

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.

H8/36049 Group

Hardware Manual

Renesas 16-Bit Single-Chip
Microcomputer

H8 Family/H8/300H Tiny Series

H8/36049F	HD64F36049, HD64F36049G,
H8/36049	HD64336049, HD64336049G,
H8/36048	HD64336048, HD64336048G,
H8/36047	HD64336047, HD64336047G

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

are in their open states, intermediate levels are induced by noise in the vicinity, and 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 throughout the chip and a low level is input on the reset pin. During the period where the states are undefined, the register settings and the output state of each pin are also undefined. Be careful of your system so that it does not malfunction because of processing while it is in the undefined state. For those products which have a reset function, reset the LSI immediately 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 registers that may have been 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 to module. However, the generic style includes the following items:

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

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

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 versions. This does not include all of the revised contents. For details, see the actual locations in the manual.

11. Index

characteristics of the H8/36049 Group to the target users.
Refer to the H8/300H Series Software Manual for a detailed description of the instruction set.

Notes on reading this manual:

- In order to understand the overall functions of the chip
Read the manual according to the contents. This manual can be roughly categorized into the CPU, system control functions, peripheral functions and electrical characteristics.
- 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 for the register. The addresses, bits, and initial values of the registers are summarized in section List of Registers.

Example: Register name: The following notation is used for cases when the same function is implemented on more than one channel:
XXX_N (XXX is the register name and N is the channel number)

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

Number notation: Binary is B'xxxx, hexadecimal is H'xxxx, and decimal is xxxx.

Signal notation: An overbar is added to a low-active signal: $\overline{\text{xxxx}}$

5. When the E7 or E8 is used, $\overline{\text{NMI}}$ is an input/output pin (open-drain in output mode), P87 are input pins, and P86 is an output pin.
6. In on-board programming mode by boot mode, channel 1 (P21/RXD and P22/TXD) is used.

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

H8/36049 Group manuals:

Document Title	Document ID
H8/36049 Group Hardware Manual	This manual
H8/300H Series Software Manual	REJ09B0001

User's manuals for development tools:

Document Title	Document ID
H8S, H8/300 Series C/C++ Compiler, Assembler, Optimizing Linkage Editor User's Manual	REJ10B0001
H8S, H8/300 Series Simulator/Debugger User's Manual	REJ10B0001
H8S, H8/300 Series High-performance Embedded Workshop 3 Tutorial	REJ10B0001
H8S, H8/300 Series High-performance Embedded Workshop 3 User's Manual	REJ10B0001

2.1	Address Space and Memory Map
2.2	Register Configuration.....
2.2.1	General Registers
2.2.2	Program Counter (PC)
2.2.3	Condition-Code Register (CCR).....
2.3	Data Formats.....
2.3.1	General Register Data Formats
2.3.2	Memory Data Formats
2.4	Instruction Set.....
2.4.1	List of Instructions Classified by Function
2.4.2	Basic Instruction Formats
2.5	Addressing Modes and Effective Address Calculation.....
2.5.1	Addressing Modes
2.5.2	Effective Address Calculation
2.6	Basic Bus Cycle
2.6.1	Access to On-Chip Memory (RAM, ROM).....
2.6.2	On-Chip Peripheral Modules
2.7	CPU States
2.8	Usage Notes
2.8.1	Notes on Data Access to Empty Areas
2.8.2	EPMOV Instruction.....
2.8.3	Bit Manipulation Instruction.....
Section 3 Exception Handling	
3.1	Exception Sources and Vector Address
3.2	Register Descriptions.....
3.2.1	Interrupt Edge Select Register 1 (IEGR1)
3.2.2	Interrupt Edge Select Register 2 (IEGR2)
3.2.3	Interrupt Enable Register 1 (IENR1)

3.5.1	Interrupts after Reset.....
3.5.2	Notes on Stack Area Use
3.5.3	Notes on Rewriting Port Mode Registers
Section 4 Address Break	
4.1	Register Descriptions.....
4.1.1	Address Break Control Register (ABRKCR)
4.1.2	Address Break Status Register (ABRKSR)
4.1.3	Break Address Registers E, H, L (BARE, BARH, BARL)
4.1.4	Break Data Registers H, L (BDRH, BDRL).....
4.2	Operation
Section 5 Clock Pulse Generators	
5.1	System Clock Generator
5.1.1	Connecting Crystal Resonator
5.1.2	Connecting Ceramic Resonator
5.1.3	External Clock Input Method.....
5.2	Subclock Generator
5.2.1	Connecting 32.768-kHz Crystal Resonator
5.2.2	Pin Connection when not Using Subclock.....
5.3	Prescalers
5.3.1	Prescaler S
5.3.2	Prescaler W
5.4	Usage Notes
5.4.1	Notes on Resonators
5.4.2	Notes on Board Design

6.2.4	Subactive Mode
6.3	Operating Frequency in Active Mode.....
6.4	Direct Transition.....
6.4.1	Direct Transition from Active Mode to Subactive Mode.....
6.4.2	Direct Transition from Subactive Mode to Active Mode.....
6.5	Module Standby Function.....
Section 7 ROM	
7.1	Block Configuration
7.2	Register Descriptions.....
7.2.1	Flash Memory Control Register 1 (FLMCR1).....
7.2.2	Flash Memory Control Register 2 (FLMCR2).....
7.2.3	Erase Block Register 1 (EBR1)
7.2.4	Flash Memory Power Control Register (FLPWCR).....
7.2.5	Flash Memory Enable Register (FENR).....
7.3	On-Board Programming Modes.....
7.3.1	Boot Mode
7.3.2	Programming/Erasing in User Program Mode.....
7.4	Flash Memory Programming/Erasing.....
7.4.1	Program/Program-Verify
7.4.2	Erase/Erase-Verify.....
7.4.3	Interrupt Handling when Programming/Erasing Flash Memory.....
7.5	Program/Erase Protection
7.5.1	Hardware Protection
7.5.2	Software Protection.....
7.5.3	Error Protection.....
7.6	Programmer Mode.....
7.7	Power-Down States for Flash Memory.....

	9.2.2	Port Data Register 2 (PDR2)
	9.2.3	Port Mode Register 3 (PMR3)
	9.2.4	Pin Functions
9.3	Port 3.....	
	9.3.1	Port Control Register 3 (PCR3)
	9.3.2	Port Data Register 3 (PDR3)
	9.3.3	Pin Functions
9.4	Port 5.....	
	9.4.1	Port Mode Register 5 (PMR5)
	9.4.2	Port Control Register 5 (PCR5)
	9.4.3	Port Data Register 5 (PDR5)
	9.4.4	Port Pull-Up Control Register 5 (PUCR5).....
	9.4.5	Pin Functions
9.5	Port 6.....	
	9.5.1	Port Control Register 6 (PCR6)
	9.5.2	Port Data Register 6 (PDR6)
	9.5.3	Pin Functions
9.6	Port 7.....	
	9.6.1	Port Control Register 7 (PCR7)
	9.6.2	Port Data Register 7 (PDR7)
	9.6.3	Pin Functions
9.7	Port 8.....	
	9.7.1	Port Control Register 8 (PCR8)
	9.7.2	Port Data Register 8 (PDR8)
	9.7.3	Pin Functions
9.8	Port 9.....	
	9.8.1	Port Control Register 9 (PCR9)
	9.8.2	Port Data Register 9 (PDR9)
	9.8.3	Pin Functions

10.3.5	RTC Control Register 1 (RTCCR1).....
10.3.6	RTC Control Register 2 (RTCCR2).....
10.3.7	Clock Source Select Register (RTCCSR).....
10.4	Operation
10.4.1	Initial Settings of Registers after Power-On
10.4.2	Initial Setting Procedure
10.4.3	Data Reading Procedure
10.5	Interrupt Sources.....
Section 11 Timer B1	
11.1	Features.....
11.2	Input/Output Pin
11.3	Register Descriptions.....
11.3.1	Timer Mode Register B1 (TMB1).....
11.3.2	Timer Counter B1 (TCB1).....
11.3.3	Timer Load Register B1 (TLB1).....
11.4	Operation
11.4.1	Interval Timer Operation
11.4.2	Auto-Reload Timer Operation
11.4.3	Event Counter Operation
11.5	Timer B1 Operating Modes
Section 12 Timer V.....	
12.1	Features.....
12.2	Input/Output Pins.....
12.3	Register Descriptions.....
12.3.1	Timer Counter V (TCNTV).....
12.3.2	Time Constant Registers A, B (TCORA, TCORB)
12.3.3	Timer Control Register V0 (TCRV0).....

13.2	Input/Output Pins.....
13.3	Register Descriptions.....
13.3.1	Timer Mode Register W (TMRW).....
13.3.2	Timer Control Register W (TCRW).....
13.3.3	Timer Interrupt Enable Register W (TIERW).....
13.3.4	Timer Status Register W (TSRW).....
13.3.5	Timer I/O Control Register 0 (TIOR0).....
13.3.6	Timer I/O Control Register 1 (TIOR1).....
13.3.7	Timer Counter (TCNT).....
13.3.8	General Registers A to D (GRA to GRD).....
13.4	Operation.....
13.4.1	Normal Operation.....
13.4.2	PWM Operation.....
13.5	Operation Timing.....
13.5.1	TCNT Count Timing.....
13.5.2	Output Compare Timing.....
13.5.3	Input Capture Timing.....
13.5.4	Timing of Counter Clearing by Compare Match.....
13.5.5	Buffer Operation Timing.....
13.5.6	Timing of IMFA to IMFD Flag Setting at Compare Match.....
13.5.7	Timing of IMFA to IMFD Setting at Input Capture.....
13.5.8	Timing of Status Flag Clearing.....
13.6	Usage Notes.....
Section 14 Timer Z.....	
14.1	Features.....
14.2	Input/Output Pins.....
14.3	Register Descriptions.....
14.3.1	Timer Start Register (TSTR).....

	14.3.13	PWM Mode Output Level Control Register (POCR)
	14.3.14	Interface with CPU
14.4	Operation	
	14.4.1	Counter Operation.....
	14.4.2	Waveform Output by Compare Match.....
	14.4.3	Input Capture Function
	14.4.4	Synchronous Operation.....
	14.4.5	PWM Mode
	14.4.6	Reset Synchronous PWM Mode.....
	14.4.7	Complementary PWM Mode.....
	14.4.8	Buffer Operation
	14.4.9	Timer Z Output Timing
14.5	Interrupts.....	
	14.5.1	Status Flag Set Timing.....
	14.5.2	Status Flag Clearing Timing
14.6	Usage Notes	
Section 15 Watchdog Timer		
15.1	Features.....	
15.2	Register Descriptions.....	
	15.2.1	Timer Control/Status Register WD (TCSRWD).....
	15.2.2	Timer Counter WD (TCWD).....
	15.2.3	Timer Mode Register WD (TMWD)
15.3	Operation	
Section 16 14-Bit PWM.....		
16.1	Features.....	
16.2	Input/Output Pin	
16.3	Register Descriptions.....	

	17.3.4	Transmit Data Register (TDR).....
	17.3.5	Serial Mode Register (SMR)
	17.3.6	Serial Control Register 3 (SCR3)
	17.3.7	Serial Status Register (SSR)
	17.3.8	Bit Rate Register (BRR)
17.4		Operation in Asynchronous Mode
	17.4.1	Clock.....
	17.4.2	SCI3 Initialization.....
	17.4.3	Data Transmission
	17.4.4	Serial Data Reception
17.5		Operation in Clocked Synchronous Mode
	17.5.1	Clock.....
	17.5.2	SCI3 Initialization.....
	17.5.3	Serial Data Transmission
	17.5.4	Serial Data Reception (Clocked Synchronous Mode)
	17.5.5	Simultaneous Serial Data Transmission and Reception.....
17.6		Multiprocessor Communication Function
	17.6.1	Multiprocessor Serial Data Transmission.....
	17.6.2	Multiprocessor Serial Data Reception
17.7		Interrupt Requests
17.8		Usage Notes
	17.8.1	Break Detection and Processing
	17.8.2	Mark State and Break Sending
	17.8.3	Receive Error Flags and Transmit Operations (Clocked Synchronous Mode Only).....
	17.8.4	Receive Data Sampling Timing and Reception Margin in Asynchronous Mode.....

18.3.7	I ² C Bus Transmit Data Register (ICDRT).....
18.3.8	I ² C Bus Receive Data Register (ICDRT).....
18.3.9	I ² C Bus Shift Register (ICDRS).....
18.4	Operation
18.4.1	I ² C Bus Format.....
18.4.2	Master Transmit Operation
18.4.3	Master Receive Operation.....
18.4.4	Slave Transmit Operation
18.4.5	Slave Receive Operation.....
18.4.6	Clocked Synchronous Serial Format.....
18.4.7	Noise Canceller.....
18.4.8	Example of Use.....
18.5	Interrupts.....
18.6	Bit Synchronous Circuit.....
18.7	Usage Notes
18.7.1	Issue (Retransmission) of Start/Stop Conditions
18.7.2	WAIT Setting in I ² C Bus Mode Register (ICMR)
Section 19 A/D Converter.....	
19.1	Features.....
19.2	Input/Output Pins
19.3	Register Descriptions.....
19.3.1	A/D Data Registers A to D (ADDRA to ADDRD)
19.3.2	A/D Control/Status Register (ADCSR)
19.3.3	A/D Control Register (ADCR)
19.4	Operation
19.4.1	Single Mode.....
19.4.2	Scan Mode
19.4.3	Input Sampling and A/D Conversion Time

20.2.2	Low-Voltage-Detection Status Register (LVDSR).....
20.3	Operation
20.3.1	Power-On Reset Circuit
20.3.2	Low-Voltage Detection Circuit.....
Section 21	Power Supply Circuit
21.1	When Using Internal Power Supply Step-Down Circuit
21.2	When Not Using Internal Power Supply Step-Down Circuit
Section 22	List of Registers.....
22.1	Register Addresses (Address Order).....
22.2	Register Bits
22.3	Register States in Each Operating Mode
Section 23	Electrical Characteristics
23.1	Absolute Maximum Ratings
23.2	Electrical Characteristics (F-ZTAT™ Version).....
23.2.1	Power Supply Voltage and Operating Ranges
23.2.2	DC Characteristics
23.2.3	AC Characteristics
23.2.4	A/D Converter Characteristics
23.2.5	Watchdog Timer Characteristics.....
23.2.6	Flash Memory Characteristics
23.2.7	Power-Supply-Voltage Detection Circuit Characteristics (Optional)
23.2.8	Power-On Reset Circuit Characteristics (Optional).....
23.3	Electrical Characteristics (Masked ROM Version).....
23.3.1	Power Supply Voltage and Operating Ranges
23.3.2	DC Characteristics
23.3.3	AC Characteristics

- A.3 Number of Execution States
- A.4 Combinations of Instructions and Addressing Modes
- B. I/O Ports.....
 - B.1 I/O Port Block Diagrams
 - B.2 Port States in Each Operating Mode
- C. Product Code Lineup
- D. Package Dimensions

- Main Revisions and Additions in this Edition

- Index

Figure 2.4	Relationship between Stack Pointer and Stack Area.....
Figure 2.5	General Register Data Formats (1).....
Figure 2.5	General Register Data Formats (2).....
Figure 2.6	Memory Data Formats.....
Figure 2.7	Instruction Formats.....
Figure 2.8	Branch Address Specification in Memory Indirect Mode.....
Figure 2.9	On-Chip Memory Access Cycle.....
Figure 2.10	On-Chip Peripheral Module Access Cycle (3-State Access).....
Figure 2.11	CPU Operation States.....
Figure 2.12	State Transitions.....
Figure 2.13	Example of Timer Configuration with Two Registers Allocated to Same Address.....

Section 3 Exception Handling

Figure 3.1	Reset Sequence.....
Figure 3.2	Stack Status after Exception Handling.....
Figure 3.3	Interrupt Sequence.....
Figure 3.4	Port Mode Register Setting and Interrupt Request Flag Clearing Procedure.....

Section 4 Address Break

Figure 4.1	Block Diagram of Address Break.....
Figure 4.2	Address Break Interrupt Operation Example (1).....
Figure 4.2	Address Break Interrupt Operation Example (2).....

Section 5 Clock Pulse Generators

Figure 5.1	Block Diagram of Clock Pulse Generators.....
Figure 5.2	Block Diagram of System Clock Generator.....
Figure 5.3	Typical Connection to Crystal Resonator.....
Figure 5.4	Equivalent Circuit of Crystal Resonator.....
Figure 5.5	Typical Connection to Ceramic Resonator.....
Figure 5.6	Example of External Clock Input.....

Figure 7.3	Program/Program-Verify Flowchart
Figure 7.4	Erase/Erase-Verify Flowchart
Section 9 I/O Ports	
Figure 9.1	Port 1 Pin Configuration
Figure 9.2	Port 2 Pin Configuration
Figure 9.3	Port 3 Pin Configuration
Figure 9.4	Port 5 Pin Configuration
Figure 9.5	Port 6 Pin Configuration
Figure 9.6	Port 7 Pin Configuration
Figure 9.7	Port 8 Pin Configuration
Figure 9.8	Port 9 Pin Configuration
Figure 9.9	Port B Pin Configuration
Section 10 Realtime Clock (RTC)	
Figure 10.1	Block Diagram of RTC
Figure 10.2	Definition of Time Expression
Figure 10.3	Initial Setting Procedure
Figure 10.4	Example: Reading of Inaccurate Time Data
Section 11 Timer B1	
Figure 11.1	Block Diagram of Timer B1
Section 12 Timer V	
Figure 12.1	Block Diagram of Timer V
Figure 12.2	Increment Timing with Internal Clock
Figure 12.3	Increment Timing with External Clock
Figure 12.4	OVF Set Timing
Figure 12.5	CMFA and CMFB Set Timing
Figure 12.6	TMOV Output Timing
Figure 12.7	Clear Timing by Compare Match

Figure 13.4	0 and 1 Output Example (TOA = 0, TOB = 1)
Figure 13.5	Toggle Output Example (TOA = 0, TOB = 1)
Figure 13.6	Toggle Output Example (TOA = 0, TOB = 1)
Figure 13.7	Input Capture Operating Example
Figure 13.8	Buffer Operation Example (Input Capture)
Figure 13.9	PWM Mode Example (1)
Figure 13.10	PWM Mode Example (2)
Figure 13.11	Buffer Operation Example (Output Compare)
Figure 13.12	PWM Mode Example (TOB = 0, TOC = 0, TOD = 0: Initial Output Values are Set to 0)
Figure 13.13	PWM Mode Example (TOB = 1, TOC = 1, and TOD = 1: Initial Output Values are Set to 1)
Figure 13.14	Count Timing for Internal Clock Source
Figure 13.15	Count Timing for External Clock Source
Figure 13.16	Output Compare Output Timing
Figure 13.17	Input Capture Input Signal Timing
Figure 13.18	Timing of Counter Clearing by Compare Match
Figure 13.19	Buffer Operation Timing (Compare Match)
Figure 13.20	Buffer Operation Timing (Input Capture)
Figure 13.21	Timing of IMFA to IMFD Flag Setting at Compare Match
Figure 13.22	Timing of IMFA to IMFD Flag Setting at Input Capture
Figure 13.23	Timing of Status Flag Clearing by CPU
Figure 13.24	Contention between TCNT Write and Clear
Figure 13.25	Internal Clock Switching and TCNT Operation
Figure 13.26	When Compare Match and Bit Manipulation Instruction to TCRW Occur Same Timing

Section 14 Timer Z

Figure 14.1	Timer Z Block Diagram
Figure 14.2	Timer Z (Channel 0) Block Diagram

Figure 14.13	Example of 0 Output/1 Output Operation
Figure 14.14	Example of Toggle Output Operation
Figure 14.15	Output Compare Timing
Figure 14.16	Example of Input Capture Operation Setting Procedure
Figure 14.17	Example of Input Capture Operation
Figure 14.18	Input Capture Signal Timing.....
Figure 14.19	Example of Synchronous Operation Setting Procedure
Figure 14.20	Example of Synchronous Operation.....
Figure 14.21	Example of PWM Mode Setting Procedure
Figure 14.22	Example of PWM Mode Operation (1)
Figure 14.23	Example of PWM Mode Operation (2)
Figure 14.24	Example of PWM Mode Operation (3)
Figure 14.25	Example of PWM Mode Operation (4)
Figure 14.26	Example of Reset Synchronous PWM Mode Setting Procedure.....
Figure 14.27	Example of Reset Synchronous PWM Mode Operation (OLS0 = OLS1 = 1)
Figure 14.28	Example of Reset Synchronous PWM Mode Operation (OLS0 = OLS1 = 0)
Figure 14.29	Example of Complementary PWM Mode Setting Procedure.....
Figure 14.30	Canceling Procedure of Complementary PWM Mode.....
Figure 14.31	Example of Complementary PWM Mode Operation (1)
Figure 14.32 (1)	Example of Complementary PWM Mode Operation (TPSC2 = TPSC1 = TPSC0 = 0) (2).....
Figure 14.32 (2)	Example of Complementary PWM Mode Operation (TPSC2 = TPSC1 = TPSC0 ≠ 0) (3).....
Figure 14.33	Timing of Overshooting.....
Figure 14.34	Timing of Undershooting.....
Figure 14.35	Compare Match Buffer Operation.....
Figure 14.36	Input Capture Buffer Operation.....
Figure 14.37	Example of Buffer Operation Setting Procedure.....

Figure 14.45	Example of Output Disable Timing of Timer Z by External Trigger
Figure 14.46	Example of Output Inverse Timing of Timer Z by Writing to TFCR
Figure 14.47	Example of Output Inverse Timing of Timer Z by Writing to POCR
Figure 14.48	IMF Flag Set Timing when Compare Match Occurs
Figure 14.49	IMF Flag Set Timing at Input Capture
Figure 14.50	OVF Flag Set Timing
Figure 14.51	Status Flag Clearing Timing
Figure 14.52	Contention between TCNT Write and Clear Operations
Figure 14.53	Contention between TCNT Write and Increment Operations
Figure 14.54	Contention between GR Write and Compare Match
Figure 14.55	Contention between TCNT Write and Overflow
Figure 14.56	Contention between GR Read and Input Capture
Figure 14.57	Contention between Count Clearing and Increment Operations by Input Capture
Figure 14.58	Contention between GR Write and Input Capture
Figure 14.59	When Compare Match and Bit Manipulation Instruction to TOCR Occur Same Timing

Section 15 Watchdog Timer

Figure 15.1	Block Diagram of Watchdog Timer
Figure 15.2	Watchdog Timer Operation Example

Section 16 14-Bit PWM

Figure 16.1	Block Diagram of 14-Bit PWM
Figure 16.2	Waveform Output by 14-Bit PWM

Section 17 Serial Communication Interface 3 (SCI3)

Figure 17.1	Block Diagram of SCI3
Figure 17.2	Data Format in Asynchronous Communication
Figure 17.3	Relationship between Output Clock and Transfer Data Phase (Asynchronous Mode) (Example with 8-Bit Data, Parity, Two Stop Bits) ..

Figure 17.12	Example of SCI3 Reception in Clocked Synchronous Mode.....
Figure 17.13	Sample Serial Reception Flowchart (Clocked Synchronous Mode).....
Figure 17.14	Sample Flowchart of Simultaneous Serial Transmit and Receive Operation (Clocked Synchronous Mode).....
Figure 17.15	Example of Inter-Processor Communication Using Multiprocessor Format (Transmission of Data H'AA to Receiving Station A).....
Figure 17.16	Sample Multiprocessor Serial Transmission Flowchart.....
Figure 17.17	Sample Multiprocessor Serial Reception Flowchart (1).....
Figure 17.17	Sample Multiprocessor Serial Reception Flowchart (2).....
Figure 17.18	Example of SCI3 Reception Using Multiprocessor Format (Example with 8-Bit Data, Multiprocessor Bit, One Stop Bit).....
Figure 17.19	Receive Data Sampling Timing in Asynchronous Mode.....

Section 18 I²C Bus Interface 2 (IIC2)

Figure 18.1	Block Diagram of I ² C Bus Interface 2.....
Figure 18.2	External Circuit Connections of I/O Pins.....
Figure 18.3	I ² C Bus Formats.....
Figure 18.4	I ² C Bus Timing.....
Figure 18.5	Master Transmit Mode Operation Timing (1).....
Figure 18.6	Master Transmit Mode Operation Timing (2).....
Figure 18.7	Master Receive Mode Operation Timing (1).....
Figure 18.8	Master Receive Mode Operation Timing (2).....
Figure 18.9	Slave Transmit Mode Operation Timing (1).....
Figure 18.10	Slave Transmit Mode Operation Timing (2).....
Figure 18.11	Slave Receive Mode Operation Timing (1).....
Figure 18.12	Slave Receive Mode Operation Timing (2).....
Figure 18.13	Clocked Synchronous Serial Transfer Format.....
Figure 18.14	Transmit Mode Operation Timing.....
Figure 18.15	Receive Mode Operation Timing.....
Figure 18.16	Block Diagram of Noise Canceller.....

Figure 19.4	A/D Conversion Accuracy Definitions (2)
Figure 19.5	Analog Input Circuit Example
Section 20 Power-On Reset and Low-Voltage Detection Circuits (Optional)		
Figure 20.1	Block Diagram of Power-On Reset Circuit and Low-Voltage Detection Circuit
Figure 20.2	Operational Timing of Power-On Reset Circuit
Figure 20.3	Operational Timing of LVDR Circuit
Figure 20.4	Operational Timing of LVDI Circuit
Figure 20.5	Timing for Operation/Release of Low-Voltage Detection Circuit
Section 21 Power Supply Circuit		
Figure 21.1	Power Supply Connection when Internal Step-Down Circuit is Used
Figure 21.2	Power Supply Connection when Internal Step-Down Circuit is Not Used
Section 23 Electrical Characteristics		
Figure 23.1	System Clock Input Timing
Figure 23.2	$\overline{\text{RES}}$ Low Width Timing
Figure 23.3	Input Timing
Figure 23.4	I ² C Bus Interface Input/Output Timing
Figure 23.5	SCK3 Input Clock Timing
Figure 23.6	SCI Input/Output Timing in Clocked Synchronous Mode
Figure 23.7	Output Load Circuit
Appendix		
Figure B.1	Port 1 Block Diagram (P17)
Figure B.2	Port 1 Block Diagram (P14, P16)
Figure B.3	Port 1 Block Diagram (P15)
Figure B.4	Port 1 Block Diagram (P12)
Figure B.5	Port 1 Block Diagram (P11)
Figure B.6	Port 1 Block Diagram (P10)
Figure B.7	Port 2 Block Diagram (P24, P23)

Figure B.19	Port 7 Block Diagram (P74)
Figure B.20	Port 7 Block Diagram (P72)
Figure B.21	Port 7 Block Diagram (P71)
Figure B.22	Port 7 Block Diagram (P70)
Figure B.23	Port 8 Block Diagram (P87 to P85)
Figure B.24	Port 8 Block Diagram (P84 to P81)
Figure B.25	Port8 Block Diagram (P80)
Figure B.26	Port 9 Block Diagram (P97 to P93)
Figure B.27	Port 9 Block Diagram (P92)
Figure B.28	Port 9 Block Diagram (P91)
Figure B.29	Port 9 Block Diagram (P90)
Figure B.30	Port B Block Diagram (PB7 to PB0)
Figure D.1	FP-80A Package Dimensions

Table 2.4	Logic Operations Instructions.....
Table 2.5	Shift Instructions.....
Table 2.6	Bit Manipulation Instructions (1).....
Table 2.6	Bit Manipulation Instructions (2).....
Table 2.7	Branch Instructions.....
Table 2.8	System Control Instructions.....
Table 2.9	Block Data Transfer Instructions.....
Table 2.10	Addressing Modes.....
Table 2.11	Absolute Address Access Ranges.....
Table 2.12	Effective Address Calculation (1).....
Table 2.12	Effective Address Calculation (2).....

Section 3 Exception Handling

Table 3.1	Exception Sources and Vector Address.....
Table 3.2	Interrupt Wait States.....

Section 4 Address Break

Table 4.1	Access and Data Bus Used.....
-----------	-------------------------------

Section 5 Clock Pulse Generators

Table 5.1	Crystal Resonator Parameters.....
-----------	-----------------------------------

Section 6 Power-Down Modes

Table 6.1	Operating Frequency and Waiting Time.....
Table 6.2	Transition Mode after SLEEP Instruction Execution and Transition Mode Interrupt.....
Table 6.3	Internal State in Each Operating Mode.....

Section 7 ROM

Table 7.1	Setting Programming Modes.....
Table 7.2	Boot Mode Operation.....

Table 11.1	Pin Configuration.....
Table 11.2	Timer B1 Operating Modes.....
Section 12 Timer V	
Table 12.1	Pin Configuration.....
Table 12.2	Clock Signals to Input to TCNTV and Counting Conditions
Section 13 Timer W	
Table 13.1	Timer W Functions.....
Table 13.2	Pin Configuration.....
Section 14 Timer Z	
Table 14.1	Timer Z Functions
Table 14.2	Pin Configuration.....
Table 14.3	Initial Output Level of FTIOB0 Pin.....
Table 14.4	Output Pins in Reset Synchronous PWM Mode.....
Table 14.5	Register Settings in Reset Synchronous PWM Mode.....
Table 14.6	Output Pins in Complementary PWM Mode.....
Table 14.7	Register Settings in Complementary PWM Mode.....
Table 14.8	Register Combinations in Buffer Operation
Section 16 14-Bit PWM	
Table 16.1	Pin Configuration.....
Section 17 Serial Communication Interface 3 (SCI3)	
Table 17.1	Channel Configuration.....
Table 17.2	Pin Configuration.....
Table 17.3	Examples of BRR Settings for Various Bit Rates (Asynchronous Mode) (1)
Table 17.3	Examples of BRR Settings for Various Bit Rates (Asynchronous Mode) (2)
Table 17.3	Examples of BRR Settings for Various Bit Rates (Asynchronous Mode) (3)
Table 17.4	Maximum Bit Rate for Each Frequency (Asynchronous Mode)

Table 18.4	Time for Monitoring SCL.....
Section 19 A/D Converter	
Table 19.1	Pin Configuration.....
Table 19.2	Analog Input Channels and Corresponding ADDR Registers
Table 19.3	A/D Conversion Time (Single Mode).....
Section 20 Power-On Reset and Low-Voltage Detection Circuits (Optional)	
Table 20.1	LVDCR Settings and Select Functions.....
Section 23 Electrical Characteristic	
Table 23.1	Absolute Maximum Ratings
Table 23.2	DC Characteristics (1).....
Table 23.2	DC Characteristics (2).....
Table 23.3	AC Characteristics
Table 23.4	I ² C Bus Interface Timing.....
Table 23.5	Serial Communication Interface (SCI) Timing.....
Table 23.6	A/D Converter Characteristics
Table 23.7	Watchdog Timer Characteristics.....
Table 23.8	Flash Memory Characteristics
Table 23.9	Power-Supply-Voltage Detection Circuit Characteristics.....
Table 23.10	Power-On Reset Circuit Characteristics.....
Table 23.11	DC Characteristics (1).....
Table 23.11	DC Characteristics (2).....
Table 23.12	AC Characteristics
Table 23.13	I ² C Bus Interface Timing.....
Table 23.14	Serial Communication Interface (SCI) Timing.....
Table 23.15	A/D Converter Characteristics
Table 23.16	Watchdog Timer Characteristics.....
Table 23.17	Power-Supply-Voltage Detection Circuit Characteristics.....
Table 23.18	Power-On Reset Circuit Characteristics.....

RTC (can be used as a free running counter)

Timer B1 (8-bit timer)

Timer V (8-bit timer)

Timer W (16-bit timer)

Timer Z (16-bit timer)

14-bit PWM

Watchdog timer

SCI3 (Asynchronous or clocked synchronous serial communication interface) × 3 ch

I²C bus interface 2 (conforms to the I²C bus interface format that is advocated by Philips Electronics)

10-bit A/D converter

POR/LVD (Power-on reset and low voltage detection circuit)

- On-chip memory

Product Classification		Standard Version	Model		
			On-Chip Power-On Reset and Low-Voltage Detecting Circuit Version	ROM	RAM
Flash memory version (F-ZTAT™ version)	H8/36049F	HD64F36049	HD64F36049G	96 kbytes	4 kb
Masked ROM version	H8/36049	HD64336049	HD64336049G	96 kbytes	3 kb
	H8/36048	HD64336048	HD64336048G	80 kbytes	3 kb
	H8/36047	HD64336047	HD64336047G	64 kbytes	3 kb

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

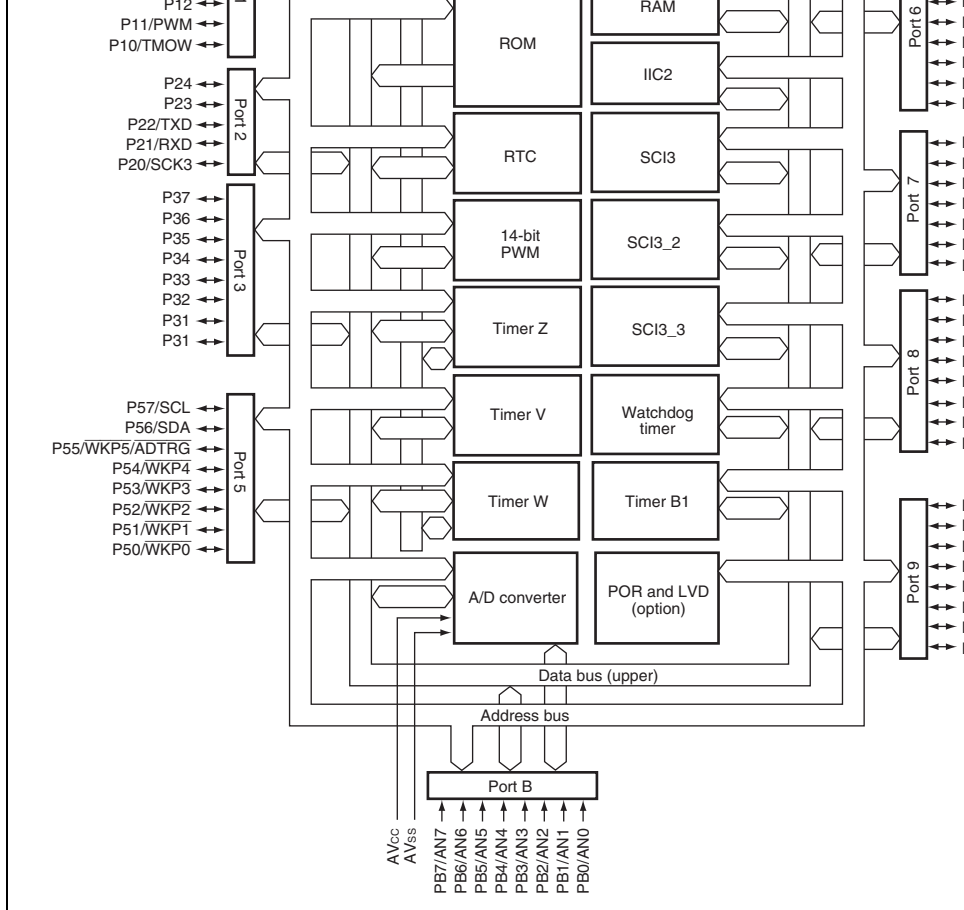


Figure 1.1 Internal Block Diagram

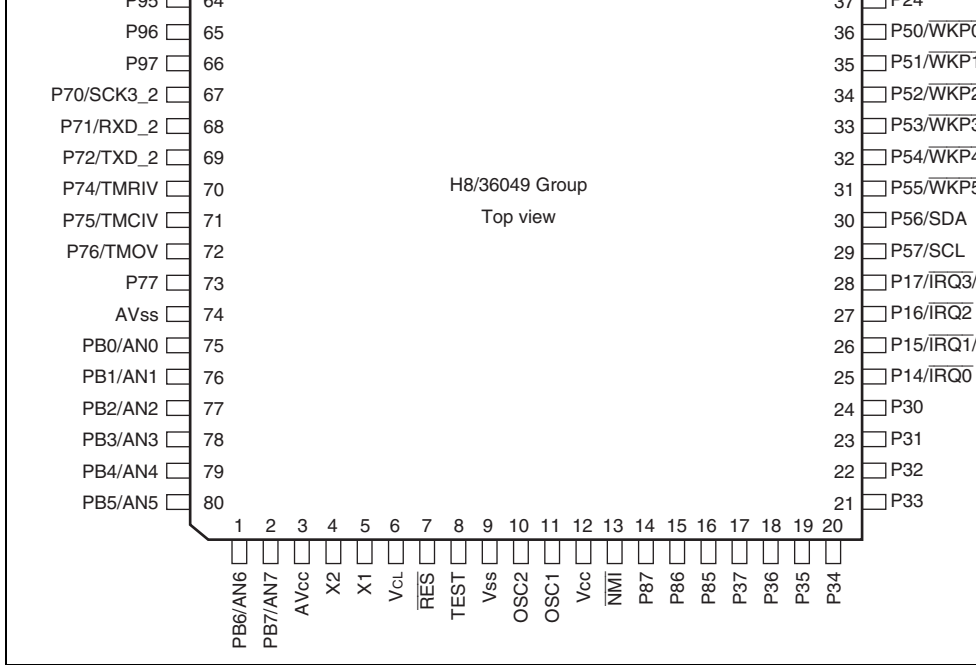


Figure 1.2 Pin Arrangements (FP-80A)

	AVcc	5	Input	Analog power supply pin for the A/D converter. When the A/D converter is not used, connect this pin to the system power supply.
	AVss	74	Input	Analog ground pin for the A/D converter. Connect this pin to the system power supply (0 V).
	VCL	6	Input	Internal step-down power supply pin. Connect a capacitor of around 0.1 μ F between this pin and the Vss pin for stabilization.
Clock pins	OSC1	11	Input	These pins connect with crystal or ceramic resonator for the system clock, or can be used to input an external clock. See section 5, Clock Pulse Generator, for a typical connection.
	OSC2	10	Output	
	X1	5	Input	
	X2	4	Output	
System control	$\overline{\text{RES}}$	7	Input	Reset pin. The pull-up resistor (typ. 15 k Ω) is incorporated. When driven low, the chip resets.
	TEST	8	Input	Test pin. Connect this pin to Vss.
External interrupt pins	$\overline{\text{NMI}}$	13	Input	Non-maskable interrupt request input pin. Be sure to pull-up by a pull-up resistor.
	$\overline{\text{IRQ0}}$ to $\overline{\text{IRQ3}}$	25 to 28	Input	External interrupt request input pins. Connect to the rising or falling edge.
	$\overline{\text{WKP0}}$ to $\overline{\text{WKP5}}$	36 to 31	Input	External interrupt request input pins. Connect to the rising or falling edge.
RTC	TMOW	56	Output	This is an output pin for divided clocks.
Timer B1	TMIB1	26	Input	External event input pin.

	FTIOC0	44	I/O	output pin Output compare output/input capture input synchronous output pin (at a reset or in complementary PWM mode)
	FTIOD0	45	I/O	Output compare output/input capture input output pin
	FTIOA1	46	I/O	Output compare output/input capture input output pin (at a reset or in complementary mode)
	FTIOB1 to FTIOD1	47 to 49	I/O	Output compare output/input capture input output pin
Timer W	FTCI	51	Input	External event input pin
	FTIOA to FTIOD	52 to 55	I/O	Output compare output/input capture input output pin
14-bit PWM	PWM	57	Output	14-bit PWM square wave output pin
I ² C bus interface 2 (IIC2)	SDA	30	I/O	IIC data I/O pin. Can directly drive a bus NMOS open-drain output. When using the external pull-up resistor is required.
	SCL	29	I/O	IIC clock I/O pin. Can directly drive a bus NMOS open-drain output. When using the external pull-up resistor is required.

A/D converter	AN7 to AN0	2, 1, 80 to 75	Input	Analog input pin
	ADTRG	31	Input	Conversion start trigger input pin
I/O ports	PB7 to PB0	2, 1, 80 to 75	Input	8-bit input port
	P17 to P14, P12 to P10	28 to 25, 58 to 56	I/O	7-bit I/O port
	P24 to P20	37 to 41	I/O	5-bit I/O port
	P37 to P30	17 to 24	I/O	8-bit I/O port
	P57 to P50	29 to 36	I/O	8-bit I/O port
	P67 to P60	49 to 42	I/O	8-bit I/O port
	P77 to P74, P72 to P70	73 to 70, 69 to 67	I/O	7-bit I/O port
	P87 to P80	14 to 16, 55 to 51	I/O	8-bit I/O port
	P97 to P90	66 to 59	I/O	8-bit I/O port

- General-register architecture
 - Sixteen 16-bit general registers also usable as sixteen 8-bit registers and eight 16-bit or eight 32-bit registers
- 62 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]
- 16-Mbyte address space
- High-speed operation
 - All frequently-used instructions execute in two to 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

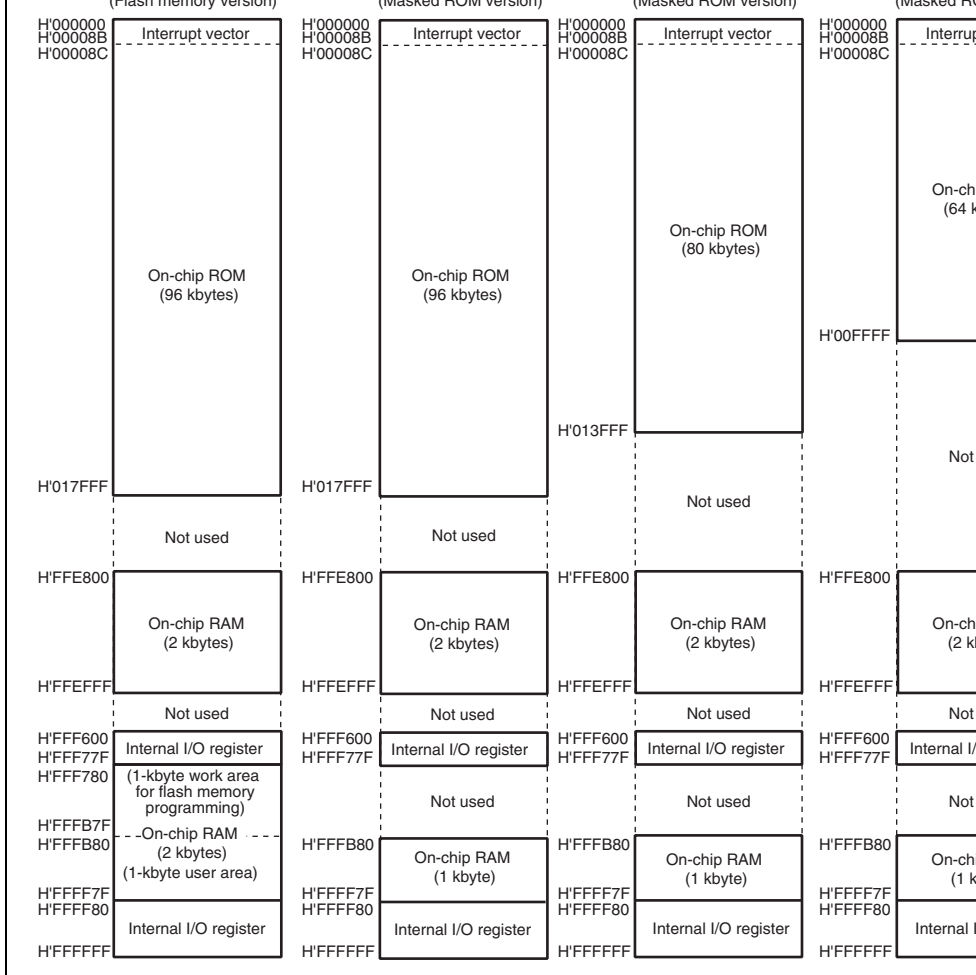
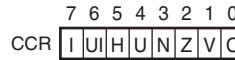
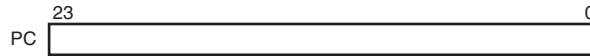


Figure 2.1 Memory Map

ER2	E2	R2H	R2L
ER3	E3	R3H	R3L
ER4	E4	R4H	R4L
ER5	E5	R5H	R5L
ER6	E6	R6H	R6L
ER7	E7	(SP) 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) and to R7L). These registers are functionally equivalent, providing a maximum of sixteen 8-bit registers.

The usage of each register can be selected independently.

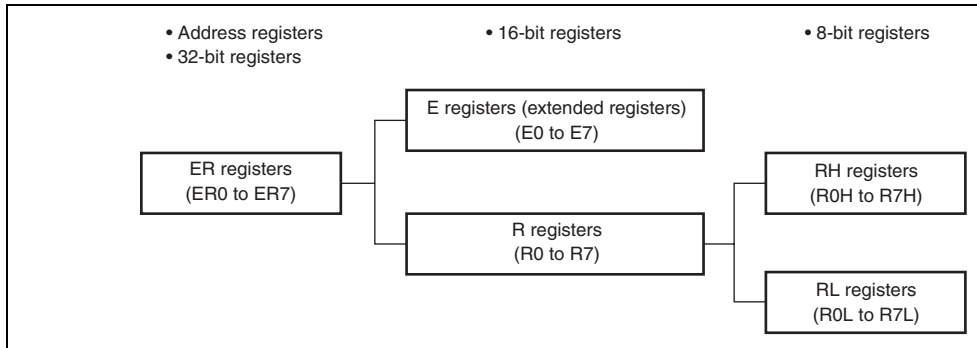


Figure 2.3 Usage of General Registers

General register ER7 has the function of stack pointer (SP) in addition to its general-register function, and is used implicitly in exception handling and subroutine calls. Figure 2.4 shows the relationship between the stack pointer and the stack area.

2.2.2 Program Counter (PC)

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

2.2.3 Condition-Code Register (CCR)

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

Some instructions leave flag bits unchanged. Operations can be performed on the CCR by LDC, STC, ANDC, ORC, and XORC instructions. The N, Z, V, and C flags are used as 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, or NEG.B instruction is executed, this flag is set to 1 if there is a carry or borrow at bit 3, and cleared to 0 otherwise. When the ADD.W, SUB.W, CMP.W, or NEG.W instruction is executed, the H flag is set to 1 if there is a carry or borrow at bit 11, and cleared to 0 otherwise. When the ADD.L, SUB.L, CMP.L, or NEG.L instruction is executed, the H flag is set to 1 if there is a carry or borrow at bit 27, and cleared to 0 otherwise.

4	U	Undefined	R/W	User Bit Can be written and read by software using the STC, ANDC, ORC, and XORC instructions.
3	N	Undefined	R/W	Negative Flag Stores the value of the most significant bit of data. Cleared to 0 if the sign bit is 0.
2	Z	Undefined	R/W	Zero Flag Set to 1 to indicate zero data, and cleared to 0 to indicate non-zero data.
1	V	Undefined	R/W	Overflow Flag Set to 1 when an arithmetic overflow occurs, and cleared to 0 at other times.
0	C	Undefined	R/W	Carry Flag Set to 1 when a carry occurs, and cleared to 0 otherwise. Used by: <ul style="list-style-type: none"> • Add instructions, to indicate a carry • Subtract instructions, to indicate a borrow • Shift and rotate instructions, to indicate a carry The carry flag is also used as a bit accumulator for bit manipulation instructions.

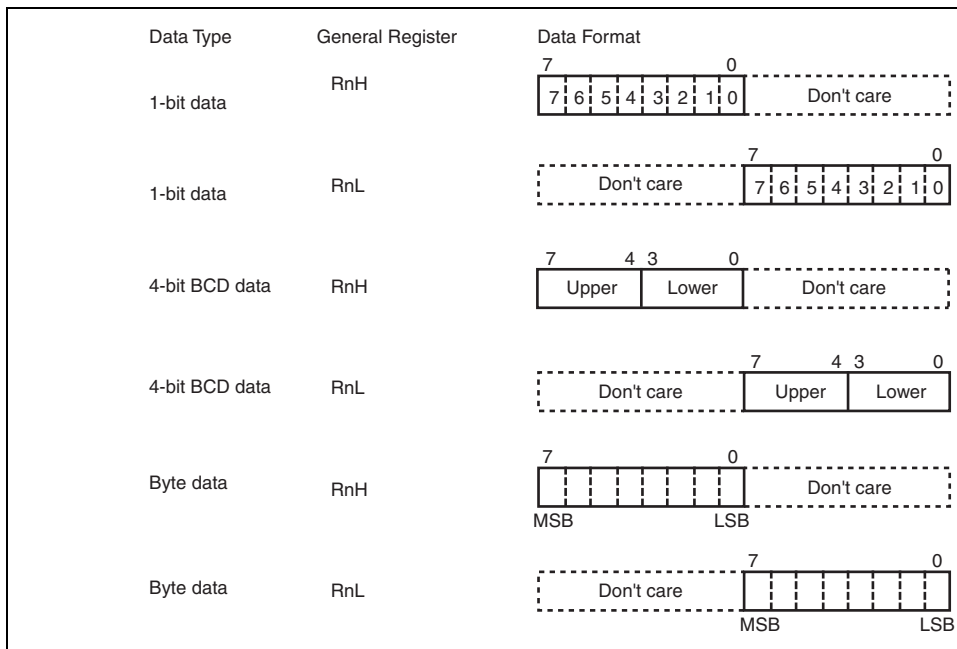


Figure 2.5 General Register Data Formats (1)

MSB

[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)

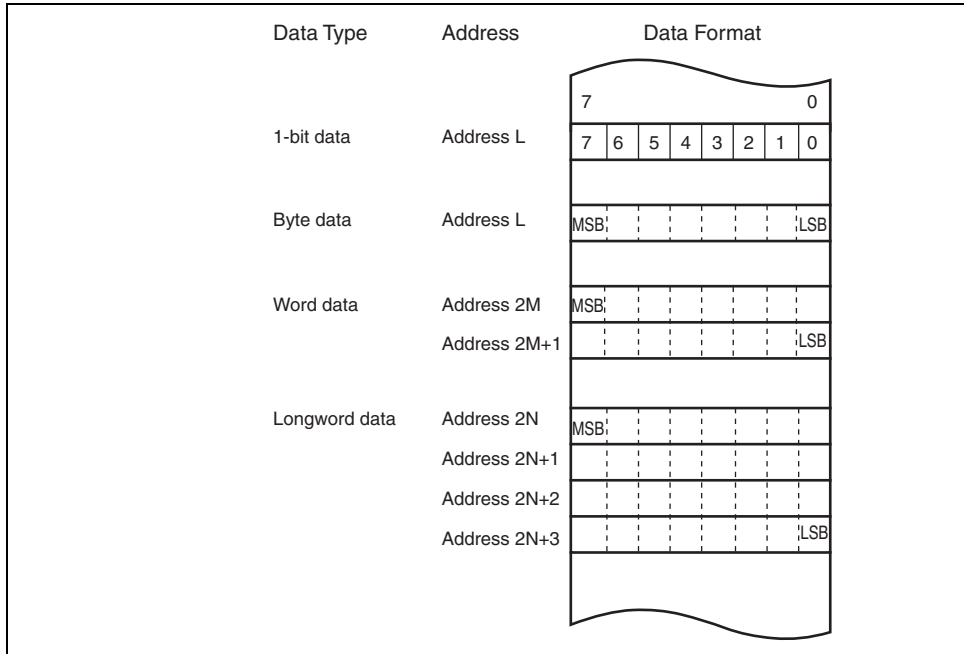


Figure 2.6 Memory Data Formats

Rd	General register (destination)
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)

MOVFP	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 MOV.W @SP+, Rn. POP.L ERn is identical to MOV.L @SP+, ERn.
PUSH	W/L	Rn → @-SP Pushes a general register onto the stack. PUSH.W Rn is identical to MOV.W Rn, @-SP. PUSH.L ERn is identical to MOV.L ERn, @-SP.

Note: * Refers to the operand size.

B: Byte

W: Word

L: Longword

DEC		Increments or decrements a general register by 1 or 2. (Byte operand can be incremented or decremented by 1 only.)
ADDS SUBS	L	$Rd \pm 1 \rightarrow Rd$, $Rd \pm 2 \rightarrow Rd$, $Rd \pm 4 \rightarrow Rd$ Adds or subtracts the value 1, 2, or 4 to or from data in a 32-bit register.
DAA DAS	B	Rd (decimal adjust) $\rightarrow Rd$ Decimal-adjusts an addition or subtraction result in a general register referring to the CCR to produce 4-bit BCD data.
MULXU	B/W	$Rd \times Rs \rightarrow Rd$ Performs unsigned multiplication on data in two general registers: 8 bits \times 8 bits \rightarrow 16 bits or 16 bits \times 16 bits \rightarrow 32 bits.
MULXS	B/W	$Rd \times Rs \rightarrow Rd$ Performs signed multiplication on data in two general registers: 8 bits \times 8 bits \rightarrow 16 bits or 16 bits \times 16 bits \rightarrow 32 bits.
DIVXU	B/W	$Rd \div Rs \rightarrow Rd$ Performs unsigned division on data in two general registers: either 8 bits \div 8 bits \rightarrow 8-bit quotient and 8-bit remainder or 32 bits \div 16 bits \rightarrow 16-bit quotient and 16-bit remainder.

Note: * Refers to the operand size.

B: Byte

W: Word

L: Longword

		Takes the two's complement (arithmetic complement) of data in general register.
EXTU	W/L	Rd (zero extension) → Rd Extends the lower 8 bits of a 16-bit register to word size, or the lower 8 bits of a 32-bit register to longword size, by padding with zeros on the left.
EXTS	W/L	Rd (sign extension) → Rd Extends the lower 8 bits of a 16-bit register to word size, or the lower 8 bits of a 32-bit register to longword size, by extending the sign.

Note: * Refers to the operand size.

B: Byte

W: Word

L: Longword

NOT	B/W/L	$\sim (Rd) \rightarrow (Rd)$ Takes the one's complement (logical 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) \rightarrow Rd Performs an arithmetic shift on general register contents.
SHLL SHLR	B/W/L	Rd (shift) \rightarrow Rd Performs a logical shift on general register contents.
ROTL ROTR	B/W/L	Rd (rotate) \rightarrow Rd Rotates general register contents.
ROTXL ROTXR	B/W/L	Rd (rotate) \rightarrow 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 or the carry flag. The bit number is specified by 3-bit immediate data or the lower three bits of a general register.
BTST	B	$\sim (\text{<bit-No.> of <EAd>}) \rightarrow Z$ Tests a specified bit in a general register or memory operand or the carry flag and sets the Z flag accordingly. The bit number is specified by 3-bit immediate data or the lower three bits of a general register.
BAND	B	$C \wedge (\text{<bit-No.> of <EAd>}) \rightarrow C$ ANDs the carry flag with a specified bit in a general register or memory operand and stores the result in the carry flag.
BIAND	B	$C \wedge \sim (\text{<bit-No.> of <EAd>}) \rightarrow C$ ANDs the carry flag with the inverse of a specified bit in a general register or memory operand and stores the result in the carry flag. The bit number is specified by 3-bit immediate data.
BOR	B	$C \vee (\text{<bit-No.> of <EAd>}) \rightarrow C$ ORs the carry flag with a specified bit in a general register or memory operand and stores the result in the carry flag.
BIOR	B	$C \vee \sim (\text{<bit-No.> of <EAd>}) \rightarrow C$ ORs the carry flag with the inverse of a specified bit in a general register or memory operand and stores the result in the carry flag. The bit number is specified by 3-bit immediate data.

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

Note: * 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	$N \oplus V = 0$
BLT	Less than	$N \oplus V = 1$
BGT	Greater than	$Z \vee (N \oplus V) = 0$
BLE	Less or equal	$Z \vee (N \oplus V) = 1$

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

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

access.

ANDC	B	$CCR \wedge \#IMM \rightarrow CCR$ Logically ANDs the CCR with immediate data.
ORC	B	$CCR \vee \#IMM \rightarrow CCR$ Logically ORs the CCR with immediate data.
XORC	B	$CCR \oplus \#IMM \rightarrow CCR$ Logically XORs the CCR with immediate data.
NOP	—	$PC + 2 \rightarrow PC$ Only increments the program counter.

Note: * Refers to the operand size.

B: Byte

W: Word

Table 2.9 Block Data Transfer Instructions

Instruction	Size	Function
EEPMOV.B	—	if R4L \neq 0 then Repeat @ER5+ \rightarrow @ER6+, R4L-1 \rightarrow R4L Until R4L = 0 else next;
EEPMOV.W	—	if R4 \neq 0 then Repeat @ER5+ \rightarrow @ER6+, R4-1 \rightarrow R4 Until R4 = 0 else next; Transfers a data block. Starting from the address set in ER5, transfer for the number of bytes set in R4L or R4 to the address location set Execution of the next instruction begins as soon as the transfer is completed.

Some instructions have two operation fields.

- Register Field

Specifies a general register. Address registers are specified by 3 bits, and data registers by 2 bits or 4 bits. Some instructions have two register fields. Some have no register field.

- Effective Address Extension

8, 16, or 32 bits specifying immediate data, an absolute address, or a displacement. An address or displacement is treated as a 32-bit data in which the upper 8 bits are 0 (H'00000000).

- Condition Field

Specifies the branching condition of Bcc instructions.

