code	FP-32D
JEDEC	Conforms
JEITA	—
Mass (reference value)	1.3 g

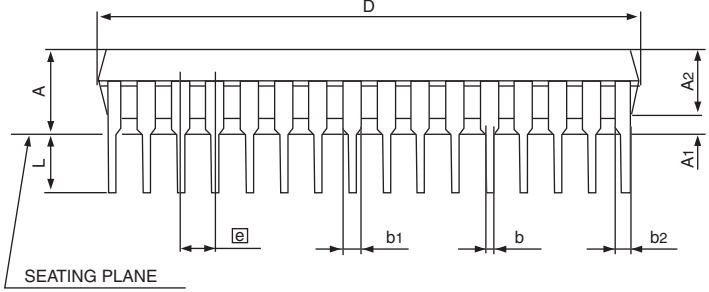
Figure D.1 FP-32D Package Dimensions



*Dimension including the plating thickness
Base material dimension

Package Code	FP-32A FP-32AV
JEDEC	—
JEITA	—
Mass (reference value)	0.2 g

Figure D.2 FP-32A Package Dimension



Symbol	Dimension in Millmeters		
	Min	Nom	Max
A	–	–	5.08
A1	0.51	–	–
A2	–	3.8	–
b	0.35	0.45	0.55
b1	0.9	1.0	1.3
b2	0.63	0.73	1.03
c	0.22	0.27	0.34
D	27.8	28.0	28.2
E	8.75	8.9	9.05
e	–	1.778	–
e1	–	10.16	–
L	3.0	–	–
θ	0°	–	15°

Package Code	32P4B
JEDEC	—
JEITA	—
Mass (reference value)	2.2

Figure D.3 32P4B Package Dimension

4. When the E7 or E8 is used, address breaks can be
either available to the user or for use by the E7 or E
address breaks are set as being used by the E7 or
address break control registers must not be access

5. When the E7 or E8 is used, $\overline{\text{NMI}}$ is an input/output
drain in output mode).

1.1 Features

1

- On-chip memory

Product Classification		Remarks
Masked ROM version	H8/36912	~~Under planning~~
	H8/36911	~~Under planning~~
	H8/36902	~~Under planning~~
	H8/36901	~~Under planning~~
	H8/36900	~~Under planning~~

2

- On-chip oscillator
  — Frequency accuracy: 8MHz ±1% (Typ.) Vcc = 5.0 V, Ta = 25°C
  (Flash memory version): 8MHz ±3% Vcc = 4.0 to 5.0 V, Ta = –
  10MHz ±4% (Typ.) Vcc = 4.0 to 5.0 V, Ta = –

2

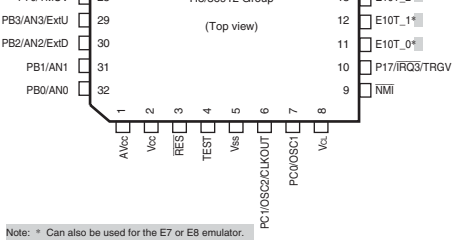
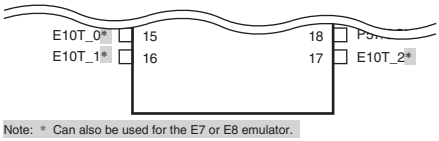
Package	Code
LQFP-32	FP-32A
SOP-32	FP-32D
SDIP-32	32P4B

2

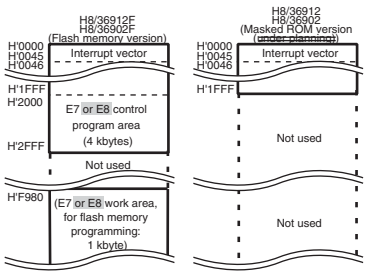
- Compact package

Package
LQFP-32
SOP-32
SDIP-32*

Note: * Flash memory version only

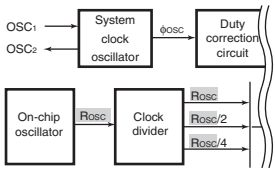
of H8/36912 Group (FP-32A) Figure 1.4 Pin Arrangement of H8/36902 Group (FP-32A)		
Figure 1.5 Pin Arrangement of H8/36912 Group (FP-32D, 32P4B), Figure 1.6 Pin Arrangement of H8/36902 Group (FP-32D, 32P4B)	7, 8	
Table 1.1 Pin Functions	9, 10	

Type	Symbol	Pin No. FP-32D, 32P4B	Pin No. FP-32A	Functi
E7, E8	E10T_0, E10T_1, E10T_2	15, 16, 17	11, 12, 13	Interfac for the emulat

Section 2 CPU	11	• High-speed operation — All frequently-used instructions execute in two or f
Figure 2.1 Memory Map (1)	12	

Note: * Available for the H8/36912 Group only.

Figure 5.1 Block Diagram of Clock Pulse Generators 69



5.2.1 RC Control Register (RCCR) 71

Bit	Bit Name	Description
1	RCPSC1	Division Ratio Select for On-chip Oscilla
0	RCPSC0	The division ratio of R_{osc} changes right a rewriting this bit. These bits can be written to only when t bit in CKCSR is 0. 0X: R_{osc} (not divided) 10: R_{osc}/2 11: R_{osc}/4

5.2.2 RC Trimming Data Protect Register (RCTRMDPR) 73

Bit	Bit Name	Description
4	TRMDRWE	Trimming Date Register Write Enable This register can be written to when the bit is 0 and this bit is 1. [Setting condition] • When writing 0 to the WRI bit while the TRMDRWE bit while the PRWE [Clearing conditions] • Reset • When writing 0 to the WRI bit and w the TRMDRWE bit while the PRWE

		6 PMMD0

5.2.4 Clock Control/Status Register (CKCSR)	74	

Bit	Bit Name	Description
7	PMRC1	Port C Function Select 1 and 0
6	PMRC0	

Figure 5.5 Timing Chart of Switching On-chip Oscillator Clock to External Clock	78	Note: * The ϕ halt duration is the duration from the timing ϕ clock stops to the first rising edge of the ϕ_{OSC} clock after six clock cycles clock have elapsed.