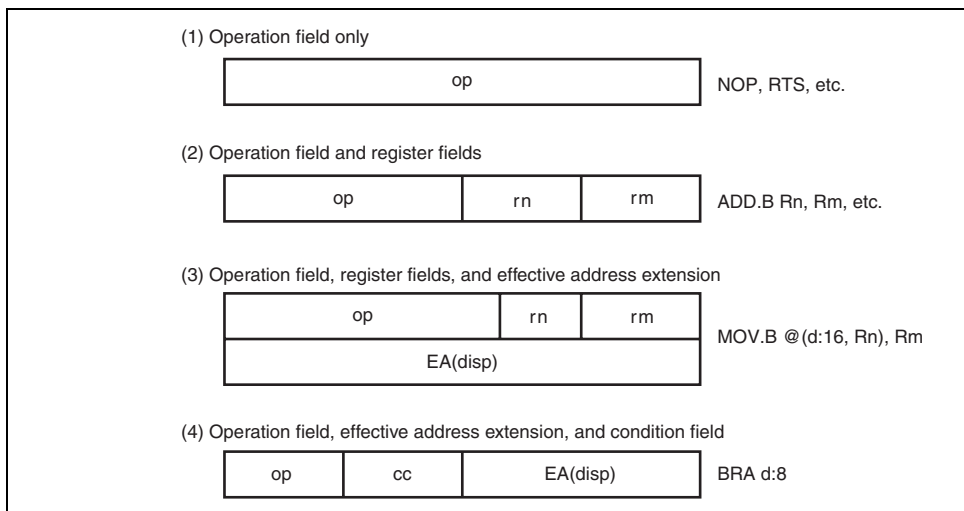


Figure 2.7 Instruction Formats

instructions can use an addressing modes except program-counter relative and memory indirect. Bit-manipulation instructions use register direct, register indirect, or the absolute addressing mode (@aa:8) to specify an operand, and register direct (BSET, BCLR, BNOT, and BTST instructions) use register direct or 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

Register Direct—Rn

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

Register Indirect—@ERn

The register field of the instruction code specifies an address register (ERn), the lower 24 bits of which contain the address of the operand on memory.

added to the address register contents (32 bits) and the sum is stored in the address register. For byte access, the value added is 1, for word access, 2, or for longword access, 4. For the word 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 register number in the instruction code, and the lower 24 bits of the result is the address of a memory operand. The result is also stored in the address register. The value subtracted is 1 for byte access, 2 for word access, or 4 for longword access. For the word or longword access, the register value should be even.

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

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

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

The access ranges of absolute addresses for the group of this LSI are those shown in table 2.11.

Table 2.11 Absolute Address Access Ranges

Absolute Address	Access Range
8 bits (@aa:8)	H'FFFF00 to H'FFFFFF
16 bits (@aa:16)	H'000000 to H'007FFF H'FF8000 to H'FFFFFF
24 bits (@aa:24)	H'000000 to H'FFFFFF

This mode is used in the BSR instruction. An 8-bit or 16-bit displacement contained in the instruction is sign-extended and added to the 24-bit PC contents to generate a branch address. The PC value to which the displacement is added is the address of the first byte of the next instruction, so the possible branching range is -126 to $+128$ bytes (-63 to $+64$ words) or -32766 to $+32768$ bytes (-16383 to $+16384$ words) from the branch instruction. The resulting value should be an even number.

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 address. The memory operand is accessed by longword access. The first byte of the memory operand is ignored, generating a 24-bit branch address. Figure 2.8 shows how to specify branch address in memory indirect mode.

The upper bits of the absolute address are all assumed to be 0, so the address range is 0 to 0xFFFF (H'0000 to H'00FF). Note that the first part of the address range is also the exception vector.

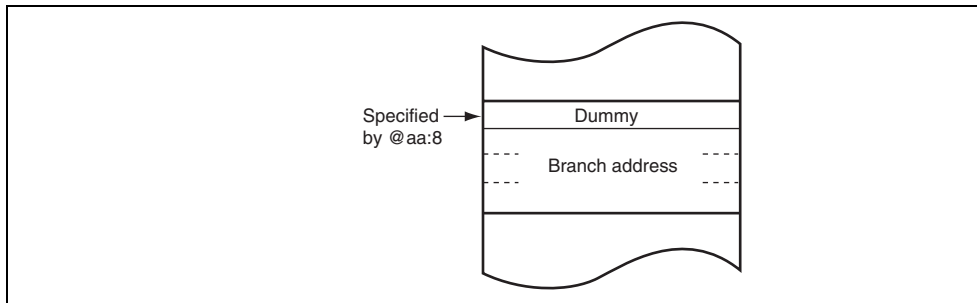
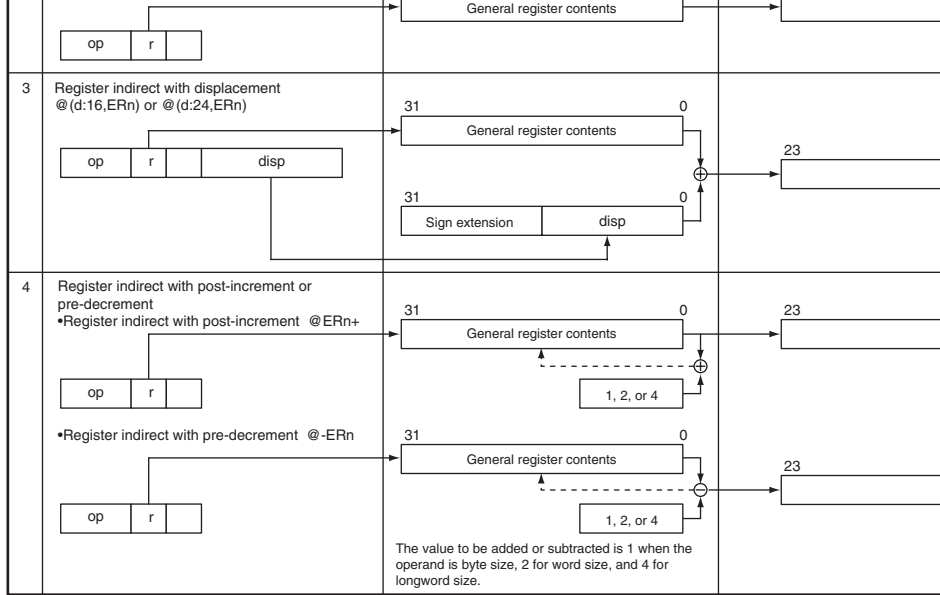
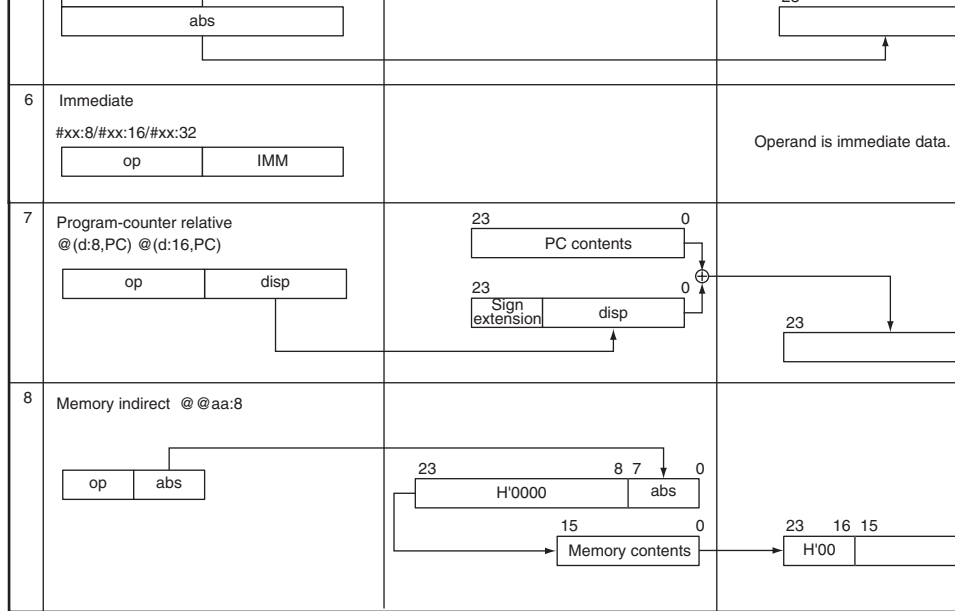


Figure 2.8 Branch Address Specification in Memory Indirect Mode





[Legend]

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

in byte or word size. Figure 2.9 shows the on-chip memory access cycle.

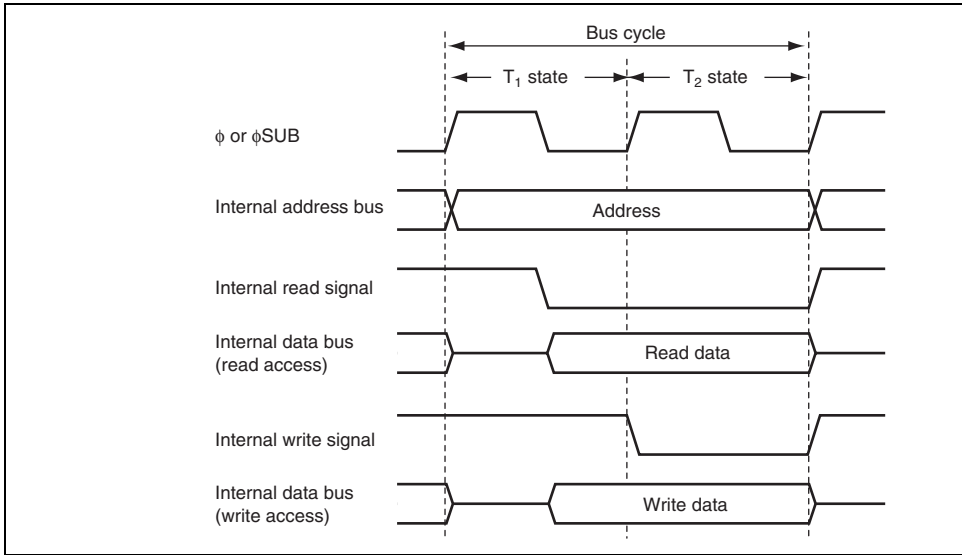


Figure 2.9 On-Chip Memory Access Cycle

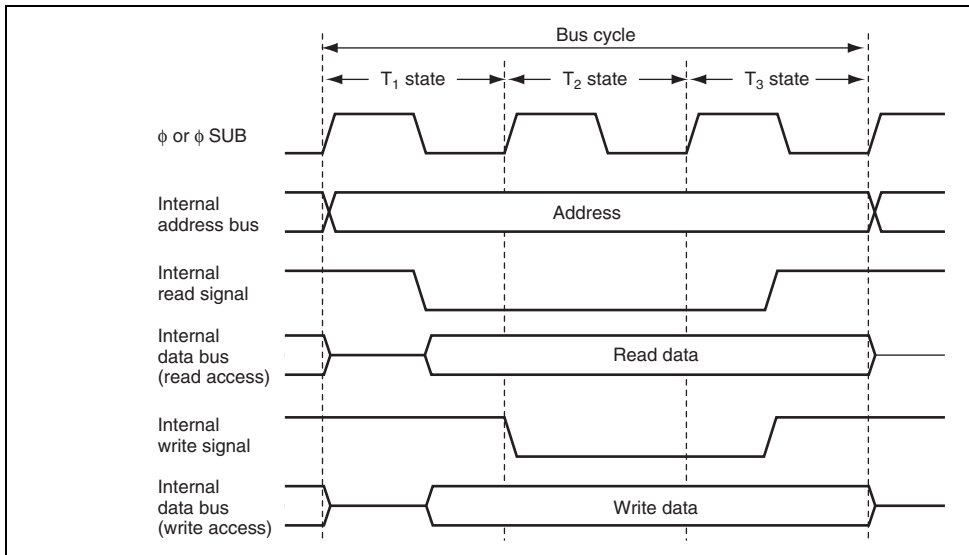


Figure 2.10 On-Chip Peripheral Module Access Cycle (3-State Access)

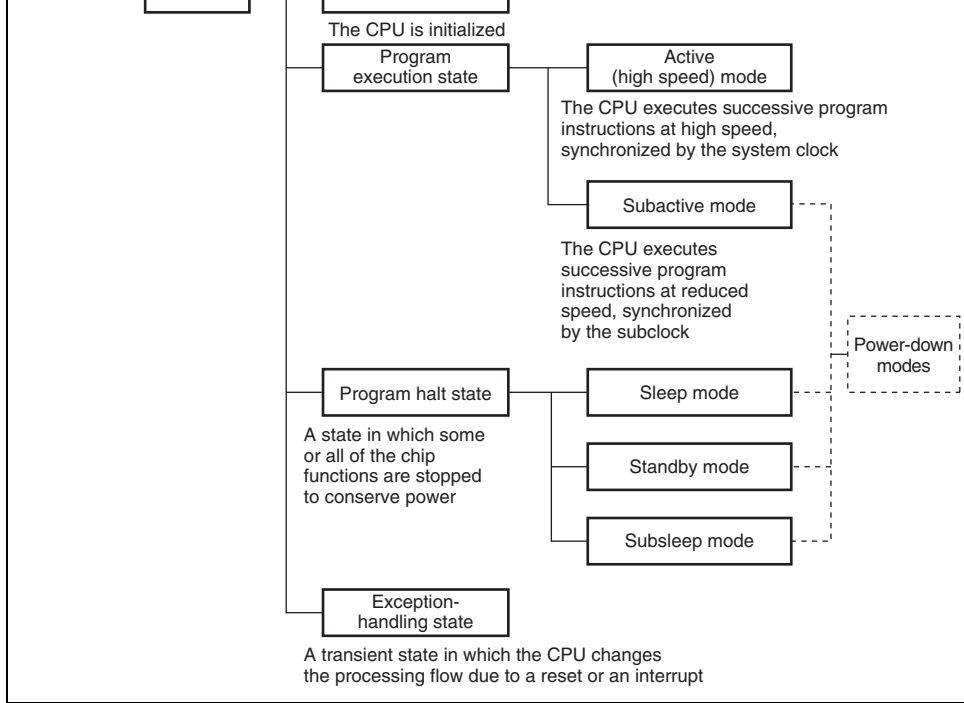


Figure 2.11 CPU Operation States

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-chip I/O registers areas available to the user. When data is transferred from CPU to empty areas, the transferred data will be lost. This action may also cause the CPU to malfunction. When data is 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 R4L, which starts from the address indicated by ER5, to the address indicated by ER6. Since R4L and ER6 so that the end address of the destination address (value of ER6 + R4 or ER6 + R4L) does not exceed H'FFFFFF (the value of ER6 must not change from H'FFFFFF to H'00000000 during execution).

(Applicable for timer B1 in the H8/36049 Group.)

Figure 2.13 shows an example of a timer in which two timer registers are assigned to the same address. When a bit-manipulation instruction accesses the timer load register and timer counter, a reloadable timer, since these two registers share the same address, the following operations take place.

1. Data is read in byte units.
2. The CPU sets or resets the bit to be manipulated with the bit-manipulation instruction.
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 timer counter register. As a result, bits other than the intended bit in the timer counter may be modified. The modified value may be written to the timer load register.

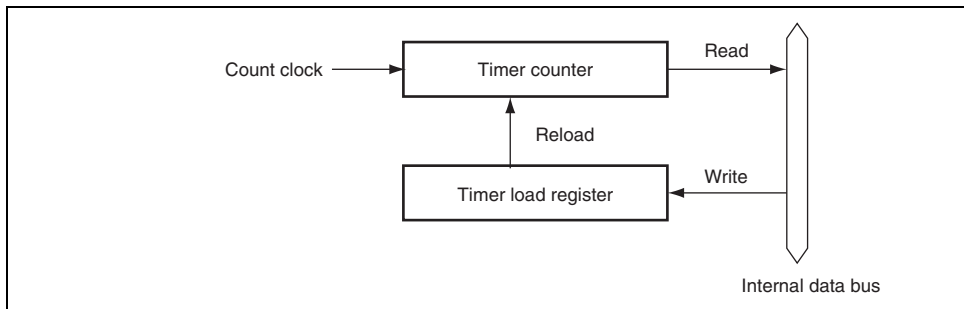


Figure 2.13 Example of Timer Configuration with Two Registers Allocated to the Same Address

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

- BSET instruction executed instruction

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

The BSET instruction is executed for port 5.

- After executing BSET instruction

	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

store a copy of the PDR5 data in a work area in memory. Perform the bit manipulation on the data in the work area, then write this data to PDR5.

- Prior to executing BSET instruction

```
MOV .B #80, R0L
MOV .B R0L, @RAM0
MOV .B R0L, @PDR5
```

The PDR5 value (H'80) is written to a work area in memory (RAM0) as well as 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

- BSET instruction executed

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

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

Bit Manipulation in a Register Containing a Write-Only Bit

Example 3: BCLR instruction executed designating port 5 control register PCR5

P57 and P56 are input pins, with a low-level signal input at P57 and a high-level signal input at P56. P55 to P50 are output pins that output low-level signals. An example of setting the P57 pin as an input pin by the BCLR instruction is shown below. It is assumed that a high-level signal is input to this input pin.

- Prior to executing BCLR instruction

	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

- BCLR instruction executed

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

The BCLR instruction is executed for PCR5.

1. When the BCLR instruction is executed, first the CPU reads PCR5. Since PCR5 is a write-only 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.

As a result of this operation, bit 0 in PCR5 becomes 0, making P50 an input port. However, bits 5 and 6 in PCR5 change to 1, so that P57 and P56 change from input pins to output pins. To solve this problem, store a copy of the PCR5 data in a work area in memory and manipulate the data. For example, clear bit 0 in the work area, then write this data to PCR5.

- Prior to executing BCLR instruction

```
MOV .B #3F, R0L
MOV .B R0L, @RAM0
MOV .B R0L, @PCR5
```

The PCR5 value (H'3F) is written to a work area in memory (RAM0) as well as 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

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, as specified in the instruction code. Exception handling can be executed at all times in the program execution, regardless of the setting of the I bit in CCR.

- Interrupts

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

—	Reserved for system use	1 to 6	H'000004 to H'00001B
External interrupt pin	NMI	7	H'00001C to H'00001F
CPU	Trap instruction #0	8	H'000020 to H'000023
	Trap instruction #1	9	H'000024 to H'000027
	Trap instruction #2	10	H'000028 to H'00002B
	Trap instruction #3	11	H'00002C to H'00002F
Address break	Break conditions satisfied	12	H'000030 to H'000033
CPU	Direct transition by executing the SLEEP instruction	13	H'000034 to H'000037
External interrupt pin	IRQ0 Low-voltage detection interrupt*	14	H'000038 to H'00003B
	IRQ1	15	H'00003C to H'00003F
	IRQ2	16	H'000040 to H'000043
	IRQ3	17	H'000044 to H'000047
	WKP	18	H'000048 to H'00004B

Timer V	Compare match A Compare match B Overflow	22	H'000058 to H'00005B
SCI3	Receive data full Transmit data empty Transmit end Receive error	23	H'00005C to H'00005F
IIC2	Transmit data empty Transmit end Receive data full Arbitration lost/overrun error NACK detection Stop conditions detected	24	H'000060 to H'000063
A/D converter	A/D conversion end	25	H'000064 to H'000067
Timer Z0	Compare match/input capture A0 to D0 Overflow	26	H'000068 to H'00006B
Timer Z1	Compare match/input capture A1 to D1 Overflow Underflow	27	H'00006C to H'00006F
Timer B1	Overflow	29	H'000074 to H'000077
—	Reserved for system use	30, 31	H'000078 to H'00007F
SCI3_2	Receive data full Transmit data empty Transmit end Receive error	32	H'000080 to H'000083

3.2 Register Descriptions

Interrupts are controlled by the following registers.

- Interrupt edge select register 1 (IEGR1)
- Interrupt edge select register 2 (IEGR2)
- Interrupt enable register 1 (IENR1)
- Interrupt enable register 2 (IENR2)
- Interrupt flag register 1 (IRR1)
- Interrupt flag register 2 (IRR2)
- Wakeup interrupt flag register (IWPR)

6	—	1	—	Reserved
5	—	1	—	These bits are always read as 1.
4	—	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	IEG2	0	R/W	IRQ2 Edge Select 0: Falling edge of $\overline{\text{IRQ2}}$ pin input is detected 1: Rising edge of $\overline{\text{IRQ2}}$ pin input is detected
1	IEG1	0	R/W	IRQ1 Edge Select 0: Falling edge of $\overline{\text{IRQ1}}$ pin input is detected 1: Rising edge of $\overline{\text{IRQ1}}$ pin input is detected
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 $\overline{WKP5}$ (ADTRG) pin input is detected 1: Rising edge of $\overline{WKP5}$ (ADTRG) pin input is detected
4	WPEG4	0	R/W	WKP4 Edge Select 0: Falling edge of $\overline{WKP4}$ pin input is detected 1: Rising edge of $\overline{WKP4}$ pin input is detected
3	WPEG3	0	R/W	WKP3 Edge Select 0: Falling edge of $\overline{WKP3}$ pin input is detected 1: Rising edge of $\overline{WKP3}$ pin input is detected
2	WPEG2	0	R/W	WKP2 Edge Select 0: Falling edge of $\overline{WKP2}$ pin input is detected 1: Rising edge of $\overline{WKP2}$ pin input is detected
1	WPEG1	0	R/W	WKP1 Edge Select 0: Falling edge of $\overline{WKP1}$ pin input is detected 1: Rising edge of $\overline{WKP1}$ pin input is detected
0	WPEG0	0	R/W	WKP0 Edge Select 0: Falling edge of $\overline{WKP0}$ pin input is detected 1: Rising edge of $\overline{WKP0}$ pin input is detected

				enabled.
5	IENWP	0	R/W	Wakeup Interrupt Enable This bit is an enable bit, which is common to WKIP5 to WKIP0. When the bit is 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 pin are enabled.
2	IEN2	0	R/W	IRQ2 Interrupt Enable When this bit is set to 1, interrupt requests of pin are enabled.
1	IEN1	0	R/W	IRQ1 Interrupt Enable When this bit is set to 1, interrupt requests of pin are enabled.
0	IEN0	0	R/W	IRQ0 Interrupt Enable When this bit is set to 1, interrupt requests of pin are enabled.

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

4	—	1	—	Reserved
3	—	1	—	These bits are always read as 1.
2	—	1	—	
1	—	1	—	
0	—	1	—	

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

3.2.5 Interrupt Flag Register 1 (IRR1)

IRR1 is a status flag register for direct transition interrupts, RTC interrupts, and $\overline{IRQ3}$ to $\overline{IRQ0}$ interrupt requests.

Bit	Bit Name	Initial Value	R/W	Description
7	IRRDT	0	R/W	Direct Transfer Interrupt Request Flag [Setting condition] When a direct transfer is made by executing a direct transfer instruction while DTON in SYSCR2 is set to 1. [Clearing condition] When IRRDT is cleared by writing 0

				<p>[Setting condition]</p> <p>When $\overline{\text{IRQ3}}$ pin is designated for interrupt input, designated signal edge is detected.</p> <p>[Clearing condition]</p> <p>When IRRI3 is cleared by writing 0</p>
2	IRRI2	0	R/W	<p>IRQ2 Interrupt Request Flag</p> <p>[Setting condition]</p> <p>When $\overline{\text{IRQ2}}$ pin is designated for interrupt input, designated signal edge is detected.</p> <p>[Clearing condition]</p> <p>When IRRI2 is cleared by writing 0</p>
1	IRRI1	0	R/W	<p>IRQ1 Interrupt Request Flag</p> <p>[Setting condition]</p> <p>When $\overline{\text{IRQ1}}$ pin is designated for interrupt input, designated signal edge is detected.</p> <p>[Clearing condition]</p> <p>When IRRI1 is cleared by writing 0</p>
0	IRRI0	0	R/W	<p>IRQ0 Interrupt Request Flag</p> <p>[Setting condition]</p> <p>When $\overline{\text{IRQ0}}$ pin is designated for interrupt input, designated signal edge is detected.</p> <p>[Clearing condition]</p> <p>When IRRI0 is cleared by writing 0</p>

When the timer B1 counter value overflows

[Clearing condition]

When IRRTB1 is cleared by writing 0

4	—	1	—	Reserved
3	—	1	—	These bits are always read as 1.
2	—	1	—	
1	—	1	—	
0	—	1	—	

				the designated signal edge is detected. [Clearing condition] When IWPF5 is cleared by writing 0.
4	IWPF4	0	R/W	WKP4 Interrupt Request Flag [Setting condition] When $\overline{WKP4}$ pin is designated for interrupt in the designated signal edge is detected. [Clearing condition] When IWPF4 is cleared by writing 0.
3	IWPF3	0	R/W	WKP3 Interrupt Request Flag [Setting condition] When $\overline{WKP3}$ pin is designated for interrupt in the designated signal edge is detected. [Clearing condition] When IWPF3 is cleared by writing 0.
2	IWPF2	0	R/W	WKP2 Interrupt Request Flag [Setting condition] When $\overline{WKP2}$ pin is designated for interrupt in the designated signal edge is detected. [Clearing condition] When IWPF2 is cleared by writing 0.
1	IWPF1	0	R/W	WKP1 Interrupt Request Flag [Setting condition] When $\overline{WKP1}$ pin is designated for interrupt in the designated signal edge is detected. [Clearing condition] When IWPF1 is cleared by writing 0.

\

When the $\overline{\text{RES}}$ pin goes low, all processing halts and this LSI enters the reset. The internal CPU and the registers of the on-chip peripheral modules are initialized by the reset. To reset this LSI at power-up, hold the $\overline{\text{RES}}$ pin low until the clock pulse generator output stabilizes. To reset the chip during operation, hold the $\overline{\text{RES}}$ pin low for at least 10 system clock cycles. When the $\overline{\text{RES}}$ pin goes high after being held low for the necessary time, this LSI starts reset exception handling. The reset exception handling sequence is shown in figure 3.1. For more details, refer to section 20, Power-On Reset and Low-Voltage Detection Circuits (Optional).

The reset exception handling sequence is as follows:

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). The data in that address is sent to the program counter (PC) as the start address, and program execution starts from that address.

NMI is the highest-priority interrupt, and can always be accepted without depending on the value in CCR.

IRQ3 to IRQ0 Interrupts:

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

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

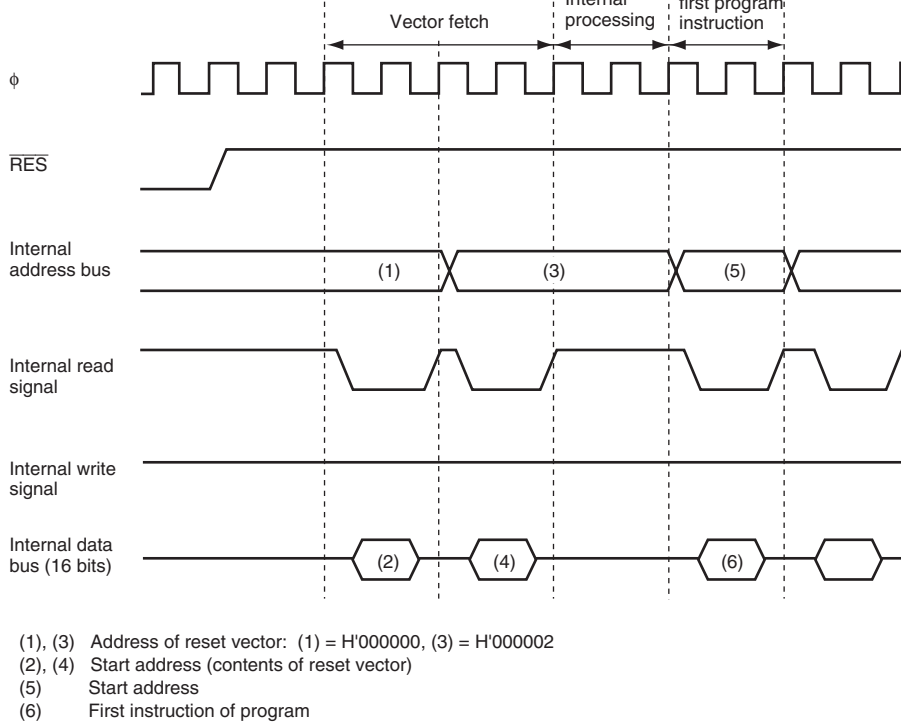


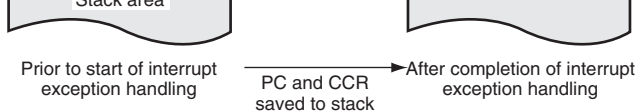
Figure 3.1 Reset Sequence

3.4.3 Interrupt Handling Sequence

Interrupts are controlled by an interrupt controller.

Interrupt operation is described as follows.

1. If an interrupt occurs while the NMI or interrupt enable bit is set to 1, an interrupt request signal is sent to the interrupt controller.
2. When multiple interrupt requests are generated, the interrupt controller requests to the CPU to handle the interrupt with the highest priority at that time according to table 3.1. Other interrupt requests are held pending.
3. The CPU accepts the NMI and address break without depending on the I bit value. Other interrupt requests are accepted, if the I bit is cleared to 0 in CCR; if the I bit is set to 1, the interrupt request is held pending.
4. If the CPU accepts the interrupt after processing of the current instruction is completed, interrupt exception handling will begin. First, both PC and CCR are pushed onto the stack. The stack status at this time is shown in figure 3.2. The PC value pushed onto the stack is the address of the first instruction to be executed upon return from interrupt handling.
5. Then, the I bit in CCR is set to 1, masking further interrupts excluding the NMI and address break. Upon return from interrupt handling, the values of I bit and other bits in CCR are 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, and transfers the address to PC as a start address of the interrupt handling-routine. Then the CPU starts executing from the address indicated in PC.



[Legend]

- PCE: Bits 23 to 16 of program counter (PC)
- PCH: Bits 15 to 8 of program counter (PC)
- PCL: Bits 7 to 0 of program counter (PC)
- CCR: Condition code register
- SP: Stack pointer

- Notes: 1. PC shows the address of the first instruction to be executed upon return from the interrupt handling routine.
2. Register contents must always be saved and restored by word length, starting from an even-numbered address.

Figure 3.2 Stack Status after Exception Handling

Saving PC and PC+1 to stack	1
Vector fetch	4
Instruction fetch	4
Internal processing	4

- Notes: 1. In case of internal interrupts, the number of states is 1.
2. Not including EEPMOV instruction.

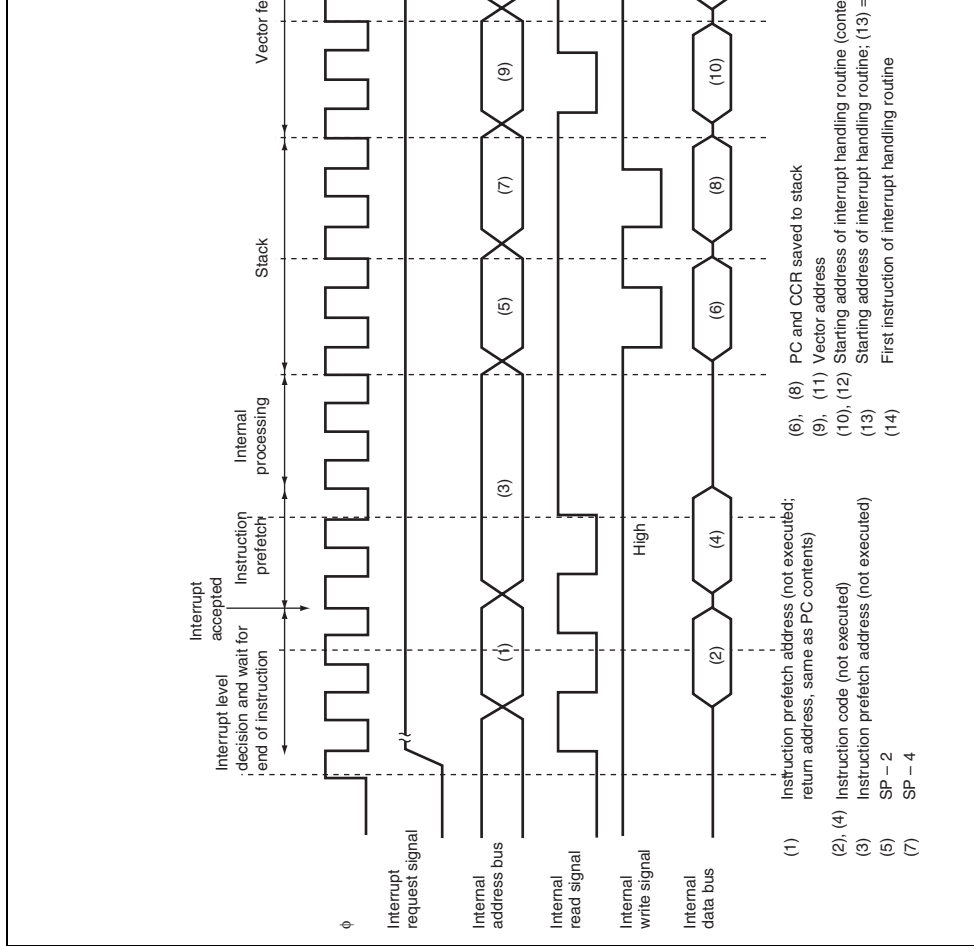


Figure 3.3 Interrupt Sequence

3.5.2 Notes on Stack Area Use

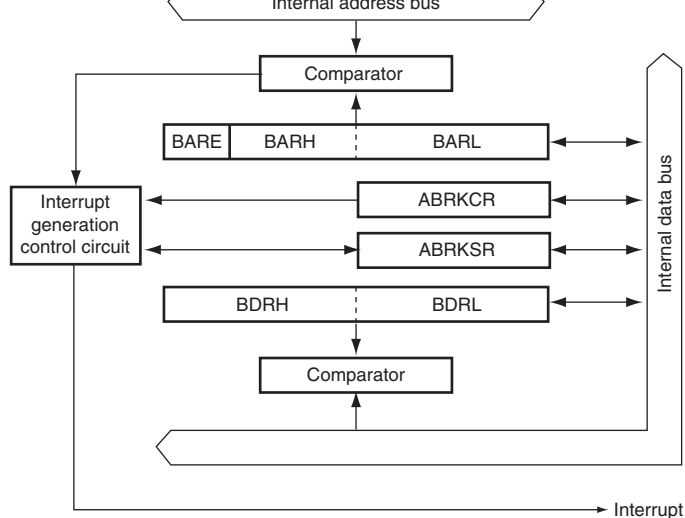
When word data is accessed, the least significant bit of the address is regarded as 0. Access to the stack always takes place in word size, so the stack pointer (SP: ER7) should never indicate an odd address. Use PUSH Rn (MOV.W Rn, @-SP) or POP Rn (MOV.W @SP+, Rn) to save and restore register values.

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 $\overline{\text{WKP5}}$ to $\overline{\text{WKP0}}$, 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 register. After accessing the port mode register, execute at least one instruction (e.g., NOP), then clear the interrupt request flag from 1 to 0.

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



[Legend]
 BARE, 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 instruction can be executed. When this bit is 1, the interrupt is not masked.
6	CSEL1	0	R/W	Condition Select 1 and 0
5	CSEL0	0	R/W	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
4	ACMP2	0	R/W	Address Compare 2 to 0
3	ACMP1	0	R/W	These bits set the comparison condition between the address set in BAR and the internal address bus.
2	ACMP0	0	R/W	000: Compares 24-bit addresses 001: Compares upper 20-bit addresses 010: Compares upper 16-bit addresses 011: Compares upper 12-bit addresses 1xx: Reserved

[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 access data bus used. When an I/O register space with an 8-bit data bus width is accessed in word access, byte access is generated twice. For details on data widths of each register, see section 22 Register Addresses (Address Order).

Table 4.1 Access and Data Bus Used

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

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

4.1.3 Break Address Registers E, H, L (BARE, BARH, BARL)

BAR (BARE, BARH, BARL) is a 24-bit readable/writable register that sets the address for generating an address break interrupt. The initial value of this register is H'FFFFFF. When the address break condition to the instruction execution cycle, set the first byte address of instruction.

4.1.4 Break Data Registers H, L (BDRH, BDRL)

BDR (BDRH, BDRL) is a 16-bit readable/writable register that sets the data for generating an address break interrupt. BDRH is compared with the upper 8-bit data bus. BDRL is compared with the lower 8-bit data bus. When memory or registers are accessed by byte, the upper 8-bit data bus is used for even and odd addresses in the data transmission. Therefore, comparison data must be set in BDRH for byte access. For word access, the data bus used depends on the address. Refer to section 4.1.1, Address Break Control Register (ABRKCR), for details. The initial value of this register is undefined.

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.

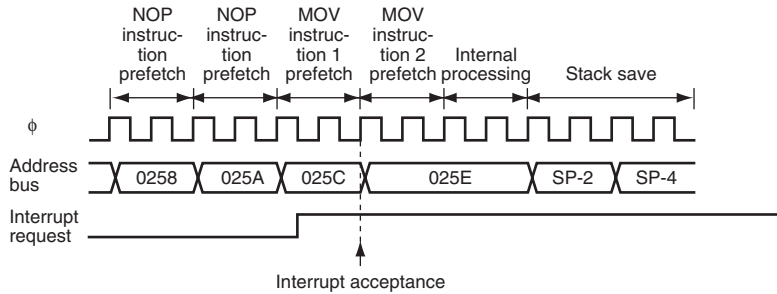


Figure 4.2 Address Break Interrupt Operation Example (1)

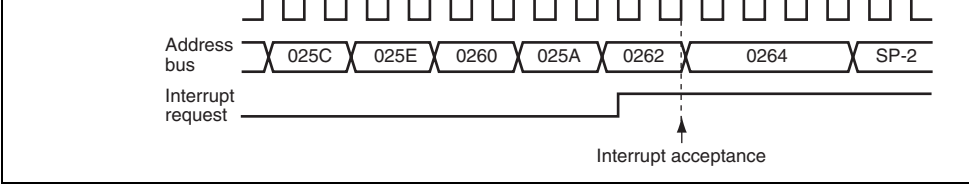


Figure 4.2 Address Break Interrupt Operation Example (2)

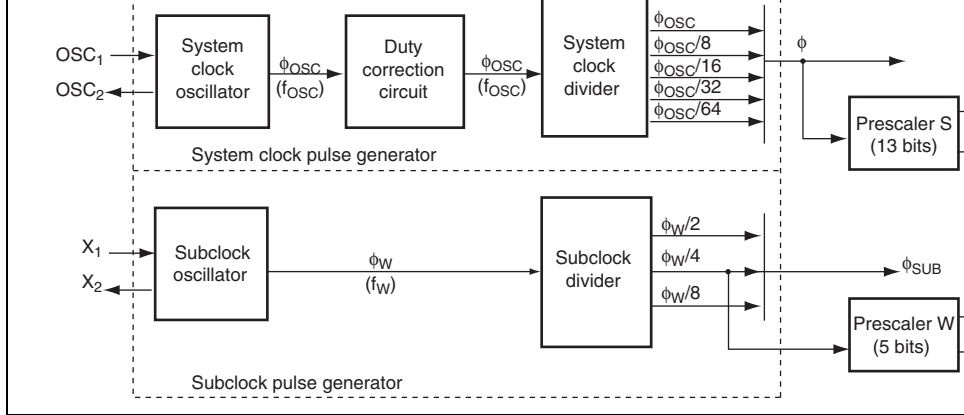


Figure 5.1 Block Diagram of Clock Pulse Generators

The basic clock signals that drive the CPU and on-chip peripheral modules are ϕ and ϕ_{SUB} . The system clock is divided by prescaler S to become a clock signal from $\phi/8192$ to $\phi/2$, and the subclock is divided by prescaler W to become a clock signal from $\phi_w/128$ to $\phi_w/8$. Both the system clock and subclock signals are provided to the on-chip peripheral modules.



LPM: Low-power mode (standby mode, subactive mode, subsleep mode)

Figure 5.2 Block Diagram of System Clock Generator

5.1.1 Connecting Crystal Resonator

Figure 5.3 shows a typical method of connecting a crystal resonator. An AT-cut parallel-resonant crystal resonator should be used. Figure 5.4 shows the equivalent circuit of a crystal resonator having the characteristics given in table 5.1 should be used.

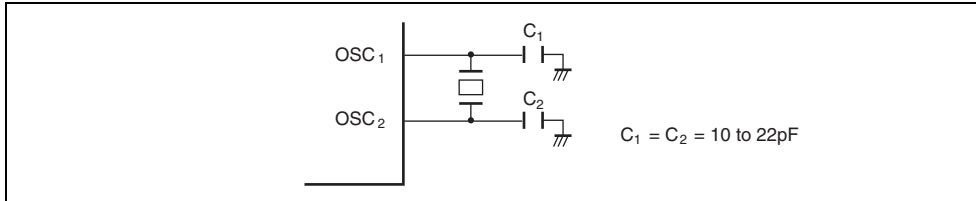


Figure 5.3 Typical Connection to Crystal Resonator

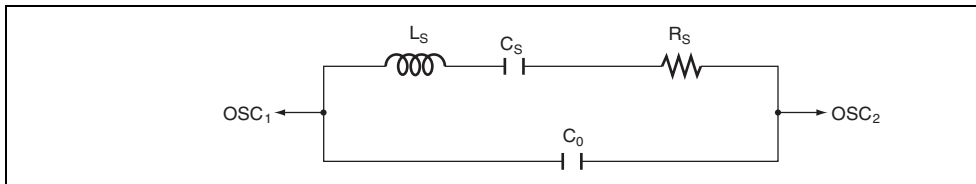


Figure 5.4 Equivalent Circuit of Crystal Resonator

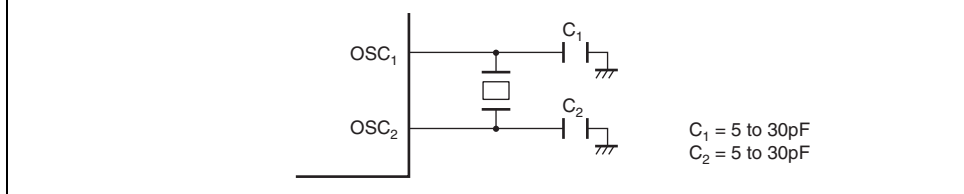


Figure 5.5 Typical Connection to Ceramic Resonator

5.1.3 External Clock Input Method

Connect an external clock signal to pin OSC₁, and leave pin OSC₂ open. Figure 5.6 shows the connection. The duty cycle of the external clock signal must be 45 to 55%.

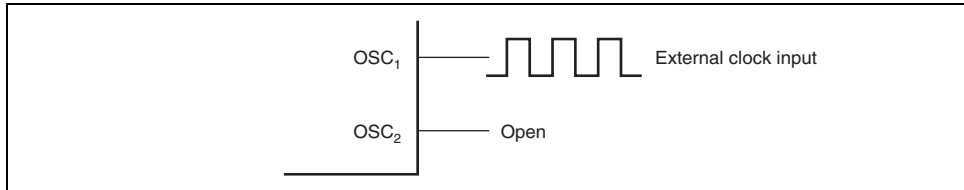


Figure 5.6 Example of External Clock Input

Figure 5.7 Block Diagram of Subclock Generator

5.2.1 Connecting 32.768-kHz Crystal Resonator

Clock pulses can be supplied to the subclock divider by connecting a 32.768-kHz crystal resonator, as shown in figure 5.8. Figure 5.9 shows the equivalent circuit of the 32.768-kHz crystal resonator.

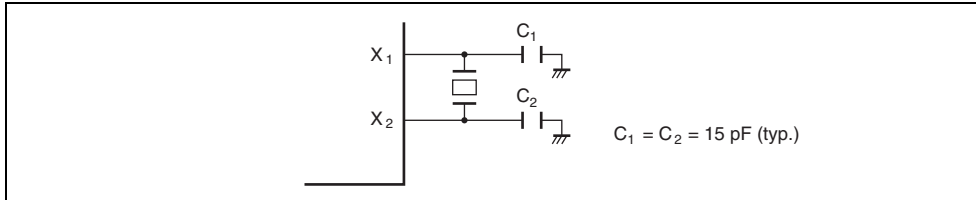


Figure 5.8 Typical Connection to 32.768-kHz Crystal Resonator

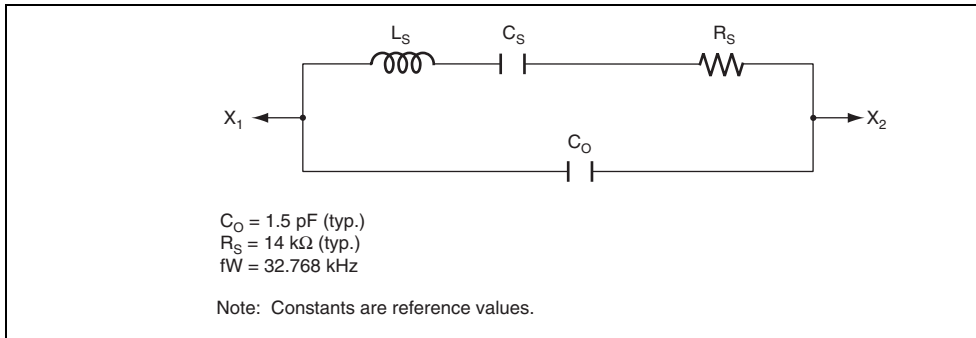


Figure 5.9 Equivalent Circuit of 32.768-kHz Crystal Resonator

5.3 Prescalers

5.3.1 Prescaler S

Prescaler S is a 13-bit counter using the system clock (ϕ) as its input clock. It is incremented by 1 per clock period. Prescaler S is initialized to H'0000 by a reset, and starts counting on exit from the reset state. In standby mode, subactive mode, and subsleep mode, the system clock from the system clock generator stops. Prescaler S also stops and is initialized to H'0000. The CPU cannot read the value of prescaler S. The output from prescaler S is shared by the on-chip peripheral modules. The division ratio can be set separately for each on-chip peripheral function. In active mode and sleep mode, the clock input to prescaler S is determined by the division factor designated by the MA[12:0] bits in SYSCR2.

5.3.2 Prescaler W

Prescaler W is a 5-bit counter using a 32.768 kHz signal divided by 4 ($\phi_w/4$) as its input clock. Its divided output is used for clock time base operation of timer A. Prescaler W is initialized to 0 by a reset, and starts counting on exit from the reset state. Even in standby mode, subactive mode, or subsleep mode, prescaler W continues functioning so long as clock signals are supplied to X₁ and X₂.

5.4.2 Notes on Board Design

When using a crystal resonator (ceramic resonator), place the resonator and its load capacitors close as possible to the OSC_1 and OSC_2 pins. Other signal lines should be routed away from resonator circuit to prevent induction from interfering with correct oscillation (see figure

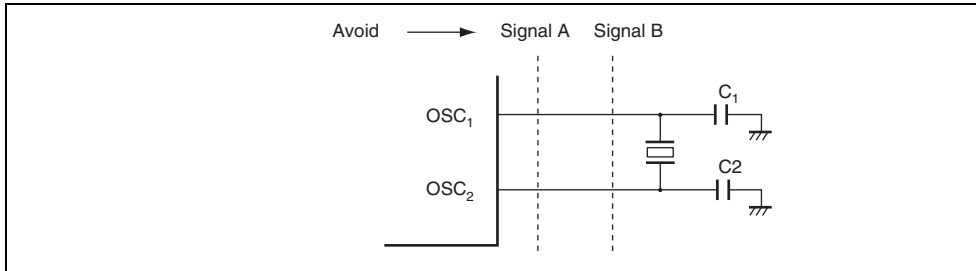


Figure 5.11 Example of Incorrect Board Design

The CPU and all on-chip peripheral modules are operable on the subclock. The subclock frequency can be selected from $\phi w/2$, $\phi w/4$, and $\phi w/8$.

- Sleep mode
The CPU halts. On-chip peripheral modules are operable on the system clock.
- Subsleep mode
The CPU halts. On-chip peripheral modules are operable on the subclock.
- Standby mode
The CPU and all on-chip peripheral modules halt. When the clock time-base function is selected, the RTC is operable.
- Module standby function
Independent of the above modes, power consumption can be reduced by halting on-chip peripheral modules that are not used in module units.

6.1 Register Descriptions

The registers related to power-down modes are listed below. For details on the serial mode control register (SCI3_3 module standby), see section 17, Serial Communication Interface 3 (SCI3).

- System control register 1 (SYSCR1)
- System control register 2 (SYSCR2)
- Module standby control register 1 (MSTCR1)
- Module standby control register 2 (MSTCR2)
- Serial Mode Control Register (SMCR)

1. Enters 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 designate the time the CPU and peripheral modules wait for stable clock operation after exiting from standby mode, subactive mode, or subsleep mode to active mode or sleep mode due to an interrupt. The designation should be made according to the clock frequency so that the waiting time is at least 6 clock cycles. The relationship between the specified value and the number of wait states is shown in table 6.1. When an external clock is to be used, the minimum value (wait = STS1 = STS0 = 1) is recommended.
4	STS0	0	R/W	
3	NESEL	0	R/W	
2	—	0	—	Reserved
1	—	0	—	These bits are always read as 0.
0	—	0	—	

1	0	128 states	0.00	0.00	0.01	0.02	0.03	0.06	0.13
	1	16 states	0.00	0.00	0.00	0.00	0.00	0.01	0.02

Note: Time unit is ms.

6.1.2 System Control Register 2 (SYSCR2)

SYSCR2 controls the power-down modes, as well as SYSCR1.

Bit	Bit Name	Initial Value	R/W	Description
7	SMSSEL	0	R/W	Sleep Mode Selection
6	LSOEN	0	R/W	Low Speed on Flag
5	DTON	0	R/W	Direct Transfer on Flag
<p>These bits select the mode to enter after the execution of a SLEEP instruction, as well as bit SSBY of SYSCR1.</p> <p>For details, see table 6.2.</p>				
4	MA2	0	R/W	Active Mode Clock Select 2 to 0
3	MA1	0	R/W	<p>These bits select the operating clock frequency in active and sleep modes. The operating clock frequency changes to the set frequency after a SLEEP instruction is executed.</p> <p>0xx: ϕ_{osc} 100: $\phi_{osc}/8$ 101: $\phi_{osc}/16$ 110: $\phi_{osc}/32$ 111: $\phi_{osc}/64$</p>
2	MA0	0	R/W	

5	MSTS3	0	R/W	SCI3 Module Standby SCI3 enters standby mode when this bit is set to 1
4	MSTAD	0	R/W	A/D Converter Module Standby A/D converter enters standby mode when this bit is set to 1
3	MSTWD	0	R/W	Watchdog Timer Module Standby Watchdog timer enters standby mode when this bit is set to 1. When the internal oscillator is selected as the watchdog timer clock, the watchdog timer operates 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 set to 1
1	MSTTV	0	R/W	Timer V Module Standby Timer V enters standby mode when this bit is set to 1
0	MSTTA	0	R/W	RTC Module Standby RTC enters standby mode when this bit is set to 1

4	MSTTB1	0	R/W	Timer B1 Module Standby Timer B1 enters standby mode when this bit is
3	—	0	—	Reserved
2	—	0	—	These bits are always read as 0.
1	MSTTZ	0	R/W	Timer Z Module Standby Timer Z enters standby mode when this bit is
0	MSTPWM	0	R/W	PWM Module Standby PWM enters standby mode when this bit is set

by an interrupt. Table 6.2 shows the internal states of the LSI in each mode.

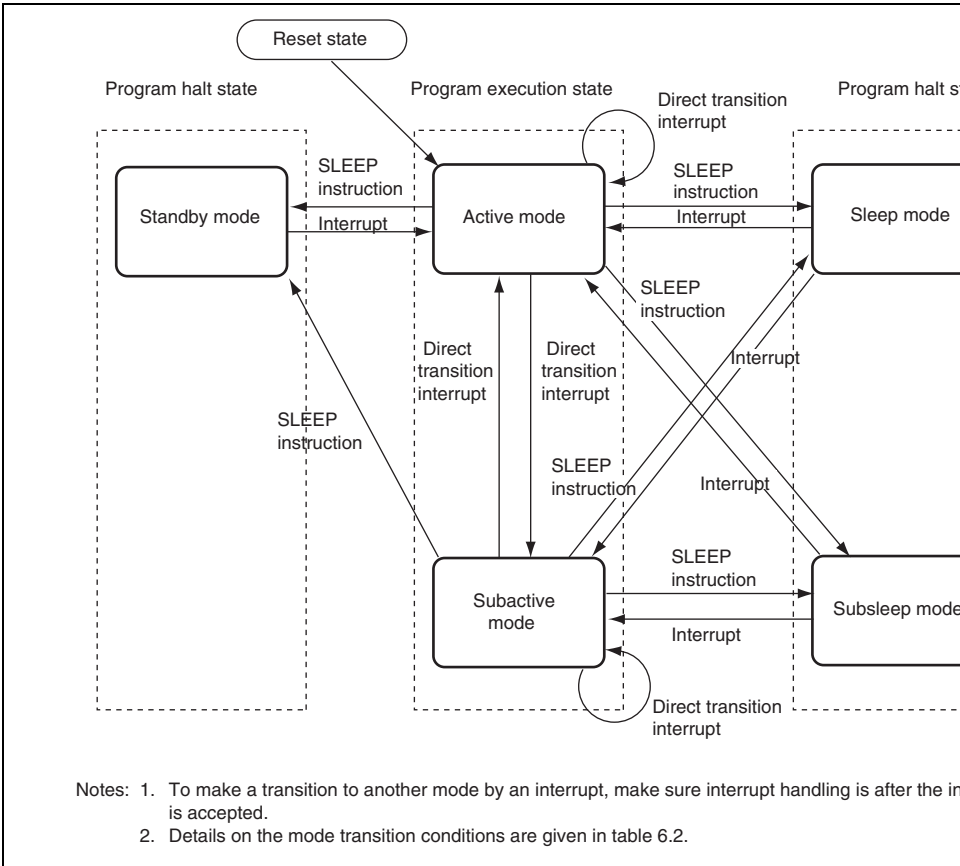


Figure 6.1 Mode Transition Diagram

1	X	0*	0	Active mode (direct transition)	—
	X	X	1	Subactive mode (direct transition)	—

[Legend] X: Don't care.

* When a state transition is performed while SMSEL is 1, timer V, SCI3, SCI3_2 and the A/D converter are reset, and all registers are set to their initial values. these functions after entering active mode, reset the registers.

External interrupts	IRQ3 to IRQ0	Functioning	Functioning	Functioning	Functioning	Functioning
	WKP5 to WKP0	Functioning	Functioning	Functioning	Functioning	Functioning
Peripheral functions	RTC	Functioning	Functioning	Functioning if the timekeeping time-base function is selected, and retained if n		
	Timer V	Functioning	Functioning	Reset	Reset	Res
	Watchdog timer	Functioning	Functioning	Retained (functioning if the internal clock selected as a count clock*)		
	SCI3, SCI3_2, SCI3_3	Functioning	Functioning	Reset	Reset	Res
	IIC2	Functioning	Functioning	Retained*	Retained	Ret
	Timer B1	Functioning	Functioning	Retained*	Retained	Ret
	Timer Z	Functioning	Functioning	Retained*	Retained	Ret
	Timer W	Functioning	Functioning	Retained (the counter is incremented by a subclock if the internal clock ϕ is selected as a count clock*)		
A/D converter	Functioning	Functioning	Reset	Reset	Res	

Note: * Registers can be read or written in subactive mode.

6.2.2 Standby Mode

In standby mode, the clock pulse generator stops, so the CPU and on-chip peripheral modules are not functioning. However, as long as the rated voltage is supplied, the contents of CPU registers, on-chip RAM, and some on-chip peripheral module registers are retained. On-chip RAM contents will be retained as long as the voltage set by the RAM data retention voltage is provided. I/O ports go to the high-impedance state.

Standby mode is cleared by an interrupt. When an interrupt is requested, the system clock pulse generator starts. After the time set in bits STS2 to STS0 in SYSCR1 has elapsed, and interrupt exception handling starts. Standby mode is not cleared if the I bit of CCR is set to 1 or the requested interrupt is disabled in the interrupt enable register.

When the $\overline{\text{RES}}$ pin goes low, the system clock pulse generator starts. Since system clock pulses are supplied to the entire chip as soon as the system clock pulse generator starts functioning, the $\overline{\text{RES}}$ pin must be kept low until the pulse generator output stabilizes. After the pulse generator output has stabilized, the CPU starts reset exception handling if the $\overline{\text{RES}}$ pin is driven high.

made to subactive mode when the bit is 1. After the time set in bits STS2 to STS0 in SYSCR1 has elapsed, a transition is made to active mode.

When the $\overline{\text{RES}}$ pin goes low, the system clock pulse generator starts. Since system clock pulses are supplied to the entire chip as soon as the system clock pulse generator starts functioning, the $\overline{\text{RES}}$ pin must be kept low until the pulse generator output stabilizes. After the pulse generator output has stabilized, the CPU starts reset exception handling if the $\overline{\text{RES}}$ pin is driven high.

6.2.4 Subactive Mode

The operating frequency of subactive mode is selected from $\phi_w/2$, $\phi_w/4$, and $\phi_w/8$ by the SA0 bits in SYSCR2. After the SLEEP instruction is executed, the operating frequency is the frequency which is set before the execution.

When the SLEEP instruction is executed in subactive mode, a transition to sleep mode, standby mode, active mode, or subactive mode is made, depending on the combination of bits in SYSCR1 and SYSCR2.

When the $\overline{\text{RES}}$ pin goes low, the system clock pulse generator starts. Since system clock pulses are supplied to the entire chip as soon as the system clock pulse generator starts functioning, the $\overline{\text{RES}}$ pin must be kept low until the pulse generator output stabilizes. After the pulse generator output has stabilized, the CPU starts reset exception handling if the $\overline{\text{RES}}$ pin is driven high.

be made by executing a SLEEP instruction while the DTION bit in SYSCR2 is set to 1. After mode transition, direct transition interrupt exception handling starts.

If the direct transition interrupt is disabled in interrupt enable register 1, a transition is made instead to sleep or subsleep mode. Note that if a direct transition is attempted while the ICCR is set to 1, sleep or subsleep mode will be entered, and the resulting mode cannot be changed by means of an interrupt.

6.4.1 Direct Transition from Active Mode to Subactive Mode

The time from the start of SLEEP instruction execution to the end of interrupt exception handling (the direct transition time) is calculated by equation (1).

Direct transition time = {(number of SLEEP instruction execution states) + (number of interrupt processing states)} × (tcyc before transition) + (number of interrupt exception handling states) × (tsubcyc after transition) (1)

Example:

$$\text{Direct transition time} = (2 + 1) \times \text{tosc} + 16 \times 8 \text{ tw} = 3 \text{ tosc} + 128 \text{ tw}$$

(when the CPU operating clock of $\phi_{\text{osc}} \rightarrow \phi_{\text{w}}/8$ is selected)

[Legend]

tosc: OSC clock cycle time
tw: Watch clock cycle time
tcyc: System clock (ϕ) cycle time
tsubcyc: Subclock (ϕ_{SUB}) cycle time

Direct transition time = $(2 + 1) \times 8 \text{ tw} + (8192 + 16) \times \text{tosc} = 24 \text{ tw} + 8208 \text{ tosc}$
(when the CPU operating clock of $\phi_w/8 \rightarrow \phi_{\text{osc}}$ and a waiting time of 8192 states are selected)

[Legend]

tosc: OSC clock cycle time

tw: Watch clock cycle time

tcyc: System clock (ϕ) cycle time

tsubcyc: Subclock (ϕ_{SUB}) cycle time

6.5 Module Standby Function

The module-standby function can be set to any peripheral module. In the module standby mode, the clock supply to modules stops to enter the power-down mode. Setting a bit in MSTCR1, MSTCR2, or SMCR that corresponds to each module to 1 enables each on-chip peripheral to enter the module standby state and the module standby state is canceled by clearing the

- Reprogramming capability
 - 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 program 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.
- Programmer mode
 - Flash memory can be programmed/erased in programmer mode using a PROM programmer, as well as in on-board programming mode.
- 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.
- Power-down mode
 - Operation of the power supply circuit can be partly halted in subactive mode. As a result, flash memory can be read with low power consumption.

Erase unit: 1 kbyte	~	H'000000	H'000001	H'000002	← Programming unit: 128 bytes →	H'00007F
		H'000380	H'000381	H'000382		H'0003FF
Erase unit: 1 kbyte	~	H'000400	H'000401	H'000402	← Programming unit: 128 bytes →	H'00047F
		H'000780	H'000781	H'000782		H'0007FF
Erase unit: 1 kbyte	~	H'000800	H'000801	H'000802	← Programming unit: 128 bytes →	H'00087F
		H'000B80	H'000B81	H'000B82		H'000BFF
Erase unit: 1 kbyte	~	H'000C00	H'000C01	H'000C02	← Programming unit: 128 bytes →	H'000C7F
		H'000F80	H'000F81	H'000F82		H'000FFF
Erase unit: 28 kbytes	~	H'001000	H'001001	H'001002	← Programming unit: 128 bytes →	H'00107F
		H'007F80	H'007F81	H'007F82		H'007FFF
Erase unit: 16 kbytes	~	H'008000	H'008001	H'008002	← Programming unit: 128 bytes →	H'00807F
		H'00BF80	H'00BF81	H'00BF82		H'00BFFF
Erase unit: 16 kbytes	~	H'00C000	H'00C001	H'00C002	← Programming unit: 128 bytes →	H'00C07F
		H'00FF80	H'00FF81	H'00FF82		H'00FFFF
Erase unit: 32 kbytes	~	H'010000	H'010001	H'010002	← Programming unit: 128 bytes →	H'01007F
		H'017F80	H'017F81	H'017F82		H'017FFF

Figure 7.1 Block Configuration of Flash Memory

7.2.1 Flash Memory Control Register 1 (FLMCR1)

FLMCR1 is a register that makes the flash memory change to program mode, program-verify 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 is cleared to 0, other FLMCR1 register bits and bits cannot be set.
5	ESU	0	R/W	Erase Setup When this bit is set to 1, the flash memory changes to the erase setup state. When it is cleared to 0, the erase setup state is cancelled. Set this bit to 1 before setting the E bit to 1 in FLMCR1.
4	PSU	0	R/W	Program Setup When this bit is set to 1, the flash memory changes to the program setup state. When it is cleared to 0, the program setup state is cancelled. Set this bit to 1 before setting the P bit in FLMCR1.
3	EV	0	R/W	Erase-Verify When this bit is set to 1, the flash memory changes to the erase-verify mode. When it is cleared to 0, the erase-verify 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. FLMCR2 is a 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 operation on flash memory (programming or erasing). When FLER is set to 1, flash memory goes to the protection state. See section 7.5.3, Error Protection, for details.
6 to 0	—	All 0	—	Reserved These bits are always read as 0.

				H'00FFFF will be erased.
5	EB5	0	R/W	When this bit is set to 1, 16 kbytes of H'008000 H'00BFFF will be erased.
4	EB4	0	R/W	When this bit is set to 1, 28 kbytes of H'001000 H'007FFF will be erased.
3	EB3	0	R/W	When this bit is set to 1, 1 kbyte of H'000C00 H'000FFF will be erased.
2	EB2	0	R/W	When this bit is set to 1, 1 kbyte of H'000800 H'000BFF will be erased.
1	EB1	0	R/W	When this bit is set to 1, 1 kbyte of H'000400 H'0007FF will be erased.
0	EB0	0	R/W	When this bit is set to 1, 1 kbyte of H'000000 H'0003FF will be erased.

When this bit is 0 and a transition is made to subactive mode, the flash memory enters the power-down mode. When this bit is 1, the flash memory remains in normal mode even after a transition is made to subactive mode.

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

7.2.5 Flash Memory Enable Register (FENR)

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

Bit	Bit Name	Initial Value	R/W	Description
7	FLSHE	0	R/W	Flash Memory Control Register Enable Flash memory control registers can be accessed when this bit is set to 1. Flash memory control registers cannot be accessed when this bit is set to 0.
6	—	0	R/W	Reserved This bit can be read from or written to, but should be set to 1.
5 to 0	—	All 0	—	Reserved These bits are always read as 0.

via SCI3. After erasing the entire flash memory, the programming control program is executed. This can be used for programming initial values in the on-board state or for a forcible reprogramming/erasing can no longer be done in user program mode. In user program mode, individual blocks can be erased and programmed by branching to the user program/erasing program prepared by the user.

Table 7.1 Setting Programming Modes

TEST	$\overline{\text{NMI}}$	P85	PB0	PB1	PB2	LSI State after Reset End
0	1	X	X	X	X	User Mode
0	0	1	X	X	X	Boot Mode
1	X	X	0	0	0	Programmer Mode

[Legend] X : Don't care.

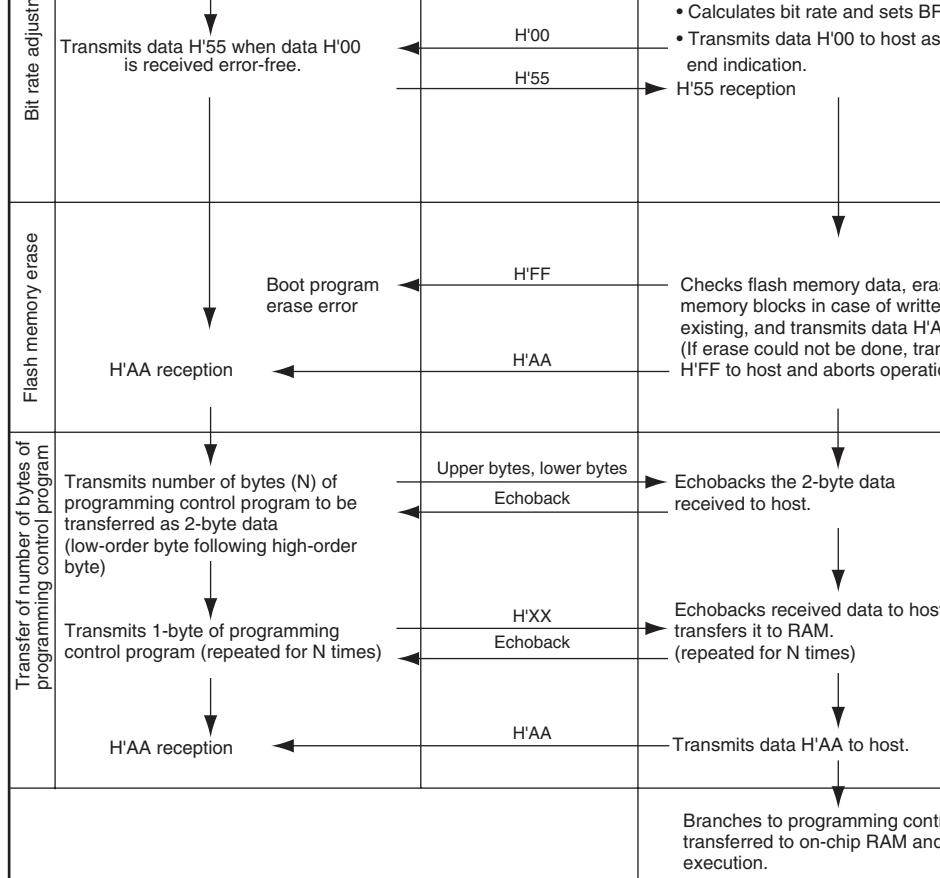
7.3.1 Boot Mode

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

1. When boot mode is used, the flash memory programming control program must be prepared on 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 data, 1 stop bit, and no parity.

the bit rates of the host and the chip. To operate the SCI properly, set the host's transfer 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 from 0x00000000 to 0xFFFFEEF is the area to which the programming control program is transferred from the host. The boot program area cannot be used until the execution state in boot mode switches to the programming control program.
6. Before branching to the programming control program, the chip terminates transfer of data by SCI3 (by clearing the RE and TE bits in SCR to 0), however the adjusted bit rate value remains set in BRR. Therefore, the programming control program can still use it for transferring program data or verify data with the host. The TxD pin is high (PCR22 = 1, P22 = 1). The contents of the CPU general registers are undefined immediately after branching to the programming control program. These registers must be initialized at the beginning of the programming control program, as the stack pointer (SP), in particular, is used implicitly in subroutine calls, etc.
7. Boot mode can be cleared by a reset. End the reset after driving the reset pin low, wait for at least 20 states, and then setting the TEST pin and $\overline{\text{NMI}}$ pin. Boot mode is also cleared if a WDT overflow occurs.
8. Do not change the TEST pin and $\overline{\text{NMI}}$ pin input levels in boot mode.



On-board programming/erasing of an individual flash memory block can also be performed in user program mode by branching to a user program/erase control program. The user must set the appropriate conditions and provide on-board means of supplying programming data. The flash memory must contain the user program/erase control program or a program that provides the user program/erase control program from external memory. As the flash memory itself cannot be read during programming/erasing, transfer the user program/erase control program to on-chip RAM, and execute it in user program mode. Figure 7.2 shows a sample procedure for programming/erasing in user program mode. Prepare a user program/erase control program in accordance with the description in section 7.5.2.2.2. Flash Memory Programming/Erasing.

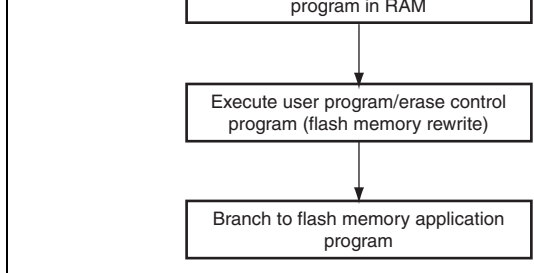
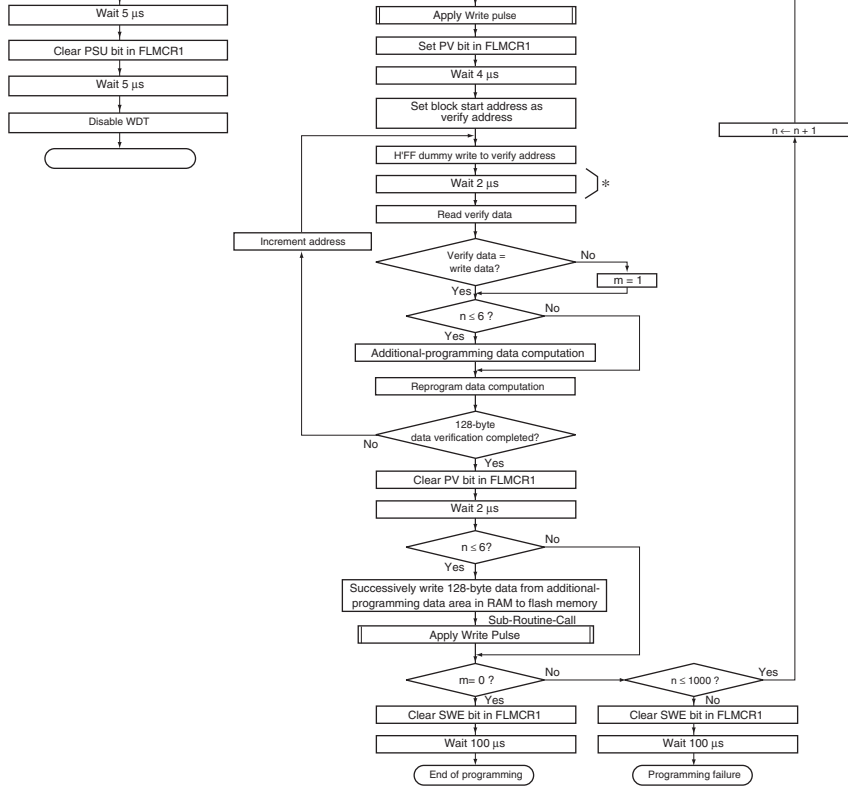


Figure 7.2 Programming/Erasing Flowchart Example in User Program M

7.4.1 Program/Program-Verify

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

1. Programming must be done to an empty address. Do not reprogram an address to which programming has already been performed.
2. Programming should be carried out 128 bytes at a time. A 128-byte data transfer must be performed even if writing fewer than 128 bytes. In this case, H'FF data must be written to the extra addresses.
3. Prepare the following data storage areas in RAM: A 128-byte programming data area, a 128-byte reprogramming data area, and a 128-byte additional-programming data area. Perform reprogramming data computation according to table 7.4, and additional programming data computation according to table 7.5.
4. Consecutively transfer 128 bytes of data in byte units from the reprogramming data area and additional-programming data area to the flash memory. The program address and 128 bytes of data are latched in the flash memory. The lower 8 bits of the start address in the flash memory destination area must be H'00 or H'80.
5. The time during which the P bit is set to 1 is the programming time. Table 7.6 shows the allowable programming times.
6. The watchdog timer (WDT) is set to prevent overprogramming due to program runaway. 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 address whose lower 8 bits are B'00. Verify data can be read in words or in longwords from the address to which the dummy write was performed.



Note: * The RTS instruction must not be used during the following 1. and 2. periods.
 1. A period between 128-byte data programming to flash memory and the P bit clearing
 2. A period between dummy writing of HFF to a verify address and verify data reading

Figure 7.3 Program/Program-Verify Flowchart

Reprogram Data	Verify Data	Additional Program Data	Comments
0	0	0	Additional-program b
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 μ s.

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 address whose low 8 bits are B'00. Verify data can be read in longwords from the address to which a dummy write was performed.
6. If the read data is not erased successfully, set erase mode again, and repeat the erase/verify sequence as before. The maximum number of repetitions of the erase/erase-verify sequence is 100.

7.4.3 Interrupt Handling when Programming/Erasing Flash Memory

All interrupts, including the NMI interrupt, are disabled while flash memory is being programmed or erased, or while the boot program is executing, for the following three reasons:

1. Interrupt during programming/erasing may cause a violation of the programming or erasing algorithm, with the result that normal operation cannot be assured.
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 cannot be carried out.

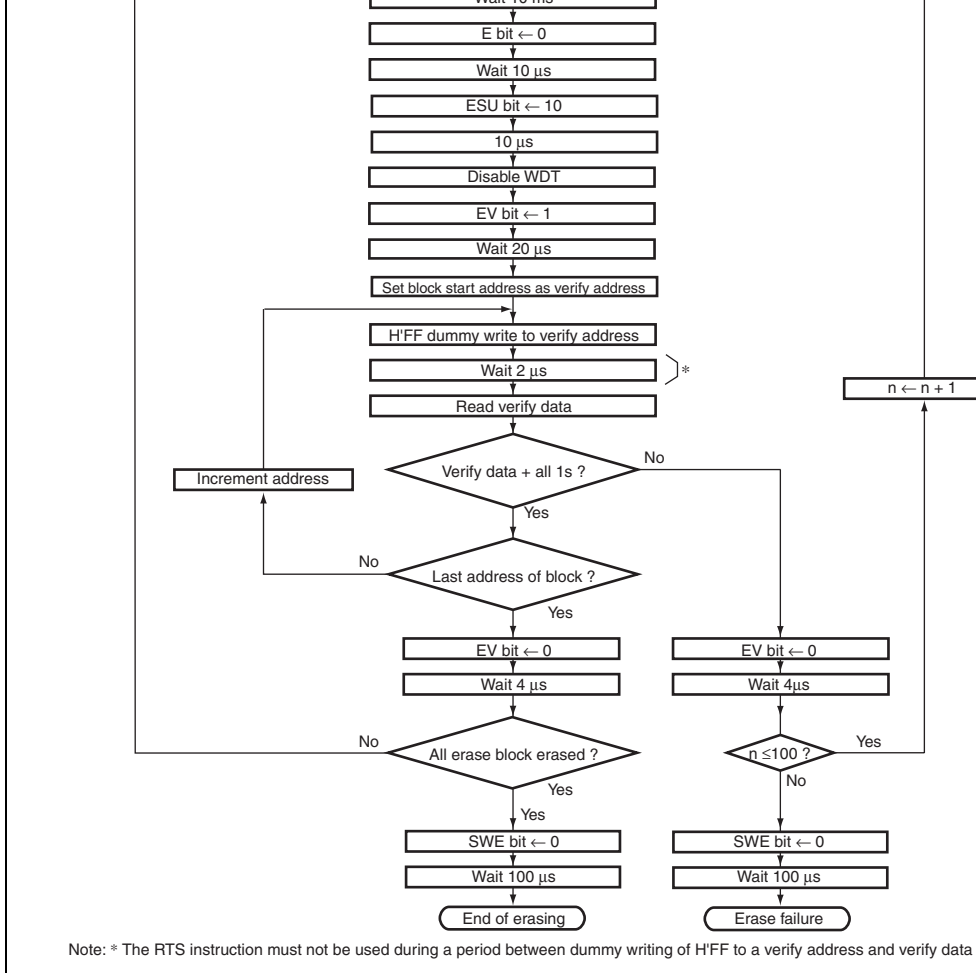


Figure 7.4 Erase/Erase-Verify Flowchart

and erase block register 1 (EBR1) are initialized. In a reset via the $\overline{\text{RES}}$ pin, the reset state is entered unless the $\overline{\text{RES}}$ pin is held low until oscillation stabilizes after powering on. In the event of a reset during operation, hold the $\overline{\text{RES}}$ pin low for the $\overline{\text{RES}}$ pulse width specified in the Characteristics section.

7.5.2 Software Protection

Software protection can be implemented against programming/erasing of all flash memory by clearing the SWE bit in FLMCR1. When software protection is in effect, setting the SWE bit in FLMCR1 does not cause a transition to program mode or erase mode. By setting the erase block register 1 (EBR1), erase protection can be set for individual blocks. When EBR1 is set to 0, 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/erase algorithm, and the program/erase operation is forcibly aborted. Aborting the program/erase operation prevents damage to the flash memory due to overprogramming or overerasing.

When the following errors are detected during programming/erasing of flash memory, the error 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/erasing (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

7.7 Power-Down States for Flash Memory

In user mode, the flash memory will operate in either of the following states:

- Normal operating mode
The flash memory can be read and written to at high speed.
- Power-down operating mode
The power supply circuit of flash memory can be partly halted. As a result, flash memory cannot be read with low power consumption.
- Standby mode
All flash memory circuits are halted.

Table 7.7 shows the correspondence between the operating modes of this LSI and the flash memory. In subactive mode, the flash memory can be set to operate in power-down mode by setting the PDWND bit in FLPWCR. When the flash memory returns to its normal operating state from power-down mode or standby mode, a period to stabilize operation of the power supply circuit that were stopped is needed. When the flash memory returns to its normal operating state from STS2 to STS0 in SYSCR1 must be set to provide a wait time of at least 20 μ s, even when an external clock is being used.

H8/36048	3 kbytes	H'FFE800 to H'FFEFFF, H'FFFB80 to H'FFFF7F
H8/36047	3 kbytes	H'FFE800 to H'FFEFFF, H'FFFB80 to H'FFFF7F

Note: * When the E7 or E8 is used, area H'FFF780 to H'FFFB7F must not be access

units.

For functions in each port, see appendix B.1, I/O Port Block Diagrams. For the execution manipulation instructions to the port control register and port data register, see section 2 Manipulation Instruction.

9.1 Port 1

Port 1 is a general I/O port also functioning as IRQ interrupt input pins, an RTC output pin, a bit PWM output pin, a timer B1 input pin, and a timer V input pin. Figure 9.1 shows its configuration.

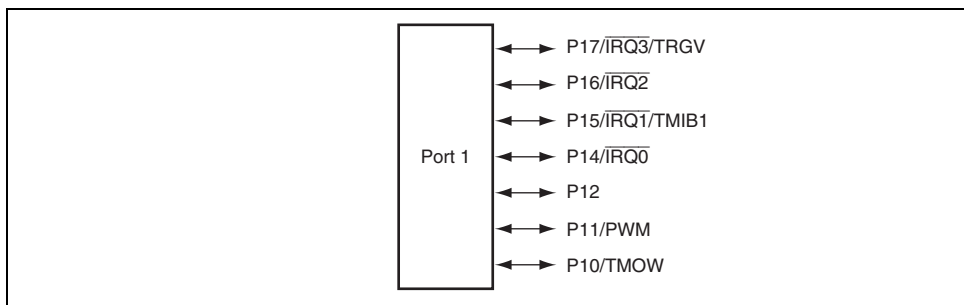


Figure 9.1 Port 1 Pin Configuration

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)

				0: General I/O port 1: $\overline{\text{IRQ2}}$ input pin
5	IRQ1	0	R/W	Selects the function of pin P15/ $\overline{\text{IRQ1}}$ /TMIB1. 0: General I/O port 1: $\overline{\text{IRQ1}}$ /TMIB1 input pin
4	IRQ0	0	R/W	Selects the function of pin P14/ $\overline{\text{IRQ0}}$. 0: General I/O port 1: $\overline{\text{IRQ0}}$ input pin
3	TXD2	0	R/W	Selects the function of pin P72/TXD_2. 0: General I/O port 1: TXD_2 output pin
2	PWM	0	R/W	Selects the function of pin P11/PWM. 0: General I/O port 1: PWM output pin
1	TXD	0	R/W	Selects the function of pin P22/TXD. 0: General I/O port 1: TXD output pin
0	TMOW	0	R/W	Selects the function of pin P10/TMOW. 0: General I/O port 1: TMOW output pin

3	—	—	—
2	PCR12	0	W
1	PCR11	0	W
0	PCR10	0	W

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	PDR1 stores output data for port 1 pins.
6	P16	0	R/W	If PDR1 is read while PCR1 bits are set to 1, the values stored in PDR1 are read. If PDR1 is read while PCR1 bits are cleared to 0, the pin states are read regardless of the value stored in PDR1.
5	P15	0	R/W	
4	P14	0	R/W	
3	—	1	—	Bit 3 is a reserved bit. This bit is always read as 1.
2	P12	0	R/W	
1	P11	0	R/W	
0	P10	0	R/W	

3	—	1	—
2	PUCR12	0	R/W
1	PUCR11	0	R/W
0	PUCR10	0	R/W

9.1.5 Pin Functions

The correspondence between the register specification and the port functions is shown below.

- P17/ $\overline{\text{IRQ3}}$ /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.

- P16/ $\overline{\text{IRQ2}}$ pin

Register	PMR1	PCR1	
Bit Name	IRQ2	PCR16	Pin Function
Setting value	0	0	P16 input pin
		1	P16 output pin
	1	X	$\overline{\text{IRQ2}}$ input pin

[Legend] X: Don't care.

Register	PMR1	PCR1	
Bit Name	IRQ0	PCR14	Pin Function
Setting value	0	0	P14 input pin
		1	P14 output pin
	1	X	$\overline{\text{IRQ0}}$ input pin

[Legend] X: Don't care.

- P12 pin

Register	PCR1	
Bit Name	PCR12	Pin Function
Setting value	0	P12 input pin
	1	P12 output pin

- P11/PWM pin

Register	PMR1	PCR1	
Bit Name	PWM	PCR11	Pin Function
Setting value	0	0	P11 input pin
		1	P11 output pin
	1	X	PWM output pin

[Legend] X: Don't care.

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

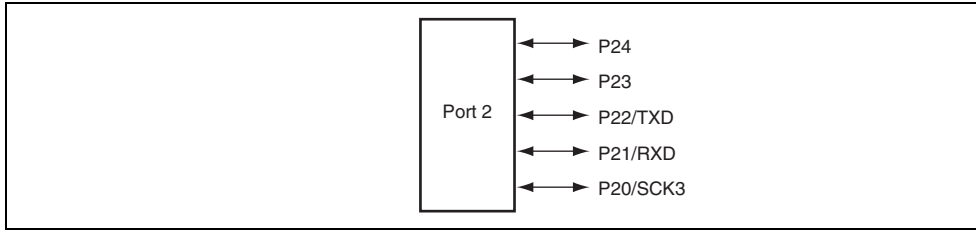


Figure 9.2 Port 2 Pin Configuration

Port 2 has the following registers.

- Port control register 2 (PCR2)
- Port data register 2 (PDR2)
- Port mode register 3 (PMR3)

3	PCR23	0	W	general I/O port, setting a PCR2 bit to 1 makes the
2	PCR22	0	W	corresponding pin an output port, while clearing
1	PCR21	0	W	to 0 makes the pin an input port.
0	PCR20	0	W	

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	—	1	—	Reserved
6	—	1	—	These bits are always read as 1.
5	—	1	—	
4	P24	0	R/W	PDR2 stores output data for port 2 pins.
3	P23	0	R/W	If PDR2 is read while PCR2 bits are set to 1, the
2	P22	0	R/W	stored in PDR2 are read. If PDR2 is read while
1	P21	0	R/W	bits are cleared to 0, the pin states are read
0	P20	0	R/W	of the value stored in PDR2.

3	POF23	0	R/W	by PMOS and it functions as the NMOS open-drain output. When cleared to 0, the pin functions as CMOS output.
2	—	1	—	Reserved
1	—	1	—	These bits are always read as 1.
0	—	1	—	

9.2.4 Pin Functions

The correspondence between the register specification and the port functions is shown below.

- P24 pin

Register		PCR2
Bit Name	PCR24	Pin Function
Setting Value	0	P24 input pin
	1	P24 output pin

- P23 pin

Register		PCR2
Bit Name	PCR23	Pin Function
Setting Value	0	P23 input pin
	1	P23 output 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 3 Pin Configuration

Port 3 has the following registers.

- Port control register 3 (PCR3)
- Port data register 3 (PDR3)

9.3.1 Port Control Register 3 (PCR3)

PCR3 selects inputs/outputs in bit units for pins to be used as general I/O ports of port 3.

Bit	Bit Name	Initial Value	R/W	Description
7	PCR37	0	W	Setting a PCR3 bit to 1 makes the corresponding pin an output port, while clearing the bit to 0 makes it an input port.
6	PCR36	0	W	
5	PCR35	0	W	
4	PCR34	0	W	
3	PCR33	0	W	
2	PCR32	0	W	
1	PCR31	0	W	
0	PCR30	0	W	

3	P33	0	R/W
2	P32	0	R/W
1	P31	0	R/W
0	P30	0	R/W

9.3.3 Pin Functions

The correspondence between the register specification and the port functions is shown below.

- P37 pin

Register		PCR3
Bit Name	PCR37	Pin Function
Setting Value	0	P37 input pin
	1	P37 output pin

- P36 pin

Register		PCR3
Bit Name	PCR36	Pin Function
Setting Value	0	P36 input pin
	1	P36 output pin

Setting Value	0	P34 input pin
	1	P34 output pin

- P33 pin

Register **PCR3**

Bit Name	PCR33	Pin Function
Setting Value	0	P33 input pin
	1	P33 output pin

- P32 pin

Register **PCR3**

Bit Name	PCR32	Pin Function
Setting Value	0	P32 input pin
	1	P32 output pin

- P31 pin

Register **PCR3**

Bit Name	PCR31	Pin Function
Setting Value	0	P31 input pin
	1	P31 output pin

pin, and a wakeup interrupt input pin. Each pin of the port 5 is shown in figure 9.4. The setting of the I²C bus interface has priority for functions of the pins P57/SCL and P56/SDA. The output buffer for pins P56 and P57 has the NMOS push-pull structure, it differs from the output buffer with the CMOS structure in the high-level output characteristics (see section 23, I/O Characteristics).

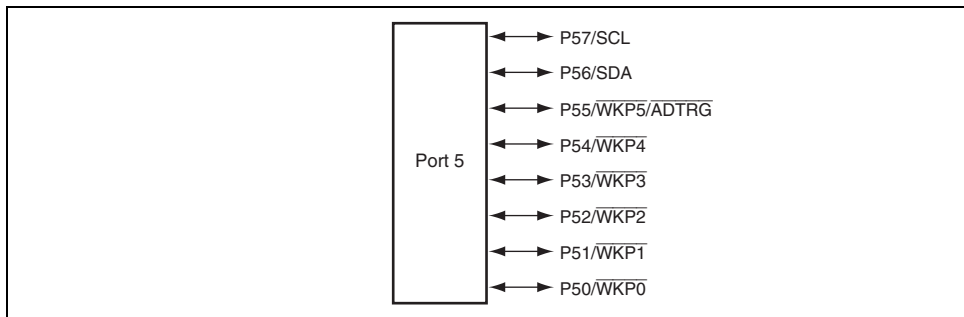


Figure 9.4 Port 5 Pin Configuration

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)

				0: General I/O port 1: $\overline{\text{WKP5/ADTRG}}$ input pin
4	WKP4	0	R/W	Selects the function of pin P54/ $\overline{\text{WKP4}}$. 0: General I/O port 1: $\overline{\text{WKP4}}$ input pin
3	WKP3	0	R/W	Selects the function of pin P53/ $\overline{\text{WKP3}}$. 0: General I/O port 1: $\overline{\text{WKP3}}$ input pin
2	WKP2	0	R/W	Selects the function of pin P52/ $\overline{\text{WKP2}}$. 0: General I/O port 1: $\overline{\text{WKP2}}$ input pin
1	WKP1	0	R/W	Selects the function of pin P51/ $\overline{\text{WKP1}}$. 0: General I/O port 1: $\overline{\text{WKP1}}$ input pin
0	WKP0	0	R/W	Selects the function of pin P50/ $\overline{\text{WKP0}}$. 0: General I/O port 1: $\overline{\text{WKP0}}$ input pin

3	PCR53	0	W
2	PCR52	0	W
1	PCR51	0	W
0	PCR50	0	W

9.4.3 Port Data Register 5 (PDR5)

PDR5 is a general I/O port data register of port 5.

Bit	Bit Name	Initial Value	R/W	Description
7	P57	0	R/W	PDR5 stores output data for port 5 pins.
6	P56	0	R/W	If PDR5 is read while PCR5 bits are set to 1, the states stored in PDR5 are read. If PDR5 is read while PCR5 bits are cleared to 0, the pin states are read regardless of the value stored in PDR5.
5	P55	0	R/W	
4	P54	0	R/W	
3	P53	0	R/W	
2	P52	0	R/W	
1	P51	0	R/W	
0	P50	0	R/W	

3	PUCR53	0	R/W	state when these bits are cleared to 0.
2	PUCR52	0	R/W	
1	PUCR51	0	R/W	
0	PUCR50	0	R/W	

9.4.5 Pin Functions

The correspondence between the register specification and the port functions is shown below.

- P57/SCL pin

Register	ICCR	PCR5	
Bit Name	ICE	PCR57	Pin Function
Setting Value	0	0	P57 input pin
		1	P57 output pin
	1	X	SCL I/O pin

[Legend] X: Don't care.

SCL performs the NMOS open-drain output, that enables a direct bus drive.

- 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.

- P54/ $\overline{\text{WKP4}}$ pin

Register	PMR5	PCR5	
Bit Name	WKP4	PCR54	Pin Function
Setting Value	0	0	P54 input pin
		1	P54 output pin
	1	X	$\overline{\text{WKP4}}$ input pin

[Legend] X: Don't care.

- P53/ $\overline{\text{WKP3}}$ pin

Register	PMR5	PCR5	
Bit Name	WKP3	PCR53	Pin Function
Setting Value	0	0	P53 input pin
		1	P53 output pin
	1	X	$\overline{\text{WKP3}}$ input pin

[Legend] X: Don't care.

Register	PMR5	PCR5	
Bit Name	WKP1	PCR51	Pin Function
Setting Value	0	0	P51 input pin
		1	P51 output pin
	1	X	$\overline{\text{WKP1}}$ input pin

[Legend] X: Don't care.

- P50/ $\overline{\text{WKP0}}$ pin

Register	PMR5	PCR5	
Bit Name	WKP0	PCR50	Pin Function
Setting Value	0	0	P50 input pin
		1	P50 output pin
	1	X	$\overline{\text{WKP0}}$ input pin

[Legend] X: Don't care.



Figure 9.5 Port 6 Pin Configuration

Port 6 has the following registers.

- Port control register 6 (PCR6)
- Port data register 6 (PDR6)

9.5.1 Port Control Register 6 (PCR6)

PCR6 selects inputs/outputs in bit units for pins to be used as general I/O ports of port 6.

Bit	Bit Name	Initial Value	R/W	Description
7	PCR67	0	W	When each of the port 6 pins P67 to P60 function as a general I/O port, setting a PCR6 bit to 1 makes the corresponding pin an output port, while clearing it to 0 makes the pin an input port.
6	PCR66	0	W	
5	PCR65	0	W	
4	PCR64	0	W	
3	PCR63	0	W	
2	PCR62	0	W	
1	PCR61	0	W	
0	PCR60	0	W	

2	P62	0	R/W
1	P61	0	R/W
0	P60	0	R/W

9.5.3 Pin Functions

The correspondence between the register specification and the port functions is shown below.

- P67/FTIOD1 pin

Register	TOER	TFCR	TPMR	TIORC1	PCR6	
Bit Name	ED1	CMD1, CMD0	PWMD1	IOD2 to IOD0	PCR67	Pin Function
Setting Value	1	00	0	000 or 1XX	0	P67 input/ input pin
					1	P67 output
	0	00	0	001 or 01X	X	FTIOD1 o
			1	XXX		
		Other than 00	X	XXX		

[Legend] X: Don't care.

Other than X
00

XXX

[Legend] X: Don't care.

- P65/FTIOB1 pin

Register	TOER	TFCR	TPMR	TIORA1	PCR6	
Bit Name	EB1	CMD1, CMD0	PWMB1	IOB2 to IOB0	PCR65	Pin Function
Setting Value	1	00	0	000 or 1XX	0	P65 input input pin
					1	P65 output
	0	00	0	001 or 01X	X	FTIOB1
			1	XXX		
		Other than 00	X	XXX		

[Legend] X: Don't care.

- P64/FTIOA1 pin

Register	TOER	TFCR	TIORA1	PCR6	
Bit Name	EA1	CMD1, CMD0	IOA2 to IOA0	PCR64	Pin Function
Setting Value	1	XX	000 or 1XX	0	P64 input input pin
				1	P64 output
	0	00	001 or 01X	X	FTIOA1

[Legend] X: Don't care.

[Legend] X: Don't care.

- P62/FTIOC0 pin

Register	TOER	TFCR	TPMR	TIORC0	PCR6	
Bit Name	EC0	CMD1, CMD0	PWMC0	IOC2 to IOC0	PCR62	Pin Function
Setting Value	1	00	0	000 or 1XX	0	P62 input/ input pin
					1	P62 output
	0	00	0	001 or 01X	X	FTIOC0 o
			1	XXX		
		Other than 00	X	XXX		

[Legend] X: Don't care.

[Legend] X: Don't care.

- P60/FTIOA0 pin

Register	TOER	TFCR	TFCR	TIORA0	PCR6	
Bit Name	EA0	CMD1, CMD0	STCLK	IOA2 to IOA0	PCR60	Pin Functi
Setting	1	XX	X	000 or	0	P60 input input pin
Value				1XX	1	P60 outp
	0	00	0	001 or 01X	X	FTIOA0

[Legend] X: Don't care.

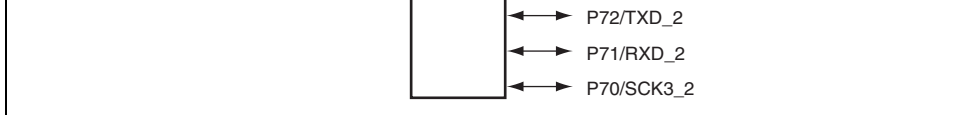


Figure 9.6 Port 7 Pin Configuration

Port 7 has the following registers.

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

9.6.1 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	PCR77	0	W	When each of the port 7 pins P77 to P74 and P70 functions as a general I/O port, setting a bit to 1 makes the corresponding pin an output port. Clearing the bit to 0 makes the pin an input port. Bit 3 is a reserved bit.
6	PCR76	0	W	
5	PCR75	0	W	
4	PCR74	0	W	
3	—	—	—	
2	PCR72	0	W	
1	PCR71	0	W	
0	PCR70	0	W	

3	—	1	—	Bit 3 is a reserved bit. This bit is always read
2	P72	0	R/W	
1	P71	0	R/W	
0	P70	0	R/W	

9.6.3 Pin Functions

The correspondence between the register specification and the port functions is shown below.

- P77 pin

Register	PCR7	
Bit Name	PCR77	Pin Function
Setting Value	0	P77 input pin
	1	P77 output pin

- P76/TMOV pin

Register	TCSR.V	PCR7	
Bit Name	OS3 to OS0	PCR76	Pin Function
Setting Value	0000	0	P76 input pin
		1	P76 output pin
	Other than above	X	TMOV output pin

[Legend] X: Don't care.

Setting Value	0	P74 input/TMRIV input pin
	1	P74 output/TMRIV input pin

- P72/TXD_2 pin

Register	PMR1	PCR7	
Bit Name	TXD2	PCR72	Pin Function
Setting Value	0	0	P72 input pin
		1	P72 output pin
	1	X	TXD_2 output pin

[Legend] X: Don't care.

- P71/RXD_2 pin

Register	SCR3_2	PCR7	
Bit Name	RE	PCR71	Pin Function
Setting Value	0	0	P71 input pin
		1	P71 output pin
	1	X	RXD_2 input pin

[Legend] X: Don't care.

9.7 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 9.7. The register setting of the timer W has priority for functions of the pins P87, P83/FTIOC, P82/FTIOB, and P81/FTIOA. The P80/FTCI pin also functions as a timer W port that is connected to the timer W regardless of the register setting of port 8.

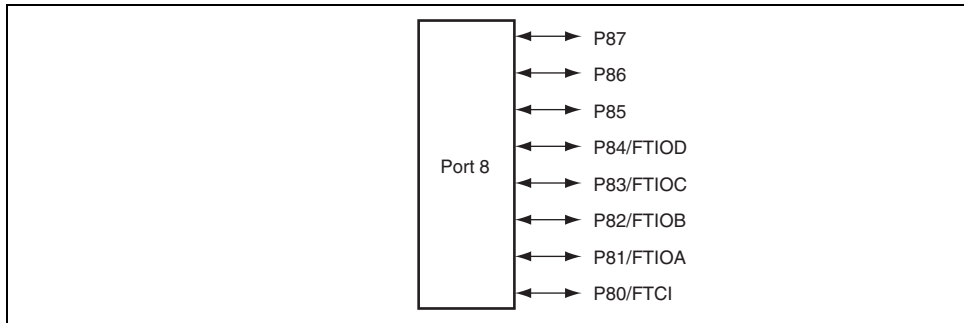


Figure 9.7 Port 8 Pin Configuration

Port 8 has the following registers.

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

3	PCR83	0	W
2	PCR82	0	W
1	PCR81	0	W
0	PCR80	0	W

9.7.2 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	P87	0	R/W	PDR8 stores output data for port 8 pins.
6	P86	0	R/W	If PDR8 is read while PCR8 bits are set to 1, the values stored in PDR8 are read. If PDR8 is read while PCR8 bits are cleared to 0, the pin states are read regardless of the value stored in PDR8.
5	P85	0	R/W	
4	P84	0	R/W	
3	P83	0	R/W	
2	P82	0	R/W	
1	P81	0	R/W	
0	P80	0	R/W	

- P86 pin

Register		PCR8
Bit Name	PCR86	Pin Function
Setting Value	0	P86 input pin
	1	P86 output pin

- P85 pin

Register		PCR8
Bit Name	PCR85	Pin Function
Setting Value	0	P85 input pin
	1	P85 output pin

- P84/FTIOD pin

Register		TMRW			TIOR1			PCR8		
Bit Name	PWMD	IOD2	IOD1	IOD0	PCR84	Pin Function				
Setting Value	0	0	0	0	0	P84 input/FTIOD inp				
					1	P84 output/FTIOD in				
	1	0	0	0	1	FTIOD output pin				
					X	FTIOD output pin				
					0	P84 input/FTIOD inp				
	1	X	X	X	X	1	P84 output/FTIOD in			
						X	PWM output			

[Legend] X: Don't care.

[Legend] X: Don't care.

- P82/FTIOB pin

Register	TMRW	TIOR0			PCR8	Pin Function				
Bit Name	PWMB	IOB2	IOB1	IOB0	PCR82					
Setting Value	0	0	0	0	0	P82 input/FTIOB input				
					1	P82 output/FTIOB input				
	0	0	1	X	X	FTIOB output pin				
						0	1	X	X	FTIOB output pin
						1	X	X	0	P82 input/FTIOB input
	1	X	X	X	X	1	P82 output/FTIOB input			
						X	X	X	X	PWM output

[Legend] X: Don't care.

[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

9.8 Port 9

Port 9 is a general I/O port also functioning as an SCI3_3 I/O pin. Each pin of the port 9 is shown in figure 9.8. The register setting of the SCI3_3 has priority for functions of the pins for

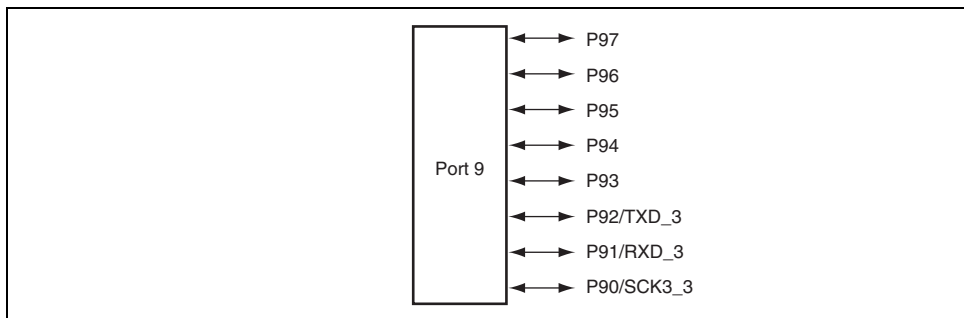


Figure 9.8 Port 9 Pin Configuration

7	PCR97	0	W	When each of the port 9 pins P97 to P90 function as a general I/O port, setting a PCR9 bit to 1 makes the corresponding pin an output port, while clearing it to 0 makes the pin an input port.
6	PCR96	0	W	
5	PCR95	0	W	
4	PCR94	0	W	
3	PCR93	0	W	
2	PCR92	0	W	
1	PCR91	0	W	
0	PCR90	0	W	

9.8.2 Port Data Register 9 (PDR9)

PDR9 is a general I/O port data register of port 9.

Bit	Bit Name	Initial Value	R/W	Description
7	P97	0	R/W	PDR9 stores output data for port 9 pins.
6	P96	0	R/W	If PDR9 is read while PCR9 bits are set to 1, the values stored in PDR9 are read. If PDR9 is read while PCR9 bits are cleared to 0, the pin states are read regardless of the value stored in PDR9.
5	P95	0	R/W	
4	P94	0	R/W	
3	P93	0	R/W	
2	P92	0	R/W	
1	P91	0	R/W	
0	P90	0	R/W	

- P96 pin

Register		PCR9
Bit Name	PCR96	Pin Function
Setting Value	0	P96 input pin
	1	P96 output pin

- P95 pin

Register		PCR9
Bit Name	PCR95	Pin Function
Setting Value	0	P95 input pin
	1	P95 output pin

- P94 pin

Register		PCR9
Bit Name	PCR94	Pin Function
Setting Value	0	P94 input pin
	1	P94 output pin

- P93 pin

Register		PCR9
Bit Name	PCR93	Pin Function
Setting Value	0	P93 input pin
	1	P93 output pin

Register	SCR3_3	PCR9	
Bit Name	RE	PCR91	Pin Function
Setting Value	0	0	P91 input pin
		1	P91 output pin
	1	X	RXD_3 input pin

[Legend] X: Don't care.

- P90/SCK3_3 pin

Register	SCR3_3		SMR3_3	PCR9	
Bit Name	CKE1	CKE0	COM	PCR90	Pin Function
Setting Value	0	0	0	0	P90 input pin
				1	P90 output pin
	0	0	1	X	SCK3_3 output pin
	0	1	X	X	SCK3_3 output pin
	1	X	X	X	SCK3_3 input pin

[Legend] X: Don't care.

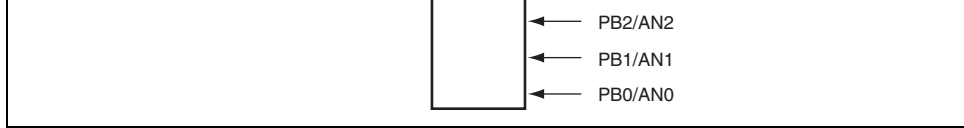


Figure 9.9 Port B Pin Configuration

Port B has the following register.

- Port data register B (PDRB)

9.9.1 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	PB7	—	R	The input value of each pin is read by reading register. However, if a port B pin is designated as an analog input channel by ADCSR of A/D converter, 0
6	PB6	—	R	
5	PB5	—	R	
4	PB4	—	R	
3	PB3	—	R	
2	PB2	—	R	
1	PB1	—	R	
0	PB0	—	R	

- Readable/writable counter of seconds, minutes, hours, and day-of-week with BCD output
- Periodic (seconds, minutes, hours, days, and weeks) interrupts
- 8-bit free running counter
- Selection of clock source

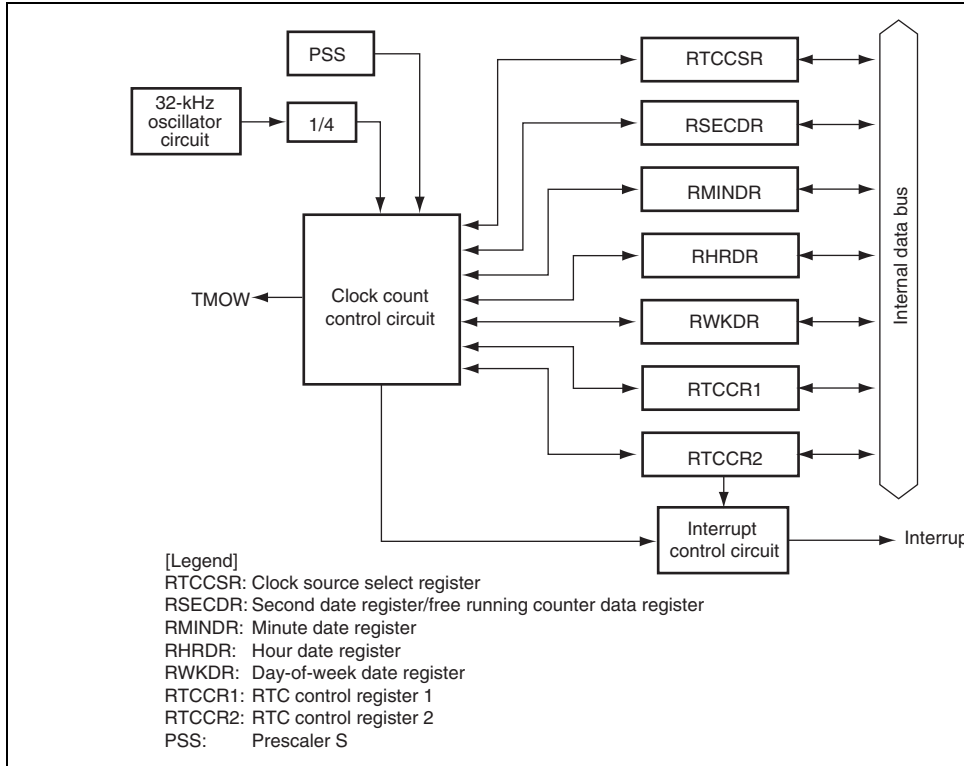


Figure 10.1 Block Diagram of RTC

The RTC has the following registers.

- Second data register/free running counter data register (RSECDR)
- Minute data register (RMINDR)
- Hour data register (RHRDR)
- Day-of-week data register (RWKDR)
- RTC control register 1 (RTCCR1)
- RTC control register 2 (RTCCR2)
- Clock source select register (RTCCSR)

data registers. When this bit is 0, the values of minute, hour, and day-of-week data registers adopted.

6	SC12	—	R/W	Counting Ten's Position of Seconds
5	SC11	—	R/W	Counts on 0 to 5 for 60-second counting.
4	SC10	—	R/W	
3	SC03	—	R/W	Counting One's Position of Seconds
2	SC02	—	R/W	Counts on 0 to 9 once per second. When a carry generated, 1 is added to the ten's position.
1	SC01	—	R/W	
0	SC00	—	R/W	

minute, hour, and day-of-week data registers n
adopted.

6	MN12	—	R/W	Counting Ten's Position of Minutes
5	MN11	—	R/W	Counts on 0 to 5 for 60-minute counting.
4	MN10	—	R/W	
3	MN03	—	R/W	Counting One's Position of Minutes
2	MN02	—	R/W	Counts on 0 to 9 once per minute. When a carry
1	MN01	—	R/W	generated, 1 is added to the ten's position.
0	MN00	—	R/W	

data registers. When this bit is 0, the values of minute, hour, and day-of-week data registers adopted.

6	—	0	—	Reserved This bit is always read as 0.
5	HR11	—	R/W	Counting Ten's Position of Hours
4	HR10	—	R/W	Counts on 0 to 2 for ten's position of hours.
3	HR03	—	R/W	Counting One's Position of Hours
2	HR02	—	R/W	Counts on 0 to 9 once per hour. When a carry generated, 1 is added to the ten's position.
1	HR01	—	R/W	
0	HR00	—	R/W	

minute, hour, and day-of-week data registers n
adopted.

6	—	0	—	Reserved
5	—	0	—	These bits are always read as 0.
4	—	0	—	
3	—	0	—	
<hr/>				
2	WK2	—	R/W	Day-of-Week Counting
1	WK1	—	R/W	Day-of-week is indicated with a binary code
0	WK0	—	R/W	000: Sunday 001: Monday 010: Tuesday 011: Wednesday 100: Thursday 101: Friday 110: Saturday 111: Reserved (setting prohibited)

				0: RTC operates in 12-hour mode. RHRDR code to 11. 1: RTC operates in 24-hour mode. RHRDR code to 23.
5	PM	—	R/W	a.m./p.m. 0: Indicates a.m. when RTC is in the 12-hour mode. 1: Indicates p.m. when RTC is in the 12-hour mode.
4	RST	0	R/W	Reset 0: Normal operation 1: Resets registers and control circuits except for the RTC and this bit. Clear this bit to 0 after having bit 1.
3	INT	—	R/W	Interrupt Generation Timing 0: Generates a second, minute, hour, or day-periodic interrupt during RTC busy period. 1: Generates a second, minute, hour, or day-periodic interrupt immediately after completing RTC busy period.
2	—	0	—	Reserved
1	—	0	—	These bits are always read as 0.
0	—	0	—	

5	FOIE	—	R/W	Free Running Counter Overflow Interrupt Enable 0: Disables an overflow interrupt 1: Enables an overflow interrupt
4	WKIE	—	R/W	Week Periodic Interrupt Enable 0: Disables a week periodic interrupt 1: Enables a week periodic interrupt
3	DYIE	—	R/W	Day Periodic Interrupt Enable 0: Disables a day periodic interrupt 1: Enables a day periodic interrupt
2	HRIE	—	R/W	Hour Periodic Interrupt Enable 0: Disables an hour periodic interrupt 1: Enables an hour periodic interrupt
1	MNIE	—	R/W	Minute Periodic Interrupt Enable 0: Disables a minute periodic interrupt 1: Enables a minute periodic interrupt
0	SEIE	—	R/W	Second Periodic Interrupt Enable 0: Disables a second periodic interrupt 1: Enables a second periodic interrupt

7	—	0	—	Reserved This bit is always read as 0.
6	RCS6	0	R/W	Clock Output Selection
5	RCS5	0	R/W	Selects a clock output from the TMOW pin when TMOW in PMR1 to 1. 00: $\phi/4$ 01: $\phi/8$ 10: $\phi/16$ 11: $\phi/32$
4	—	0	—	Reserved This bit is always read as 0.
3	RCS3	1	R/W	Clock Source Selection
2	RCS2	0	R/W	0000: $\phi/8$ Free running counter operation
1	RCS1	0	R/W	0001: $\phi/32$ Free running counter operation
0	RCS0	0	R/W	0010: $\phi/128$ Free running counter operation 0011: $\phi/256$ Free running counter operation 0100: $\phi/512$ Free running counter operation 0101: $\phi/2048$ Free running counter operation 0110: $\phi/4096$ Free running counter operation 0111: $\phi/8192$ Free running counter operation 1XXX: 32.768 kHz...RTC operation

[Legend]

X: Don't care

Figure 10.3 shows the procedure for the initial setting of the RTC. To set the RTC again, follow this procedure.

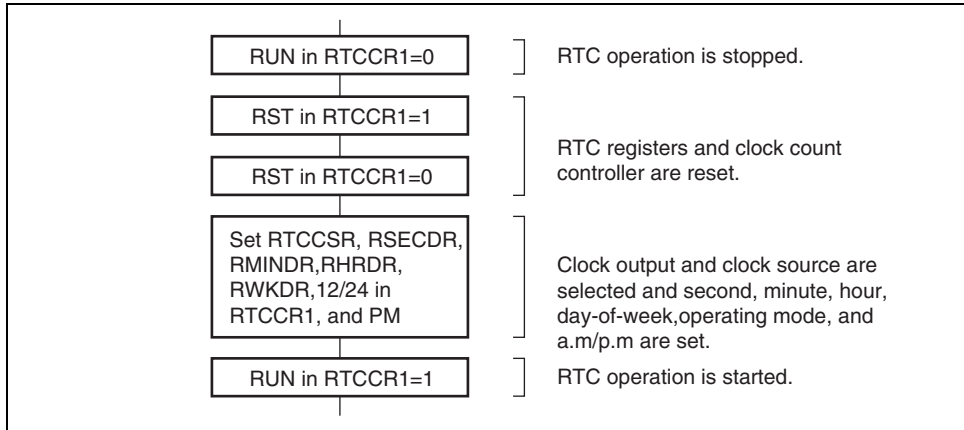


Figure 10.3 Initial Setting Procedure

bit is set to 1, the registers are updated, and the BSY bit is cleared to 0.

2. Making use of interrupts, read from the second, minute, hour, and day-of week registers. The IRRTA flag in IRR1 is set to 1 and the BSY bit is confirmed to be 0.
3. Read from the second, minute, hour, and day-of week registers twice in a row, and if there is no change in the read data, the read data is used.