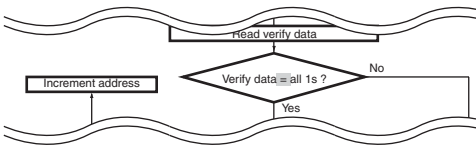
Table 5.1 Crystal Resonator Parameters	82	

Frequency (MHz)	12
R_S (Max.)	50 Ω

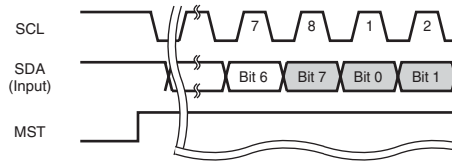
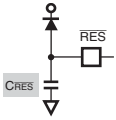
Section 7 ROM	97	The features of the 12-kbyte (including 4 kbytes as the E7 control program area) flash memory built into the HD64F3 and HD64F36902G are summarized below.
Figure 7.1 Flash Memory Block Configuration	98	← Programming unit: 64 kbytes →
Table 7.3 System Clock Frequencies for which Automatic Adjustment of LSI Bit Rate is Possible	105	

Host Bit Rate	System Clock Frequency Range of
9600bps	8 MHz (on-chip oscillator clock)
4800bps	8 MHz (on-chip oscillator clock)
2400bps	8 MHz (on-chip oscillator clock)

Figure 7.4 Erase/Erase-Verify Flowchart	111	



Section 8 RAM	115	Note: * When the E7 or E8 is used, area H'F980 to H'FD not be accessed.

Register (ICSR)		**Bit** **Bit Name** **Description** 3 STOP Stop Condition Detection Flag [Setting conditions] • In master mode, when a stop cond detected after frame transfer • In slave mode, when a stop condit detected after the general call add the first byte slave address, next t detection of start condition, accord address set in SAR ……
Figure 15.15 Receive Mode Operation Timing	265	
15.7 Usage Notes	272	Added
16.3.1 A/D Data Registers A to D (ADDRA to ADDRD)	276	There are four 16-bit read-only ADDR registers; …… Therefore, byte access to ADDR should be done by read upper byte first then the lower one. ADDR is initialized to
Figure 17.2 Block Diagram of Power-On Reset Circuit and Low-Voltage Detection Circuit	287	
20.3 Electrical Characteristics (Masked ROM Version)	331	20.3 Electrical Characteristics (Masked ROM Version) ~~[Preli~~ ~~The guarantee value for the electrical characteristics of mask version is preliminary.~~ 20.3.1 Power Supply Voltage and Operating Ranges

Product Type		Model Marking	Package Code
H8/36912	Flash memory version	HD64F36912GFH	LQFP-32 (FP-32A
		HD64F36912GTP	SOP-32 (FP-32D
		HD64F36912GP	SDIP-32 (32P4B)
	Masked ROM version	HD64336912G(***) FH	LQFP-32 (FP-32A
		HD64336912G(***) TP	SOP-32 (FP-32D
		~~HD64336912G(***) P~~	~~SDIP-32 (32P4B)~~
H8/36911	Masked ROM version	HD64336911G(***) FH	LQFP-32 (FP-32A
		HD64336911G(***) TP	SOP-32 (FP-32D
		~~HD64336911G(***) P~~	~~SDIP-32 (32P4B)~~
H8/36902	Flash memory version	HD64F36902GFH	LQFP-32 (FP-32A
		HD64F36902GTP	SOP-32 (FP-32D
		HD64F36902GP	SDIP-32 (32P4B)
	Masked ROM version	HD64336902G(***) FH	LQFP-32 (FP-32A
		HD64336902G(***) TP	SOP-32 (FP-32D
		~~HD64336902G(***) P~~	~~SDIP-32 (32P4B)~~
H8/36901	Masked ROM version	HD64336901G(***) FH	LQFP-32 (FP-32A
		HD64336901G(***) TP	SOP-32 (FP-32D
		~~HD64336901G(***) P~~	~~SDIP-32 (32P4B)~~
H8/36900	Masked ROM version	HD64336900G(***) FH	LQFP-32 (FP-32A
		HD64336900G(***) TP	SOP-32 (FP-32D
		~~HD64336900G(***) P~~	~~SDIP-32 (32P4B)~~

Package Code	32P4B
JEDEC	—
JEITA	—
Mass (reference value)	2.2

B

C

E

G

I

S

Renesas 16-Bit Single-Chip Microcomputer
Hardware Manual
H8/36912 Group, H8/36902 Group

Publication Date: Rev.1.00, Nov. 07, 2003
Rev.3.00, Sep. 14, 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 Ehsan,
Tel: <603> 7955-9390, Fax: <603> 7955-9510

H8/36912 Group, H8/36902 Group
Hardware Manual

Renesas Electronics Corporation
1753, Shimonumabe, Nakahara-ku, Kawasaki-shi, Kanagawa 211-8668 Japan

REJ09B0105-0300

X-ON Electronics

Largest Supplier of Electrical and Electronic Components

Click to view similar products for 16-bit Microcontrollers - MCU ***category:***

Click to view products by Renesas ***manufacturer:***

Other Similar products are found below :

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