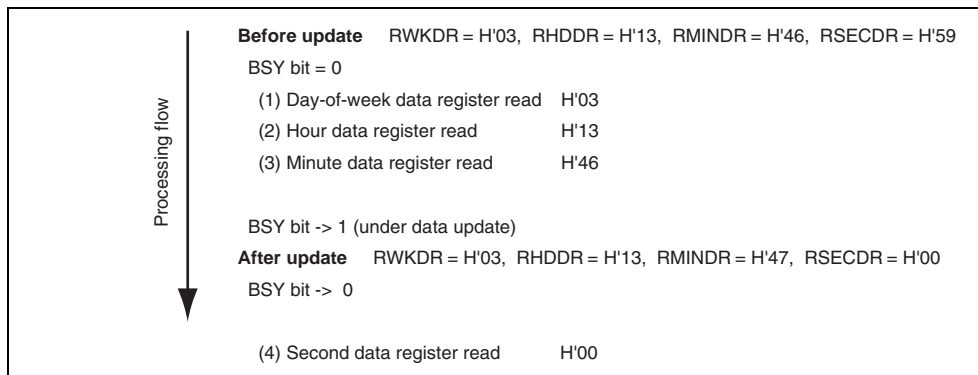


Figure 10.4 Example: Reading of Inaccurate Time Data

Table 10.2 Interrupt Sources

Interrupt Name	Interrupt Source	Interrupt Enable
Overflow interrupt	Occurs when the free running counter is overflowed.	FOIE
Week periodic interrupt	Occurs every week when the day-of-week date register value becomes 0.	WKIE
Day periodic interrupt	Occurs every day when the day-of-week date register is counted.	DYIE
Hour periodic interrupt	Occurs every hour when the hour date register is counted.	HRIE
Minute periodic interrupt	Occurs every minute when the minute date register is counted.	MNIE
Second periodic interrupt	Occurs every second when the second date register is counted.	SCIE

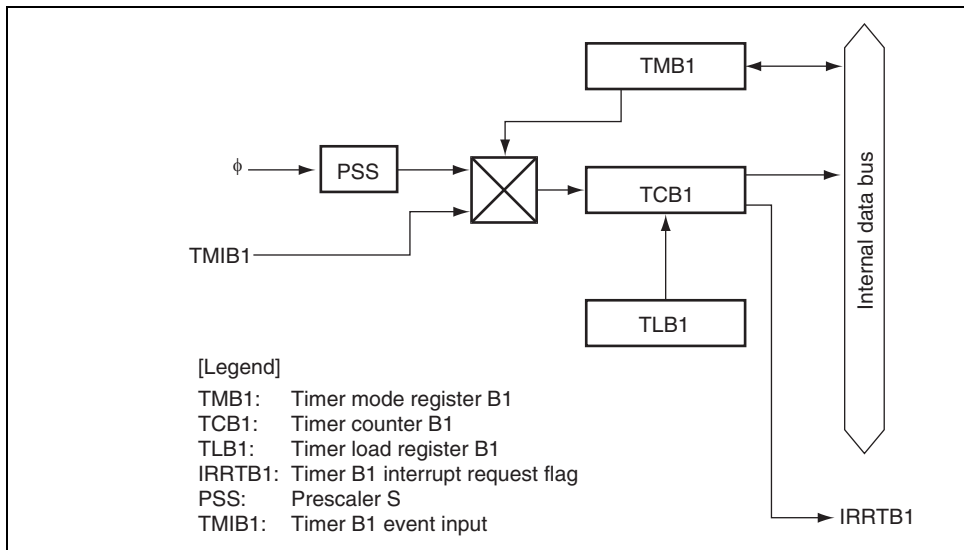


Figure 11.1 Block Diagram of Timer B1

11.2 Input/Output Pin

Table 11.1 shows the timer B1 pin configuration.

Table 11.1 Pin Configuration

Name	Abbreviation	I/O	Function
Timer B1 event input	TMIB1	Input	Event input to TCB1

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	—	Reserved
5	—	1	—	These bits are always read as 1.
4	—	1	—	
3	—	1	—	
2	TMB12	0	R/W	Clock Select
1	TMB11	0	R/W	000: Internal clock: $\phi/8192$
0	TMB10	0	R/W	001: Internal clock: $\phi/2048$ 010: Internal clock: $\phi/512$ 011: Internal clock: $\phi/256$ 100: Internal clock: $\phi/64$ 101: Internal clock: $\phi/16$ 110: Internal clock: $\phi/4$ 111: External event (TMIB1): rising or falling edge

Note: * The edge of the external event signal selected by bit IEG1 in the interrupt edge select register 1 (IEGR1). See section Interrupt Edge Select Register 1 (IEGR1) details. Before setting TMB12 to TMB10, bit IEG1 in the port mode register 1 (PMR1) should be set to 1.

TLB1 is an 8-bit write-only register for setting the reload value of TCB1. When a reload value is set in TLB1, the same value is loaded into TCB1 as well, and TCB1 starts counting up from that value. When TCB1 overflows during operation in auto-reload mode, the TLB1 value is loaded into TCB1. Accordingly, overflow periods can be set within the range of 1 to 256 input clock cycles. TLB1 is allocated to the same address as TCB1. TLB1 is initialized to H'00.

11.4 Operation

11.4.1 Interval Timer Operation

When bit TMB17 in TMB1 is cleared to 0, timer B1 functions as an 8-bit interval timer. After a reset, TCB1 is cleared to H'00 and bit TMB17 is cleared to 0, so up-counting and interval timer operation resume immediately. The operating clock of timer B1 is selected from seven internal clock sources by prescaler S, or an external clock input at pin TMB1. The selection is made by bits TMB12 to TMB10 in TMB1.

After the count value in TMB1 reaches H'FF, the next clock signal input causes timer B1 overflow, setting flag IRRTB1 in IRR2 to 1. If IENTB1 in IENR2 is 1, an interrupt is recognized by the CPU.

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

TCB1.

11.4.3 Event Counter Operation

Timer B1 can operate as an event counter in which TMIB1 is set to an event input pin. External event counting is selected by setting bits TMB12 to TMB10 in TMB1 to 1. TCB1 counts rising or falling edge of an external event signal input at pin TMB1.

When timer B1 is used to count external event input, bit IRQ1 in PMR1 should be set to 1. IEN1 in IENR1 should be cleared to 0 to disable IRQ1 interrupt requests.

11.5 Timer B1 Operating Modes

Table 11.2 shows the timer B1 operating modes.

Table 11.2 Timer B1 Operating Modes

	Operating Mode	Reset	Active	Sleep	Subactive	Subsleep	Standby
TCB1	Interval	Reset	Functions	Functions	Halted	Halted	Halting
	Auto-reload	Reset	Functions	Functions	Halted	Halted	Halting
TMB1		Reset	Functions	Retained	Retained	Retained	Retained

- Choice of seven clock signals is available.
Choice of six internal clock sources ($\phi/128$, $\phi/64$, $\phi/32$, $\phi/16$, $\phi/8$, $\phi/4$) or an external clock source.
- Counter can be cleared by compare match A or B, or by an external reset signal. If the stop function is selected, the counter can be halted when cleared.
- Timer output is controlled by two independent compare match signals, enabling pulse width modulation (PWM) 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 edge, or both edges of the TRGV input can be selected.

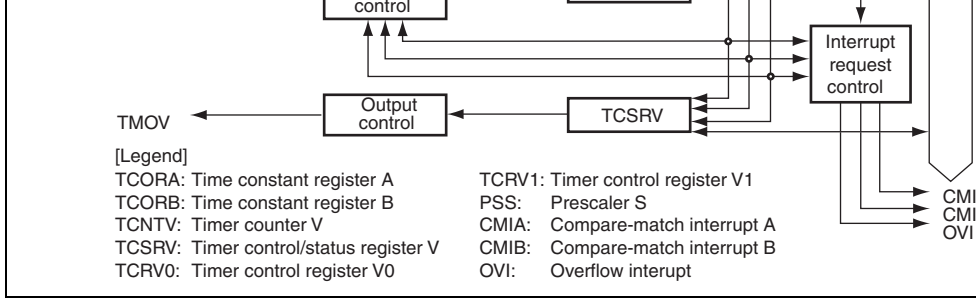


Figure 12.1 Block Diagram of Timer V

12.3 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 (TCSR V)
- Timer control register V1 (TCRV1)

12.3.1 Timer Counter V (TCNTV)

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

TCNTV is initialized to H'00.

and the settings of bits OS3 to OS0 in TCSR.V.

TCORA and TCORB are initialized to H'FF.

				When this bit is set to 1, interrupt request from CMFA bit in TCSR _V is enabled.
5	OVIE	0	R/W	Timer Overflow Interrupt Enable When this bit is set to 1, interrupt request from bit in TCSR _V is enabled.
4	CCLR1	0	R/W	Counter Clear 1 and 0
3	CCLR0	0	R/W	These bits specify the clearing conditions of TCRV1. 00: Clearing is disabled 01: Cleared by compare match A 10: Cleared by compare match B 11: Cleared on the rising edge of the TMRIV pin during operation of TCNTV after clearing depends on TRGE in TCRV1.
2	CKS2	0	R/W	Clock Select 2 to 0
1	CKS1	0	R/W	These bits select clock signals to input to TCRV1.
0	CKS0	0	R/W	the counting condition in combination with ICH in TCRV1. Refer to table 12.2.

		1	0	Internal clock: counts on $\phi/64$, falling edge
			1	Internal clock: counts on $\phi/128$, falling edge
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 falling edge

12.3.4 Timer Control/Status Register V (TCSR_V)

TCSR_V indicates the status flag and controls outputs by using a compare match.

Bit	Bit Name	Initial Value	R/W	Description
7	CMFB	0	R/W	Compare Match Flag B Setting condition: When the TCNTV value matches the TCORB value Clearing condition: After reading CMFB = 1, cleared by writing 0 to CMFB
6	CMFA	0	R/W	Compare Match Flag A Setting condition: When the TCNTV value matches the TCORA value Clearing condition: After reading CMFA = 1, cleared by writing 0 to CMFA

2	OS2	0	R/W	These bits select an output method for the TC the compare match of TCORB and TCNTV. 00: No change 01: 0 output 10: 1 output 11: Output toggles
1	OS1	0	R/W	Output Select 1 and 0
0	OS0	0	R/W	These bits select an output method for the TC the compare match of TCORA 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 output level for compare match A. The two output levels can be controlled independently. After a reset, the timer output is 0 until the first compare match.

3	TVEG0	0	R/W	<p>These bits select the TRGV input edge.</p> <p>00: TRGV trigger input is prohibited</p> <p>01: Rising edge is selected</p> <p>10: Falling edge is selected</p> <p>11: Rising and falling edges are both selected</p>
2	TRGE	0	R/W	<p>TCNT starts counting up by the input of the edge selected by TVEG1 and TVEG0.</p> <p>0: Disables starting counting-up TCNTV by the TRGV pin and halting counting-up TCNTV. TCNTV is cleared by a compare match.</p> <p>1: Enables starting counting-up TCNTV by the TRGV pin and halting counting-up TCNTV. TCNTV is cleared by a compare match.</p>
1	—	1	—	<p>Reserved</p> <p>This bit is always read as 1.</p>
0	ICKS0	0	R/W	<p>Internal Clock Select 0</p> <p>This bit selects clock signals to input to TCNTV in combination with CKS2 to CKS0 in TCRV0.</p> <p>Refer to table 12.2.</p>

will be set. The timing at this time is shown in figure 12.4. An interrupt request is set for the CPU when OVIE in TCRV0 is 1.

3. TCNTV is constantly compared with TCORA and TCORB. Compare match flag A or B (CMFA or CMFB) is set to 1 when TCNTV matches TCORA or TCORB, respectively. A compare-match signal is generated in the last state in which the values match. Figure 12.5 shows the timing. An interrupt request is generated for the CPU when CMIEA or CMIEB in TCRV0 is 1.
4. When a compare match A or B is generated, the TMOV responds with the output value selected by bits OS3 to OS0 in TCSR. Figure 12.6 shows the timing when the output is toggled by compare match A.
5. When CCLR1 or CCLR0 in TCRV0 is 01 or 10, TCNTV can be cleared by the corresponding compare match. Figure 12.7 shows the timing.
6. When CCLR1 or CCLR0 in TCRV0 is 11, TCNTV can be cleared by the rising edge of the input of TMRIV pin. A TMRIV input pulse-width of at least 1.5 system clocks is needed. Figure 12.8 shows the timing.
7. When a counter-clearing source is generated with TRGE in TCRV1 set to 1, the counter is halted as soon as TCNTV is cleared. TCNTV resumes counting-up when the edge selector TVEG1 or TVEG0 in TCRV1 is input from the TGRV pin.

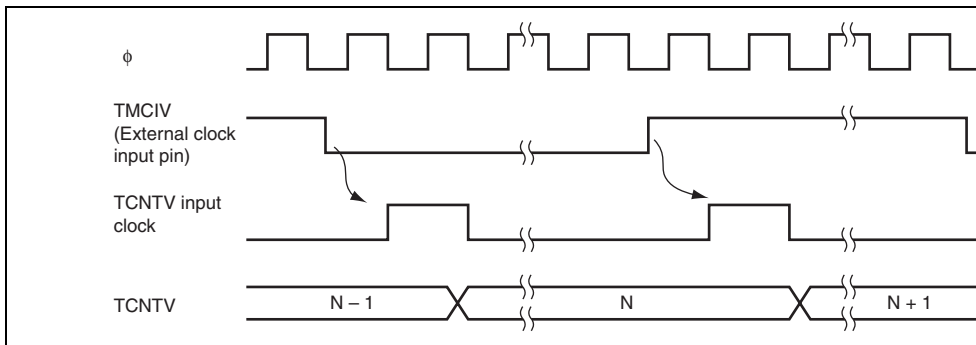


Figure 12.3 Increment Timing with External Clock

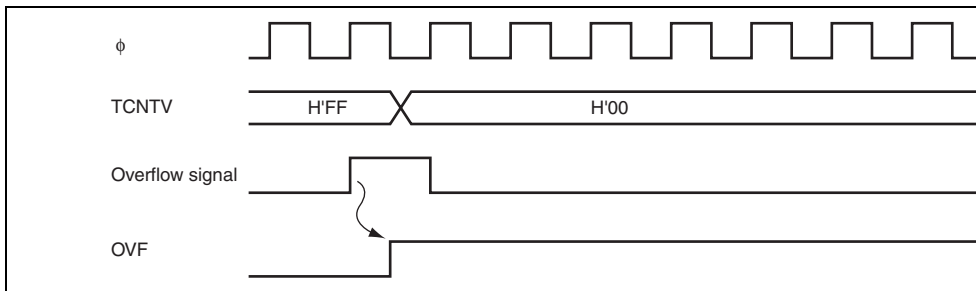


Figure 12.4 OVF Set Timing

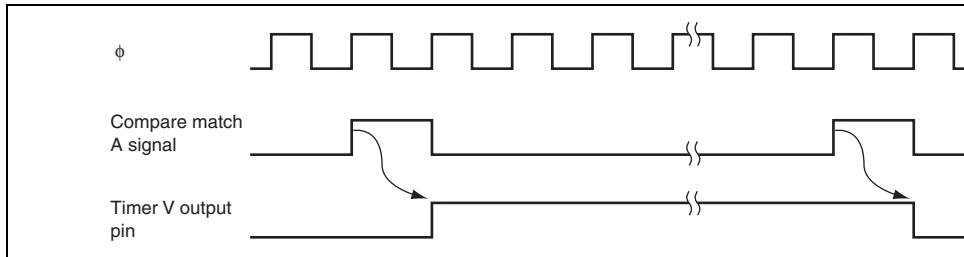


Figure 12.6 TMOV Output Timing

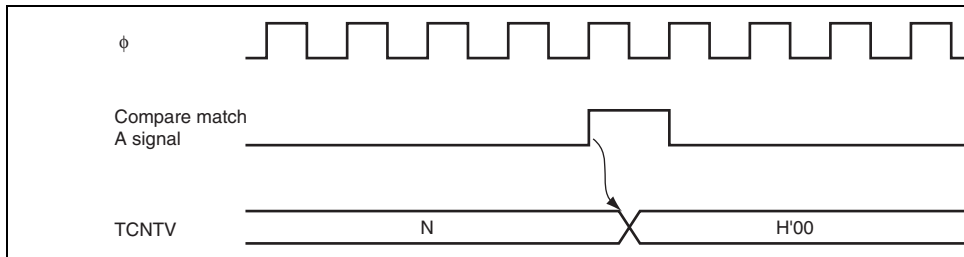


Figure 12.7 Clear Timing by Compare Match

12.5 Timer V Application Examples

12.5.1 Pulse Output with Arbitrary Duty Cycle

Figure 12.9 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 match with TCORA.
2. Set bits OS3 to OS0 in TCSRV so that the output will go to 1 at compare match with TCORA 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 clock.
4. With these settings, a waveform is output without further software intervention, with a period determined by TCORA and a pulse width determined by TCORB.

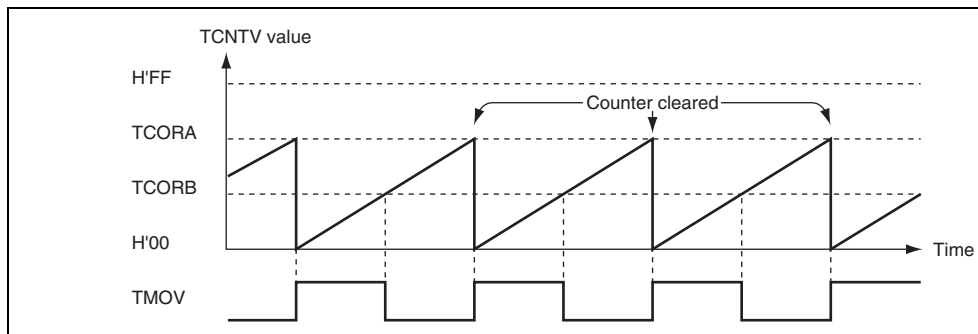


Figure 12.9 Pulse Output Example

- input.
- Set bits CKS2 to CKS0 in TCRV0 and bit ICKS0 in TCRV1 to select the desired clock.
 - With these settings, a pulse waveform will be output without further software intervention with a delay determined by TCORA from the TRGV input, and a pulse width determined by TCORB - TCORA (TCORB - TCORA).

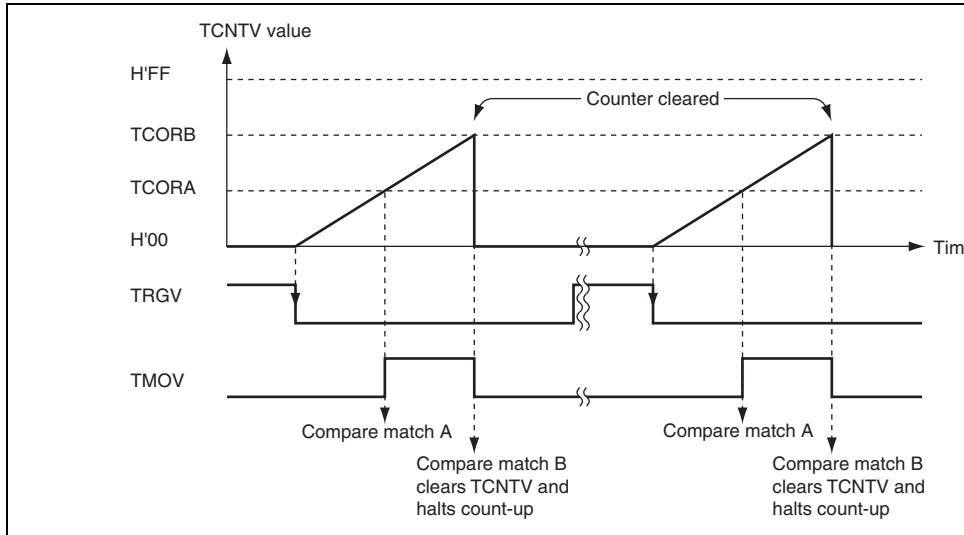


Figure 12.10 Example of Pulse Output Synchronized to TRGV Input

3. If compare matches A and B occur simultaneously, any conflict between the output signal for compare match A and compare match B is resolved by the following priority: toggle output > output 1 > output 0.
4. Depending on the timing, TCNTV may be incremented by a switch between different clock sources. When TCNTV is internally clocked, an increment pulse is generated from the falling edge of an internal clock signal, that is divided system clock (ϕ). Therefore, as seen in figure 12.3 the switch is from a high clock signal to a low clock signal, the switch is seen as a falling edge, causing TCNTV to increment. TCNTV can also be incremented by a switch between internal and external clocks.

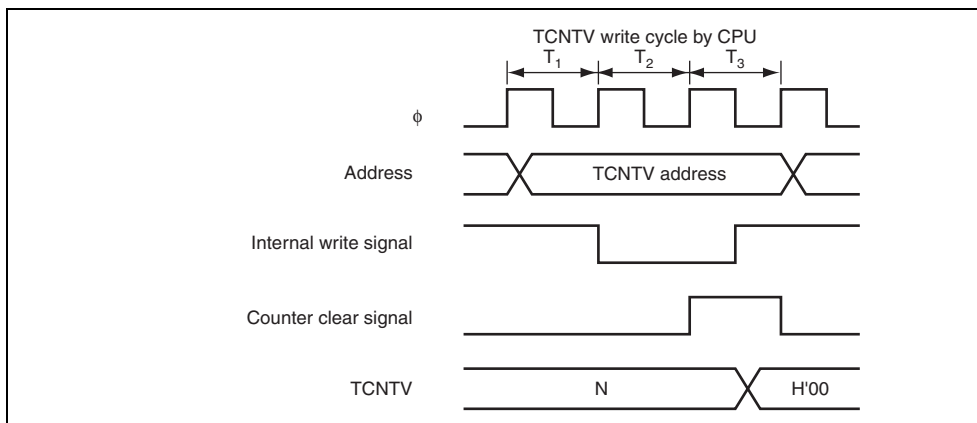


Figure 12.11 Contention between TCNTV Write and Clear

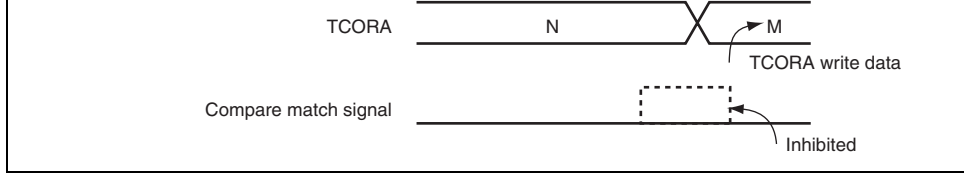


Figure 12.12 Contention between TCORA Write and Compare Match

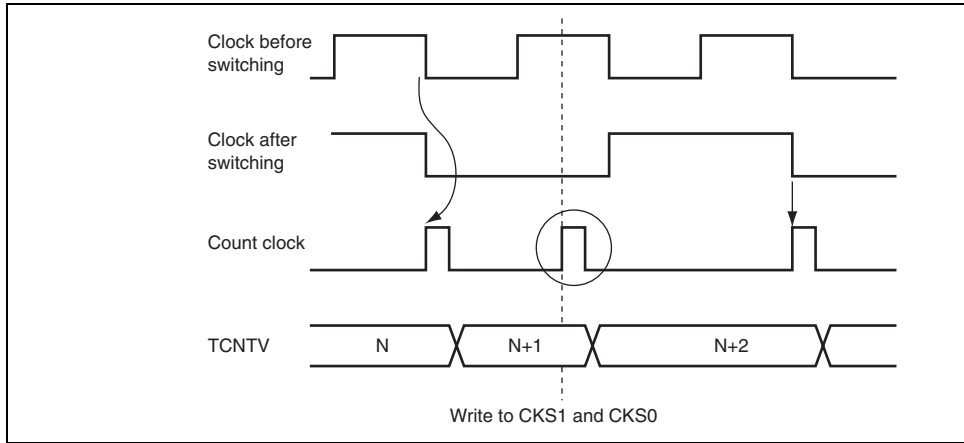


Figure 12.13 Internal Clock Switching and TCNTV Operation

- 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 the compare or input capture register
- Timer input/output functions
 - Waveform output by compare match:
 - Selection 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.

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)

Figure 13.1 Block Diagram of Timer W

compare B			input pin for GRB input capture, output pin in PWM mode
Input capture/output compare C	FTIOC	Input/output	Output pin for GRC output compare, input pin for GRC input capture, output pin in PWM mode
Input capture/output compare D	FTIOD	Input/output	Output pin for GRD output compare, input pin for GRD input capture, output pin in PWM mode

13.3 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)

5	BUFEB	0	R/W	Buffer Operation B Selects the GRD function. 0: GRD operates as an input capture/output compare register 1: GRD operates as the buffer register for GRD
4	BUFEA	0	R/W	Buffer Operation A Selects the GRC function. 0: GRC operates as an input capture/output compare register 1: GRC operates as the buffer register for GRC
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 mode) 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 mode) 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 mode) 1: PWM output

6	CKS2	0	R/W	Clock Select 2 to 0
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 event (FTCI) When the internal clock source (ϕ) is selected, sources are counted in subactive and subsleep
3	TOD	0	R/W	Timer Output Level Setting D Sets the output value of the FTIOD pin until the compare match D is generated. 0: Initial output value is 0* 1: Initial output value is 1*
2	TOC	0	R/W	Timer Output Level Setting C Sets the output value of the FTIOC pin until the compare match C is generated. 0: Initial output value is 0* 1: Initial output value is 1*
1	TOB	0	R/W	Timer Output Level Setting B Sets the output value of the FTIOB pin until the compare match B is generated. 0: Initial output value is 0* 1: Initial output value is 1*

13.3.3 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 request 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 request 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 request 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 request flag in TSRW is enabled.

6 to 4	—	All 1	—	Reserved	<ul style="list-style-type: none"> Read OVF when OVF=1, then write 0 in OVF. <p>These bits are always read as 1.</p>
3	IMFD	0	R/W	Input Capture/Compare Match Flag D	<p>[Setting conditions]</p> <ul style="list-style-type: none"> TCNT=GRD when GRD functions as an output compare register The TCNT value is transferred to GRD by a capture signal when GRD functions as an input capture register <p>[Clearing condition]</p> <ul style="list-style-type: none"> Read IMFD when IMFD=1, then write 0 in IMFD.
2	IMFC	0	R/W	Input Capture/Compare Match Flag C	<p>[Setting conditions]</p> <ul style="list-style-type: none"> TCNT=GRC when GRC functions as an output compare register The TCNT value is transferred to GRC by a capture signal when GRC functions as an input capture register <p>[Clearing condition]</p> <ul style="list-style-type: none"> Read IMFC when IMFC=1, then write 0 in IMFC.

0	IMFA	0	R/W	<ul style="list-style-type: none"> Read IMFB when IMFB=1, then write 0 in Input Capture/Compare Match Flag A [Setting conditions] <ul style="list-style-type: none"> TCNT=GRA when GRA functions as an output compare register The TCNT value is transferred to GRA by the capture signal when GRA functions as an input capture register [Clearing condition] <ul style="list-style-type: none"> Read IMFA when IMFA=1, then write 0 in
---	------	---	-----	---

13.3.5 Timer I/O Control Register 0 (TIOR0)

TIOR0 selects the functions of GRA and GRB, and specifies the functions of the FTIOA and FTIOB pins.

Bit	Bit Name	Initial Value	R/W	Description
7	—	1	—	Reserved This bit is always read as 1.
6	IOB2	0	R/W	I/O Control B2 Selects the GRB function. 0: GRB functions as an output compare register 1: GRB functions as an input capture register

00: Input capture at rising edge at the FTIOB pin
 01: Input capture at falling edge at the FTIOB pin
 1x: Input capture at rising edge and falling edge at the FTIOB 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
1	IOA1	0	R/W	I/O Control A1 and A0
0	IOA0	0	R/W	When IOA2 = 0, 00: No output at compare match 01: 0 output to the FTIOA pin at GRA compare match 10: 1 output to the FTIOA pin at GRA compare match 11: Output toggles to the FTIOA pin at GRA compare match When IOA2 = 1, 00: Input capture at rising edge of the FTIOA pin 01: Input capture at falling edge of the FTIOA pin 1x: Input capture at rising edge and falling edge of the FTIOA pin

[Legend] X: Don't care.

				0: GRD functions as an output compare register 1: GRD functions as an input capture register
5	IOD1	0	R/W	I/O Control D1 and D0
4	IOD0	0	R/W	When IOD2 = 0, 00: No output at compare match 01: 0 output to the FTIOD pin at GRD compare match 10: 1 output to the FTIOD pin at GRD compare match 11: Output toggles to the FTIOD pin at GRD compare match When IOD2 = 1, 00: Input capture at rising edge at the FTIOD pin 01: Input capture at falling edge at the FTIOD pin 1x: Input capture at rising edge and falling edge at the FTIOD 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

[Legend] X: Don't care.

13.3.7 Timer Counter (TCNT)

TCNT is a 16-bit readable/writable up-counter. The clock source is selected by bits CKS0 and CKS1 in TCRW. TCNT can be cleared to H'0000 through a compare match with GRA by setting the CCLR bit in TCRW to 1. When TCNT overflows (changes from H'FFFF to H'0000), the OFIF flag in TSRW is set to 1. If the OVIE bit in TIERW is set to 1 at this time, an interrupt request is generated. TCNT must always be read or written in 16-bit units; 8-bit access is not allowed. TCNT is initialized to H'0000.

When a general register is used as an input-capture register, an external input-capture signal is detected and the current TCNT value is stored in the general register. The corresponding interrupt-enable bit (IMFA, IMFB, IMFC, or IMFD) in TSRW is set to 1. If the corresponding interrupt-enable bit (IMIEA, IMIEB, IMIEC, or IMIED) in TSRW is set to 1 at this time, an interrupt request is 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 setting BUFEB and BUFEB in TMRW.

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

When GRA is set as an input-capture register and GRC is set as the buffer register for GRA, the value in TCNT is transferred to GRA and the value in GRA is transferred to the buffer register GRC 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 GRD is initialized to H'FFFF.

running counter. When the CST bit in TMRW is set to 1, TCNT starts incrementing the counter. 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 13.2 shows free-running counter operation.

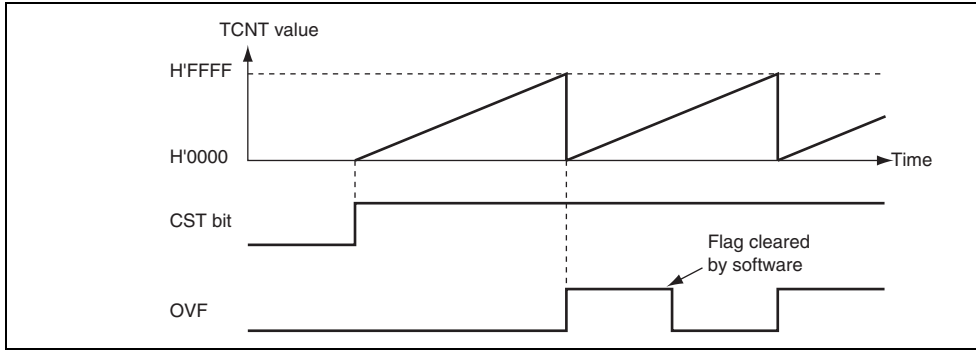


Figure 13.2 Free-Running Counter Operation

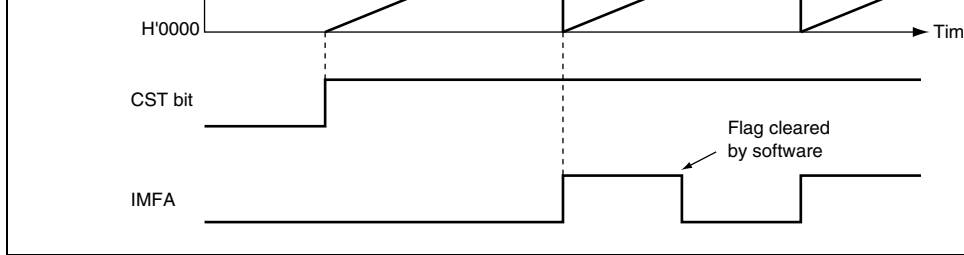


Figure 13.3 Periodic Counter Operation

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 toggle. Figure 13.4 shows an example of 0 and 1 output when TCNT operates as a free-running counter. Output 1 is selected for compare match A, and 0 output is selected for compare match B. When the output signal is already at the selected output level, the signal level does not change at compare

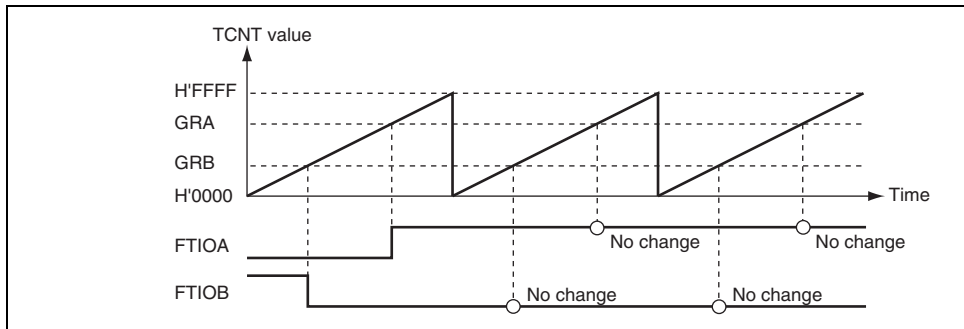


Figure 13.4 0 and 1 Output Example (TOA = 0, TOB = 1)

Figure 13.5 Toggle Output Example (TOA = 0, TOB = 1)

Figure 13.6 shows another example of toggle output when TCNT operates as a periodic counter cleared by compare match A. Toggle output is selected for both compare match A and B.

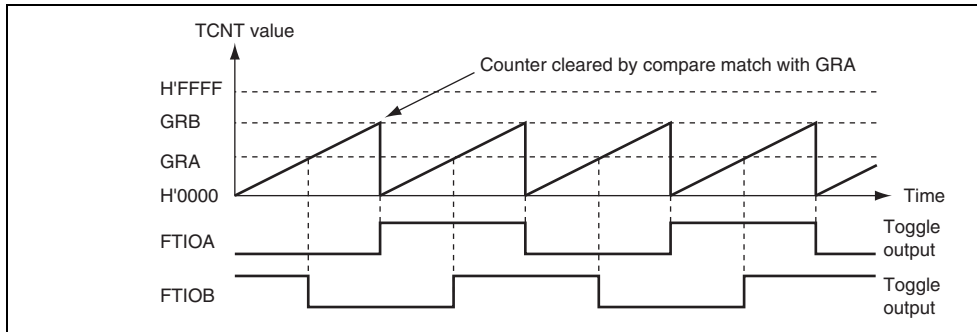


Figure 13.6 Toggle Output Example (TOA = 0, TOB = 1)

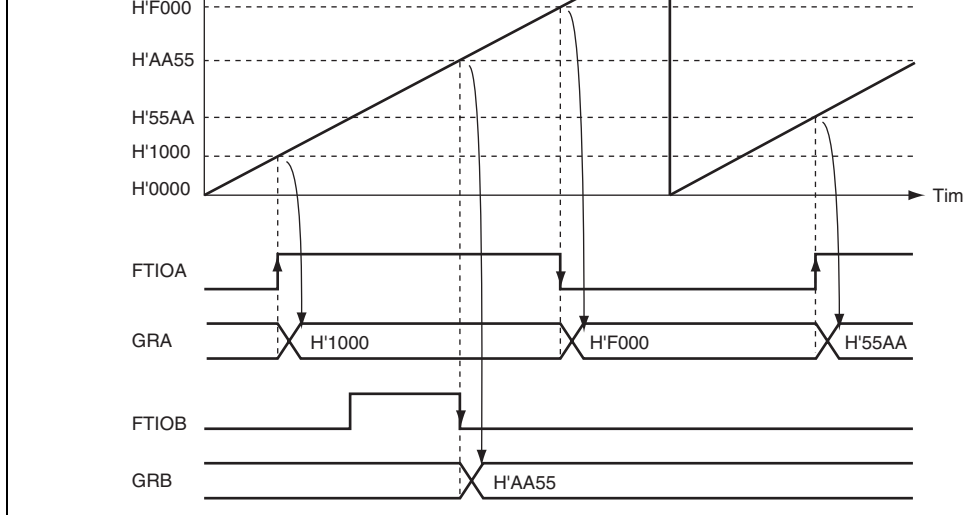


Figure 13.7 Input Capture Operating Example

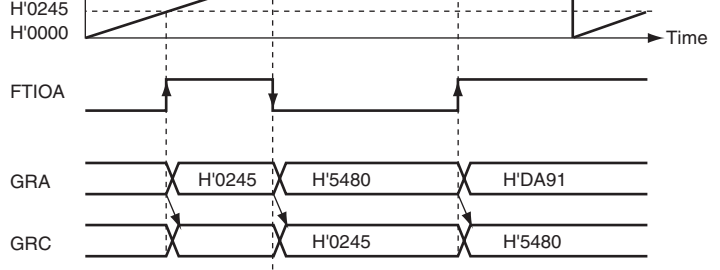


Figure 13.8 Buffer Operation Example (Input Capture)

same value is set in the cycle register and the duty register, the output does not change when a compare match occurs.

Figure 13.9 shows an example of operation in PWM mode. The output signals go to 1 at compare match A, and are cleared at compare match B, C, and D (TOB, TOC, and TOD = 1).

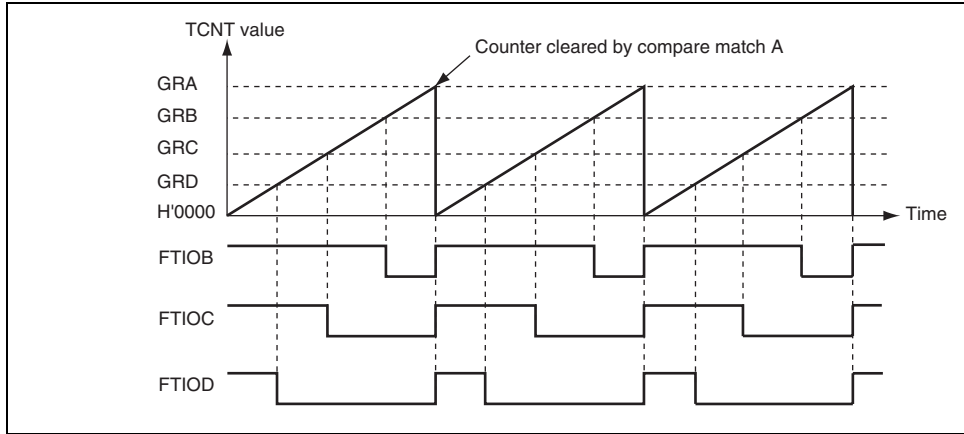


Figure 13.9 PWM Mode Example (1)

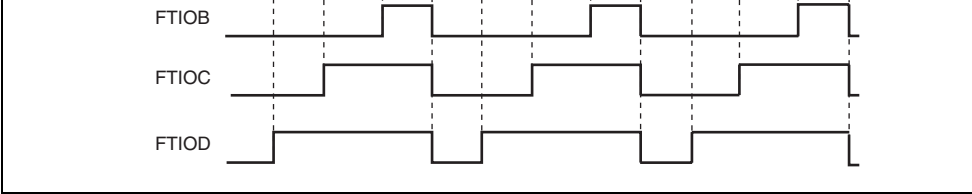


Figure 13.10 PWM Mode Example (2)

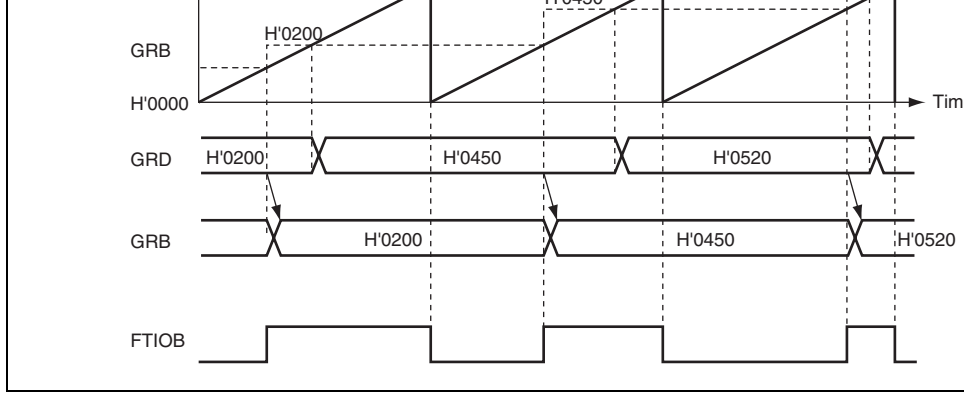


Figure 13.11 Buffer Operation Example (Output Compare)

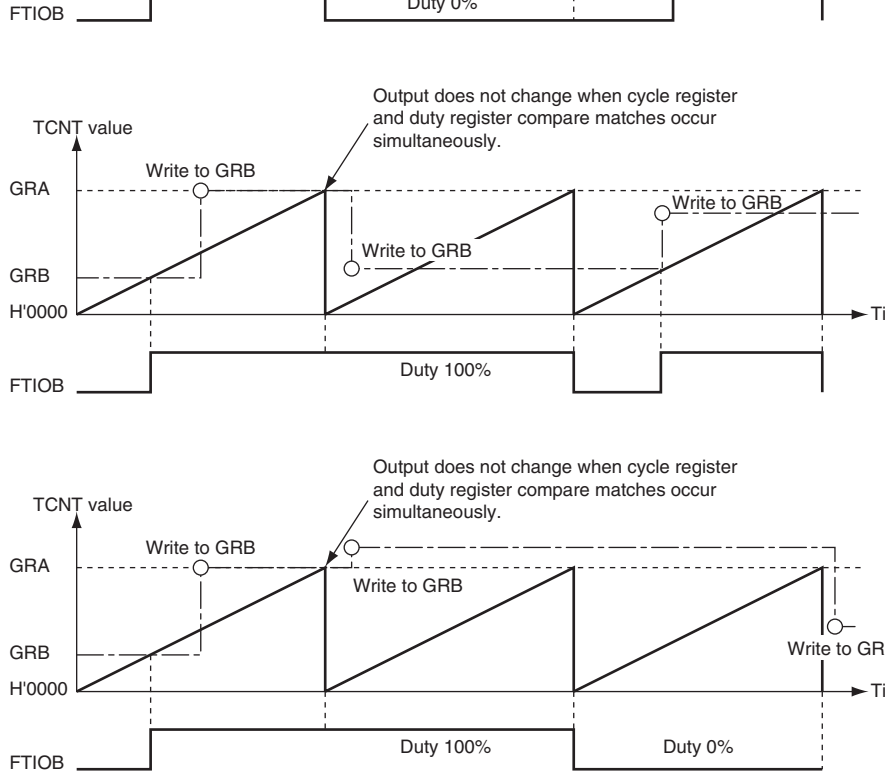


Figure 13.12 PWM Mode Example
(TOB = 0, TOC = 0, TOD = 0: Initial Output Values are Set to 0)

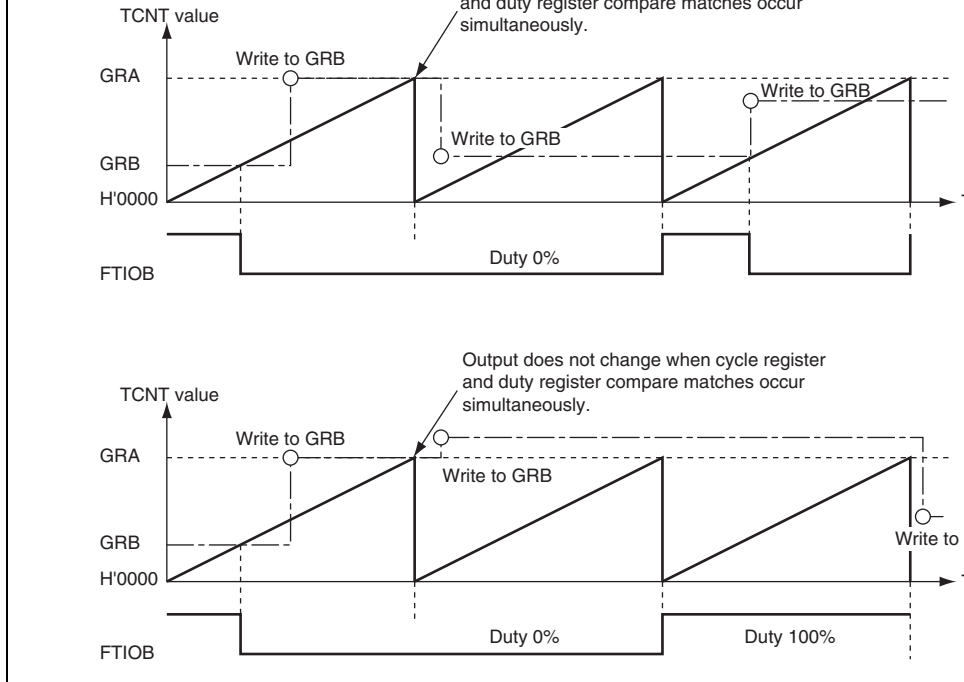


Figure 13.13 PWM Mode Example
(TOB = 1, TOC = 1, and TOD = 1: Initial Output Values are Set to 1)

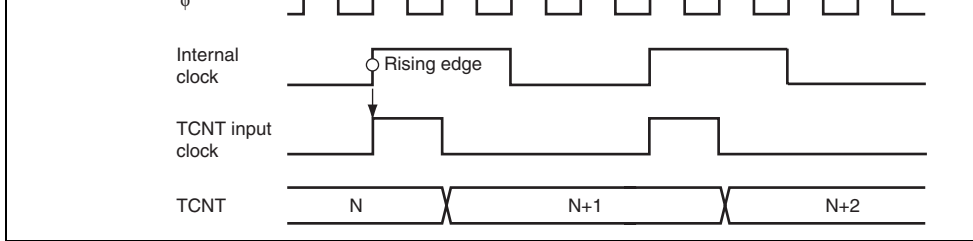


Figure 13.14 Count Timing for Internal Clock Source

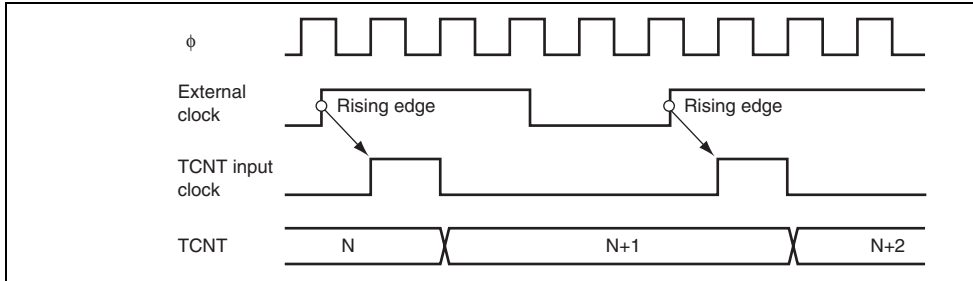


Figure 13.15 Count Timing for External Clock Source

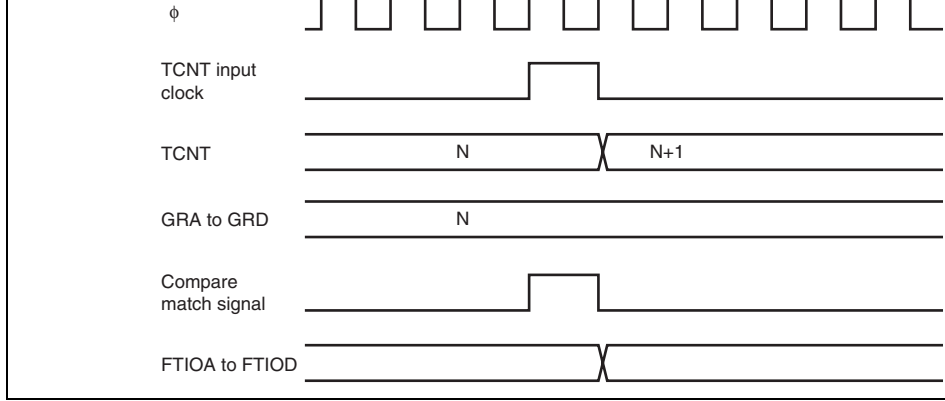


Figure 13.16 Output Compare Output Timing

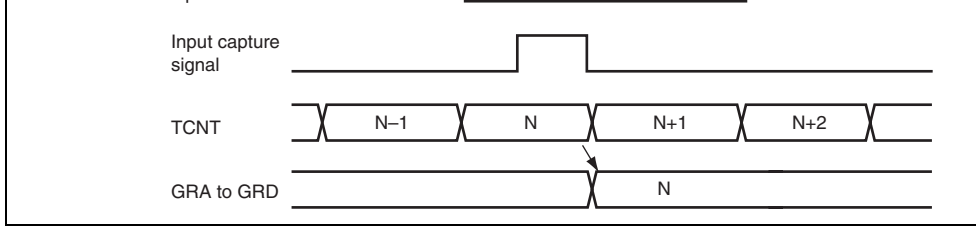


Figure 13.17 Input Capture Input Signal Timing

13.5.4 Timing of Counter Clearing by Compare Match

Figure 13.18 shows the timing when the counter is cleared by compare match A. When the value is N, the counter counts from 0 to N, and its cycle is N + 1.

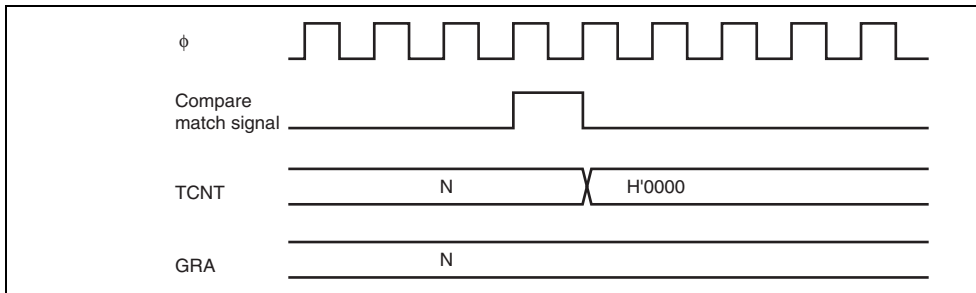


Figure 13.18 Timing of Counter Clearing by Compare Match

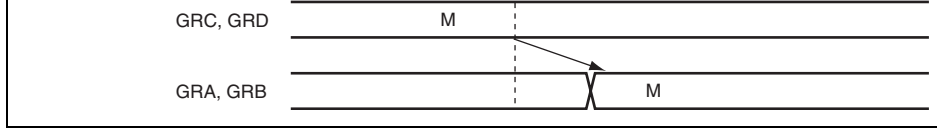


Figure 13.19 Buffer Operation Timing (Compare Match)

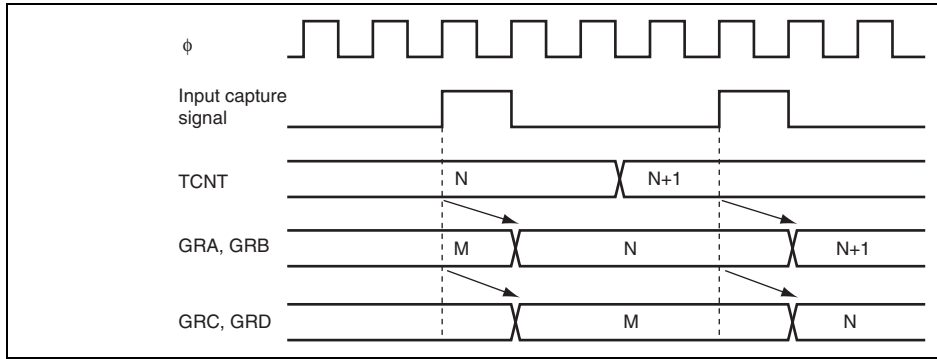


Figure 13.20 Buffer Operation Timing (Input Capture)

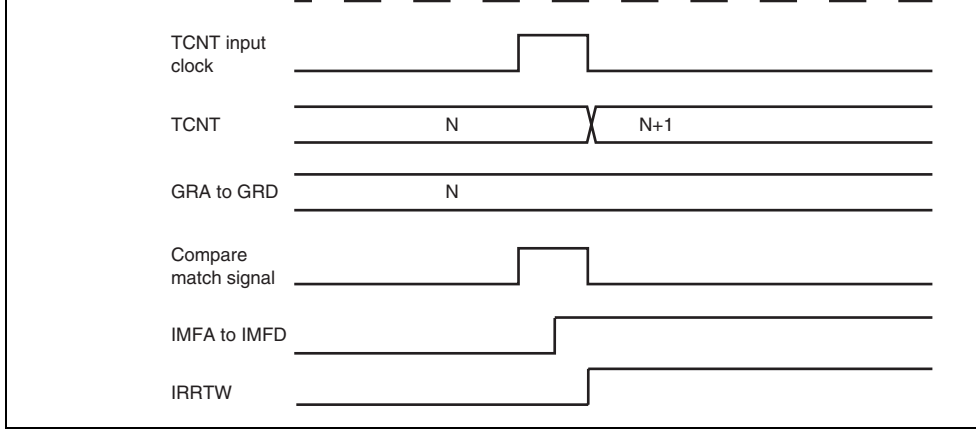


Figure 13.21 Timing of IMFA to IMFD Flag Setting at Compare Match

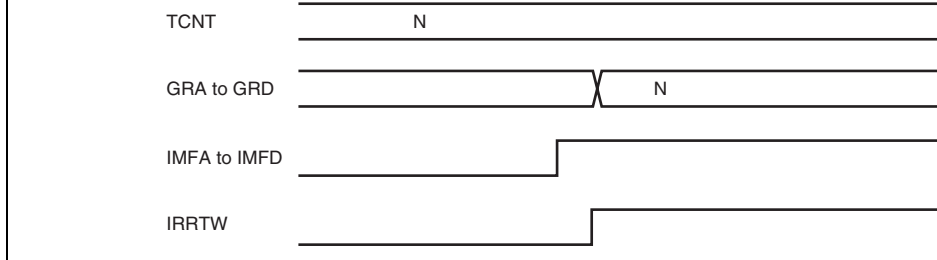


Figure 13.22 Timing of IMFA to IMFD Flag Setting at Input Capture

13.5.8 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 flag is cleared. Figure 13.23 shows the status flag clearing timing.

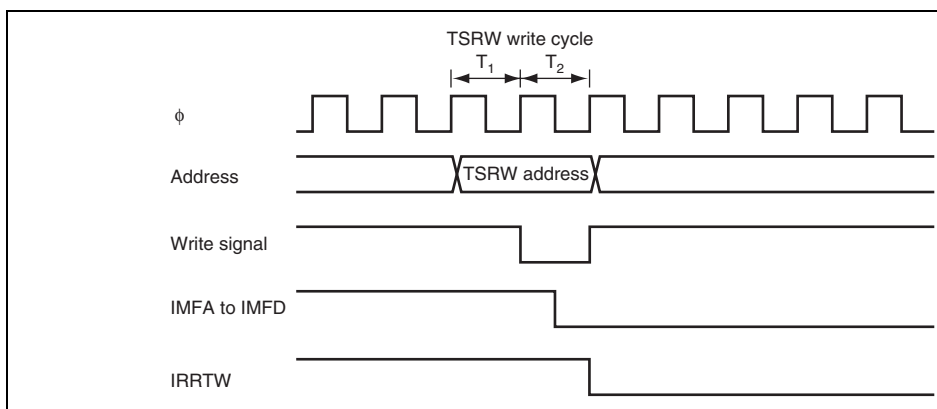


Figure 13.23 Timing of Status Flag Clearing by CPU

- precedence.
- 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 from the rising edge of an internal clock signal, that is divided system clock (ϕ). Therefore, as shown in figure 13.25 the switch is from a low clock signal to a high clock signal, the switch occurs as a rising edge, causing TCNT to increment.
 - If timer W enters module standby mode while an interrupt request is generated, the interrupt request cannot be cleared. Before entering module standby mode, disable interrupt requests.

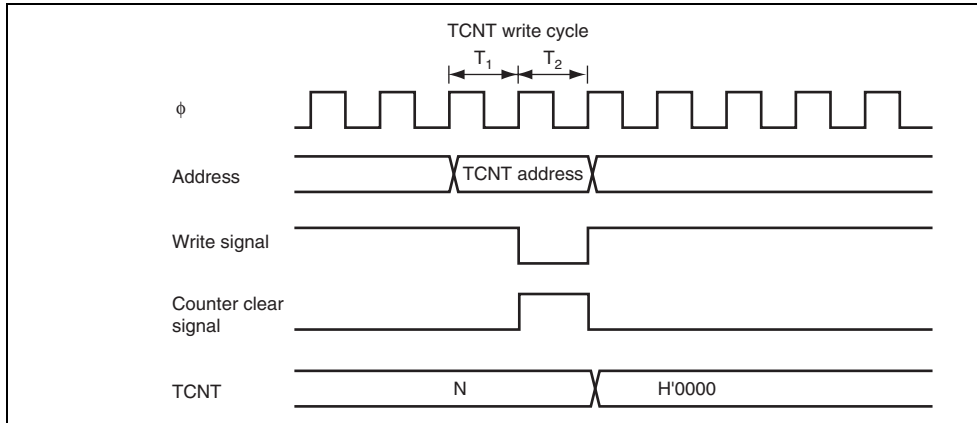


Figure 13.24 Contention between TCNT Write and Clear

bit manipulation instruction to TCRW occur at the same timing.

TOCR has been set to H'06. Compare match B and compare match C are used.

The FTIOB pin is in the 1 output state, and is set to the toggle output or the 0 output by compare match B. When BCLR#2, @TOCR is executed to clear the TOC bit (the FTIOC0 signal is low) and compare match B occurs at the same timing as shown below, the H'02 writing to TOCR has priority and compare match B does not drive the FTIOB signal low; the FTIOB signal remains high.

Bit	7	6	5	4	3	2	1	0
TCRW	CCLR	CKS2	CKS1	CKS0	TOD	TOC	TOB	TOA
Set value	0	0	0	0	0	1	1	0

BCLR#2, @TCRW

- (1) TCRW read operation: Read H'06
- (2) Modify operation: Modify H'06 to H'02
- (3) Write operation to TOCR: Write H'02

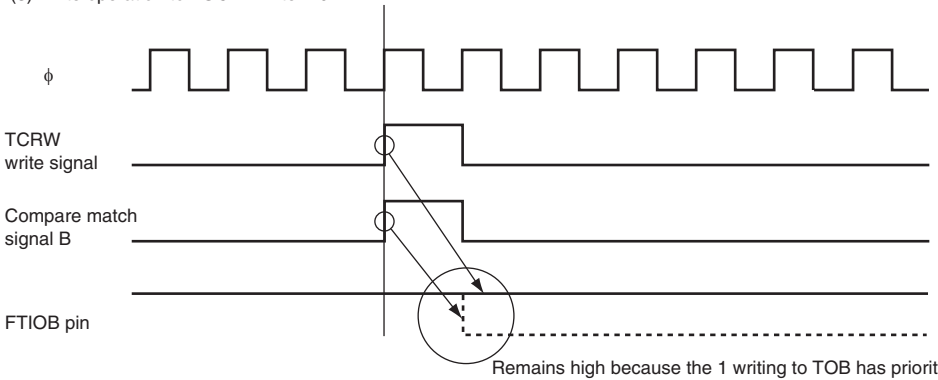
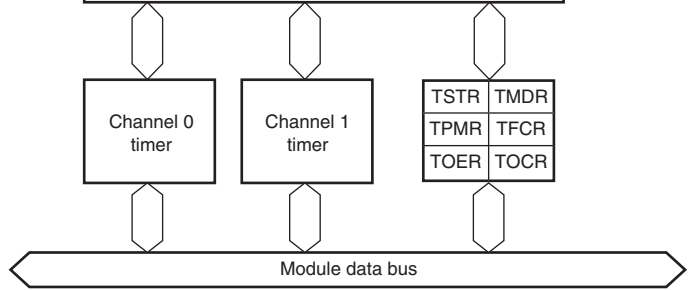


Figure 13.26 When Compare Match and Bit Manipulation Instruction to TCRW Occur at the Same Timing

- Independently assignable output compare or input capture functions
- Selection of five counter clock sources: four internal clocks (ϕ , $\phi/2$, $\phi/4$, and $\phi/8$) and external clock
- Seven selectable operating modes
 - Output compare function
 - Selection of 0 output, 1 output, or toggle output
 - Input capture function
 - Rising edge, falling edge, or both edges
 - Synchronous operation
 - Timer counters_0 and _1 (TCNT_0 and TCNT_1) can be written simultaneously
 - Simultaneous clearing by compare match or input capture is possible.
 - PWM mode
 - Up to six-phase PWM output can be provided with desired duty ratio.
 - Reset synchronous PWM mode
 - Three-phase PWM output for normal and counter phases
 - Complementary PWM mode
 - Three-phase PWM output for non-overlapped normal and counter phases
 - The A/D conversion start trigger can be set for PWM cycles.
 - Buffer operation
 - The input capture register can be consisted of double buffers.
 - The output compare register can automatically be modified.
- High-speed access by the internal 16-bit bus
 - 16-bit TCNT and GR registers can be accessed in high speed by a 16-bit bus interne
- Any initial timer output value can be set
- Output of the timer is disabled by external trigger

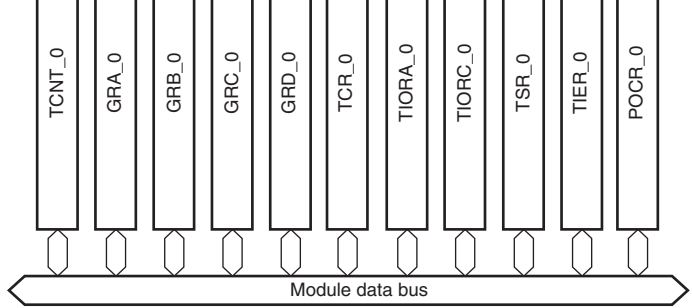
Buffer register	GRC_0, GRD_0		GRC_1, GRD_1
I/O pins	FTIOA0, FTIOB0, FTIOC0, FTIOD0		FTIOA1, FTIOB1, FTIOC1, FTIOD1
Counter clearing function	Compare match/input capture of GRA_0, GRB_0, GRC_0, or GRD_0		Compare match/input capture of GRA_1, GRB_1, GRC_1, or GRD_1
Compare match output	0 output	Yes	Yes
	1 output	Yes	Yes
	output	Yes	Yes
Input capture function	Yes		Yes
Synchronous operation	Yes		Yes
PWM mode	Yes		Yes
Reset synchronous PWM mode	Yes		Yes
Complementary PWM mode	Yes		Yes
Buffer function	Yes		Yes
Interrupt sources	Compare match/input capture A0 to D0 Overflow		Compare match/input capture to D1 Overflow Underflow



[Legend]

- TSTR: Timer start register (8 bits)
- TMDR: Timer mode register (8 bits)
- TPMR: Timer PWM mode register (8 bits)
- TFCR: Timer function control register (8 bits)
- TOER: Timer output master enable register (8 bits)
- TOCR: Timer output control register (8 bits)
- ADTRG: A/D conversion start trigger output signal
- ITMZ0: Channel 0 interrupt
- ITMZ1: Channel 1 interrupt

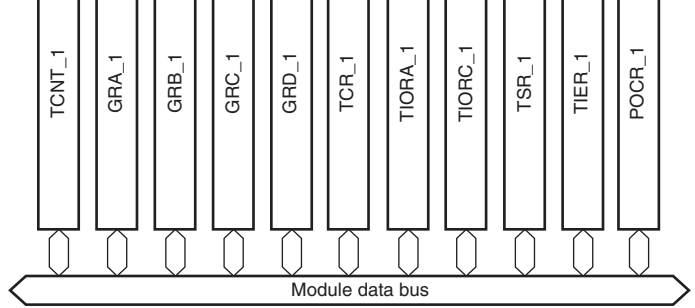
Figure 14.1 Timer Z Block Diagram



[Legend]

TCNT_0	Timer counter_0 (16 bits)
GRA_0, GRB_0:	General registers A_0, B_0, C_0, and D_0 (input capture/output compare registers: 16 bits × 4)
GRC_0, GRD_0	
TCR_0:	Timer control register_0 (8 bits)
TIORA_0:	Timer I/O control register A_0 (8 bits)
TIORC_0:	Timer I/O control register C_0 (8 bits)
TSR_0:	Timer status register_0 (8 bits)
TIER_0:	Timer interrupt enable register_0 (8 bits)
POCCR_0:	PWM mode output level control register_0 (8 bits)
ITMZ0:	Channel 0 interrupt

Figure 14.2 Timer Z (Channel 0) Block Diagram



[Legend]

- TCNT_1: Timer counter_1 (16 bits)
- GRA_1, GRB_1: General registers A_1, B_1, C_1, and D_1 (input capture/output compare registers: 16 bits × 4)
- GRC_1, GRD_1: 16 bits × 4
- TCR_1: Timer control register_1 (8 bits)
- TIOA_1: Timer I/O control register A_1 (8 bits)
- TIORC_1: Timer I/O control register C_1 (8 bits)
- TSR_1: Timer status register_1 (8 bits)
- TIER_1: Timer interrupt enable register_1 (8 bits)
- POOCR_1: PWM mode output level control register_1 (8 bits)
- ITMZ1: Channel 1 interrupt

Figure 14.3 Timer Z (Channel 1) Block Diagram

compare B0			input capture input, or PWM ou
Input capture/output compare C0	FTIOC0	Input/output	GRC_0 output compare output, input capture input, or PWM synchronous output (in reset synchronous PWM and comple PWM modes)
Input capture/output compare D0	FTIOD0	Input/output	GRD_0 output compare output, input capture input, or PWM ou
Input capture/output compare A1	FTIOA1	Input/output	GRA_1 output compare output, input capture input, or PWM ou reset synchronous PWM and complementary PWM modes)
Input capture/output compare B1	FTIOB1	Input/output	GRB_1 output compare output, input capture input, or PWM ou
Input capture/output compare C1	FTIOC1	Input/output	GRC_1 output compare output, input capture input, or PWM ou
Input capture/output compare D1	FTIOD1	Input/output	GRD_1 output compare output, input capture input, or PWM ou

- Timer output control register (TOCR)

Channel 0

- Timer control register_0 (TCR_0)
- Timer I/O control register A_0 (TIORA_0)
- Timer I/O control register C_0 (TIORC_0)
- Timer status register_0 (TSR_0)
- Timer interrupt enable register_0 (TIER_0)
- PWM mode output level control register_0 (POCR_0)
- Timer counter_0 (TCNT_0)
- General register A_0 (GRA_0)
- General register B_0 (GRB_0)
- General register C_0 (GRC_0)
- General register D_0 (GRD_0)

Channel 1

- Timer control register_1 (TCR_1)
- Timer I/O control register A_1 (TIORA_1)
- Timer I/O control register C_1 (TIORC_1)
- Timer status register_1 (TSR_1)
- Timer interrupt enable register_1 (TIER_1)
- PWM mode output level control register_1 (POCR_1)
- Timer counter_1 (TCNT_1)
- General register A_1 (GRA_1)
- General register B_1 (GRB_1)

modified.

1	STR1	0	R/W	Channel 1 Counter Start 0: TCNT_1 halts counting 1: TCNT_1 starts counting
0	STR0	0	R/W	Channel 0 Counter Start 0: TCNT_0 halts counting 1: TCNT_0 starts counting

				0: GRC_1 operates normally 1: GRA_1 and GRD_1 are used together for k operation
5	BFD0	0	R/W	Buffer Operation D0 0: GRD_0 operates normally 1: GRB_0 and GRD_0 are used together for k operation
4	BFC0	0	R/W	Buffer Operation C0 0: GRC_0 operates normally 1: GRA_0 and GRC_0 are used together for k operation
3 to 1	—	All 1	—	Reserved These bits are always read as 1, and cannot modified.
0	SYNC	0	R/W	Timer Synchronization 0: TCNT_1 and TCNT_0 operate as a differen 1: TCNT_1 and TCNT_0 are synchronized TCNT_1 and TCNT_0 can be pre-set or clear synchronously

				1: FTIOD1 operates in PWM mode
5	PWMC1	0	R/W	PWM Mode C1 0: FTIOC1 operates normally 1: FTIOC1 operates in PWM mode
4	PWMB1	0	R/W	PWM Mode B1 0: FTIOB1 operates normally 1: FTIOB1 operates in PWM mode
3	—	1	—	Reserved This bit is always read as 1, and cannot be mo
2	PWMD0	0	R/W	PWM Mode D0 0: FTIOD0 operates normally 1: FTIOD0 operates in PWM mode
1	PWMC0	0	R/W	PWM Mode C0 0: FTIOC0 operates normally 1: FTIOC0 operates in PWM mode
0	PWMB0	0	R/W	PWM Mode B0 0: FTIOB0 operates normally 1: FTIOB0 operates in PWM mode

				1: External clock input is enabled
5	ADEG	0	R/W	<p>A/D Trigger Edge Select</p> <p>A/D module should be set to start an A/D conversion at the external trigger</p> <p>0: A/D trigger at the crest in complementary PWM mode</p> <p>1: A/D trigger at the trough in complementary PWM mode</p>
4	ADTRG	0	R/W	<p>External Trigger Disable</p> <p>0: A/D trigger for PWM cycles is disabled in complementary PWM mode</p> <p>1: A/D trigger for PWM cycles is enabled in complementary PWM mode</p>
3	OLS1	0	R/W	<p>Output Level Select 1</p> <p>Selects the counter-phase output levels in complementary PWM mode or complementary PWM mode.</p> <p>0: Initial output is high and the active level is low</p> <p>1: Initial output is low and the active level is high</p>
2	OLS0	0	R/W	<p>Output Level Select 0</p> <p>Selects the normal-phase output levels in complementary PWM mode or complementary PWM mode.</p> <p>0: Initial output is high and the active level is low</p> <p>1: Initial output is low and the active level is high</p> <p>Figure 14.4 shows an example of outputs in complementary PWM mode and complementary PWM mode when OLS1 = 0 and OLS0 = 0.</p>

at the crest)

Note: When reset synchronous PWM mode or complementary PWM mode is selected by bits, this setting has the priority to the selected PWM mode by each bit in TPMR. Stop TCNT_0 and TCNT_1 before making settings for reset synchronous PWM mode or complementary PWM mode.

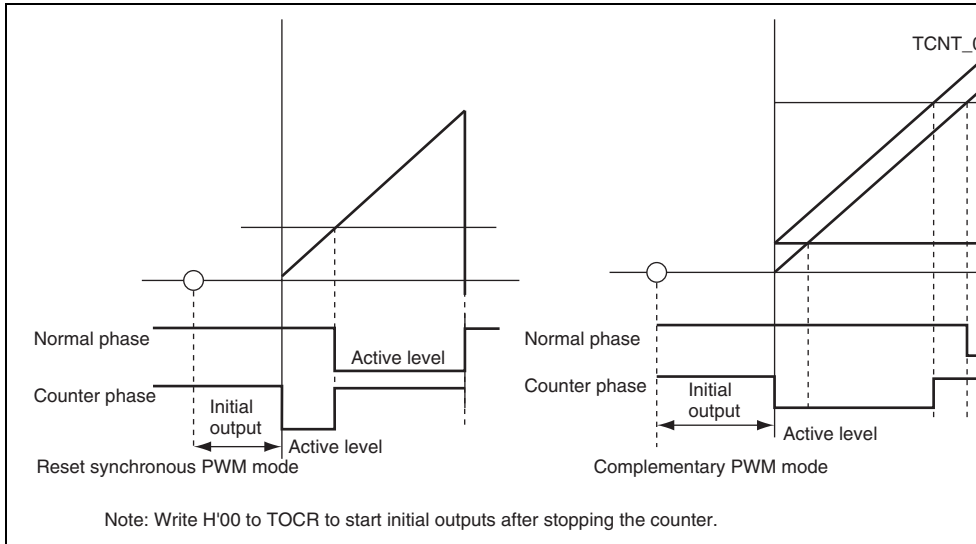


Figure 14.4 Example of Outputs in Reset Synchronous PWM Mode and Complementary PWM Mode

				1: FTIOD1 pin output is disabled regardless of TPMPR, TFCR, and TIORC_1 settings (FTIOD1 operated as an I/O port).
6	EC1	1	R/W	<p>Master Enable C1</p> <p>0: FTIOC1 pin output is enabled according to TPMPR, TFCR, and TIORC_1 settings</p> <p>1: FTIOC1 pin output is disabled regardless of TPMPR, TFCR, and TIORC_1 settings (FTIOC1 operated as an I/O port).</p>
5	EB1	1	R/W	<p>Master Enable B1</p> <p>0: FTIOB1 pin output is enabled according to TPMPR, TFCR, and TIORA_1 settings</p> <p>1: FTIOB1 pin output is disabled regardless of TPMPR, TFCR, and TIORA_1 settings (FTIOB1 operated as an I/O port).</p>
4	EA1	1	R/W	<p>Master Enable A1</p> <p>0: FTIOA1 pin output is enabled according to TPMPR, TFCR, and TIORA_1 settings</p> <p>1: FTIOA1 pin output is disabled regardless of TPMPR, TFCR, and TIORA_1 settings (FTIOA1 operated as an I/O port).</p>
3	ED0	1	R/W	<p>Master Enable D0</p> <p>0: FTIOD0 pin output is enabled according to TPMPR, TFCR, and TIORC_0 settings</p> <p>1: FTIOD0 pin output is disabled regardless of TPMPR, TFCR, and TIORC_0 settings (FTIOD0 operated as an I/O port).</p>

1: FTIOB0 pin output is disabled regardless of TPMR, TFCR, and TIOA_0 settings (FTIO operated as an I/O port).

0	EA0	1	R/W	Master Enable A0
---	-----	---	-----	------------------

0: FTIOA0 pin output is enabled according to the TPMR, TFCR, and TIOA_0 settings

1: FTIOA0 pin output is disabled regardless of the TPMR, TFCR, and TIOA_0 settings (FTIO operated as an I/O port).

6	TOC1	0	R/W	Output Level Select C1 0: 0 output at the FTIOC1 pin* 1: 1 output at the FTIOC1 pin*
5	TOB1	0	R/W	Output Level Select B1 0: 0 output at the FTIOB1 pin* 1: 1 output at the FTIOB1 pin*
4	TOA1	0	R/W	Output Level Select A1 0: 0 output at the FTIOA1 pin* 1: 1 output at the FTIOA1 pin*
3	TOD0	0	R/W	Output Level Select D0 0: 0 output at the FTIOD0 pin* 1: 1 output at the FTIOD0 pin*
2	TOC0	0	R/W	Output Level Select C0 0: 0 output at the FTIOC0 pin* 1: 1 output at the FTIOC0 pin*
1	TOB0	0	R/W	Output Level Select B0 0: 0 output at the FTIOB0 pin* 1: 1 output at the FTIOB0 pin*
0	TOA0	0	R/W	Output Level Select A0 0: 0 output at the FTIOA0 pin* 1: 1 output at the FTIOA0 pin*

Note: * The change of the setting is immediately reflected in the output value.

bit units; they must always be accessed as a 16-bit unit. TCNT is initialized to H'0000.

14.3.8 General Registers A, B, C, and D (GRA, GRB, GRC, and GRD)

GR are 16-bit registers. Timer Z has eight general registers (GR), four for each channel. 7 registers are dual function 16-bit readable/writable registers, functioning as either output or input capture registers. Functions can be switched by TIORA and TIORC.

The values in GR and TCNT are constantly compared with each other when the GR registers are used as output compare registers. When the both values match, the IMFA to IMFD flags are set to 1. Compare match outputs can be selected by TIORA and TIORC.

When the GR registers are used as input capture registers, the TCNT value is stored after external signals. At this point, IMFA to IMFD flags in the corresponding TSR are set to 1. Detection edges for input capture signals can be selected by TIORA and TIORC.

When PWM mode, complementary PWM mode, or reset synchronous PWM mode is selected, values in TIORA and TIORC are ignored. Upon reset, the GR registers are set as output compare registers (no output) and initialized to H'FFFF. The GR registers cannot be accessed in 8-bit mode; they must always be accessed as a 16-bit unit.

capture
 010: Clears TCNT by GRB compare match/input capture*¹
 011: Synchronization clear; Clears TCNT in syn with counter clearing of the other channel
 100: Disables TCNT clearing
 101: Clears TCNT by GRC compare match/input capture*¹
 110: Clears TCNT by GRD compare match/input capture*¹
 111: Synchronization clear; Clears TCNT in syn with counter clearing of the other channel

4	CKEG1	0	R/W	Clock Edge 1 and 0
3	CKEG0	0	R/W	00: Count at rising edge 01: Count at falling edge 1X: Count at both edges
2	TPSC2	0	R/W	Time Prescaler 2 to 0
1	TPSC1	0	R/W	000: Internal clock: count by ϕ
0	TPSC0	0	R/W	001: Internal clock: count by $\phi/2$ 010: Internal clock: count by $\phi/4$ 011: Internal clock: count by $\phi/8$ 1XX: External clock: count by FTIOA0 (TCLK)

- Notes: 1. When GR functions as an output compare register, TCNT is cleared by compare match.
 When GR functions as input capture, TCNT is cleared by input capture.
 2. Synchronous operation is set by TMDR.
 3. X: Don't care

Bit	Bit Name	Initial Value	R/W	Description
7	—	1	—	Reserved This bit is always read as 1.
6	IOB2	0	R/W	I/O Control B2 to B0
5	IOB1	0	R/W	GRB is an output compare register:
4	IOB0	0	R/W	000: Disables pin output by compare match 001: 0 output by GRB compare match 010: 1 output by GRB compare match 011: Toggle output by GRB compare match GRB is an input capture register: 100: Input capture to GRB at the rising edge 101: Input capture to GRB at the falling edge 11X: Input capture to GRB at both rising and falling edges
3	—	1	—	Reserved This bit is always read as 1.
2	IOA2	0	R/W	I/O Control A2 to A0
1	IOA1	0	R/W	GRA is an output compare register:
0	IOA0	0	R/W	000: Disables pin output by compare match 001: 0 output by GRA compare match 010: 1 output by GRA compare match 011: Toggle output by GRA compare match GRA is an input capture register: 100: Input capture to GRA at the rising edge 101: Input capture to GRA at the falling edge 11X: Input capture to GRA at both rising and falling edges

[Legend] X: Don't care

4	IOD0	0	R/W	<p>000: Disables pin output by compare match</p> <p>001: 0 output by GRD compare match</p> <p>010: 1 output by GRD compare match</p> <p>011: Toggle output by GRD compare match</p> <p>GRD is an input capture register:</p> <p>100: Input capture to GRD at the rising edge</p> <p>101: Input capture to GRD at the falling edge</p> <p>11X: Input capture to GRD at both rising and edges</p>
3	—	1	—	<p>Reserved</p> <p>This bit is always read as 1.</p>
2	IOC2	0	R/W	I/O Control C2 to C0
1	IOC1	0	R/W	GRC is an output compare register:
0	IOC0	0	R/W	<p>000: Disables pin output by compare match</p> <p>001: 0 output by GRC compare match</p> <p>010: 1 output by GRC compare match</p> <p>011: Toggle Output by GRC compare match</p> <p>GRC is an input capture register:</p> <p>100: Input capture to GRC at the rising edge</p> <p>101: Input capture to GRC at the falling edge</p> <p>11X: Input capture to GRC at both rising and edges</p>

[Legend] X: Don't care

5	UDF*	0	R/W	Underflow Flag [Setting condition] <ul style="list-style-type: none"> When TCNT_1 underflows [Clearing condition] <ul style="list-style-type: none"> When 0 is written to UDF after reading UDF
4	OVF	0	R/W	Overflow Flag [Setting condition] <ul style="list-style-type: none"> When the TCNT value underflows [Clearing condition] <ul style="list-style-type: none"> When 0 is written to OVF after reading OVF
3	IMFD	0	R/W	Input Capture/Compare Match Flag D [Setting conditions] <ul style="list-style-type: none"> When TCNT = GRD and GRD is functioning as input capture register When TCNT value is transferred to GRD by input capture signal and GRD is functioning as input capture register [Clearing condition] <ul style="list-style-type: none"> When 0 is written to IMFD after reading IMFD

1	IMFB	0	R/W	<ul style="list-style-type: none"> When 0 is written to IMFC after reading IMFC Input Capture/Compare Match Flag B [Setting conditions] <ul style="list-style-type: none"> When TCNT = GRB and GRB is functioning as output compare register When TCNT value is transferred to GRB by capture signal and GRB is functioning as capture register [Clearing condition] <ul style="list-style-type: none"> When 0 is written to IMFB after reading IMFB
0	IMFA	0	R/W	Input Capture/Compare Match Flag A [Setting conditions] <ul style="list-style-type: none"> When TCNT = GRA and GRA is functioning as output compare register When TCNT value is transferred to GRA by capture signal and GRA is functioning as capture register [Clearing condition] <ul style="list-style-type: none"> When 0 is written to IMFA after reading IMFA

Note: Bit 5 is not the UDF flag in TSR_0. It is a reserved bit. It is always read as 1.

				0: Interrupt requests (OVI) by OVF or UDF flag disabled 1: Interrupt requests (OVI) by OVF or UDF flag enabled
3	IMIED	0	R/W	Input Capture/Compare Match Interrupt Enable 0: Interrupt requests (IMID) by IMFD flag are disabled 1: Interrupt requests (IMID) by IMFD flag are enabled
2	IMIEC	0	R/W	Input Capture/Compare Match Interrupt Enable 0: Interrupt requests (IMIC) by IMFC flag are disabled 1: Interrupt requests (IMIC) by IMFC flag are enabled
1	IMIEB	0	R/W	Input Capture/Compare Match Interrupt Enable 0: Interrupt requests (IMIB) by IMFB flag are disabled 1: Interrupt requests (IMIB) by IMFB flag are enabled
0	IMIEA	0	R/W	Input Capture/Compare Match Interrupt Enable 0: Interrupt requests (IMIA) by IMFA flag are disabled 1: Interrupt requests (IMIA) by IMFA flag are enabled

				0: The output level of FTIOD is low-active 1: The output level of FTIOD is high-active
1	POLC	0	R/W	PWM Mode Output Level Control C 0: The output level of FTIOC is low-active 1: The output level of FTIOC is high-active
0	POLB	0	R/W	PWM Mode Output Level Control B 0: The output level of FTIOB is low-active 1: The output level of FTIOB is high-active

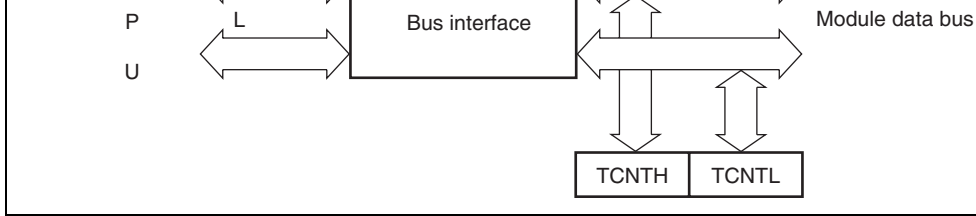


Figure 14.5 Accessing Operation of 16-Bit Register (between CPU and TCNT (16-bit))

2. 8-bit register

Registers other than TCNT and GR are 8-bit registers that are connected internally with CPU in an 8-bit width. Figure 14.6 shows an example of accessing the 8-bit registers.

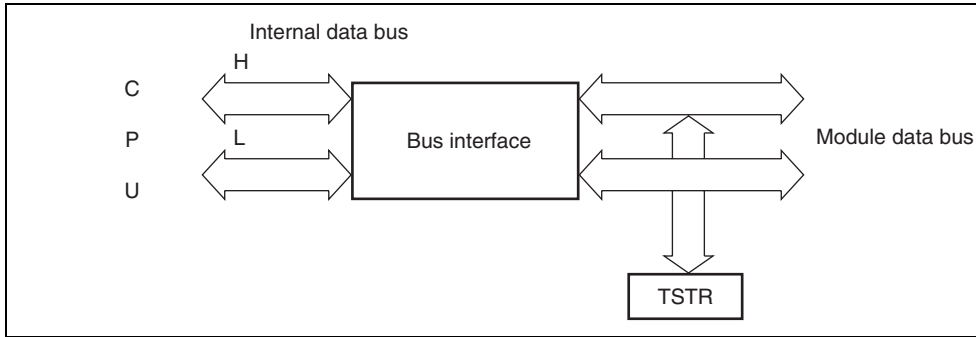
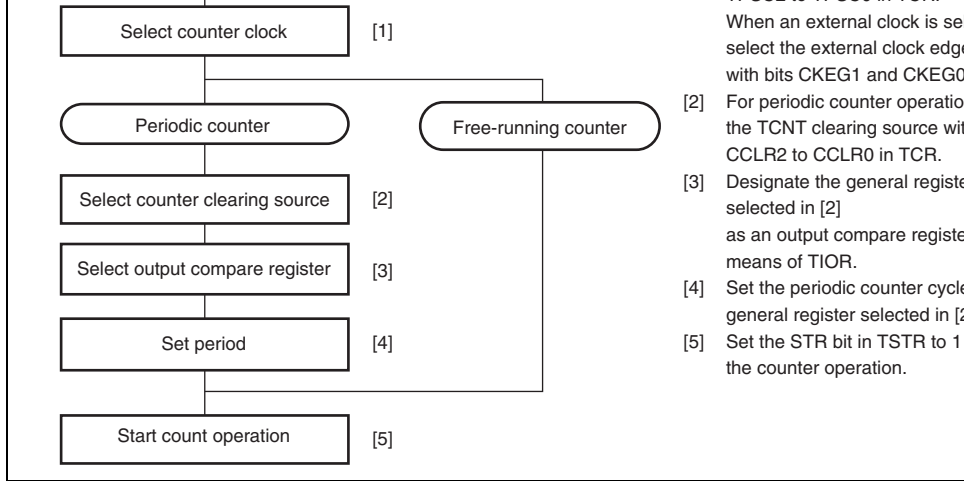


Figure 14.6 Accessing Operation of 8-Bit Register (between CPU and TSTR (8-bit))



- [1] When an external clock is selected, select the external clock edge with bits CKEG1 and CKEG0.
- [2] For periodic counter operation, the TCNT clearing source with CCLR2 to CCLR0 in TCR.
- [3] Designate the general register selected in [2] as an output compare register means of TIOR.
- [4] Set the periodic counter cycle general register selected in [2].
- [5] Set the STR bit in TSTR to 1 to start the counter operation.

Figure 14.7 Example of Counter Operation Setting Procedure

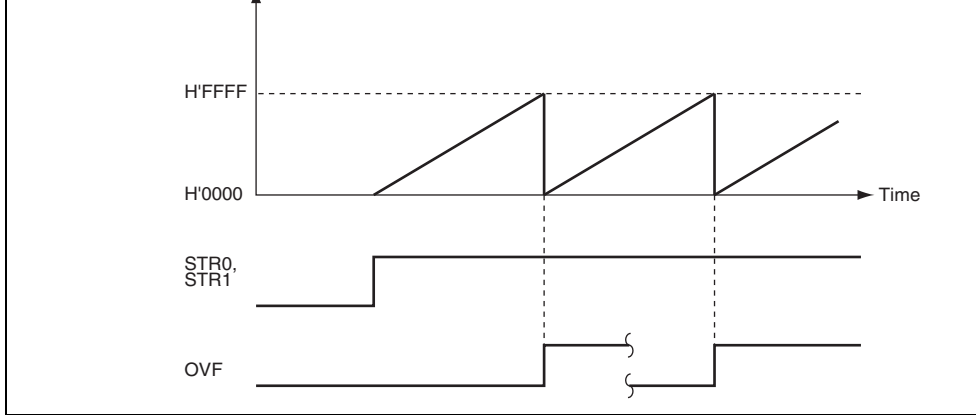


Figure 14.8 Free-Running Counter Operation

When compare match is selected as the TCNT clearing source, the TCNT counter for the channel performs periodic count operation. The GR registers for setting the period are designated as output compare registers, and counter clearing by compare match is selected by means of CCLR1 and CCLR0 in TCR. After the settings have been made, TCNT starts an increment operation as a periodic counter when the corresponding bit in TSTR is set to 1. When the counter value matches the value in GR, the IMFA, IMFB, IMFC, or IMFD flag in TSR is set to 1 and TCNT is cleared to H'0000.

If the value of the corresponding IMIEA, IMIEB, IMIEC, or IMIED bit in TIER is 1 at the time the timer Z requests an interrupt. After a compare match, TCNT starts an increment operation again from H'0000.

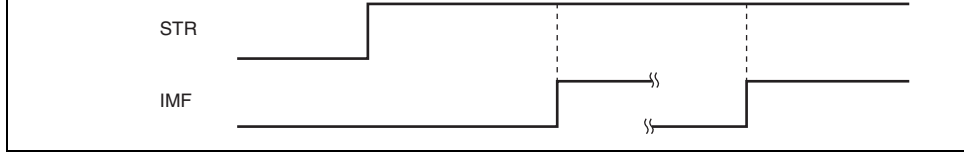


Figure 14.9 Periodic Counter Operation

2. TCNT count timing

A. Internal clock operation

A system clock (ϕ) or three types of clocks ($\phi/2$, $\phi/4$, or $\phi/8$) that divides the system clock can be selected by bits TPSC2 to TPSC0 in TCR.

Figure 14.10 illustrates this timing.

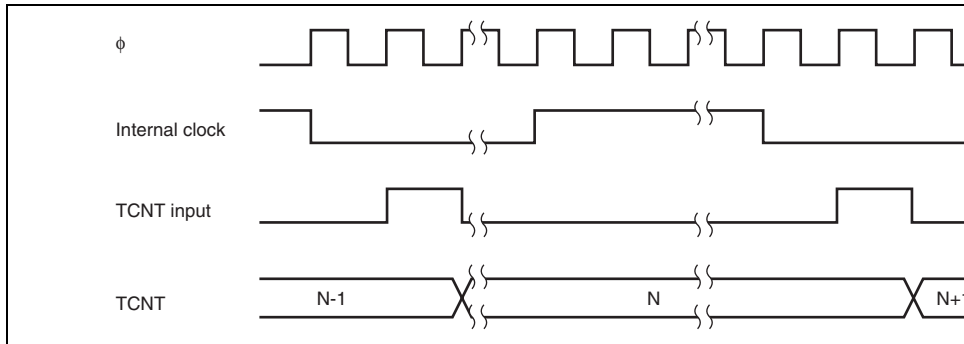


Figure 14.10 Count Timing at Internal Clock Operation

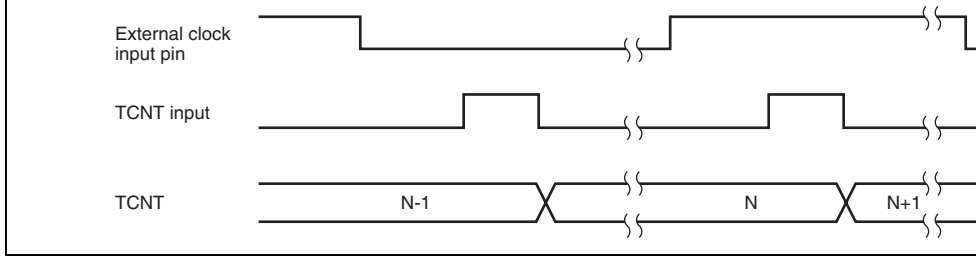


Figure 14.11 Count Timing at External Clock Operation (Both Edges Detect)

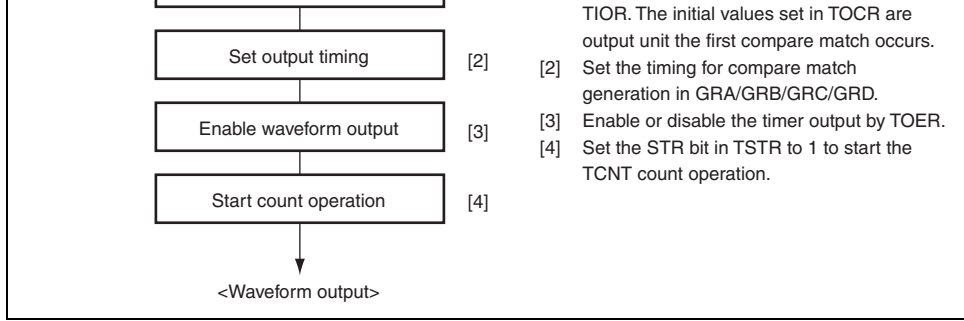


Figure 14.12 Example of Setting Procedure for Waveform Output by Compare

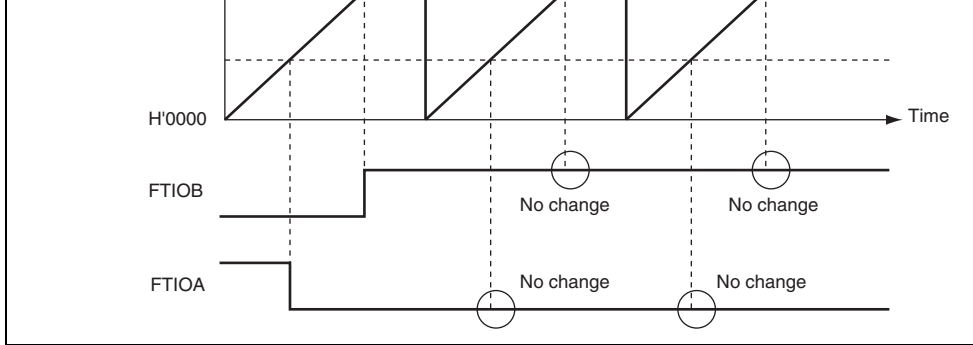


Figure 14.13 Example of 0 Output/1 Output Operation

Figure 14.14 shows an example of toggle output.

In this example, TCNT has been designated as a periodic counter (with counter clearing on compare match B), and settings have been made such that the output is toggled by both compare match A and compare match B.

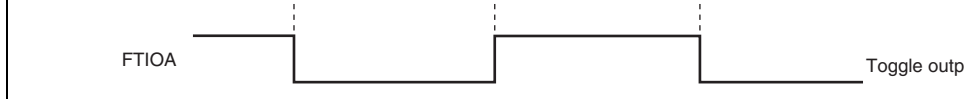


Figure 14.14 Example of Toggle Output Operation

2. Output compare timing

The compare match signal is generated in the last state in which TCNT and GR match (TCNT changes from the matching value to the next value). When the compare match signal is generated, the output value selected in TIOR is output at the compare match output pin (FTIOA, FTIOB, FTIOC, or FTIOD). When TCNT matches GR, the compare match signal is generated only after the next TCNT input clock pulse is input.

Figure 14.15 shows an example of the output compare timing.

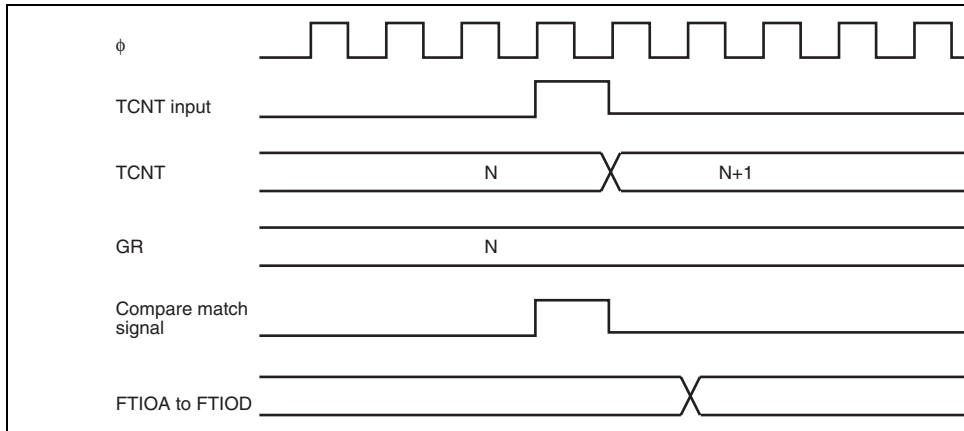


Figure 14.15 Output Compare Timing

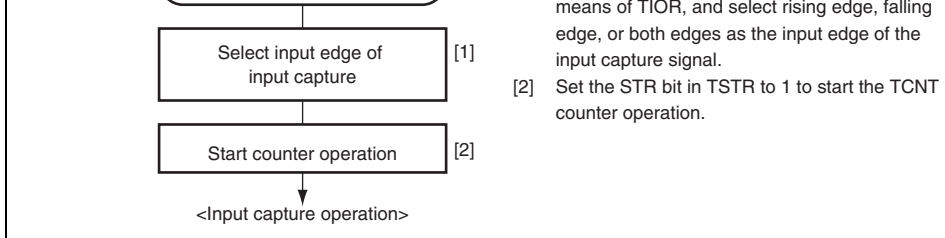


Figure 14.16 Example of Input Capture Operation Setting Procedure

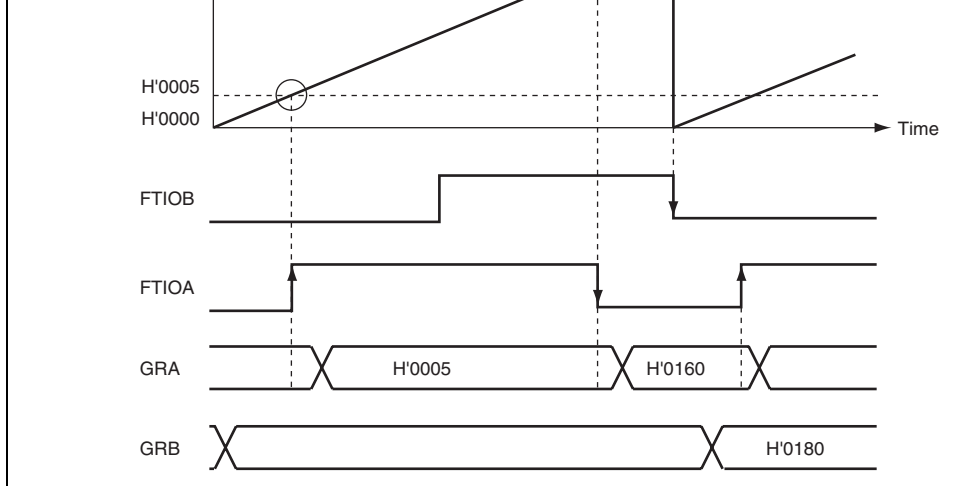


Figure 14.17 Example of Input Capture Operation

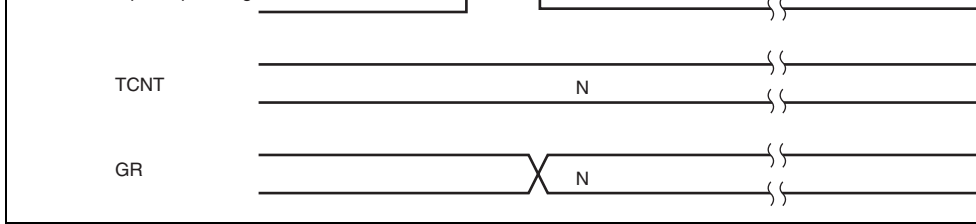


Figure 14.18 Input Capture Signal Timing

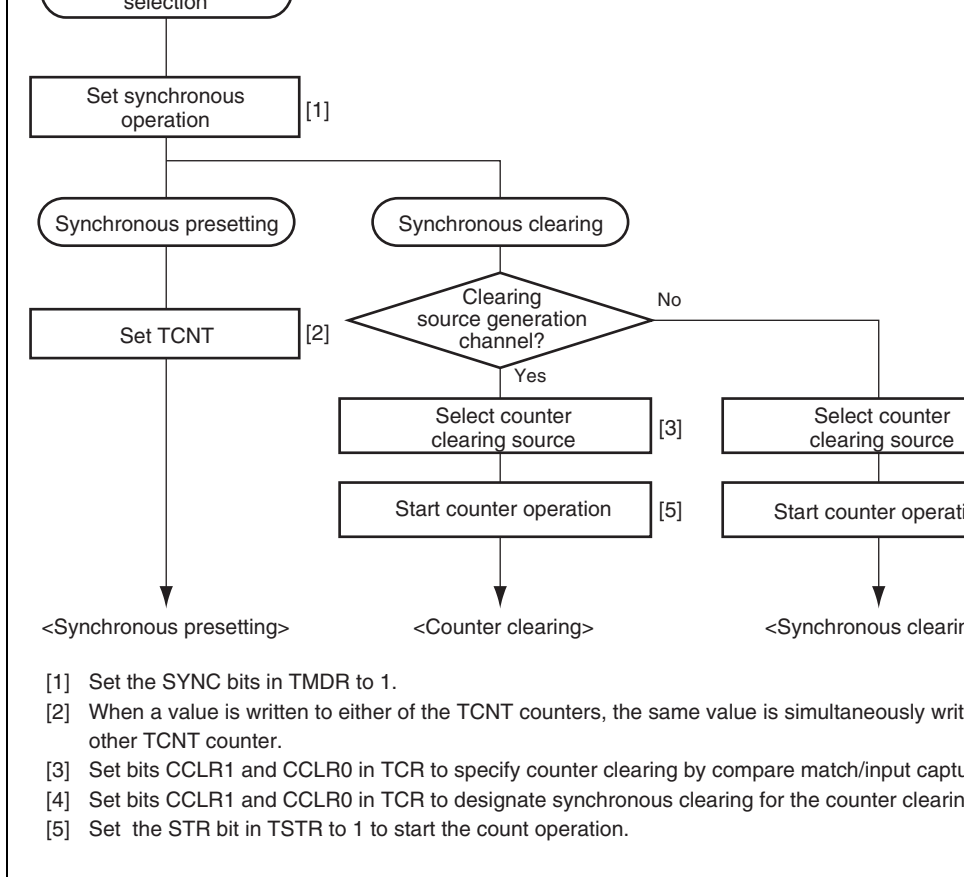


Figure 14.19 Example of Synchronous Operation Setting Procedure

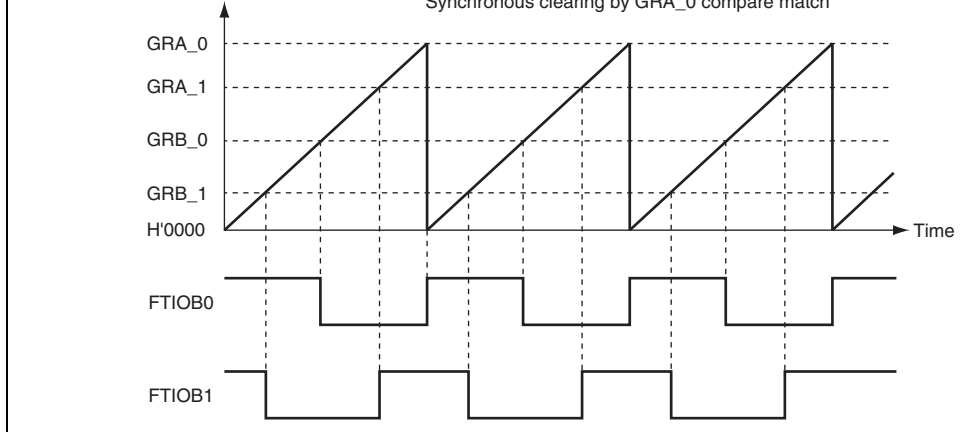


Figure 14.20 Example of Synchronous Operation

14.4.5 PWM Mode

In PWM mode, PWM waveforms are output from the FTIOB, FTIOC, and FTIOD outputs with GRA as a cycle register and GRB, GRC, and GRD as duty registers. The initial output of the corresponding pin depends on the setting values of TOCR and POCCR. Table 14.3 shows an example of the initial output level of the FTIOB0 pin.

The output level is determined by the POLB to POLD bits corresponding to POCCR. When POLB is 0, the FTIOB output pin is set to 0 by compare match B and set to 1 by compare match A. When POLB is 1, the FTIOB output pin is set to 1 by compare match B and cleared to 0 by compare match A. In PWM mode, maximum 6-phase PWM outputs are possible.

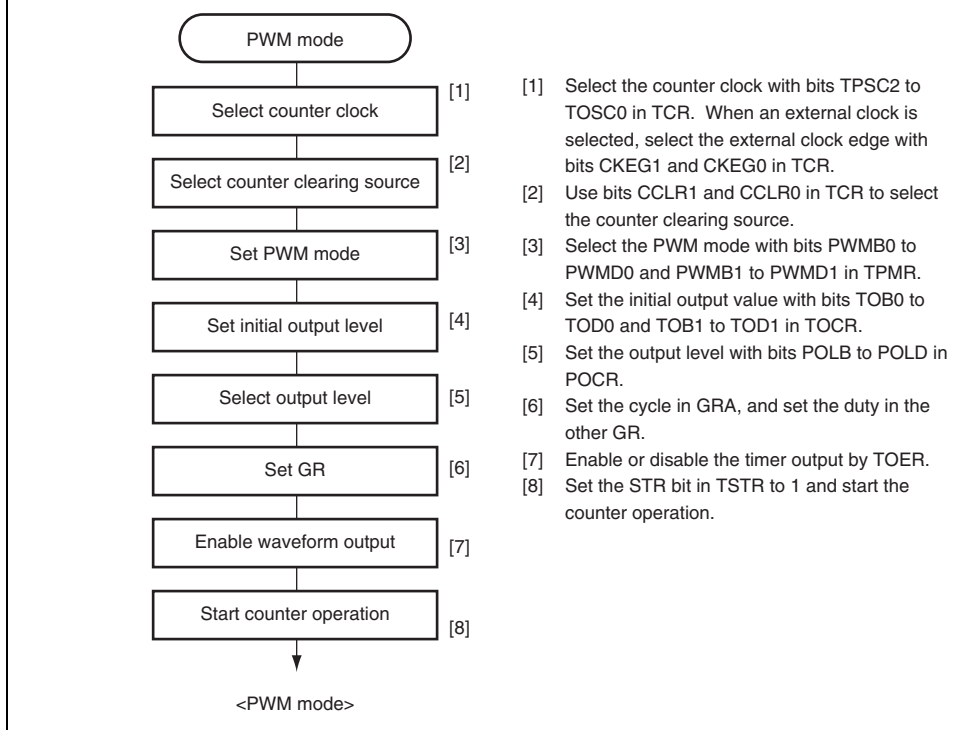


Figure 14.21 Example of PWM Mode Setting Procedure

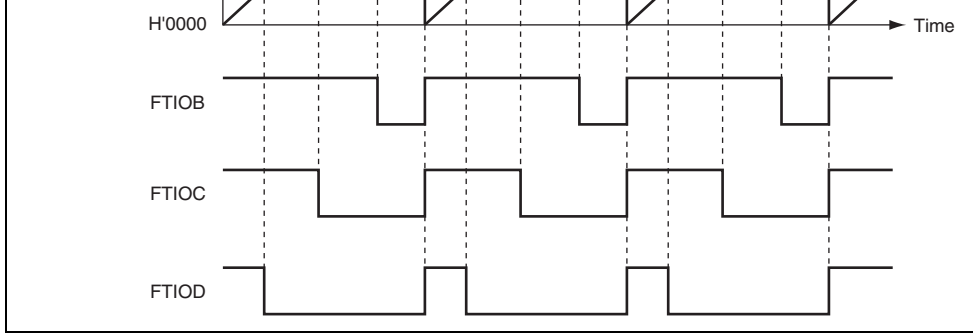


Figure 14.22 Example of PWM Mode Operation (1)

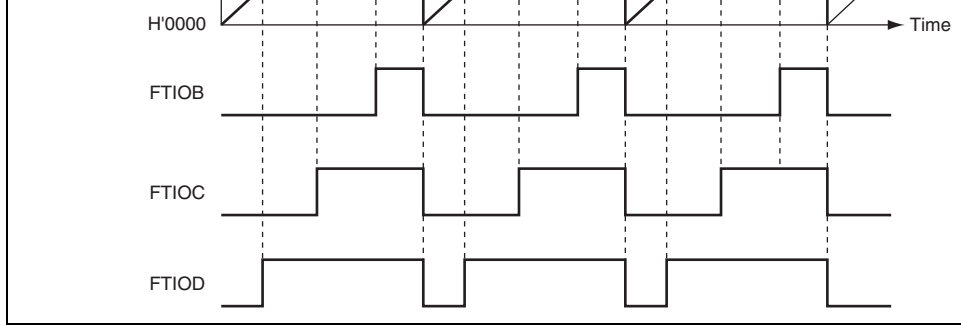


Figure 14.23 Example of PWM Mode Operation (2)

Figures 14.24 (when TOB, TOC, and TOD = 0, POLB, POLC, and POLD = 0) and 14.25 (when TOB, TOC, and TOD = 0, POLB, POLC, and POLD = 1) show examples of the output waveforms with duty cycles of 0% and 100% in PWM mode.

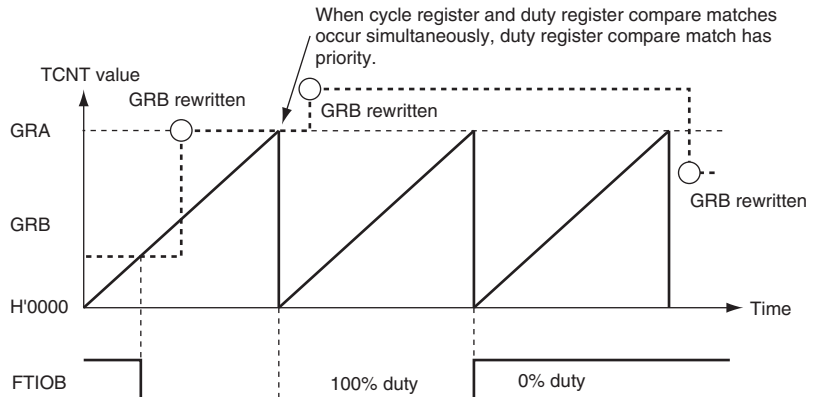
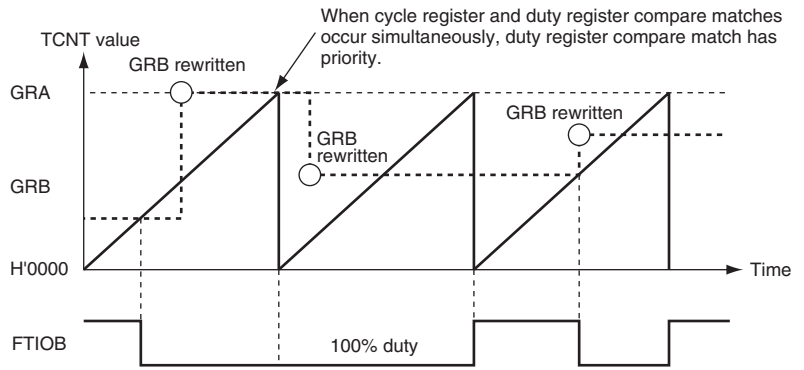


Figure 14.24 Example of PWM Mode Operation (3)

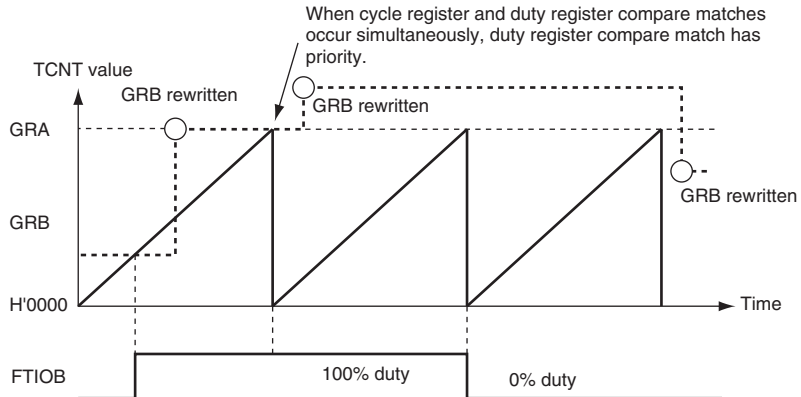
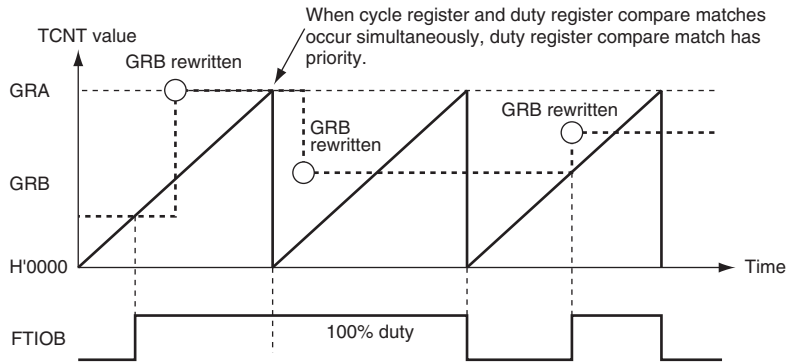


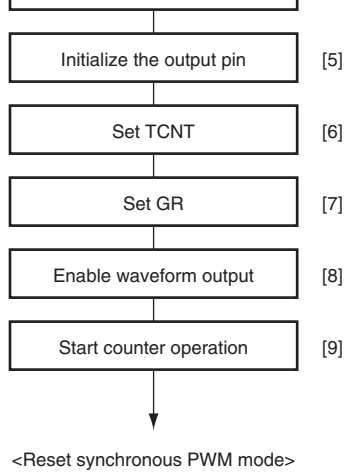
Figure 14.25 Example of PWM Mode Operation (4)

Table 14.4 Output Pins in Reset Synchronous PWM Mode

Channel	Pin Name	Input/Output	Pin Function
0	FTIOC0	Output	Toggle output in synchronous with PWM cycle
0	FTIOB0	Output	PWM output 1
0	FTIOD0	Output	PWM output 1 (counter-phase waveform of PWM output 1)
1	FTIOA1	Output	PWM output 2
1	FTIOC1	Output	PWM output 2 (counter-phase waveform of PWM output 2)
1	FTIOB1	Output	PWM output 3
1	FTIOD1	Output	PWM output 3 (counter-phase waveform of PWM output 3)

Table 14.5 Register Settings in Reset Synchronous PWM Mode

Register	Description
TCNT_0	Initial setting of H'0000
TCNT_1	Not used (independently operates)
GRA_0	Sets counter cycle of TCNT_0
GRB_0	Set a changing point of the PWM waveform output from pins FTIOB0 and FT
GRA_1	Set a changing point of the PWM waveform output from pins FTIOA1 and FT
GRB_1	Set a changing point of the PWM waveform output from pins FTIOB1 and FT



- bits CMD1 and CMD0 in TFCR. FTIOB0 to FTIOD0 and FTIOA1 to FTIOD1 become PWM output pins automatically.
- [5] Set H'00 to TOCR.
- [6] Set TCNT_0 as H'0000. TCNT1 does not need to be set.
- [7] GRA_0 is a cycle register. Set a cycle for GRA_0. Set the changing point timing of the PWM output waveform for GRB_0, GRA_1, and GRB_1.
- [8] Enable or disable the timer output by TOER.
- [9] Set the STR bit in TSTR to 1 and start the counter operation.

Figure 14.26 Example of Reset Synchronous PWM Mode Setting Procedure

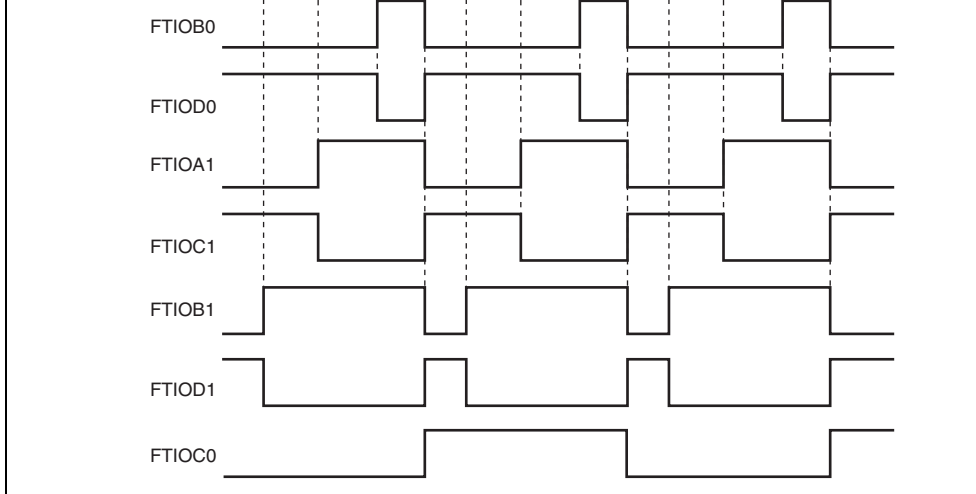


Figure 14.27 Example of Reset Synchronous PWM Mode Operation (OLS0 = OL

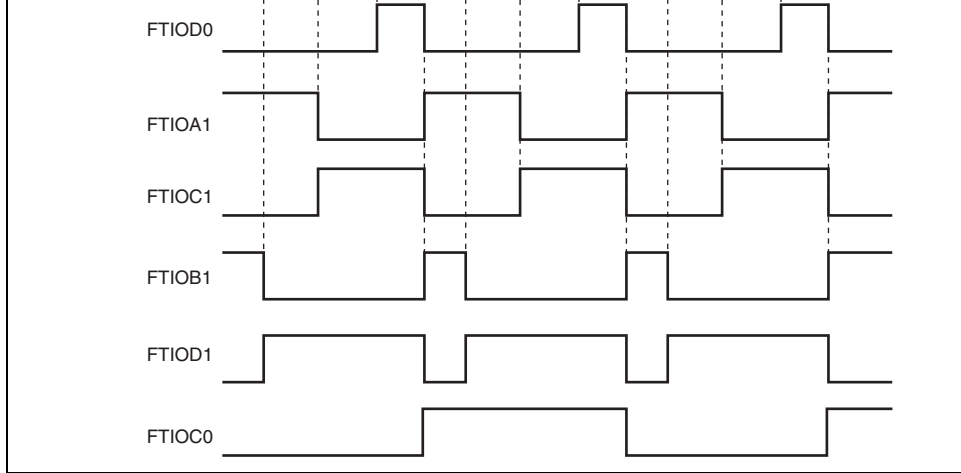


Figure 14.28 Example of Reset Synchronous PWM Mode Operation (OLS0 = 0)

In reset synchronous PWM mode, TCNT_0 and TCNT_1 perform increment and independent operations, respectively. However, GRA_1 and GRB_1 are separated from TCNT_1. When a compare match occurs between TCNT_0 and GRA_0, a counter is cleared and an increment operation is restarted from H'0000.

The PWM pin outputs 0 or 1 whenever a compare match between GRB_0, GRA_1, GRB_1, TCNT_0 or counter clearing occur.

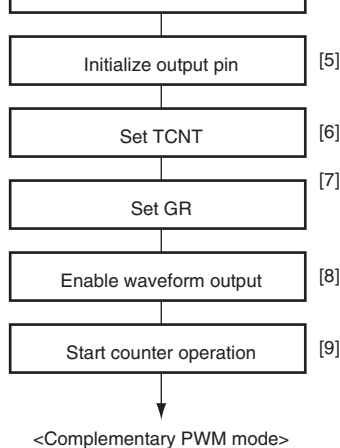
For details on operations when reset synchronous PWM mode and buffer operation are simultaneously set, see section 14.4.8, Buffer Operation.

Table 14.6 Output Pins in Complementary PWM Mode

Channel	Pin Name	Input/Output	Pin Function
0	FTIOC0	Output	Toggle output in synchronous with PWM cycle
0	FTIOB0	Output	PWM output 1
0	FTIOD0	Output	PWM output 1 (counter-phase waveform non-overlapped with PWM output 1)
1	FTIOA1	Output	PWM output 2
1	FTIOC1	Output	PWM output 2 (counter-phase waveform non-overlapped with PWM output 2)
1	FTIOB1	Output	PWM output 3
1	FTIOD1	Output	PWM output 3 (counter-phase waveform non-overlapped with PWM output 3)

Table 14.7 Register Settings in Complementary PWM Mode

Register	Description
TCNT_0	Initial setting of non-overlapped periods (non-overlapped periods are different from TCNT_1)
TCNT_1	Initial setting of H'0000
GRA_0	Sets (upper limit value – 1) of TCNT_0
GRB_0	Set a changing point of the PWM waveform output from pins FTIOB0 and FTIOD0
GRA_1	Set a changing point of the PWM waveform output from pins FTIOA1 and FTIOC1
GRB_1	Set a changing point of the PWM waveform output from pins FTIOB1 and FTIOD1



- and FTIOA1 to FTIOD1 automatically become PWM output pins.
- [5] Set H'00 to TOCR.
 - [6] TCNT_1 must be H'0000. Set a non-overlapping period to TCNT_0.
 - [7] GRA_0 is a cycle register. Set the cycle to GRA_0. Set the timing to change the PWM output waveform to GRB_0, GRA_1, and GRB_1. Note that the timing must be set within the range of compare match carried out for TCNT_0 and TCNT_1. For GR settings, see 3. Setting GR Value in Complementary PWM Mode in section 14.4.7, Complementary PWM Mode.
 - [8] Use TOER to enable or disable the timer output.
 - [9] Set the STR0 and STR1 bits in TSTR to 1 to start the count operation.

Note: To re-enter complementary PWM mode, first, enter a mode other than the complementary PWM mode. After that, repeat the setting procedures from step [1]. For settings of waveform outputs with a duty cycle of 0% and 100%, see the settings shown in 2. Examples of Complementary PWM Mode Operation and 3. Setting GR Value in Complementary PWM Mode in section 14.4.7, Complementary PWM Mode.

Figure 14.29 Example of Complementary PWM Mode Setting Procedure

<Normal operation>

Figure 14.30 Canceling Procedure of Complementary PWM Mode

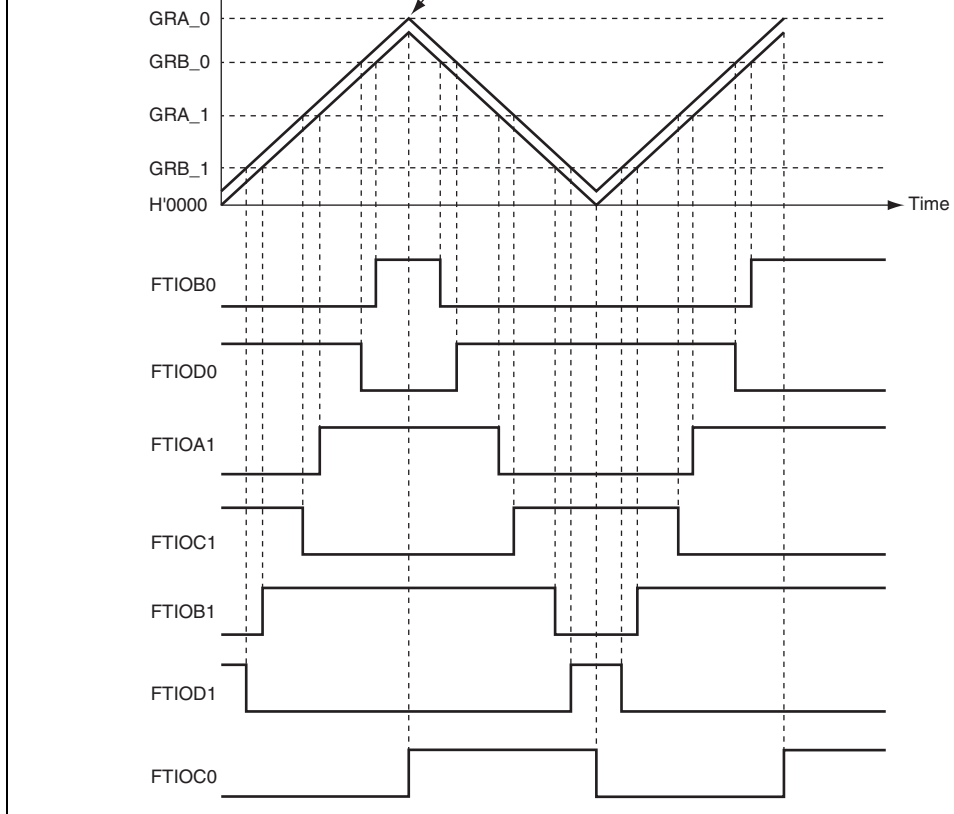


Figure 14.31 Example of Complementary PWM Mode Operation (1)

cycle waveform output, see 3. C., Outputting a waveform with a duty cycle of 0% and section 14.4.7, Complementary PWM Mode.

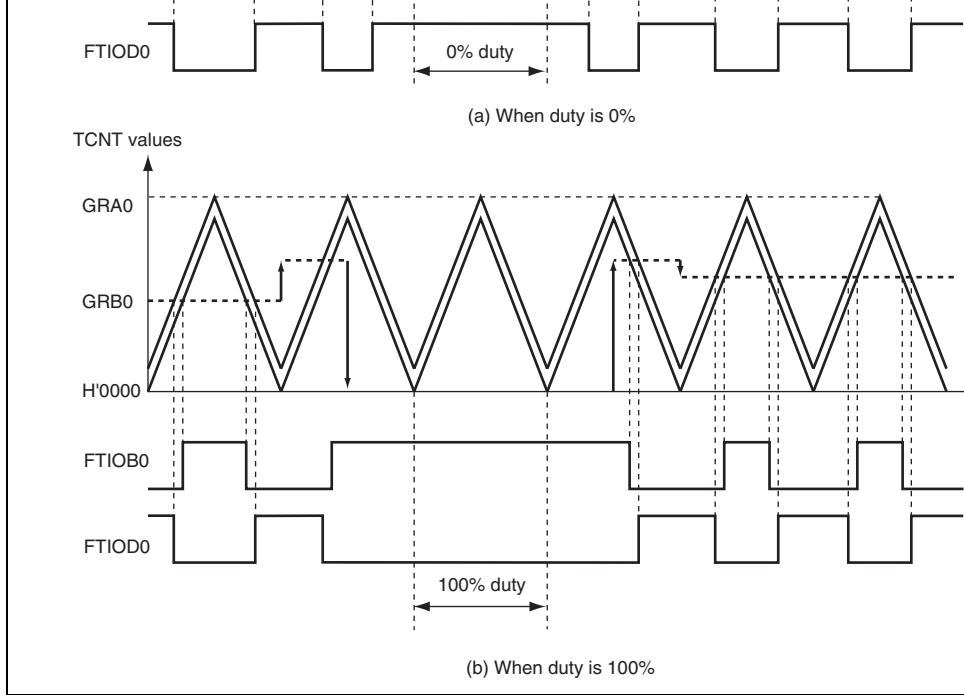


Figure 14.32 (1) Example of Complementary PWM Mode Operation (TPSC2 = TPSC1 = TPSC0 = 0) (2)

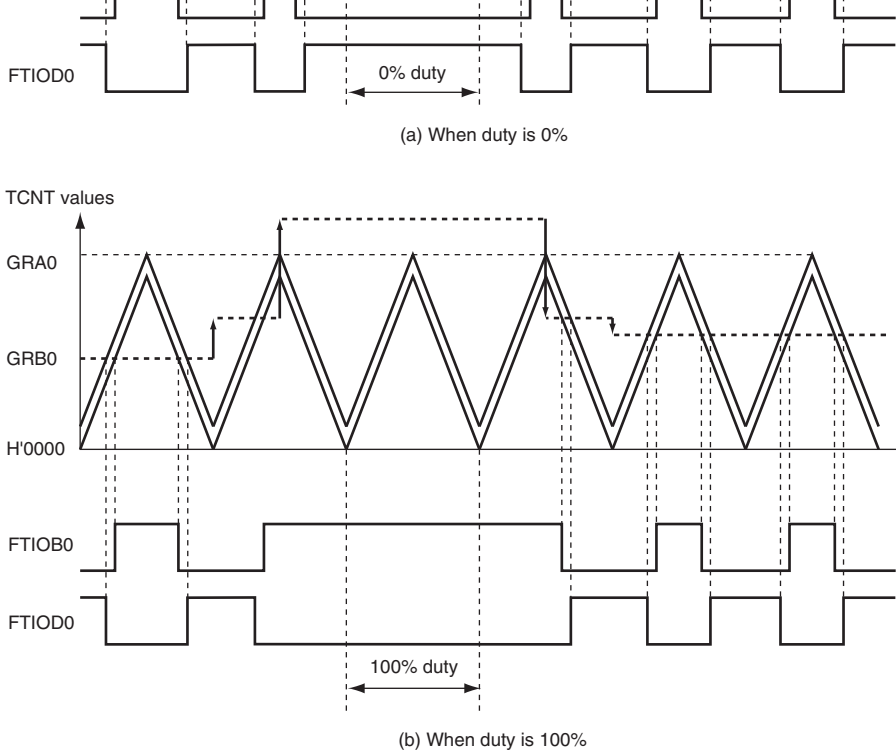


Figure 14.32 (2) Example of Complementary PWM Mode Operation (TPSC2 = TPSC1 = TPSC0 ≠ 0) (3)

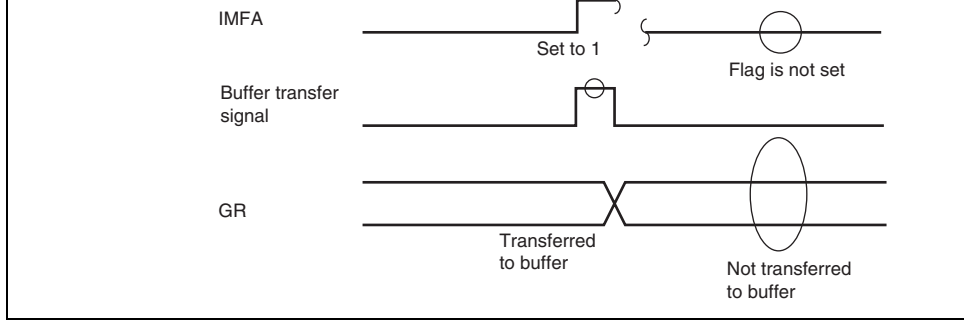


Figure 14.33 Timing of Overshooting

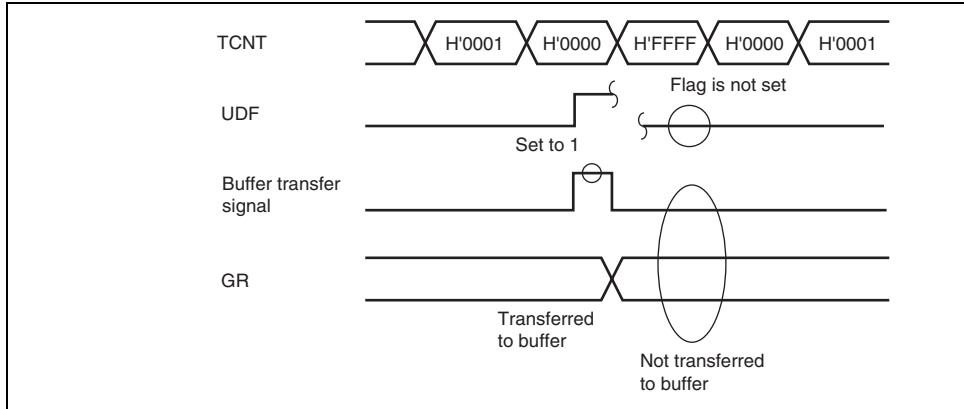


Figure 14.34 Timing of Undershooting

H'FFFC or less. When TPSC2 = TPSC1 = TPSC0 = 0, the GRA_0 value can be H'FFFF or less.

- b. H'0000 to T - 1 (T: Initial value of TCNT0) must not be set for the initial value.
- c. GRA_0 - (T - 1) or more must not be set for the initial value.
- d. When using buffer operation, the same values must be set in the buffer registers and corresponding general registers.

B. Modifying the setting value

- a. Writing to GR directly must be performed while the TCNT_1 and TCNT_0 values should satisfy the following expression: $H'0000 \leq TCNT_1 < \text{previous GR value}$ and $\text{previous GR value} < TCNT_0 \leq GRA_0$. Otherwise, a waveform is not output correctly. For details on outputting a waveform with a duty cycle of 0% and 100%, see Section 10.2.2.1. C., Outputting a waveform with a duty cycle of 0% and 100%.
- b. Do not write the following values to GR directly. When writing the values, a waveform is not output correctly.
 $H'0000 \leq GR \leq T - 1$ and $GRA_0 - (T - 1) \leq GR < GRA_0$ when TPSC2 = TPSC1 = TPSC0 = 0
 $H'0000 < GR \leq T - 1$ and $GRA_0 - (T - 1) \leq GR < GRA_0 + 1$ when TPSC2 = TPSC1 = TPSC0 = 0
- c. Do not change settings of GRA_0 during operation.

C. Outputting a waveform with a duty cycle of 0% and 100%

- a. Buffer operation is not used and TPSC2 = TPSC1 = TPSC0 = 0
Write H'0000 or a value equal to or more than the GRA_0 value to GR directly. The output timing is shown below.
 - To output a 0%-duty cycle waveform, write a value equal to or more than the GRA_0 value while $H'0000 \leq TCNT_1 < \text{previous GR value}$

b. Buffer operation is used and $TPSC2 = TPSC1 = TPSC0 = 0$

Write H'0000 or a value equal to or more than the GRA_0 value to the buffer register

- To output a 0%-duty cycle waveform, write a value equal to or more than the GRA_0 value to the buffer register
 - To output a 100%-duty cycle waveform, write H'0000 to the buffer register
- For details on buffer operation, see section 14.4.8, Buffer Operation.

c. Buffer operation is not used and other than $TPSC2 = TPSC1 = TPSC0 = 0$

Write a value which satisfies $GRA_0 + 1 < GR < H'FFFF$ to GR directly at the address shown below.

- To output a 0%-duty cycle waveform, write the value while $H'0000 \leq TCNT_0 < previous\ GR\ value$
- To output a 100%-duty cycle waveform, write the value while $previous\ GR\ value < TCNT_0 \leq GRA_0$

To change duty cycles while a waveform with a duty cycle of 0% and 100% is being output, the following procedure must be followed.

- To change duty cycles while a 0%-duty cycle waveform is being output, write the value while $H'0000 \leq TCNT_1 < previous\ GR\ value$
- To change duty cycles while a 100%-duty cycle waveform is being output, write the value while $previous\ GR\ value < TCNT_0 \leq GRA_0$

Note that changing from a 0%-duty cycle waveform to a 100%-duty cycle waveform and vice versa is not possible.

Register of an output compare register, or in reset by non-maskable PWM mode or complementary PWM mode.

Table 14.8 shows the register combinations used in buffer operation.

Table 14.8 Register Combinations in Buffer Operation

General Register	Buffer Register
GRA	GRC
GRB	GRD

1. When GR is an output compare register
When a compare match occurs, the value in the buffer register of the corresponding channel is transferred to the general register.
This operation is illustrated in figure 14.35.

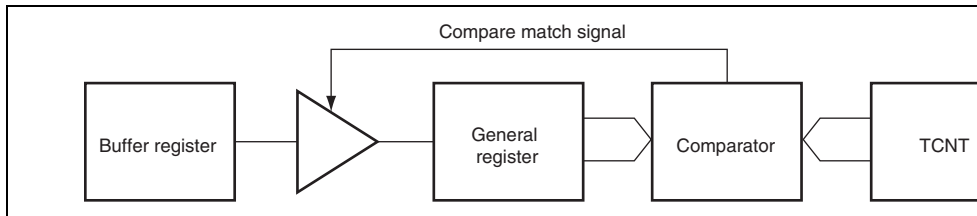


Figure 14.35 Compare Match Buffer Operation

Figure 14.36 Input Capture Buffer Operation

3. Complementary PWM Mode

When the counter switches from counting up to counting down or vice versa, the value of the buffer register is transferred to the general register. Here, the value of the buffer register is transferred to the general register in the following timing:

- A. When TCNT_0 and GRA_0 are compared and their contents match
- B. When TCNT_1 underflows

4. Reset Synchronous PWM Mode

The value of the buffer register is transferred from compare match A0 to the general register.

5. Example of Buffer Operation Setting Procedure

Figure 14.37 shows an example of the buffer operation setting procedure.

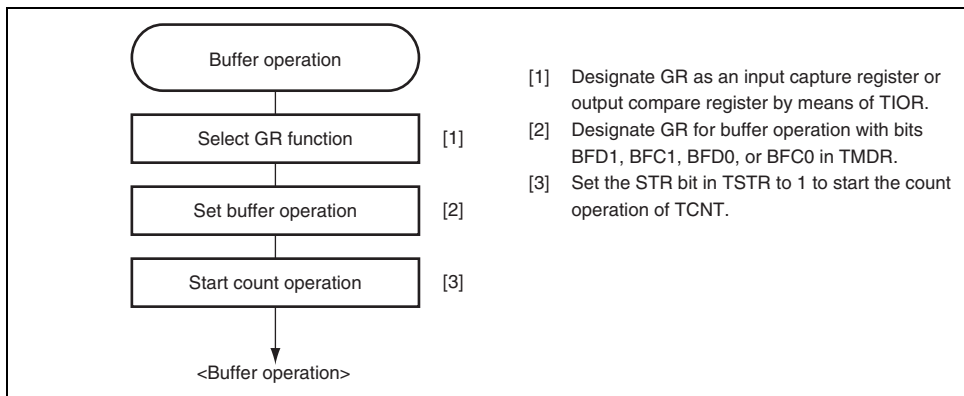
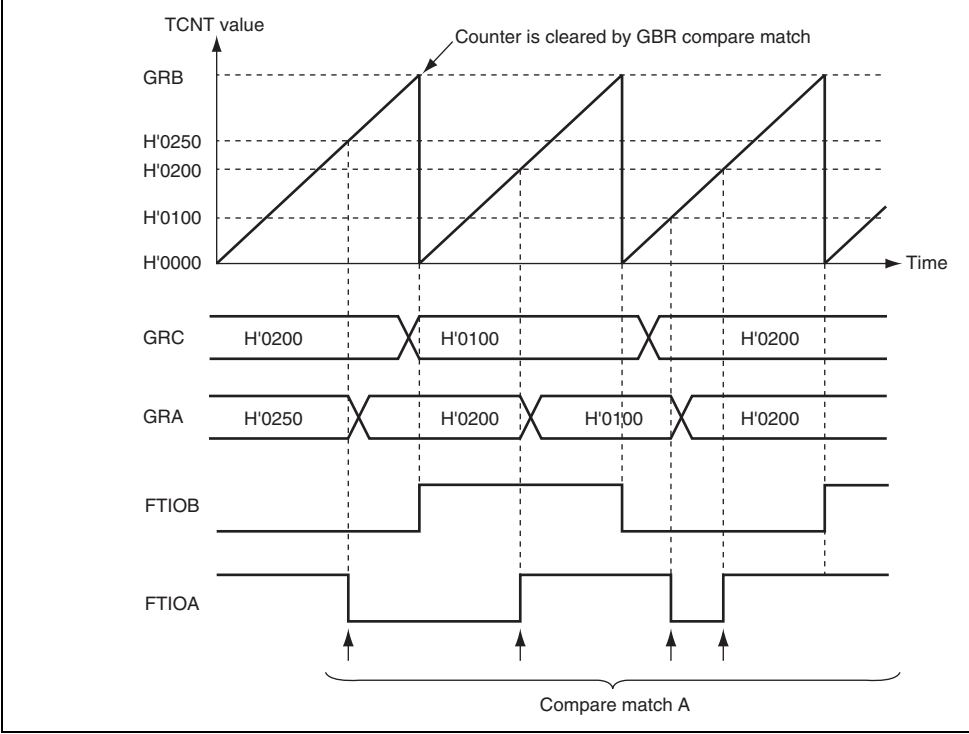


Figure 14.37 Example of Buffer Operation Setting Procedure



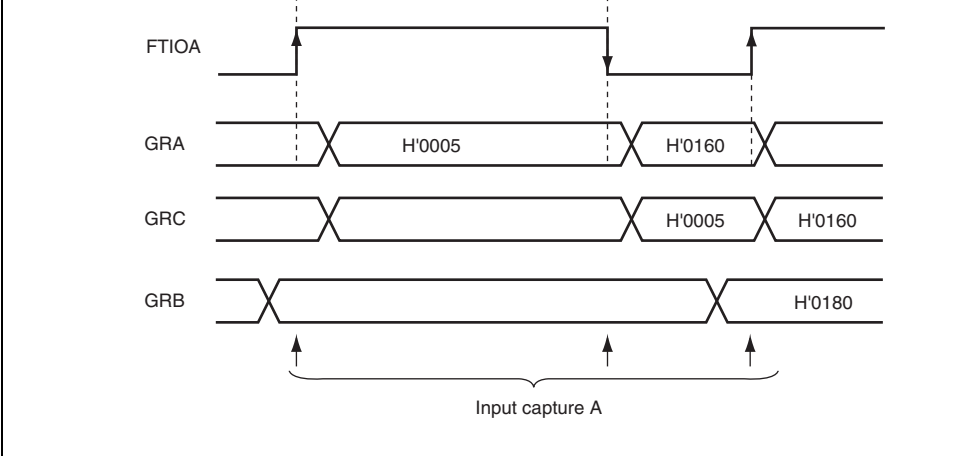
**Figure 14.38 Example of Buffer Operation (1)
(Buffer Operation for Output Compare Register)**

Figure 14.39 Example of Compare Match Timing for Buffer Operation

Figure 14.40 shows an operation example in which GRA has been designated as an input register, and buffer operation has been designated for GRA and GRC.

Counter clearing by input capture B has been set for TCNT, and falling edges have been set as the FIOCB pin input capture input edge. And both rising and falling edges have been set as the FIOCA pin input capture input edge.

As buffer operation has been set, when the TCNT value is stored in GRA upon the occurrence of input capture A, the value previously stored in GRA is simultaneously transferred to GRC. The transfer timing is shown in figure 14.41.



**Figure 14.40 Example of Buffer Operation (2)
(Buffer Operation for Input Capture Register)**

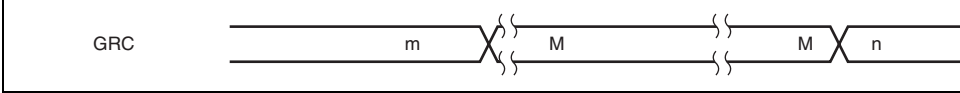


Figure 14.41 Input Capture Timing of Buffer Operation

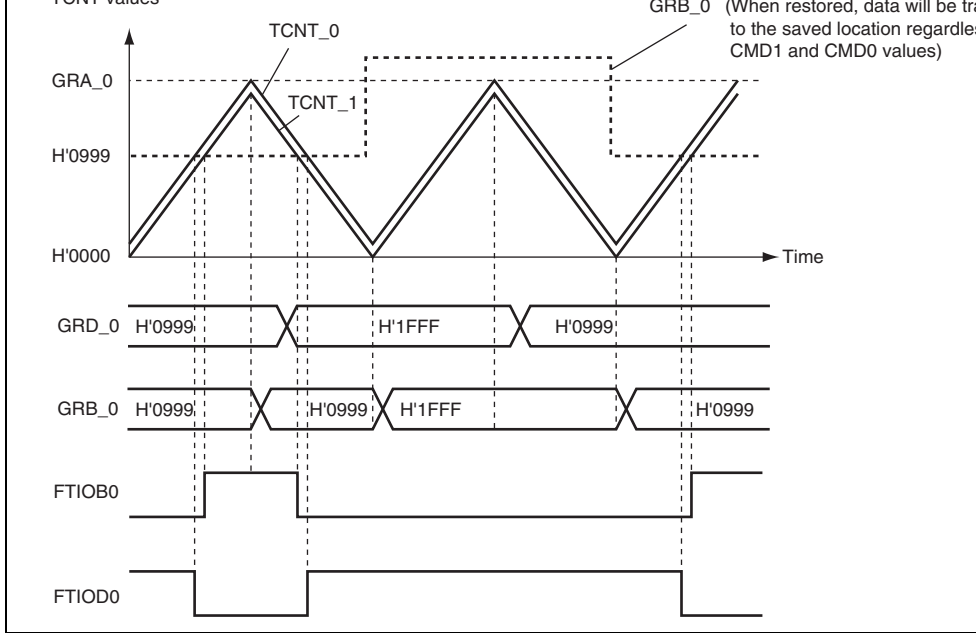


Figure 14.42 Buffer Operation (3)
(Buffer Operation in Complementary PWM Mode CMD1 = CMD0 = 1)

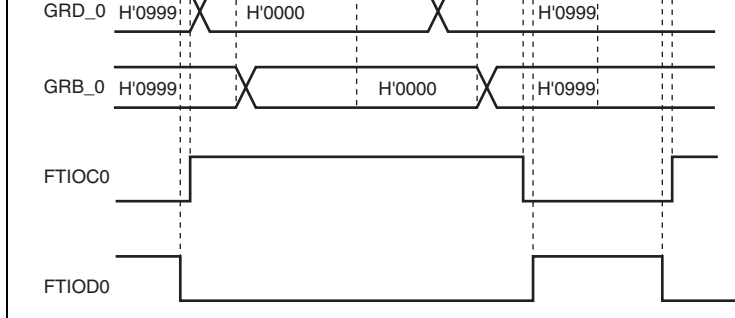


Figure 14.43 Buffer Operation (4)
(Buffer Operation in Complementary PWM Mode CMD1 = CMD0 = 1)

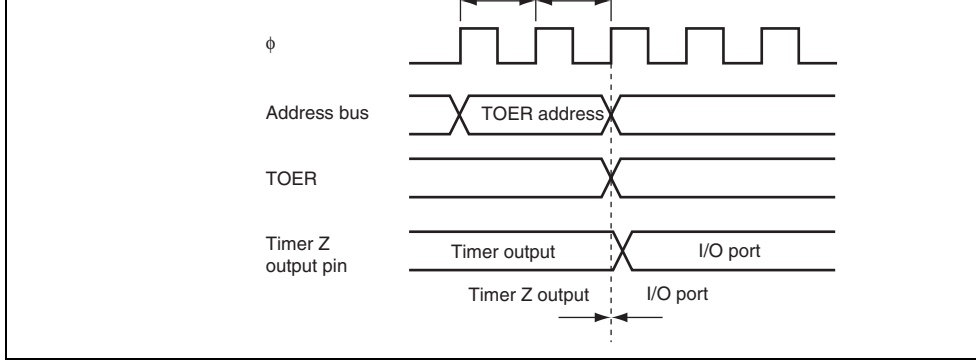


Figure 14.44 Example of Output Disable Timing of Timer Z by Writing to TOER

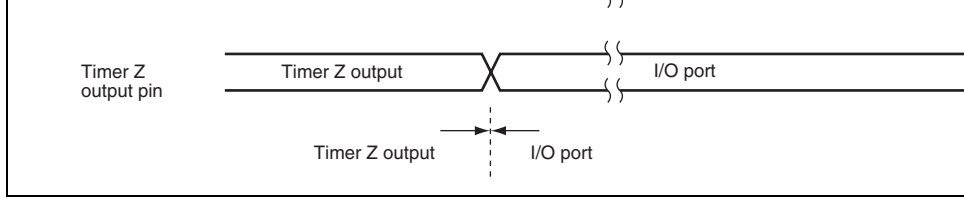


Figure 14.45 Example of Output Disable Timing of Timer Z by External Tri

3. Output Inverse Timing by TFCR: The output level can be inverted by inverting the OLS0 bits in TFCR in reset synchronous PWM mode or complementary PWM mode. Figure 14.46 shows the timing.

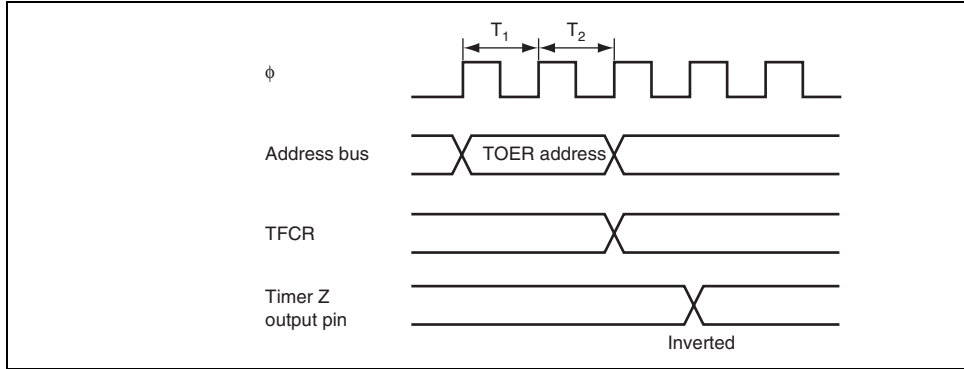


Figure 14.46 Example of Output Inverse Timing of Timer Z by Writing to T

Figure 14.47 Example of Output Inverse Timing of Timer Z by Writing to PC

14.5 Interrupts

There are three kinds of timer Z interrupt sources; input capture/compare match, overflow/underflow. An interrupt is requested when the corresponding interrupt request flag is set to 1. The corresponding interrupt enable bit is set to 1.

14.5.1 Status Flag Set Timing

1. **IMF Flag Set Timing:** The IMF flag is set to 1 by the compare match signal that is generated when the GR matches with the TCNT. The compare match signal is generated at the time of matching (timing to update the counter value when the GR and TCNT match). The IMF flag is cleared to 0 when the TCNT and GR matches, the compare match signal will not be generated until the next TCNT input clock is generated. Figure 14.48 shows the timing to set the IMF flag.



Figure 14.48 IMF Flag Set Timing when Compare Match Occurs

2. IMF Flag Set Timing at Input Capture: When an input capture signal is generated, the GR is set to 1 and the value of TCNT is simultaneously transferred to corresponding GR. Figure 14.49 shows the timing.

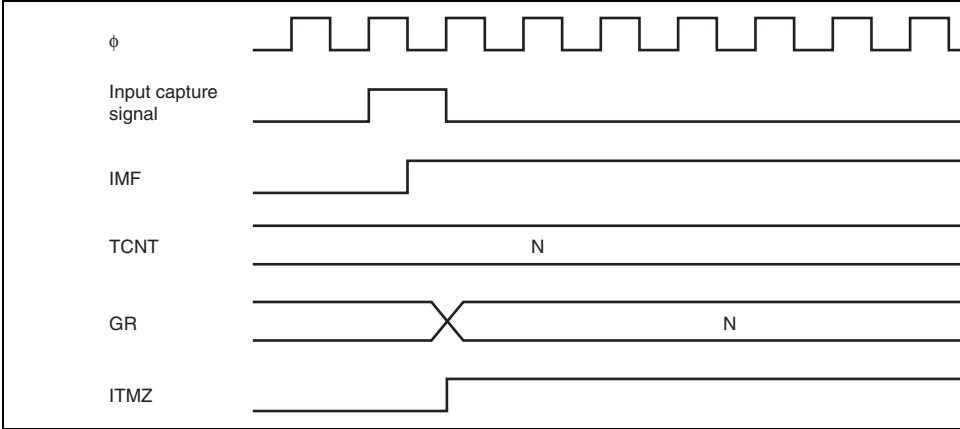


Figure 14.49 IMF Flag Set Timing at Input Capture



Figure 14.50 OVF Flag Set Timing

14.5.2 Status Flag Clearing Timing

The status flag can be cleared by writing 0 after reading 1 from the CPU. Figure 14.51 shows timing in this case.

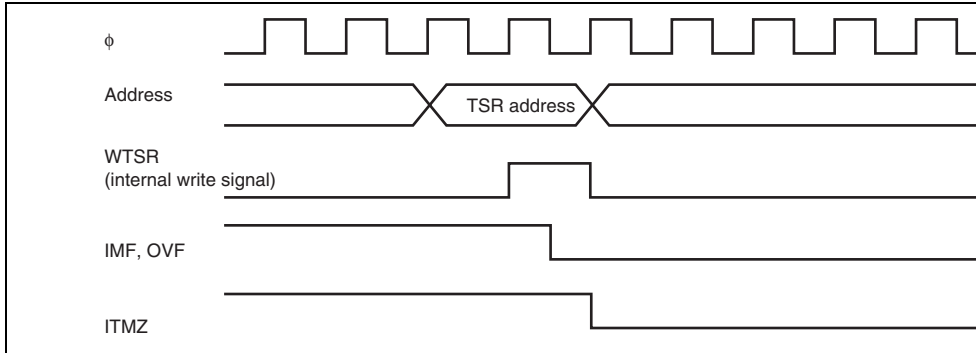


Figure 14.51 Status Flag Clearing Timing

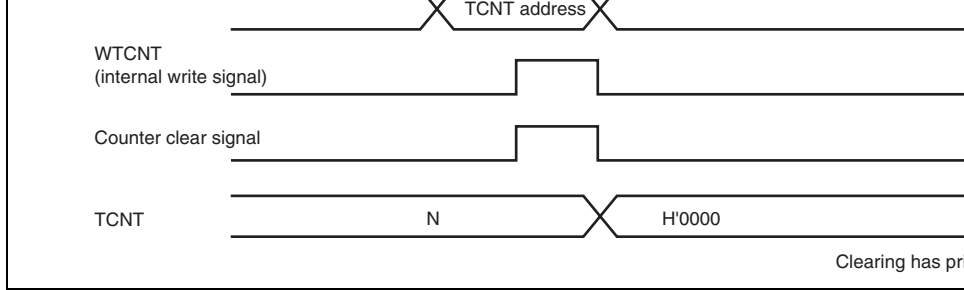


Figure 14.52 Contention between TCNT Write and Clear Operations

2. Contention between TCNT Write and Increment Operations: If incrementation is done at the start of a TCNT write cycle, TCNT writing has priority. Figure 14.53 shows the timing case.

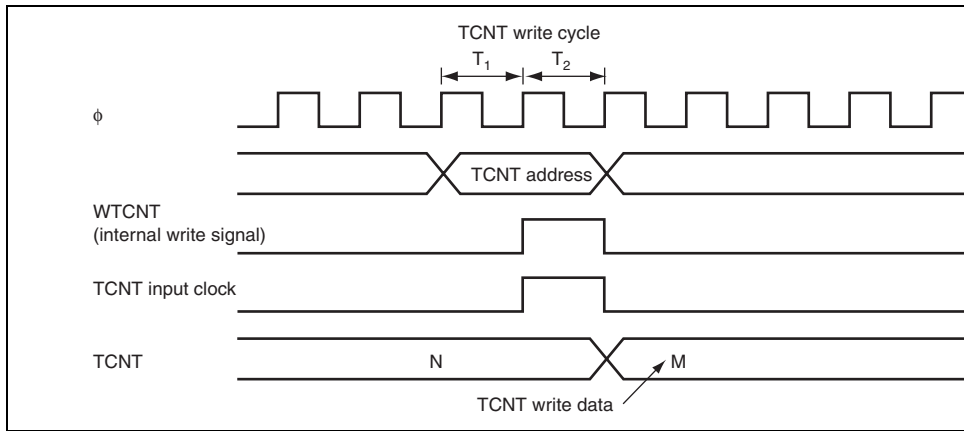


Figure 14.53 Contention between TCNT Write and Increment Operation

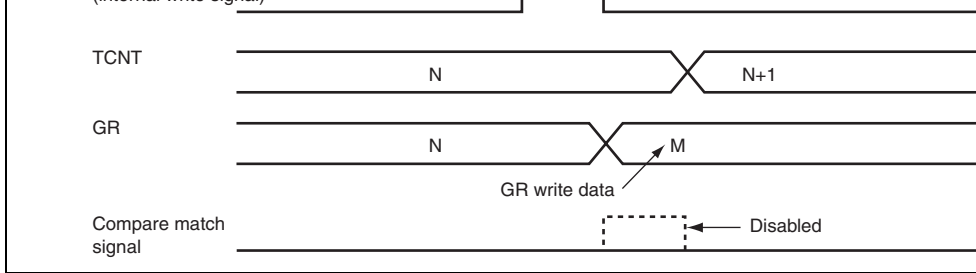


Figure 14.54 Contention between GR Write and Compare Match

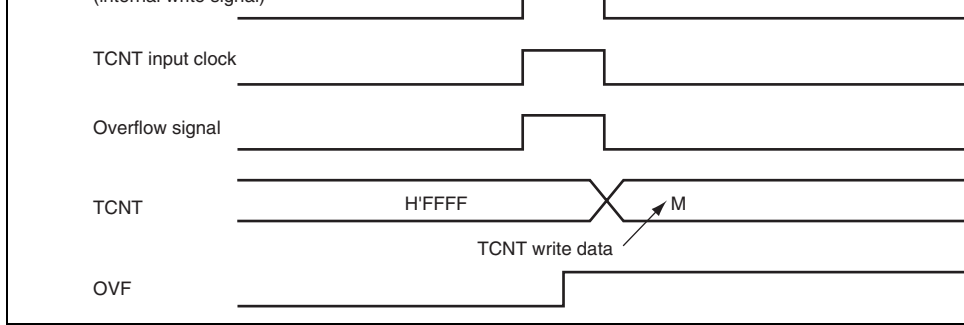


Figure 14.55 Contention between TCNT Write and Overflow

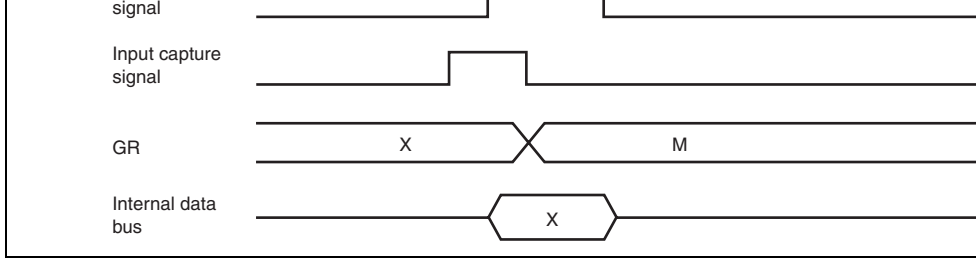


Figure 14.56 Contention between GR Read and Input Capture

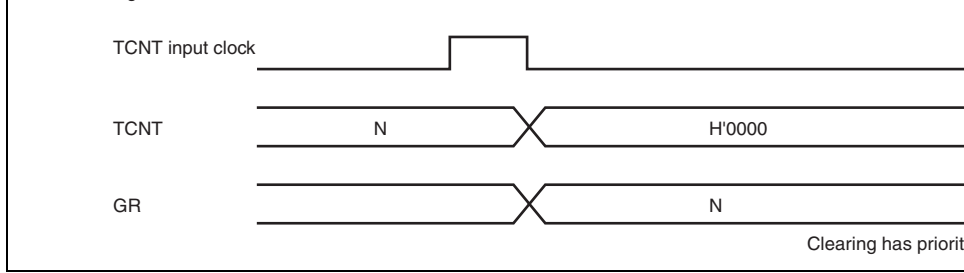


Figure 14.57 Contention between Count Clearing and Increment Operation by Input Capture

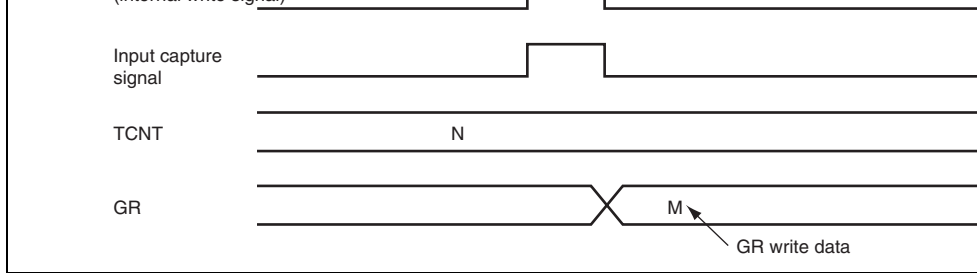


Figure 14.58 Contention between GR Write and Input Capture

8. Notes on Setting Reset Synchronous PWM Mode/Complementary PWM Mode: When CMD1 and CMD0 in TFCR are set, note the following:
 - A. Write bits CMD1 and CMD0 while TCNT_1 and TCNT_0 are halted.
 - B. Changing the settings of reset synchronous PWM mode to complementary PWM mode and vice versa is disabled. Set reset synchronous PWM mode or complementary PWM mode after the normal operation (bits CMD1 and CMD0 are cleared to 0) has been set.
9. Notes on Writing to the TOA0 to TOD0 Bits and the TOA1 to TOD1 Bits in TOCR:

The TOA0 to TOD0 bits and the TOA1 to TOD1 bits in TOCR decide the value of the pin, which is output until the first compare match occurs. Once a compare match occurs, this compare match changes the values of FTIOA0 to FTIOD0 and FTIOA1 to FTIOD1 pin output, the values of the FTIOA0 to FTIOD0 and FTIOA1 to FTIOD1 pin output and values read from the TOA0 to TOD0 and TOA1 to TOD1 bits may differ. Moreover, when writing to TOCR and the generation of the compare match A0 to D0 and A1 to D1 occur at the same timing, the writing to TOCR has the priority. Thus, output change due to the compare match is not reflected to the FTIOA0 to FTIOD0 and FTIOA1 to FTIOD1 pins. Therefore, when bit manipulation instruction is used to write to TOCR, the values of the FTIOA0 to FTIOD0 and FTIOA1 to FTIOD1 pin output may result in an unexpected result. When writing to TOCR while compare match is operating, stop the counter once before accessing TOCR.

TOCR	TOB1	TOB0	TOA1	TOA0	TOB0	TOB0	TOB0	TOB0
Set value	0	0	0	0	0	1	1	0

BCLR#2, @TOCR

- (1) TOCR read operation: Read H'06
- (2) Modify operation: Modify H'06 to H'02
- (3) Write operation to TOCR: Write H'02

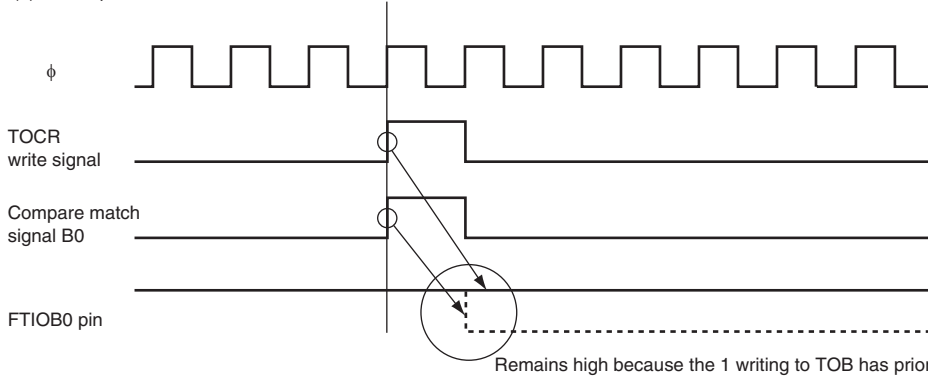


Figure 14.59 When Compare Match and Bit Manipulation Instruction to TOCR Occur at the Same Timing

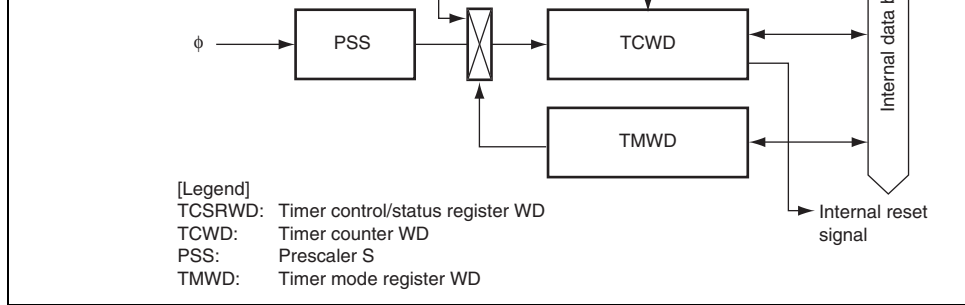


Figure 15.1 Block Diagram of Watchdog Timer

15.1 Features

- Selectable from nine counter input clocks.
 Eight clock sources ($\phi/64$, $\phi/128$, $\phi/256$, $\phi/512$, $\phi/1024$, $\phi/2048$, $\phi/4096$, and $\phi/8192$) internal oscillator can be selected as the timer-counter clock. When the internal oscillator is selected, it can operate as the watchdog timer in any operating mode.
- Reset signal generated on counter overflow
 An overflow period of 1 to 256 times the selected clock can be set.

watchdog timer operation and indicates the operating state. TCSRWD must be rewritten with the MOV instruction. The bit manipulation instruction cannot be used to change the setting.

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

				<ul style="list-style-type: none"> When 0 is written to the WDON bit while writing the B2WI when the TCSRWE bit=1
1	BOWI	1	R/W	<p>Bit 0 Write Inhibit</p> <p>This bit can be written to the WRST bit only when the write value of the BOWI bit is 0. This bit is always 1.</p>
0	WRST	0	R/W	<p>Watchdog Timer Reset</p> <p>[Setting condition]</p> <p>When TCWD overflows and an internal reset is generated</p> <p>[Clearing conditions]</p> <ul style="list-style-type: none"> Reset by $\overline{\text{RES}}$ pin When 0 is written to the WRST bit while writing the BOWI bit when the TCSRWE bit=1

Bit	Bit Name	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 $\phi/64$
0	CKS0	1	R/W	1001: Internal clock: counts on $\phi/128$ 1010: Internal clock: counts on $\phi/256$ 1011: Internal clock: counts on $\phi/512$ 1100: Internal clock: counts on $\phi/1024$ 1101: Internal clock: counts on $\phi/2048$ 1110: Internal clock: counts on $\phi/4096$ 1111: Internal clock: counts on $\phi/8192$ 0xxx: Internal oscillator For the internal oscillator overflow periods, see 23, Electrical Characteristics.

[Legend] x: Don't care.

Figure 15.2 shows an example of watchdog timer operation.

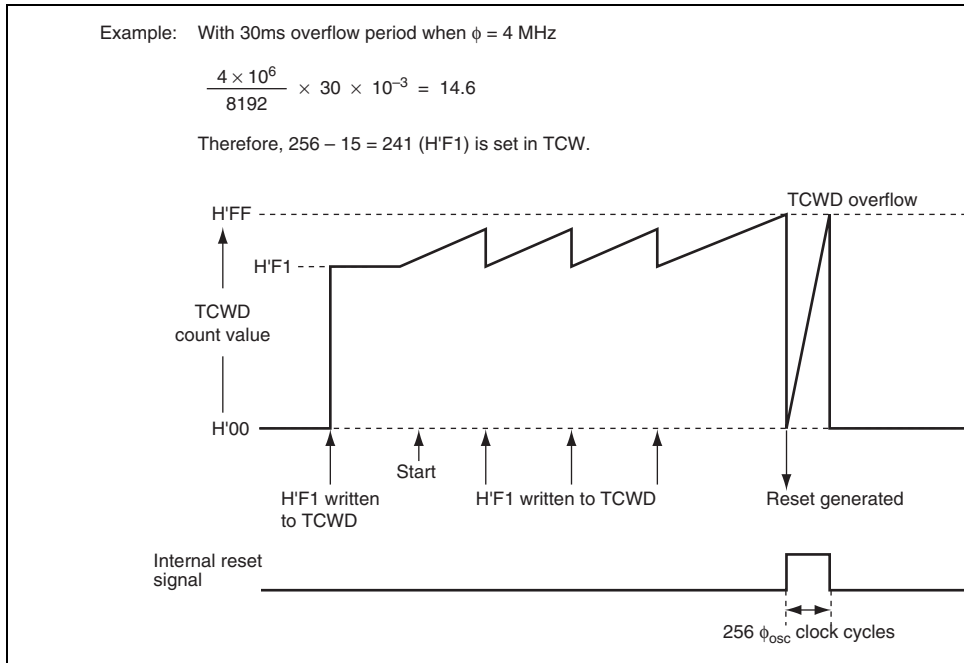


Figure 15.2 Watchdog Timer Operation Example

- Pulse division method for less ripple

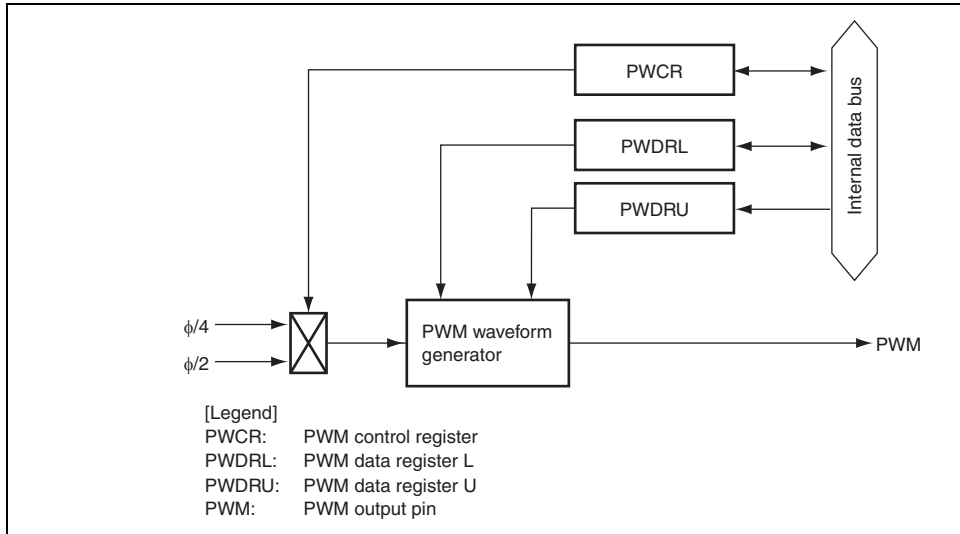


Figure 16.1 Block Diagram of 14-Bit PWM

16.2 Input/Output Pin

Table 16.1 shows the 14-bit PWM pin configuration.

Table 16.1 Pin Configuration

Name	Abbreviation	I/O	Function
14-bit PWM square-wave output	PWM	Output	14-bit PWM square-wave

PWCR selects the conversion period.

Bit	Bit Name	Initial Value	R/W	Description
7	—	1	—	Reserved
6	—	1	—	These bits are always read as 1, and cannot be modified.
5	—	1	—	
4	—	1	—	
3	—	1	—	
2	—	1	—	
1	—	1	—	
0	PWCR0	0	R/W	Clock Select 0: The input clock is $\phi/2$ ($t_{\phi} = 2/\phi$) — The conversion period is $16384/\phi$, with minimum modulation width of $1/\phi$ 1: The input clock is $\phi/4$ ($t_{\phi} = 4/\phi$) — The conversion period is $32768/\phi$, with minimum modulation width of $2/\phi$

[Legend] t_{ϕ} : Period of PWM clock input

PWDRU and PWDRL are initialized to H'C000.

16.4 Operation

When using the 14-bit PWM, set the registers in this sequence:

1. Set the PWM bit in the port mode register 1 (PMR1) to set the P11/PWM pin to function as a PWM output pin.
2. Set the PWCR0 bit in PWCR to select a conversion period of either.
3. Set the output waveform data in PWDRU and PWDRL. Be sure to write byte data first to PWDRL and then to PWDRU. When the data is written in PWDRU, the contents of both registers are latched in the PWM waveform generator, and the PWM waveform generation data is updated in synchronization with internal signals.

One conversion period consists of 64 pulses, as shown in figure 16.2. The total high-level time during this period (T_H) corresponds to the data in PWDRU and PWDRL. This relation can be expressed as follows:

$$T_H = (\text{data value in PWDRU and PWDRL} + 64) \times t\phi/2$$

where $t\phi$ is the period of PWM clock input: $2/\phi$ (bit PWCR0 = 0) or $4/\phi$ (bit PWCR0 = 1). If the data value in PWDRU and PWDRL is from H'FFC0 to H'FFFF, the PWM output is high. When the data value is H'C000, T_H is calculated as follows:

$$T_H = 64 \times t\phi/2 = 32 t\phi$$

Table 17.1 shows the SCI3 channel configuration and figure 17.1 shows a block diagram of SCI3. Since basic pin functions are identical for each of the three channels (SCI3, SCI3_1, SCI3_2, SCI3_3), separate explanations are not given in this section.

17.1 Features

- Choice of asynchronous or clocked synchronous serial communication mode
- Full-duplex communication capability
The transmitter and receiver are mutually independent, enabling transmission and reception to be executed simultaneously.
Double-buffering is used in both the transmitter and the receiver, enabling continuous 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 source
- Six interrupt sources
Transmit-end, transmit-data-empty, receive-data-full, overrun error, framing error, and parity error.
- Noise canceller (only for SCI3_3)

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 case of a framing error

			SCR3	H'FFFFAA	
			TDR	H'FFFFAB	
			SSR	H'FFFFAC	
			RDR	H'FFFFAD	
			RSR	—	
			TSR	—	
Channel 2	SCI3_2	SCK3_2	SMR_2	H'FFF740	Non
		RXD_2	BRR_2	H'FFF741	
		TXD_2	SCR3_2	H'FFF742	
			TDR_2	H'FFF743	
			SSR_2	H'FFF744	
			RDR_2	H'FFF745	
			RSR_2	—	
			TSR_2	—	
Channel 3	SCI3_3	SCK3_3	SMR_3	H'FFF600	Yes
		RXD_3	BRR_3	H'FFF601	
		TXD_3	SCR3_3	H'FFF602	
			TDR_3	H'FFF603	
			SSR_3	H'FFF604	
			RDR_3	H'FFF605	
			RSR_3	—	
			TSR_3	—	
			SMCR_3* ¹	H'FFF608	

When COM in SMR is cleared to 0, bit is set to 1, noise in the RXD_3 input is taken.

1	TXD_3	0	R/W	TXD_3 Pin Select Selects P92/TXD_3 pin function. 0: General input pin is selected 1: TXD_3 output pin is selected
0	MSTS3_3	0	R/W	SCI3_3 Module Standby When this bit is set to 1, SCI3_3 enters standby state.

- Noise canceller

The RXD_3 input signal is loaded internally via the noise canceller. The noise canceller consists of three latch circuits and match detection circuit connected in series. The RXD_3 input signal is sampled on the basic clock with a frequency 16 times the baud rate transfer rate, and the level is passed forward to the next circuit when outputs of all three latches match. When the outputs are not match, previous value is retained. In other words, when the same level is retained more than three clocks, the input signal is acknowledged as a signal. When the level is changed within three clocks, the signal is acknowledged as not a signal change but noise.

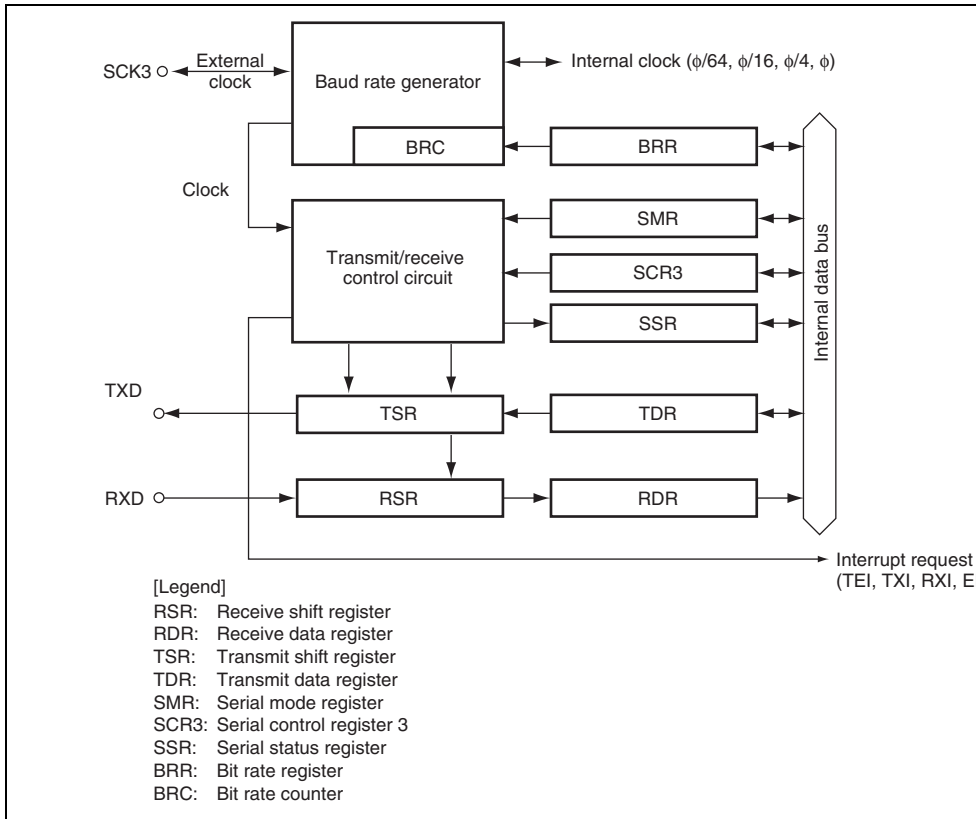


Figure 17.1 Block Diagram of SCI3

17.3 Register Descriptions

The 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)
- Serial mode control register 3 (SMCR3)

17.3.1 Receive Shift Register (RSR)

RSR is a shift register that is used to receive serial data input from the RxD pin and convert it to parallel data. When one frame of data has been received, it is transferred to RDR automatically. RSR cannot be directly accessed by the CPU.

17.3.2 Receive Data Register (RDR)

RDR is an 8-bit register that stores received data. When the SCI3 has received one frame of data, it transfers the received serial data from RSR to RDR, where it is stored. After this transfer, RDR is receive-enabled. As RSR and RDR function as a double buffer in this way, continuous receive 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.

data has already been written to TDR during transmission of one-frame data, the SCI3 transmits the written data to TSR to continue transmission. To achieve reliable serial transmission, transmit data to TDR only once after confirming that the TDRE bit in SSR is set to 1. TDR is initialized to H'FF.

17.3.5 Serial Mode Register (SMR)

SMR is used to set the SCI3's serial transfer format and select the baud rate generator clock source.

Bit	Bit Name	Initial Value	R/W	Description
7	COM	0	R/W	Communication Mode 0: Asynchronous mode 1: Clocked synchronous mode
6	CHR	0	R/W	Character Length (enabled only in asynchronous mode) 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 mode) When this bit is set to 1, the parity bit is added to the data to be transmitted and checked in reception.
4	PM	0	R/W	Parity Mode (enabled only when the PE bit is 1 in asynchronous mode) 0: Selects even parity. 1: Selects odd parity.

Bit	Bit Name	Initial Value	Access	Description
2	MP	0	R/W	Multiprocessor mode When this bit is set to 1, the multiprocessor communication function is enabled. The PE bit settings are invalid in multiprocessor mode. In clocked synchronous mode, clear this bit to 0.
1	CKS1	0	R/W	Clock Select 0 and 1
0	CKS0	0	R/W	These bits select the clock source for the baud rate generator. 00: ϕ clock (n = 0) 01: $\phi/4$ clock (n = 1) 10: $\phi/16$ clock (n = 2) 11: $\phi/64$ clock (n = 3) For the relationship between the bit rate register and the baud rate, see section 17.3.8, Bit Rate Register (BRR). n is the decimal representation of the value in BRR (see section 17.3.8, Bit Rate Register).

6	RIE	0	R/W	Receive Interrupt Enable When this bit is set to 1, RXI and ERI interrupt are enabled.
5	TE	0	R/W	Transmit Enable When this bit is set to 1, transmission is enabled.
4	RE	0	R/W	Receive Enable When this bit is set to 1, reception is enabled.
3	MPIE	0	R/W	Multiprocessor Interrupt Enable (enabled only if MP bit in SMR is 1 in asynchronous mode) When this bit is set to 1, receive data in which the multiprocessor bit is 0 is skipped, and setting of RDRF, FER, and OER status flags in SSR is disabled. On receiving data in which the multiprocessor bit is 0, this bit is automatically cleared and normal reception is resumed. For details, see section 17.6, Multiprocessor Communication Function.
2	TEIE	0	R/W	Transmit End Interrupt Enable When this bit is set to 1, TEI interrupt request is enabled.

Inputs a clock with a frequency 16 times that of 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

17.3.7 Serial Status Register (SSR)

SSR is a register containing status flags of the SCI3 and multiprocessor bits for transfer. Only the bits that can be written to flags TDRE, RDRF, OER, PER, and FER; they can only be cleared.

Bit	Bit Name	Initial Value	R/W	Description
7	TDRE	1	R/W	Transmit Data Register Empty Indicates whether TDR contains transmit data. [Setting conditions] <ul style="list-style-type: none">• When the TE bit in SCR3 is 0• When data is transferred from TDR to TSI [Clearing conditions] <ul style="list-style-type: none">• When 0 is written to TDRE after reading TDR• When the transmit data is written to TDR

5	OER	0	R/W	<p>Overrun Error</p> <p>[Setting condition]</p> <ul style="list-style-type: none"> When an overrun error occurs in reception <p>[Clearing condition]</p> <ul style="list-style-type: none"> When 0 is written to OER after reading OER
4	FER	0	R/W	<p>Framing Error</p> <p>[Setting condition]</p> <ul style="list-style-type: none"> When a framing error occurs in reception <p>[Clearing condition]</p> <ul style="list-style-type: none"> When 0 is written to FER after reading FER
3	PER	0	R/W	<p>Parity Error</p> <p>[Setting condition]</p> <ul style="list-style-type: none"> When a parity error is detected during reception <p>[Clearing condition]</p> <ul style="list-style-type: none"> When 0 is written to PER after reading PER
2	TEND	1	R	<p>Transmit End</p> <p>[Setting conditions]</p> <ul style="list-style-type: none"> When the TE bit in SCR3 is 0 When TDRE = 1 at transmission of the last 1-frame serial transmit character <p>[Clearing conditions]</p> <ul style="list-style-type: none"> When 0 is written to TDRE after reading TDRE When the transmit data is written to TDR

17.3.8 Bit Rate Register (BRR)

BRR is an 8-bit register that adjusts the bit rate. The initial value of BRR is H'FF. Table shows the relationship between the N setting in BRR and the n setting in bits CKS1 and SMR in asynchronous mode. Table 17.4 shows the maximum bit rate for each frequency asynchronous mode. The values shown in both tables 17.3 and 17.4 are values in active (speed) mode. Table 17.5 shows the relationship between the N setting in BRR and the n bits CKS1 and CKS0 of SMR in clocked synchronous mode. The values shown in table values in active (high-speed) mode. The N setting in BRR and error for other operating frequencies and bit rates can be obtained by the following formulas:

[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	31	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

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

2400	0	103	0.16	0	127	0.00	0	129	0.16	0	13
4800	0	51	0.16	0	63	0.00	0	64	0.16	0	77
9600	0	25	0.16	0	31	0.00	0	32	-1.36	0	38
19200	0	12	0.16	0	15	0.00	0	15	1.73	0	19
31250	0	7	0.00	0	9	-1.70	0	9	0.00	0	11
38400	0	6	-6.99	0	7	0.00	0	7	1.73	0	9

[Legend]

—: A setting is available but error occurs.

1200	1	79	0.00	1	90	0.16	1	95	0.00	1	103
2400	0	159	0.00	0	181	0.16	0	191	0.00	0	207
4800	0	79	0.00	0	90	0.16	0	95	0.00	0	103
9600	0	39	0.00	0	45	-0.93	0	47	0.00	0	51
19200	0	19	0.00	0	22	-0.93	0	23	0.00	0	25
31250	0	11	2.40	0	13	0.00	0	14	-1.70	0	15
38400	0	9	0.00	—	—	—	0	11	0.00	0	12

2400	0	233	0.16	1	64	0.16
4800	0	116	0.16	0	129	0.16
9600	0	58	-0.96	0	64	0.16
19200	0	28	1.02	0	32	-1.36
31250	0	17	0.00	0	19	0.00
38400	0	14	-2.34	0	15	1.73

[Legend]

—: A setting is available but error occurs.

Table 17.4 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	8	250000	0
2.097152	65536	0	0	9.8304	307200	0
2.4576	76800	0	0	10	312500	0
3	93750	0	0	12	375000	0
3.6864	115200	0	0	12.288	384000	0
4	125000	0	0	14	437500	0
4.9152	153600	0	0	14.7456	460800	0
5	156250	0	0	16	500000	0
6	187500	0	0	17.2032	537600	0
6.144	192000	0	0	18	562500	0
7.3728	230400	0	0	20	625000	0

10k	0	49	0	99	0	199	0	249	1
25k	0	19	0	39	0	79	0	99	0
50k	0	9	0	19	0	39	0	49	0
100k	0	4	0	9	0	19	0	24	0
250k	0	1	0	3	0	7	0	9	0
500k	0	0*	0	1	0	3	0	4	0
1M			0	0*	0	1	—	—	0
2M					0	0*	—	—	0
2.5M							0	0*	—
4M									0

[Legend]

Blank: No setting is available.

—: A setting is available but error occurs.

*: Continuous transfer is not possible.

5k	1	224	1	249
10k	1	112	1	124
25k	0	179	0	199
50k	0	89	0	99
100k	0	44	0	49
250k	0	17	0	19
500k	0	8	0	9
1M	0	4	0	4
2M	—	—	—	—
2.5M	—	—	0	1
4M	—	—	—	—

[Legend]

Blank: No setting is available.

—: A setting is available but error occurs.

*: Continuous transfer is not possible.

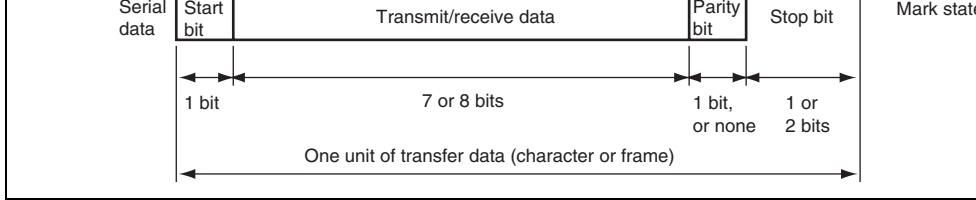


Figure 17.2 Data Format in Asynchronous Communication

17.4.1 Clock

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

When the SCI3 is operated on an internal clock, the clock can be output from the SCK3 pin. In this case, the frequency of the clock output is equal to the bit rate, and the phase is such that the rising edge of the clock is in the middle of the transmit data, as shown in figure 17.3.

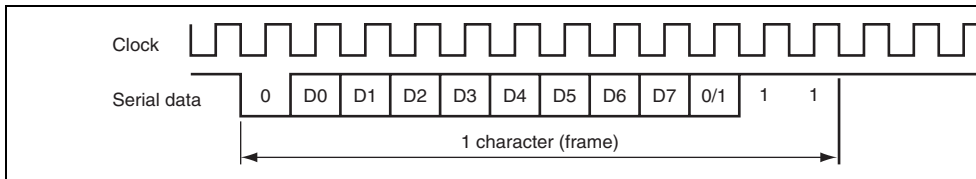


Figure 17.3 Relationship between Output Clock and Transfer Data Phase (Asynchronous Mode) (Example with 8-Bit Data, Parity, Two Stop Bits)

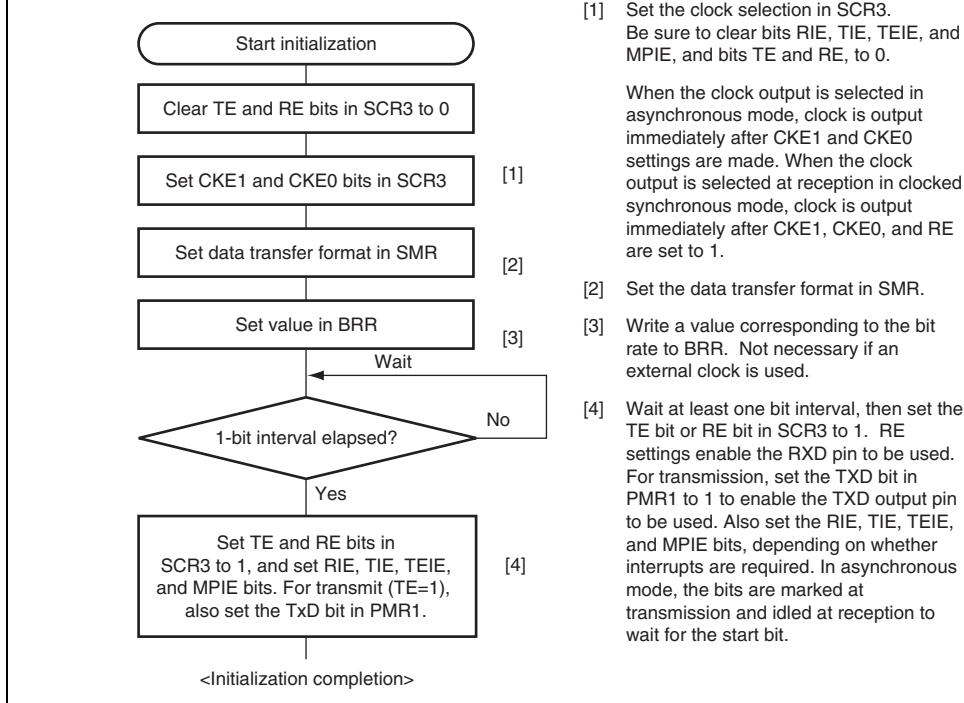


Figure 17.4 Sample SCI3 Initialization Flowchart

3. The 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, and 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 the “idle state” is entered, in which 1 is output. If the TEIE bit in SCR3 is set to 1 at this time, an interrupt request is generated.
6. Figure 17.6 shows a sample flowchart for transmission in asynchronous mode.

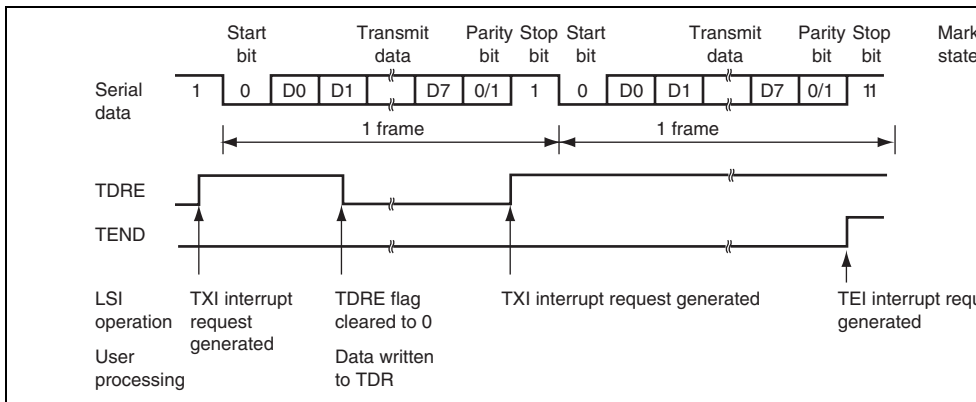


Figure 17.5 Example of SCI3 Transmission in Asynchronous Mode (8-Bit Data, Parity, One Stop Bit)

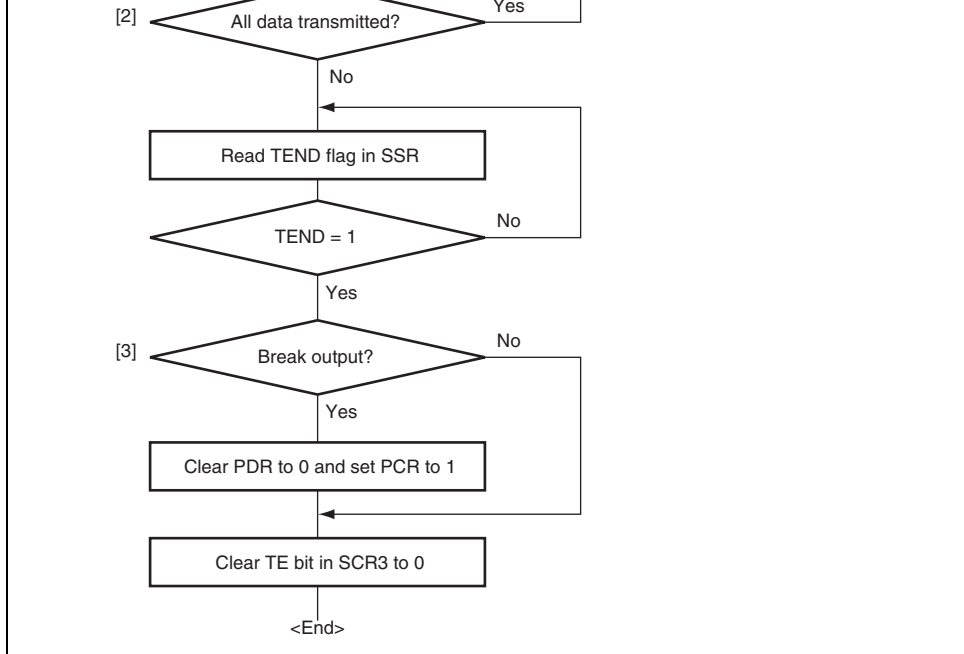


Figure 17.6 Sample Serial Transmission Data Flowchart (Asynchronous Mode)

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

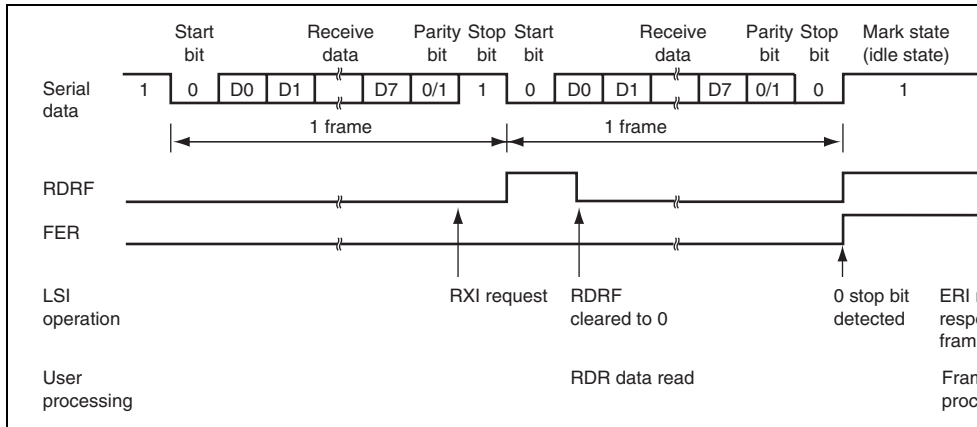
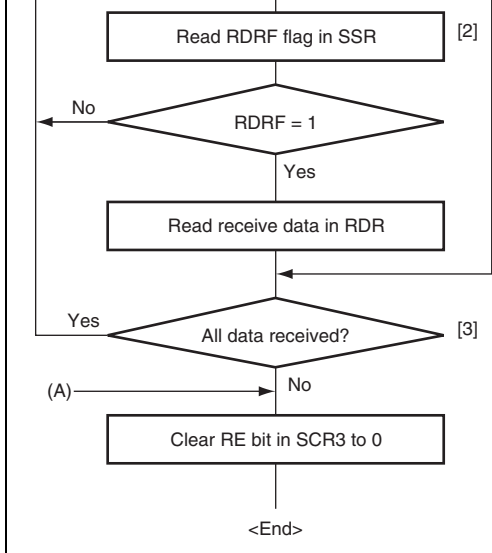


Figure 17.7 Example of SCI3 Reception in Asynchronous Mode (8-Bit Data, Parity, One Stop Bit)

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 + fram
1	1	0	1	Lost	Overrun error + parit
0	0	1	1	Transferred to RDR	Framing error + parit
1	1	1	1	Lost	Overrun error + fram parity error

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



[4] If a receive error occurs, read the OER, FER, and RDRF flags in SSR to identify the error. After performing the appropriate error processing, ensure that the OER, FER, and RDRF flags are all cleared to 0. Reception cannot be resumed if any of these flags is set to 1. In the case of a framing error, a break can be detected by reading the status of the input port corresponding to the pin.

Figure 17.8 Sample Serial Reception Data Flowchart (Asynchronous Mode)

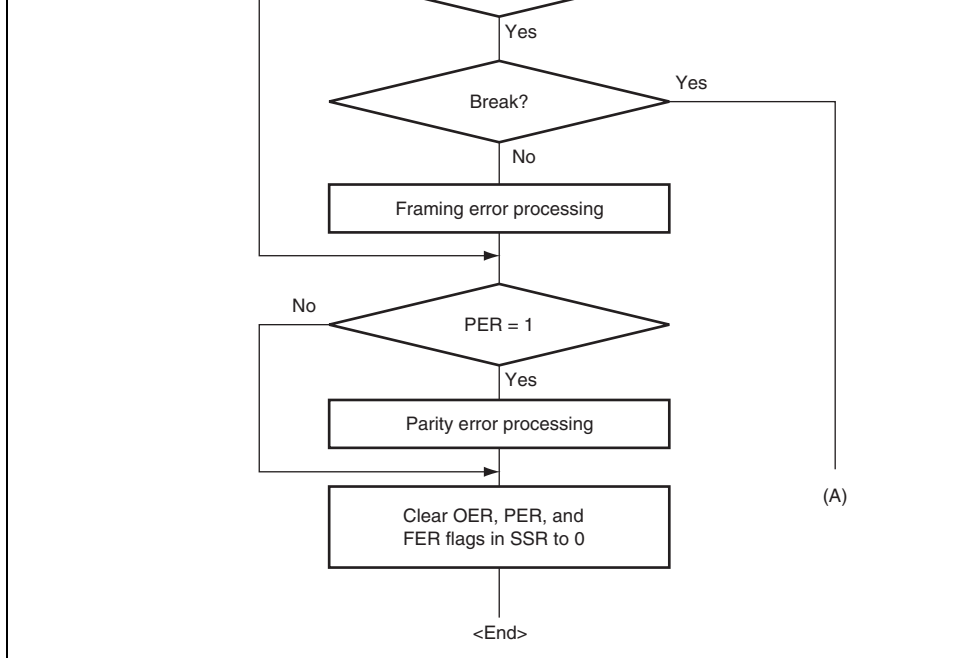


Figure 17.8 Sample Serial Reception Data Flowchart (Asynchronous Mode)

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

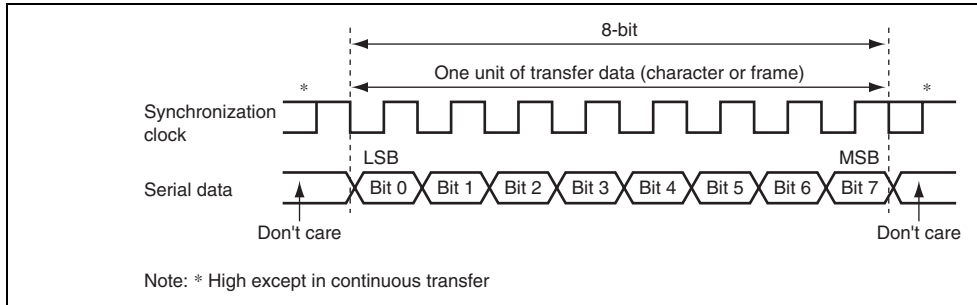


Figure 17.9 Data Format in Clocked Synchronous Communication

17.5.1 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 the bit in SMR and CKE0 and CKE1 bits in SCR3. When the SCI3 is operated on an internal clock, the synchronization clock is output from the SCK3 pin. Eight synchronization clock pulses are output in the transfer of one character, and when no transfer is performed the clock is fixed at a high level.

17.5.2 SCI3 Initialization

Before transmitting and receiving data, the SCI3 should be initialized as described in a flowchart in figure 17.4.

mode has been specified, and synchronized with the input clock when use of an external clock pin.

4. The 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 transmission of the next frame is started.
6. If the TDRE flag is set to 1, the TEND flag in SSR is set to 1, and the TDRE flag matches the output state of the last bit. If the TEIE bit in SCR3 is set to 1 at this time, a TEI interrupt request is generated.
7. The SCK3 pin is fixed high at the end of transmission.

Figure 17.11 shows a sample flow chart for serial data transmission. Even if the TDRE flag is cleared to 0, transmission will not start while a receive error flag (OER, FER, or PER) is set. Make sure that the receive error flags are cleared to 0 before starting transmission.

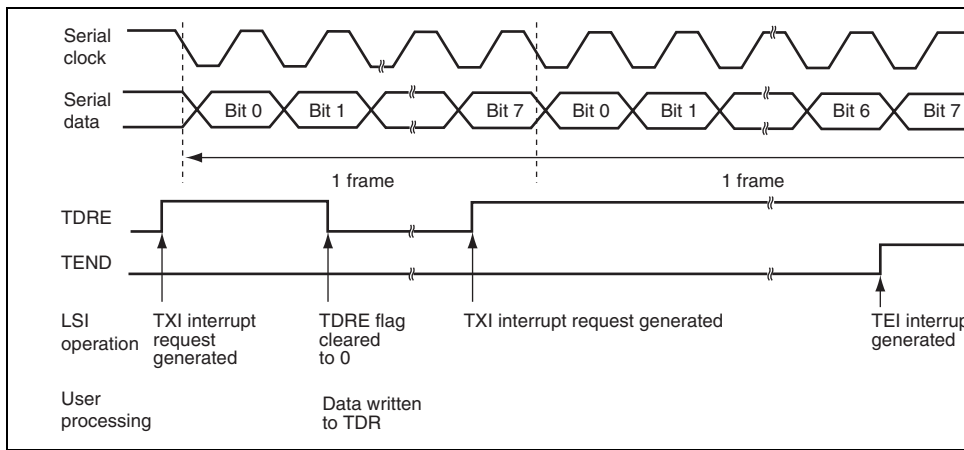


Figure 17.10 Example of SCI3 Transmission in Clocked Synchronous Mode

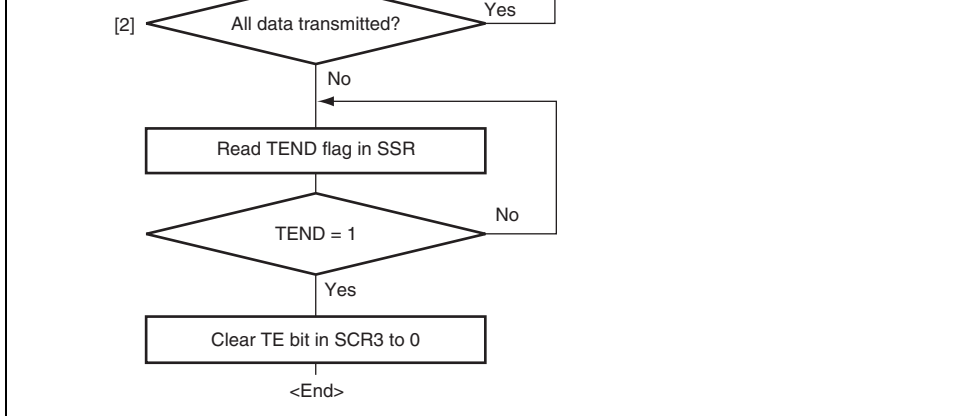


Figure 17.11 Sample Serial Transmission Flowchart (Clocked Synchronous M

time, an ERI interrupt request is generated, receive data is not transferred to RDR, and 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 data is transferred to RDR. If the RIE bit in SCR3 is set to 1 at this time, an RXI interrupt is generated.

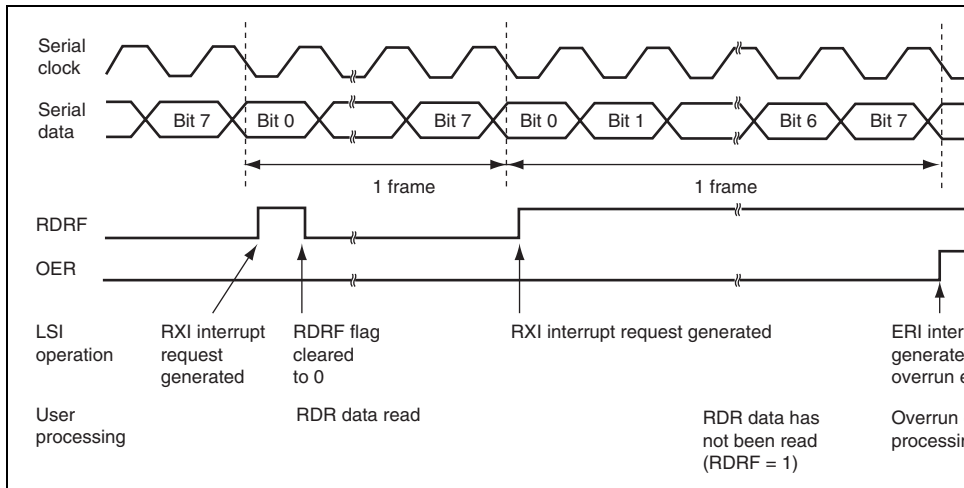


Figure 17.12 Example of SCI3 Reception in Clocked Synchronous Mode

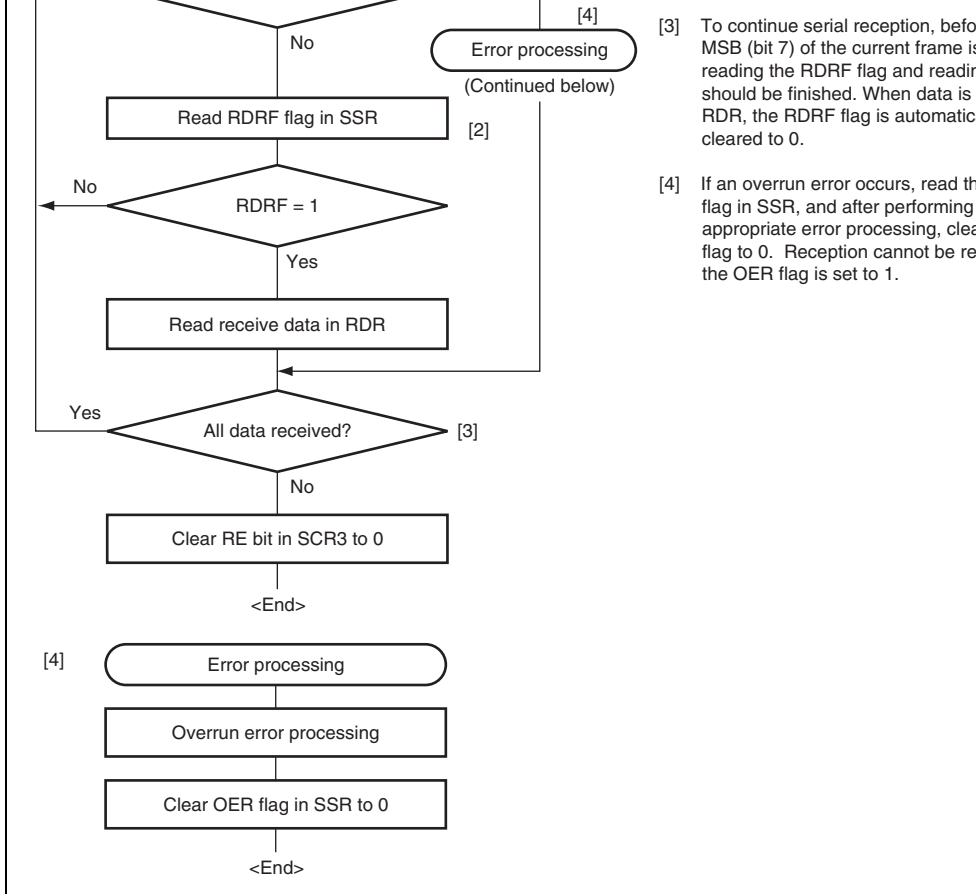


Figure 17.13 Sample Serial Reception Flowchart (Clocked Synchronous Mode)

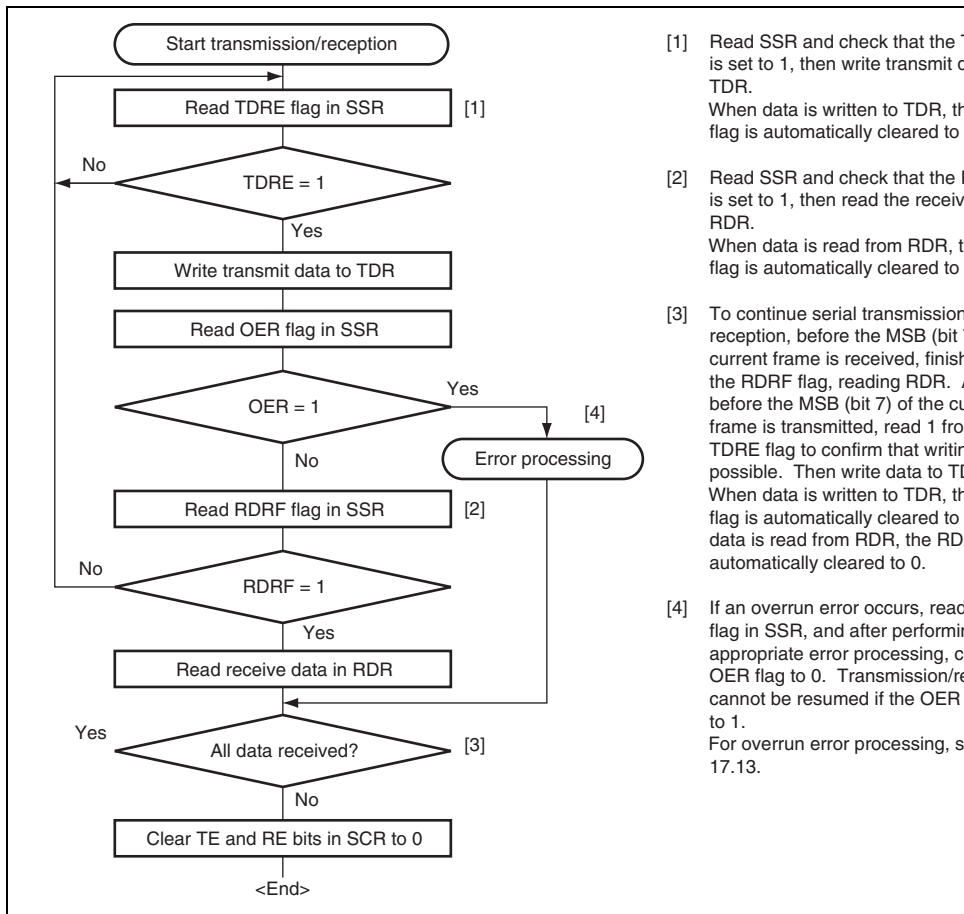


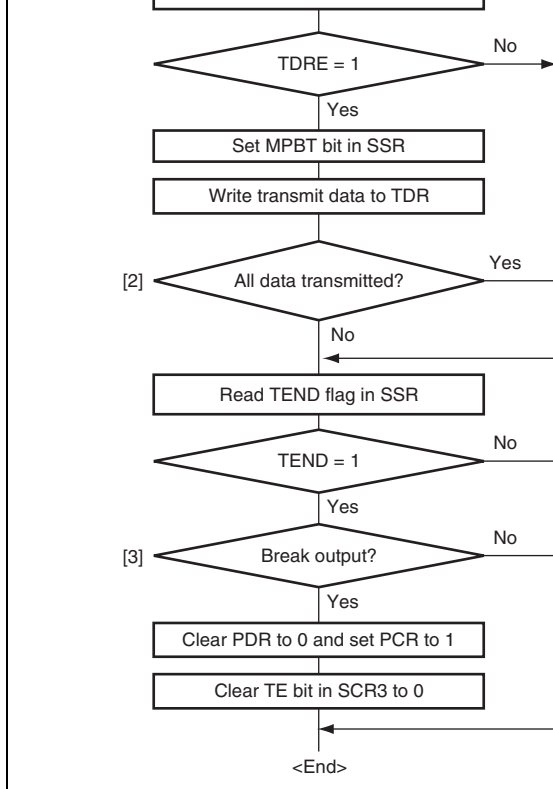
Figure 17.14 Sample Flowchart of Simultaneous Serial Transmit and Receive Operation (Clocked Synchronous Mode)

cycle is a data transmission cycle. Figure 17.15 shows an example of inter-processor communication using the multiprocessor format. The transmitting station first sends the ID of the receiving station with which it wants to perform serial communication as data with a 0 multiprocessor bit added. It then sends transmit data as data with a 0 multiprocessor bit added. When data with a 1 multiprocessor bit is received, the receiving station compares that data with its own ID. The station whose ID matches then receives the data sent next. Stations whose ID does not match continue to skip data until data with a 1 multiprocessor bit is again received.

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

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

**Figure 17.15 Example of Inter-Processor Communication Using Multiprocessor
(Transmission of Data H'AA to Receiving Station A)**



TDR, the TDRE flag is automatically cleared to 0.

[2] To continue serial transmission, be sure to read 1 from the TDRE flag to confirm that writing is possible, then write data to TDR. When data is written to TDR, the TDRE flag is automatically cleared to 0.

[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 17.16 Sample Multiprocessor Serial Transmission Flowchart

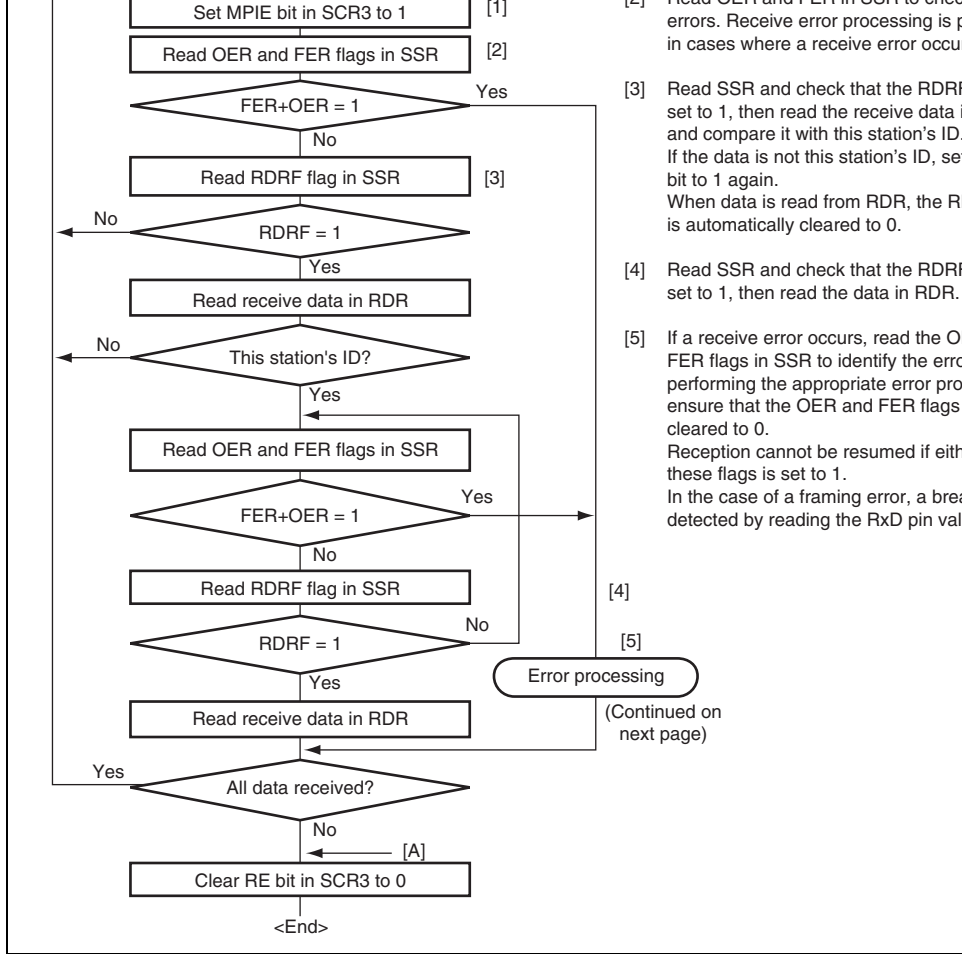


Figure 17.17 Sample Multiprocessor Serial Reception Flowchart (1)

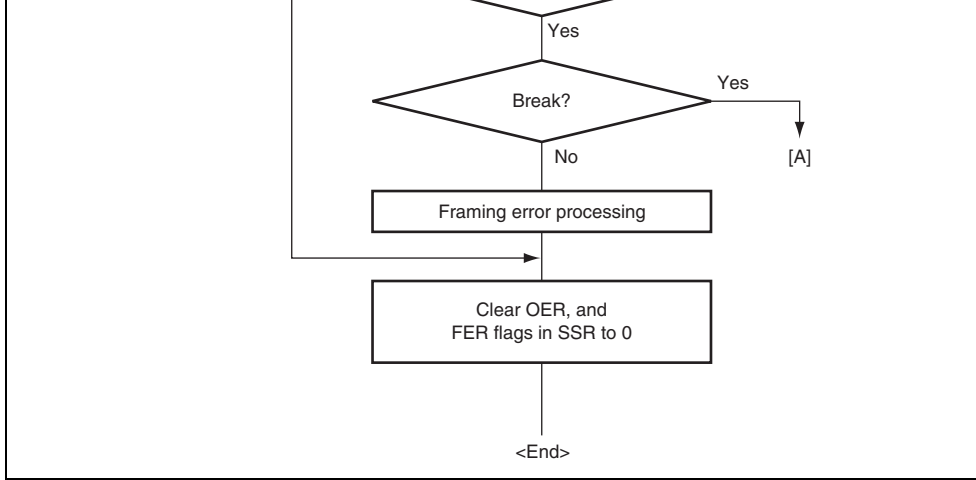


Figure 17.17 Sample Multiprocessor Serial Reception Flowchart (2)

LSI operation
 User processing

RXI interrupt request
 MPIE cleared to 0

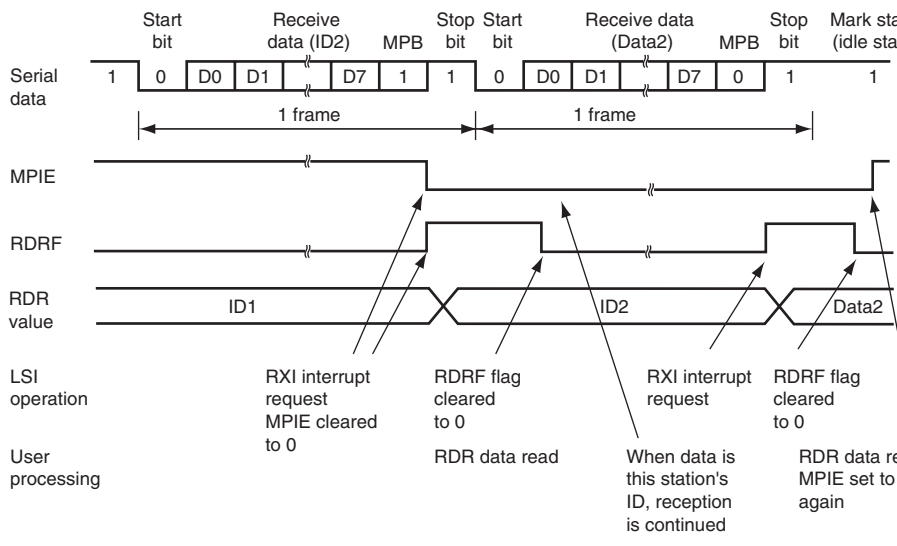
RDRF flag cleared to 0

RDR data read

When data is not this station's ID, MPIE is set to 1 again

RXI interrupt is not generated
 RDR retains its value

(a) When data does not match this receiver's ID



(b) When data matches this receiver's ID

**Figure 17.18 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 1 before transferring the transmit data to TDR, a TXI interrupt request is generated even if the transmit data is not ready. The initial value of the TEND flag in SSR is 1. Thus, when the TEIE bit in SCR3 is set to 1 before transferring the transmit data to TDR, a TEI interrupt request is generated even if the transmit data has not been sent. It is possible to make use of the most of these interrupt requests efficiently by transferring the transmit data to TDR in the interrupt routine. To prevent the generation of these interrupt requests (TXI and TEI), set the enable bits (TIE and TEIE) in SCR3 correspond to these interrupt requests to 0, after transferring the transmit data to TDR.

When TE is 0, the TxD pin is used as an I/O port whose direction (input or output) and level are determined by PCR and PDR. This can be used to set the TxD pin to mark state (high level) to send a break during serial data transmission. To maintain the communication line at mark state until TE is set to 1, set both PCR and PDR to 1. As TE is cleared to 0 at this point, the TxD pin becomes an I/O port, and 1 is output from the TxD pin. To send a break during serial data transmission, first set PCR to 1 and clear PDR to 0, and then clear TE to 0. When TE is cleared to 0, the transmitter is initialized regardless of the current transmission state, the TxD pin becomes an I/O port, and 0 is output from the TxD pin.

17.8.3 Receive Error Flags and Transmit Operations (Clocked Synchronous Mode)

Transmission cannot be started when a receive error flag (OER, PER, or FER) is set to 1. Before starting transmission, 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 set 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) = 0.5, 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 allowed in system design.

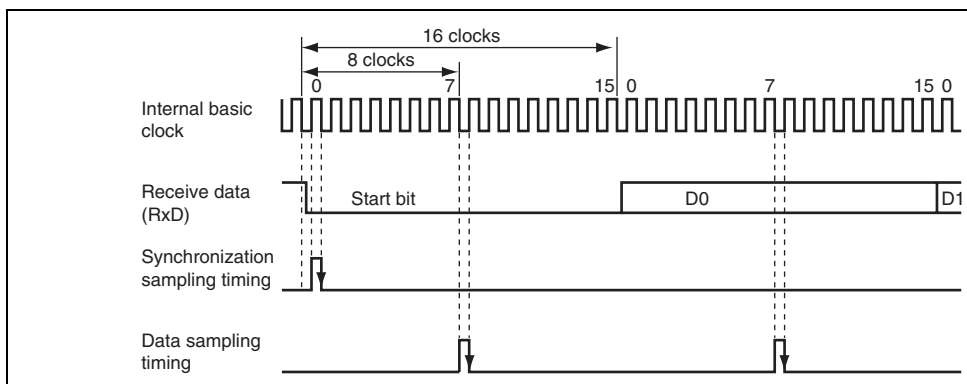


Figure 17.19 Receive Data Sampling Timing in Asynchronous Mode

16.1 Features

- Selection of I²C format or clocked synchronous serial format
- Continuous transmission/reception

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

I²C 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 are completed.

- Six interrupt sources

Transmit data empty (including slave-address match), transmit end, receive data full (including 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 control function is selected.

Clocked synchronous format

- Four interrupt sources

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

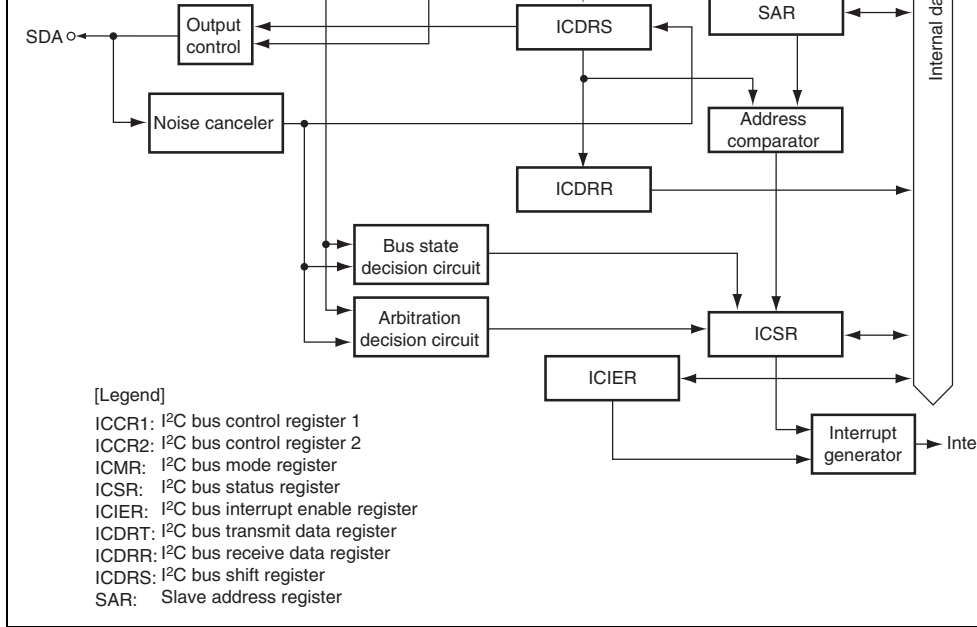


Figure 18.1 Block Diagram of I²C Bus Interface 2

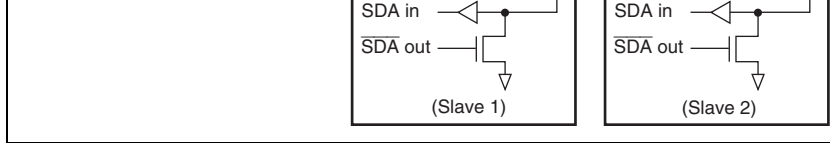


Figure 18.2 External Circuit Connections of I/O Pins

18.2 Input/Output Pins

Table 18.1 summarizes the input/output pins used by the I²C bus interface 2.

Table 18.1 Pin Configuration

Name	Abbreviation	I/O	Function
Serial clock	SCL	I/O	IIC serial clock input/output
Serial data	SDA	I/O	IIC serial data input/output

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

18.3.1 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 mode.

Bit	Bit Name	Initial Value	R/W	Description
7	ICE	0	R/W	I ² C Bus Interface Enable 0: This module is halted. (SCL and SDA pins are in port function.) 1: This bit is enabled for transfer operations. (SCL and SDA pins are bus drive state.)
6	RCVD	0	R/W	Reception Disable This bit enables or disables the next operation when TRS is 0 and ICDRR is read. 0: Enables next reception 1: Disables next reception

overflow error occurs in master mode with the synchronous serial format, MST is cleared to slave receive mode is entered.

Operating modes are described below according to MST and TRS combination. When clocked synchronous serial format is selected and MST is 1, clock is

00: Slave receive mode

01: Slave transmit mode

10: Master receive mode

11: Master transmit mode

3	CKS3	0	R/W	Transfer Clock Select 3 to 0
2	CKS2	0	R/W	These bits should be set according to the needed transfer rate (see table 18.2) in master mode.
1	CKS1	0	R/W	In slave mode, these bits are used for reservation of the time in transmit mode. The time is $10 t_{cyc}$ when CKS1 = 0 and $20 t_{cyc}$ when CKS1 = 1.
0	CKS0	0	R/W	

			1	0	$\phi/112$	44.6 kHz	71.4 kHz	89.3 kHz	143 kHz	
				1	$\phi/128$	39.1 kHz	62.5 kHz	78.1 kHz	125 kHz	
1	0	0	0	0	$\phi/56$	89.3 kHz	143 kHz	179 kHz	286 kHz	
				1	$\phi/80$	62.5 kHz	100 kHz	125 kHz	200 kHz	
			1	0	$\phi/96$	52.1 kHz	83.3 kHz	104 kHz	167 kHz	
				1	$\phi/128$	39.1 kHz	62.5 kHz	78.1 kHz	125 kHz	
	1	0	0	0	$\phi/160$	31.3 kHz	50.0 kHz	62.5 kHz	100 kHz	
				1	$\phi/200$	25.0 kHz	40.0 kHz	50.0 kHz	80.0 kHz	
			1	0	$\phi/224$	22.3 kHz	35.7 kHz	44.6 kHz	71.4 kHz	8
				1	$\phi/256$	19.5 kHz	31.3 kHz	39.1 kHz	62.5 kHz	7

format, this bit has no meaning. With the I²C bit, this bit is set to 1 when the SDA level changes from high to low under the condition of SCL = high, indicating that the start condition has been issued. This bit is cleared to 0 when the SDA level changes from low to high under the condition of SCL = high, assuming the stop condition has been issued. Write 1 to BBSY and 0 to SCP to issue a start condition. Follow the same procedure when also re-transmitting a start condition. Write 0 in BBSY and 0 in SCP to issue a stop condition. To issue start/stop conditions, use the MOV instruction.

6	SCP	1	W	<p>Start/Stop Issue Condition Disable</p> <p>The SCP bit controls the issue of start/stop conditions in master mode.</p> <p>To issue a start condition, write 1 in BBSY and 0 in SCP. A retransmit start condition is issued in the same way. To issue a stop condition, write 0 in BBSY and 0 in SCP. This bit is always read as 1. If 1 is written, the data is not stored.</p>
5	SDAO	1	R/W	<p>SDA Output Value Control</p> <p>This bit is used with SDAOP when modifying the output level of SDA. This bit should not be manipulated during data transfer.</p> <p>0: When reading, SDA pin outputs low. When writing, SDA pin is changed to output high.</p> <p>1: When reading, SDA pin outputs high. When writing, SDA pin is changed to output low. (outputs high by external pull-up resistance)</p>

2	—	1	—	Reserved
				This bit is always read as 1.
1	IICRST	0	R/W	IIC Control Part Reset
				This bit resets the control part except for I ² C re this bit is set to 1 when hang-up occurs because communication failure during I ² C operation, I ² C part can be reset without setting ports and initial registers.
0	—	1	—	Reserved
				This bit is always read as 1.

18.3.3 I²C Bus Mode Register (ICMR)

ICMR selects whether the MSB or LSB is transferred first, performs master mode wait count, 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 I ² C bus format is used.

5	—	1	—	Reserved																		
4	—	1	—	These bits are always read as 1.																		
3	BCWP	1	R/W	<p>BC Write Protect</p> <p>This bit controls the BC2 to BC0 modifications. When modifying BC2 to BC0, this bit should be cleared and use the MOV instruction. In clock synchronous serial mode, BC should not be modified.</p> <p>0: When writing, values of BC2 to BC0 are set to 0.</p> <p>1: When reading, 1 is always read.</p> <p>When writing, settings of BC2 to BC0 are invalid.</p>																		
2	BC2	0	R/W	Bit Counter 2 to 0																		
1	BC1	0	R/W	<p>These bits specify the number of bits to be transferred next. When read, the remaining number of bits to be transferred is indicated. With the I²C bus format, the data is transferred with one additional acknowledge bit. BC2 to BC0 settings should be made during an interrupt between transfer frames. If bits BC2 to BC0 are set to a value other than 000, the setting should be maintained when the SCL pin is low. The value returns to 000 at the start of a data transfer, including the acknowledge bit. In the clock synchronous serial format, these bits should not be modified.</p> <table border="0"> <tr> <td>I²C Bus Format</td> <td>Clock Synchronous Serial Mode</td> </tr> <tr> <td>000: 9 bits</td> <td>000: 8 bits</td> </tr> <tr> <td>001: 2 bits</td> <td>001: 1 bits</td> </tr> <tr> <td>010: 3 bits</td> <td>010: 2 bits</td> </tr> <tr> <td>011: 4 bits</td> <td>011: 3 bits</td> </tr> <tr> <td>100: 5 bits</td> <td>100: 4 bits</td> </tr> <tr> <td>101: 6 bits</td> <td>101: 5 bits</td> </tr> <tr> <td>110: 7 bits</td> <td>110: 6 bits</td> </tr> <tr> <td>111: 8 bits</td> <td>111: 7 bits</td> </tr> </table>	I ² C Bus Format	Clock Synchronous Serial Mode	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
I ² C Bus Format	Clock Synchronous Serial Mode																					
000: 9 bits	000: 8 bits																					
001: 2 bits	001: 1 bits																					
010: 3 bits	010: 2 bits																					
011: 4 bits	011: 3 bits																					
100: 5 bits	100: 4 bits																					
101: 6 bits	101: 5 bits																					
110: 7 bits	110: 6 bits																					
111: 8 bits	111: 7 bits																					
0	BC0	0	R/W																			

disabled.
1: Transmit data empty interrupt request (TXI) enabled.

6	TEIE	0	R/W	<p>Transmit End Interrupt Enable</p> <p>This bit enables or disables the transmit end interrupt request (TEI) at the rising of the ninth clock while the TEI in ICSR is 1. TEI can be canceled by clearing the TXI bit or the TEIE bit to 0.</p> <p>0: Transmit end interrupt request (TEI) is disabled. 1: Transmit end interrupt request (TEI) is enabled.</p>
<hr/>				
5	RIE	0	R/W	<p>Receive Interrupt Enable</p> <p>This bit enables or disables the receive data full interrupt request (RXI) and the overrun error interrupt request (ERI) with the clocked synchronous format when a receive data is transferred from ICDRS to ICDRR and the RDRF bit in ICSR is set to 1. RXI and ERI can be canceled by clearing the RDRF or RIE bit to 0.</p> <p>0: Receive data full interrupt request (RXI) and the overrun error interrupt request (ERI) with the clocked synchronous format are disabled. 1: Receive data full interrupt request (RXI) and the overrun error interrupt request (ERI) with the clocked synchronous format are enabled.</p>

3	STIE	0	R/W	<p>Stop Condition Detection Interrupt Enable</p> <p>0: Stop condition detection interrupt request (disabled).</p> <p>1: Stop condition detection interrupt request (enabled).</p>
2	ACKE	0	R/W	<p>Acknowledge Bit Judgement Select</p> <p>0: The value of the receive acknowledge bit is 0 and continuous transfer is performed.</p> <p>1: If the receive acknowledge bit is 1, continuous transfer is halted.</p>
1	ACKBR	0	R	<p>Receive Acknowledge</p> <p>In transmit mode, this bit stores the acknowledge bits that are returned by the receive device. This bit cannot be modified.</p> <p>0: Receive acknowledge = 0</p> <p>1: Receive acknowledge = 1</p>
0	ACKBT	0	R/W	<p>Transmit Acknowledge</p> <p>In receive mode, this bit specifies the bit to be sent at the acknowledge timing.</p> <p>0: 0 is sent at the acknowledge timing.</p> <p>1: 1 is sent at the acknowledge timing.</p>

- When TRS is set
- When a start condition (including re-transmission) has been issued
- When transmit mode is entered from receive mode in slave mode

[Clearing conditions]

- When 0 is written in TDRE after reading TDRE
- When data is written to ICDRT with an instruction

6	TEND	0	R/W	Transmit End
---	------	---	-----	--------------

[Setting conditions]

- When the ninth clock of SCL rises with the start condition in I2C format while the TDRE flag is 1
- When the final bit of transmit frame is sent in I2C clock synchronous serial format

[Clearing conditions]

- When 0 is written in TEND after reading TEND
- When data is written to ICDRT with an instruction

5	RDRF	0	R/W	Receive Data Register Full
---	------	---	-----	----------------------------

[Setting condition]

- When a receive data is transferred from ICDRR to ICDRT

[Clearing conditions]

- When 0 is written in RDRF after reading RDRF
- When ICDRR is read with an instruction

3	STOP	0	R/W	<p>Stop Condition Detection Flag</p> <p>[Setting conditions]</p> <ul style="list-style-type: none"> • In master mode, when a stop condition is detected after frame transfer • In slave mode, when a stop condition is detected after the general call address or the first block address, next to detection of start condition with the address set in SAR <p>[Clearing condition]</p> <ul style="list-style-type: none"> • When 0 is written in STOP after reading S
---	------	---	-----	---

[Setting conditions]

- If the internal SDA and SDA pin disagree a of SCL in master transmit mode
- When the SDA pin outputs high in master r while a start condition is detected
- When the final bit is received with the clock synchronous format while RDRF = 1

[Clearing condition]

- When 0 is written in AL/OVE after reading AL/OVE=1

1	AAS	0	R/W	Slave Address Recognition Flag
---	-----	---	-----	--------------------------------

In slave receive mode, this flag is set to 1 if the frame following a start condition matches bits S SVA0 in SAR.

[Setting conditions]

- When the slave address is detected in slav mode
- When the general call address is detected receive mode.

[Clearing condition]

- When 0 is written in AAS after reading AAS

18.3.6 Slave Address Register (SAR)

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

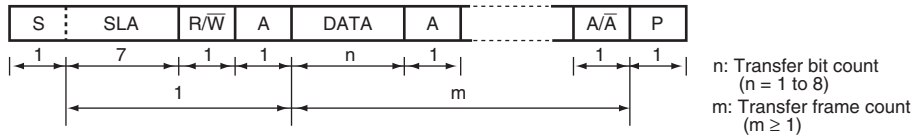
Bit	Bit Name	Initial Value	R/W	Description
7 to 1	SVA6 to SVA0	All 0	R/W	Slave Address 6 to 0 These bits set a unique address in bits SVA6 to SVA0, differing from the addresses of other slave devices connected to the I ² C bus.
0	FS	0	R/W	Format Select 0: I ² C bus format is selected. 1: Clocked synchronous 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. ICDRR is a receive-only register, therefore the CPU cannot write to this register. The initial value of ICDRR is 0xFF.

18.3.9 I²C Bus Shift Register (ICDRS)

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

(a) I²C bus format (FS = 0)



(b) I²C bus format (Start condition retransmission, FS = 0)

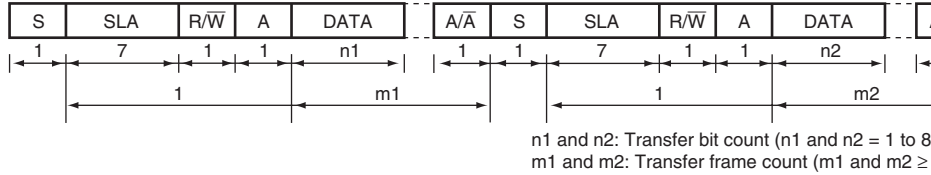


Figure 18.3 I²C Bus Formats

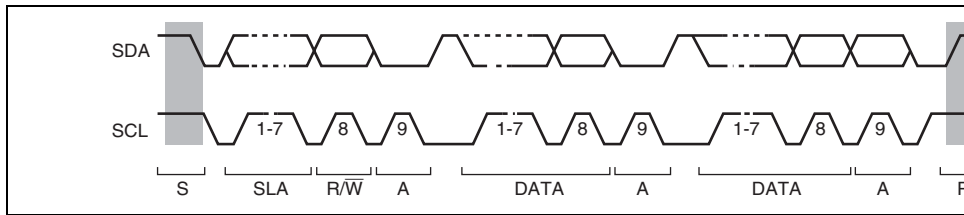


Figure 18.4 I²C Bus Timing

In master transmit mode, the master device outputs the transmit clock and transmit data, and the slave device returns an acknowledge signal. For master transmit mode operation timing, see figures 18.5 and 18.6. The transmission procedure and operations in master 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 to 1 and the CKS2 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 bits in ICCR1 to select master transmit mode. Then, write 1 to BBSY and 0 to SCP using MOV 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 byte shows the slave address and $\overline{R/\overline{W}}$) to ICDRT. At this time, TDRE is automatically cleared 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 is set at the rise of the 9th transmit clock pulse. Read the ACKBR bit in ICIER, and confirm that the slave device has been selected. Then, write second byte data to ICDRT. When ACKBR is 0, the slave device has not been acknowledged, so issue the stop condition. To issue the stop condition, write 0 to BBSY and SCP using MOV instruction. SCL is fixed low until the 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 end of byte data transmission) while TDRE is 1, or wait for NACK (NACKF in ICSR = 1) from the receive device while ACKE in ICIER is 1. Then, issue the stop condition to clear TEND and NACKF.
7. When the STOP bit in ICSR is set to 1, the operation returns to the slave receive mode.

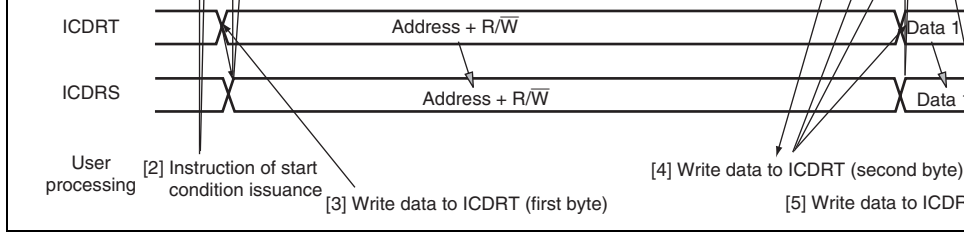


Figure 18.5 Master Transmit Mode Operation Timing (1)

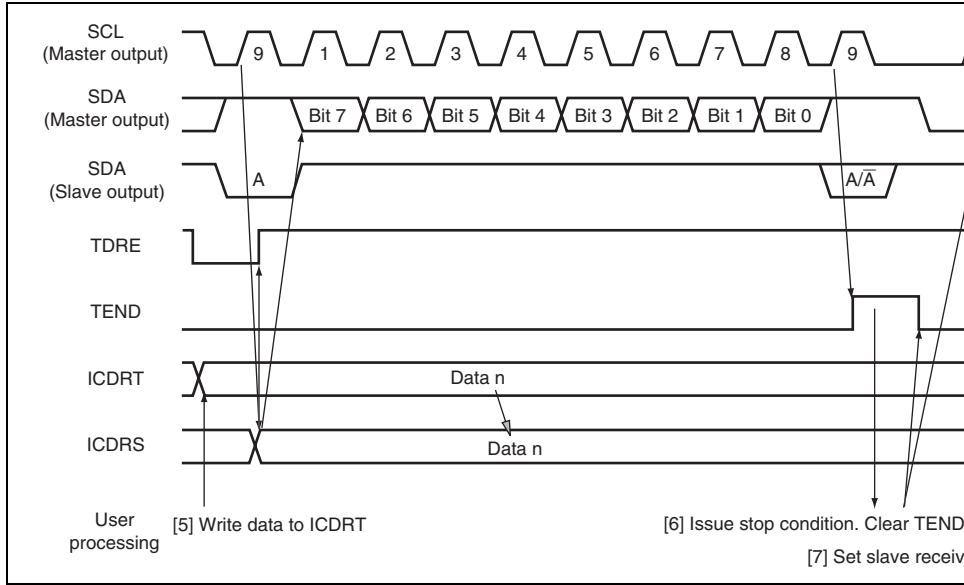


Figure 18.6 Master Transmit Mode Operation Timing (2)

3. After the reception of first frame data is completed, the RDRF bit in ICST is set to 1 at the 9th receive clock pulse. At this time, the receive data is read by reading ICDRR, and is cleared to 0.
4. The continuous reception is performed by reading ICDRR every time RDRF is set. If the receive clock pulse falls after reading ICDRR by the other processing while RDRF is 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 ICDRR. 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 completion condition.
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.

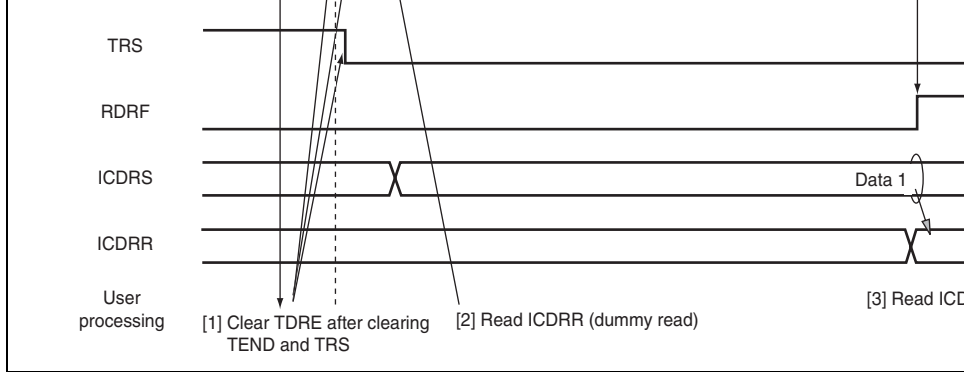


Figure 18.7 Master Receive Mode Operation Timing (1)

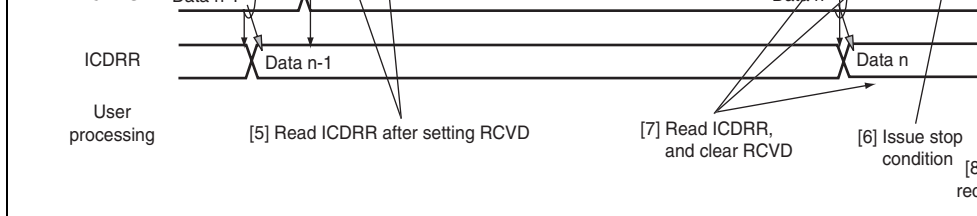


Figure 18.8 Master Receive Mode Operation Timing (2)

18.4.4 Slave Transmit Operation

In slave transmit mode, the slave device outputs the transmit data, while the master device provides the receive clock and returns an acknowledge signal. For slave transmit mode operation timing, see figures 18.9 and 18.10.

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 to 1. (Initial setting) Set the MST and TRS bits in ICCR1 to select slave transmit mode, and wait until the slave address matches.
2. When the slave address matches in the first frame following detection of the start condition, the slave device outputs the level specified by ACKBT in ICIER to SDA, at the rise of the receive clock pulse. At this time, if the 8th bit data ($\overline{R/W}$) is 1, the TRS and ICSR bits in ICCR1 are set to 1, and the mode changes to slave transmit mode automatically. The continuous transmission is performed by writing transmit data to ICDRT every time TDRE is set.
3. If TDRE is set after writing last transmit data to ICDRT, wait until TEND in ICSR is set 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.

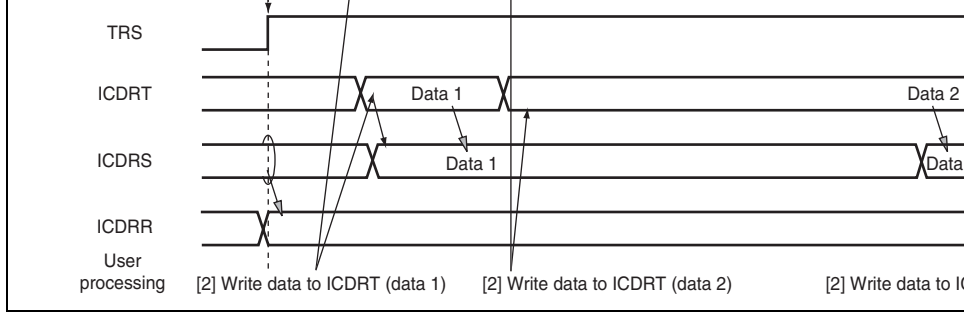


Figure 18.9 Slave Transmit Mode Operation Timing (1)

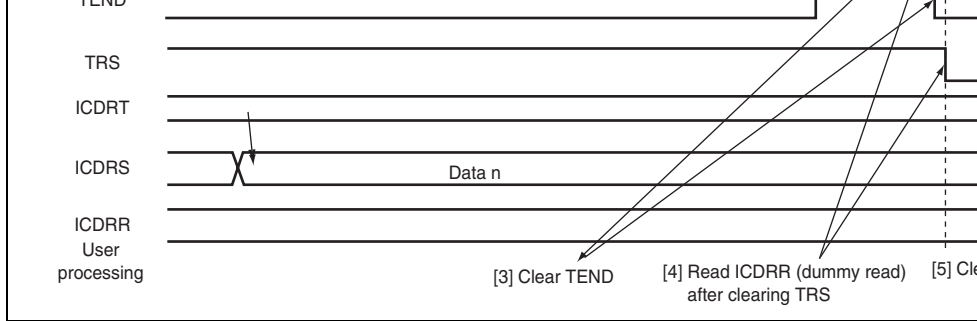


Figure 18.10 Slave Transmit Mode Operation Timing (2)

18.4.5 Slave Receive Operation

In slave receive mode, the master device outputs the transmit clock and transmit data, and the slave device returns an acknowledge signal. For slave receive mode operation timing, see Figures 18.11 and 18.12. The reception procedure and operations in slave receive mode are described below.

1. Set the ICE bit in ICCR1 to 1. Set the MLS and WAIT bits in ICMR and the CKS3 to 3 bits in ICCR1 to 1. (Initial setting) Set the MST and TRS bits in ICCR1 to select slave receive mode, and wait until the slave address matches.
2. When the slave address matches in the first frame following detection of the start condition, the slave device outputs the level specified by ACKBT in ICIEP to SDA, at the rise of the receive clock pulse. At the same time, RDRF in ICSR is set to read ICDRR (dummy read). (Since the 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 set, the ICDRR is fixed low until ICDRR is read. The change of the acknowledge before reading ICDRR is returned to the master device, is reflected to the next transmit frame.

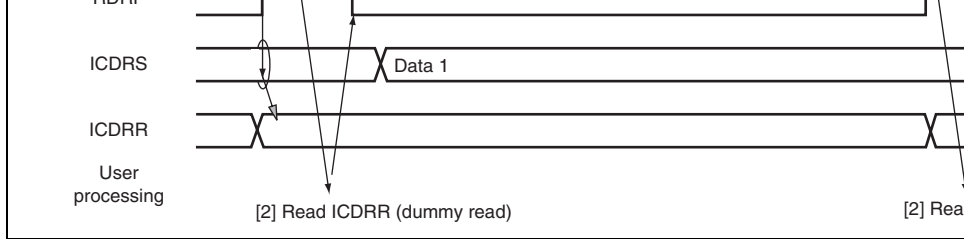


Figure 18.11 Slave Receive Mode Operation Timing (1)

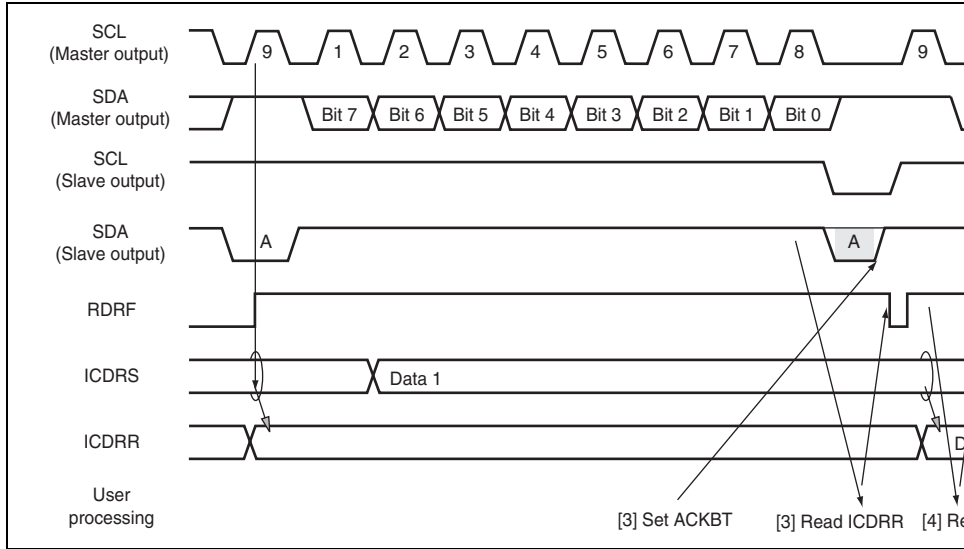


Figure 18.12 Slave Receive Mode Operation Timing (2)

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

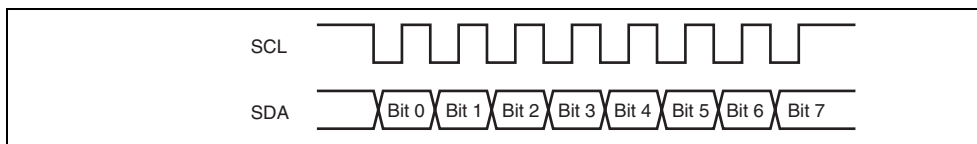


Figure 18.13 Clocked Synchronous Serial Transfer Format

Transmit Operation

In transmit mode, transmit data is output from SDA, in synchronization with the fall of the clock. The transfer clock is output when MST in ICCR1 is 1, and is input when MST is 0. For transmit mode operation timing, see figure 18.14. The transmission procedure and operation 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. (Initial 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 is 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 changing from transmit mode to receive mode, clear TRS while TDRE is 1.

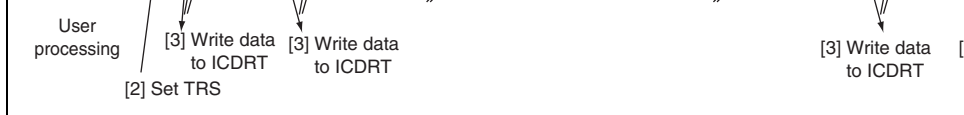


Figure 18.14 Transmit Mode Operation Timing

Receive Operation

In receive mode, data is latched at the rise of the transfer clock. The transfer clock is output when MST in ICCR1 is 1, and is input when MST is 0. For receive mode operation timing, see Figure 18.15. The reception procedure and operations in receive 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. (Clock 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. 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 time 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 ICDRR.
4. To stop receiving when MST = 1, set RCVD in ICCR1 to 1, then read ICDRR. Then set RCVD to 0 (fixed high after receiving the next byte data).

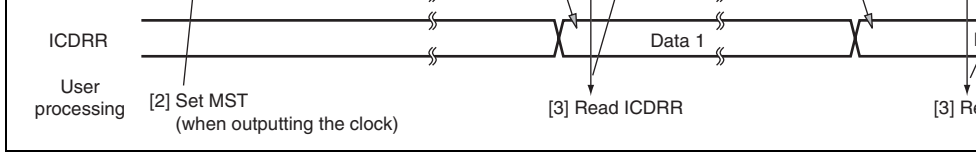


Figure 18.15 Receive Mode Operation Timing

18.4.7 Noise Canceller

The logic levels at the SCL and SDA pins are routed through noise cancellers before being internally. Figure 18.16 shows a block diagram of the noise canceller circuit.

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

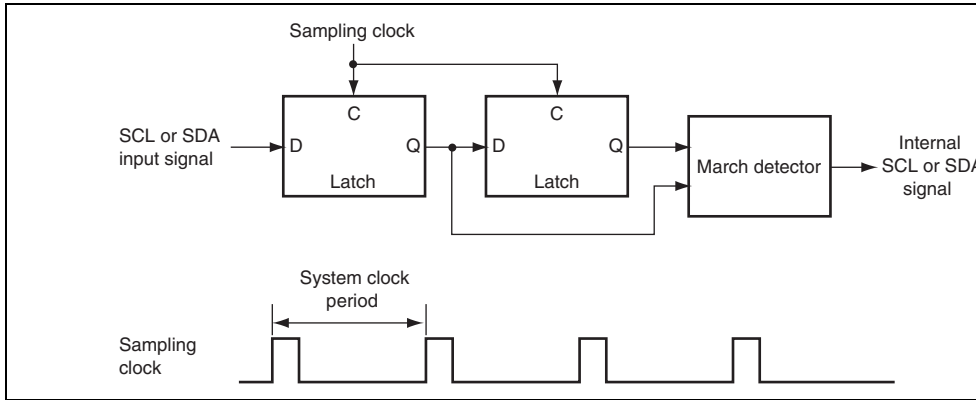


Figure 18.16 Block Diagram of Noise Canceller

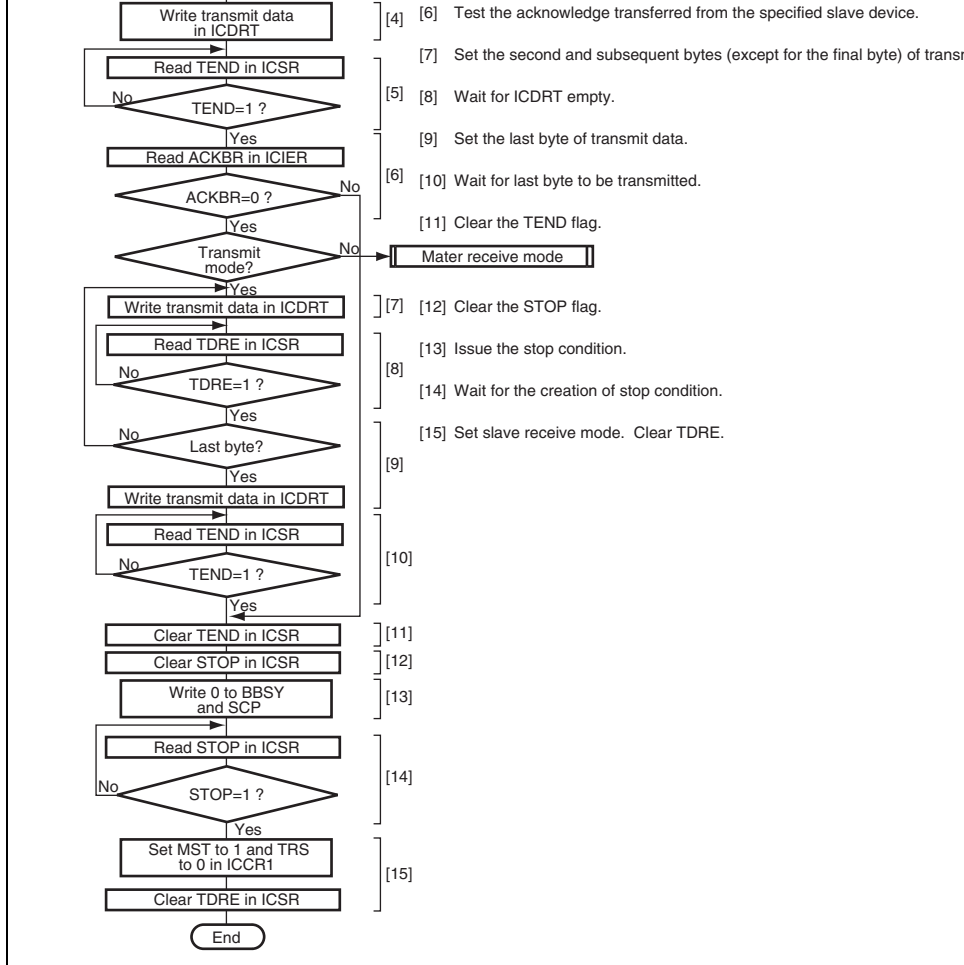
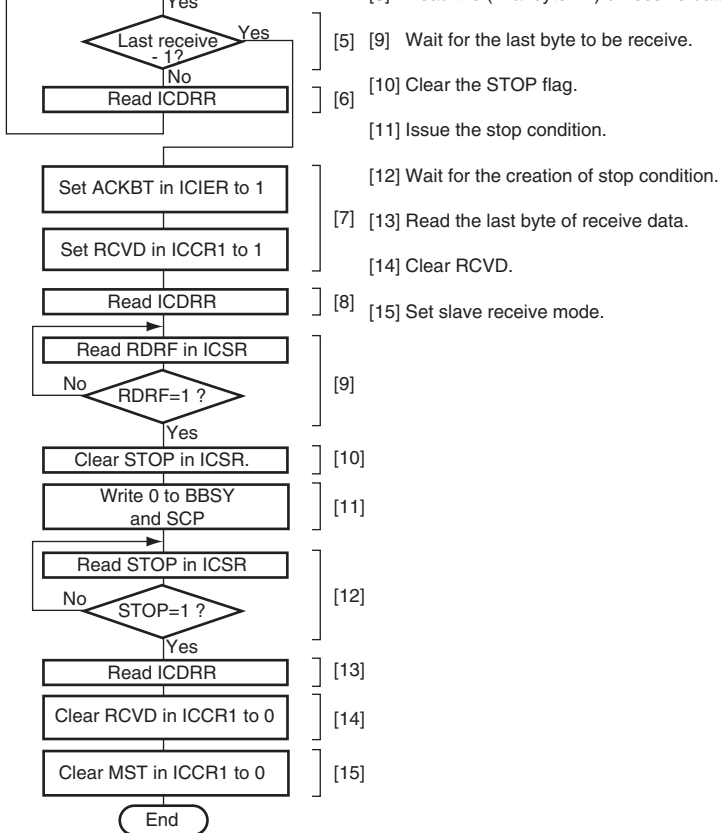


Figure 18.17 Sample Flowchart for Master Transmit Mode



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 dummy-read in ICDRR.

Figure 18.18 Sample Flowchart for Master Receive Mode

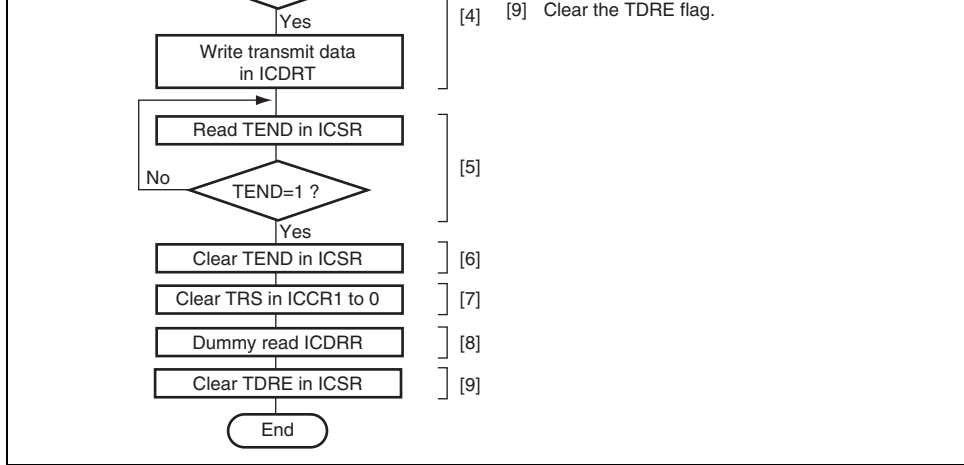
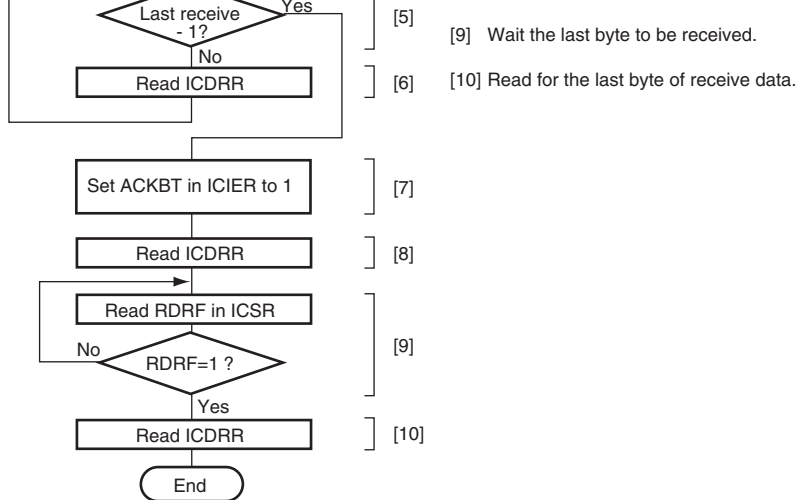


Figure 18.19 Sample Flowchart for Slave Transmit Mode



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 dummy-read in ICDRR.

Figure 18.20 Sample Flowchart for Slave Receive Mode

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

When interrupt conditions described in table 18.3 are 1 and the I bit in CCR is 0, the CPU executes an interrupt exception processing. Interrupt sources should be cleared in the exception processing. TDRE and TEND are automatically cleared to 0 by writing the transmit data to ICDRT. RDRF are automatically cleared to 0 by reading ICDRR. TDRE is set to 1 again at the same time when transmit data is written to ICDRT. When TDRE is cleared to 0, then another data of one byte may be transmitted.

Figure 18.21 shows the timing of the bit synchronous circuit and table 18.4 shows the time for monitoring SCL. SCL output changes from low to Hi-Z then SCL is monitored.

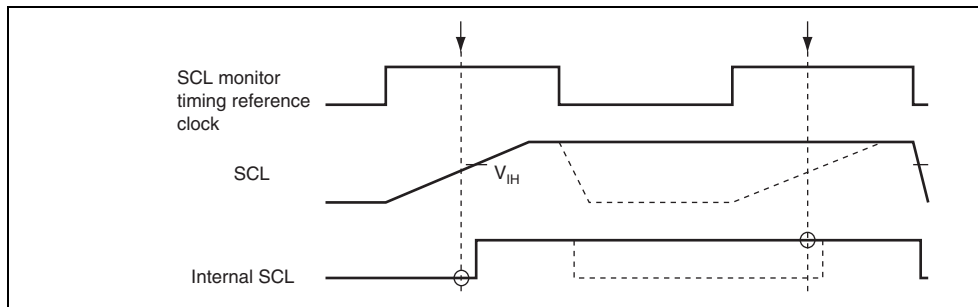


Figure 18.21 Timing of Bit Synchronous Circuit

Table 18.4 Time for Monitoring SCL

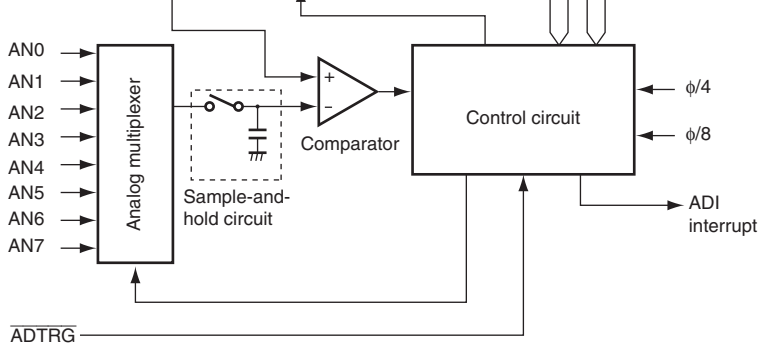
CKS3	CKS2	Time for Monitoring SCL
0	0	7.5 tcyc
	1	19.5 tcyc
1	0	17.5 tcyc
	1	41.5 tcyc

- 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 and ninth clocks, that is driven by the slave device

18.7.2 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 clocks by the slave device at the eighth and ninth clocks, the high period of ninth clock may be shorter than one clock period. To avoid this, set the WAIT bit in ICMR to 0.

- Conversion time: at least 3.5 μ s per channel (at 20-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 data register for each channel
- Sample-and-hold function
- Two conversion start methods
 - Software
 - External trigger signal
- Interrupt source
 - An A/D conversion end interrupt (ADI) request 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 19.1 Block Diagram of A/D Converter

Analog input pin 0	AN0	Input	Group 0 analog input
Analog input pin 1	AN1	Input	
Analog input pin 2	AN2	Input	
Analog input pin 3	AN3	Input	
Analog input pin 4	AN4	Input	Group 1 analog input
Analog input pin 5	AN5	Input	
Analog input pin 6	AN6	Input	
Analog input pin 7	AN7	Input	
A/D external trigger input pin	$\overline{\text{ADTRG}}$	Input	External trigger input for sta conversion

19.3.1 A/D Data Registers A to D (ADDRA to ADDR D)

There are four 16-bit read-only ADDR registers; ADDRA to ADDR D, used to store the result of the A/D conversion. The ADDR registers, which store a conversion result for each analog input channel, are shown in table 19.2.

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

The data bus width between the CPU and the A/D converter is 8 bits. The upper byte can be read directly from the CPU, however the lower byte should be read via a temporary register. The lower byte of the temporary register contents are transferred from the ADDR when the upper byte data is read. Therefore, byte access to ADDR should be done by reading the upper byte first then the lower byte. Word access is also possible. ADDR is initialized to H'0000.

Table 19.2 Analog Input Channels and Corresponding ADDR Registers

Analog Input Channel		
Group 0	Group 1	A/D Data Register to be Stored Results of A/D Conversion
AN0	AN4	ADDRA
AN1	AN5	ADDRB
AN2	AN6	ADDRC
AN3	AN7	ADDRD

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 request (ADI) is by ADF when this bit is set to 1
5	ADST	0	R/W	A/D Start Setting this bit to 1 starts A/D conversion. In single mode, this bit is cleared to 0 automatically when conversion on the specified channel is complete. In scan mode, conversion continues sequentially on the specified channels until this bit is cleared to 0 by software, a reset, or a transition to standby mode.
4	SCAN	0	R/W	Scan Mode Selects single mode or scan mode as the A/D conversion operating mode. 0: Single mode 1: Scan mode
3	CKS	0	R/W	Clock Select Selects the A/D conversions time. 0: Conversion time = 134 states (max.) 1: Conversion time = 70 states (max.) Clear the ADST bit to 0 before switching the clock time.

19.3.3 A/D Control Register (ADCR)

ADCR enables A/D conversion started by an external trigger signal.

3, 2	—	All 0	R/W	Reserved Although these bits are readable/writable, they should not be set to 1.
1	—	1	—	Reserved This bit is always read as 1.
0	—	0	R/W	Reserved Although this bit is readable/writable, this bit should not be set to 1.

channel as follows:

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

19.4.2 Scan Mode

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

1. When the ADST bit in ADCSR is set to 1 by software or external trigger input, A/D conversion starts on the first channel in the group (AN0 when CH2 = 0, AN4 when CH2 = 1).
2. When A/D conversion for each channel is completed, the result is sequentially transferred to the A/D data register corresponding to each channel.
3. When conversion of all the selected channels is completed, the ADF flag in ADCSR is set to 1. If the ADIE bit is set to 1 at this time, an ADI interrupt request is generated. A/D conversion starts again on the first channel in the group.
4. The ADST bit is not automatically cleared to 0. Steps [2] and [3] are repeated as long as the 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 19.3 apply to the first conversion time. In the subsequent conversions, the conversion time is 128 states (fixed) when CKS = 0 and 66 (fixed) when CKS = 1.

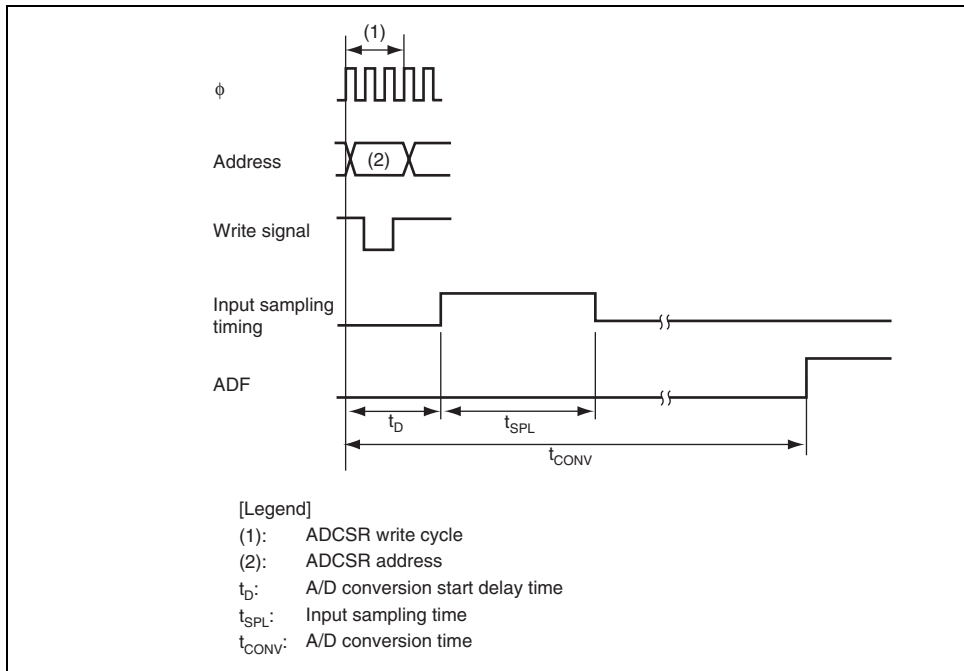


Figure 19.2 A/D Conversion Timing

A/D conversion can also be started by an external trigger input. When the TRGE bit in ADCSR is set to 1, external trigger input is enabled at the $\overline{\text{ADTRG}}$ pin. A falling edge at the $\overline{\text{ADTRG}}$ pin sets the ADST bit in ADCSR to 1, starting A/D conversion. Other operations, in both single and scan modes, are the same as when the bit ADST has been set to 1 by software. Figure 19.3 shows the timing.

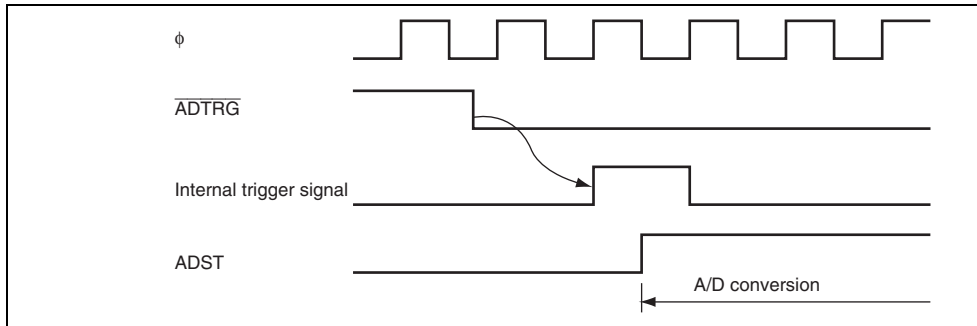


Figure 19.3 External Trigger Input Timing

when the digital output changes from the minimum voltage value 0000000000 to 0000000001 (see figure 19.5).

- Full-scale error

The deviation of the analog input voltage value from the ideal A/D conversion characteristic when the digital output changes from 1111111110 to 1111111111 (see figure 19.5).

- Nonlinearity error

The deviation from the ideal A/D conversion characteristic as the voltage changes from 0 to full scale. This does not include the offset error, full-scale error, or quantization error.

- Absolute accuracy

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

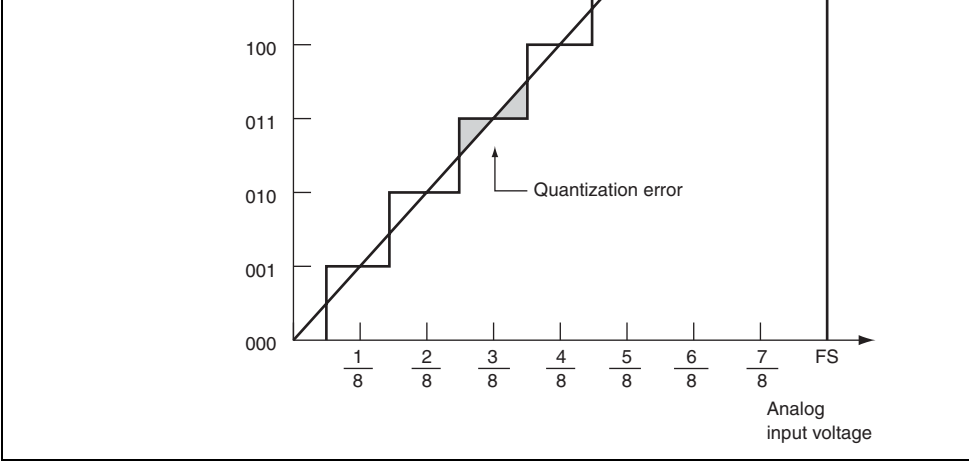


Figure 19.4 A/D Conversion Accuracy Definitions (1)

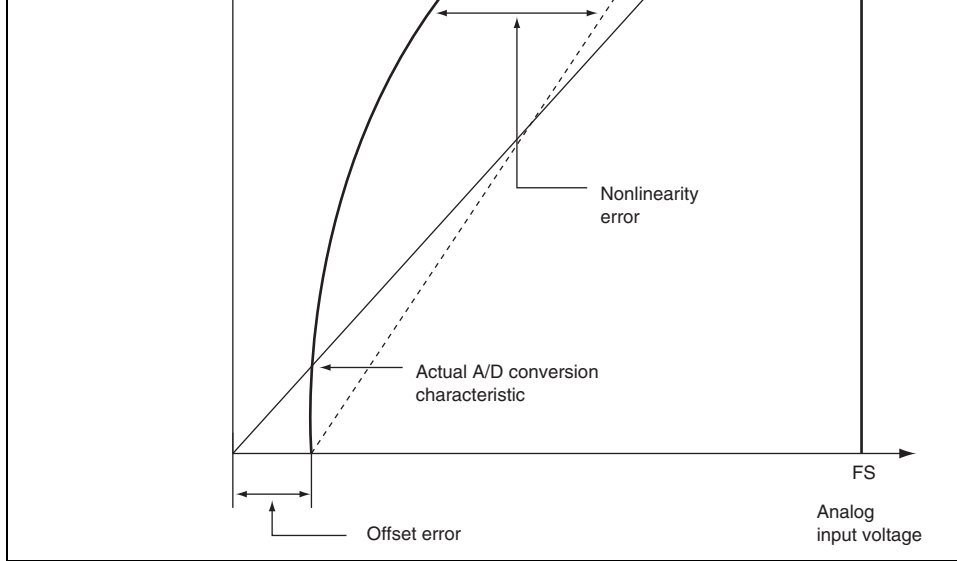


Figure 19.4 A/D Conversion Accuracy Definitions (2)

input resistance of $10\text{ k}\Omega$, and the signal source impedance is ignored. However, as a low-pass filter effect is obtained in this case, it may not be possible to follow an analog signal with a high differential coefficient (e.g., $5\text{ mV}/\mu\text{s}$ or greater) (see figure 19.5). When converting a high-frequency analog signal or converting in scan mode, a low-impedance buffer should be inserted.

19.6.2 Influences on Absolute Accuracy

Adding capacitance results in coupling with GND, and therefore noise in GND may adversely 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 act as antennas on the mounting board.

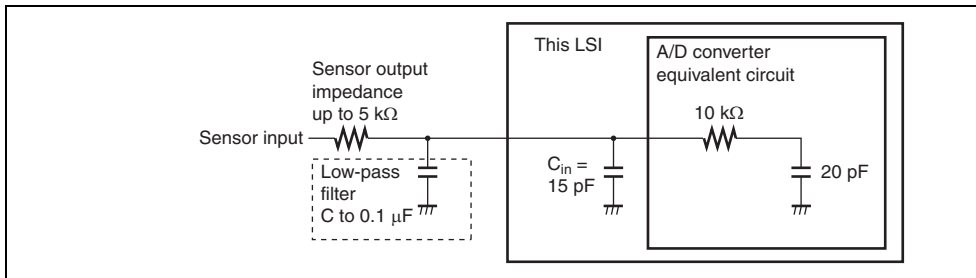


Figure 19.5 Analog Input Circuit Example

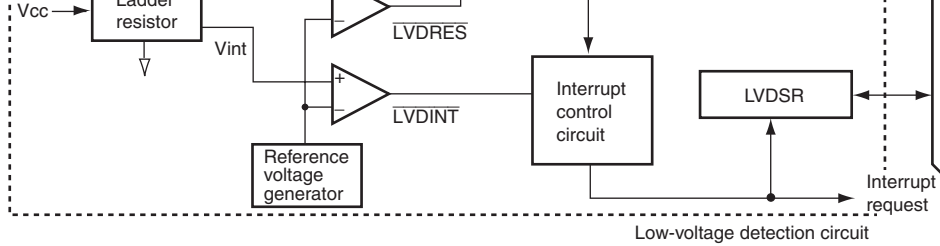
power supply voltage rises again.

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

Figure 20.1 is a block diagram of the power-on reset circuit and the low-voltage detection circuit.

20.1 Features

- Power-on reset circuit
Uses an external capacitor to generate an internal reset signal when power is first supplied.
- Low-voltage detection circuit
LVDR: Monitors the power-supply voltage, and generates an internal reset signal when the voltage falls below a specified value.
LVDI: Monitors the power-supply voltage, and generates an interrupt when the voltage falls below or rises above respective specified values.
Two pairs of detection levels for reset generation voltage are available: when only the LVDR circuit is used, or when the LVDI and LVDR circuits are both used.



[Legend]

- PSS: Prescaler S
- LVDCR: Low-voltage-detection control register
- LVDSR: Low-voltage-detection status register
- LVDRES: Low-voltage-detection reset signal
- LVDINT: Low-voltage-detection interrupt signal
- Vreset: Reset detection voltage
- Vint: Power-supply fall/rise detection voltage

Figure 20.1 Block Diagram of Power-On Reset Circuit and Low-Voltage Detection Circuit

the LVDR function, enable or disable the LVDR function, and enable or disable general interrupt when the power-supply voltage rises above or falls below the respective levels.

Table 20.1 shows the relationship between the LVDCR settings and select functions. LVDCR should be set according to table 20.1.

Bit	Bit Name	Initial Value	R/W	Description
7	LVDE	0*	R/W	LVD Enable 0: The low-voltage detection circuit is not used (standby mode) 1: The low-voltage detection circuit is used
6 to 4	—	All 1	—	Reserved These bits are always read as 1, and cannot be modified.
3	LVDSSEL	0*	R/W	LVDR Detection Level Select 0: Reset detection voltage is 2.3 V (typ.) 1: Reset detection voltage is 3.6 V (typ.) When the falling or rising voltage detection interrupt is used, reset detection voltage of 2.3 V (typ.) should be used. When only a reset detection interrupt is used, reset detection voltage of 3.6 V (typ.) should be used.
2	LVDRRE	0*	R/W	LVDR Enable 0: Disables the LVDR function 1: Enables the LVDR function

Note: * Not initialized by LVDR but initialized by a power-on reset or WDT reset.

Table 20.1 LVDCR Settings and Select Functions

LVDCR Settings					Select Functions			
LVDE	LVDSSEL	LVDRE	LVDDE	LVDUE	Power-On Reset	LVDR	Low-Voltage-Detection Falling Interrupt	Low Voltage-Detection Rising Interrupt
0	*	*	*	*	0	—	—	—
1	1	1	0	0	0	0	—	—
1	0	0	1	0	0	—	0	—
1	0	0	1	1	0	—	0	0
1	0	1	1	1	0	0	0	0

[Legend] * means invalid.

	LVDUF	0	R/W	<p>LVD Power-Supply Voltage Fall Flag</p> <p>[Setting condition]</p> <p>When the power-supply voltage falls below V_{int} (typ. = 3.7 V)</p> <p>[Clearing condition]</p> <p>Writing 0 to this bit after reading it as 1</p>
0	LVDUF	0*	R/W	<p>LVD Power-Supply Voltage Rise Flag</p> <p>[Setting condition]</p> <p>When the power supply voltage falls below V_{int} while the LVDUE bit in LVDCCR is set to 1, then the power supply voltage rises above V_{int} (U) (typ. = 4.0 V) before falling below V_{reset1} (typ. = 2.3 V)</p> <p>[Clearing condition]</p> <p>Writing 0 to this bit after reading it as 1</p>

Note: * Initialized by LVDR.

131,072 clock (ϕ) cycles. The noise cancellation circuit of approximately 100 ns is incorporated to prevent the incorrect operation of the chip by noise on the $\overline{\text{RES}}$ pin.

To achieve stable operation of this LSI, the power supply needs to rise to its full level and settle within the specified time. The maximum time required for the power supply to rise and settle after power has been supplied (t_{PWON}) is determined by the oscillation frequency (f_{OSC}) and capacitor (C_{RES}) which is connected to $\overline{\text{RES}}$ pin (C_{RES}). If t_{PWON} means the time required to reach 90 % of power supply voltage, the power supply circuit should be designed to satisfy the following formula:

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

$(t_{\text{PWON}} \leq 3000 \text{ ms, } C_{\text{RES}} \geq 0.22 \text{ }\mu\text{F, and } f_{\text{OSC}} = 10 \text{ in 2-MHz to 10-MHz operation})$

Note that the power supply voltage (V_{CC}) must fall below $V_{\text{por}} = 100 \text{ mV}$ and rise after the $\overline{\text{RES}}$ pin is removed. To remove charge on the $\overline{\text{RES}}$ pin, it is recommended that the diode should be placed near V_{CC} . If the power supply voltage (V_{CC}) rises from the point above V_{por} , power-on reset may not occur.

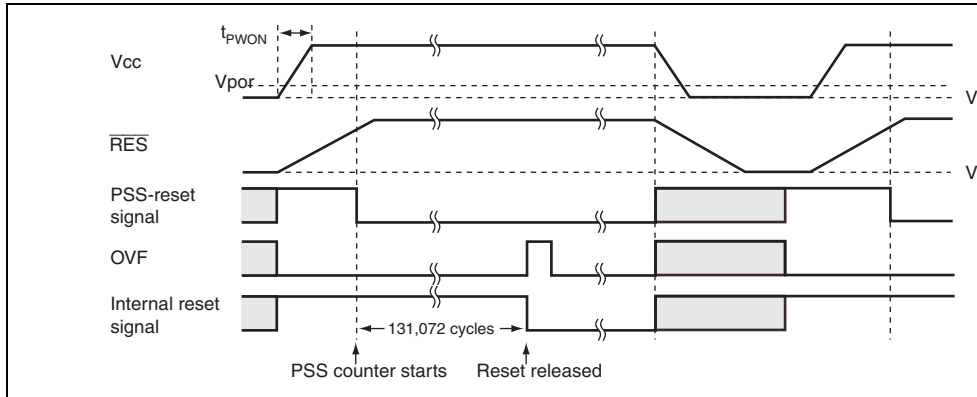


Figure 20.2 Operational Timing of Power-On Reset Circuit

settings of ports must be made. To cancel the low-voltage detection circuit, first the LVDE bit should be cleared to 0 and then the LVDE bit should be cleared to 0. The LVDE and LVDESEL must not be cleared to 0 simultaneously because incorrect operation may occur.

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

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

If the power supply voltage (Vcc) falls below $V_{\text{por}} = 100 \text{ mV}$, a power-on reset occurs.

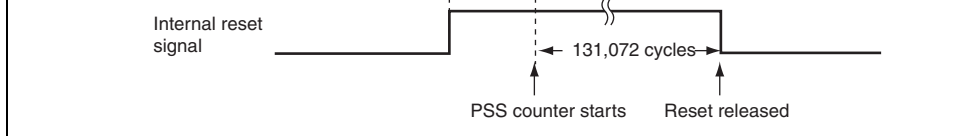


Figure 20.3 Operational Timing of LVDR Circuit

LVDI (Interrupt by Low Voltage Detect) Circuit:

Figure 20.4 shows the timing of LVDI functions. The LVDI enters the module-standby state when a power-on reset is canceled. To operate the LVDI, set the LVDE bit in LVDCR to 1, wait t_{LVDON} until the reference voltage and the low-voltage-detection power supply have stabilized by a software timer, etc., then set the LVDDE and LVDUE bits in LVDCR to 1. After the output settings of ports must be made. To cancel the low-voltage detection circuit, first the LVDDE and LVDUE bits should all be cleared to 0 and then the LVDE bit should be cleared. The LVDE bit must not be cleared to 0 at the same timing as the LVDDE and LVDUE bits because incorrect operation may occur.

When the power-supply voltage falls below $V_{int(D)}$ (typ. = 3.7 V) voltage, the LVDI clears the \overline{LVDINT} signal to 0 and the LVDDF bit in LVDSR is set to 1. If the LVDDE bit is 1 at this time, an IRQ0 interrupt request is simultaneously generated. In this case, the necessary data must be saved in the external EEPROM, etc, and a transition must be made to standby mode or suspend mode. Until this processing is completed, the power supply voltage must be higher than the limit of the guaranteed operating voltage.

When the power-supply voltage does not fall below V_{reset1} (typ. = 2.3 V) voltage but rises above $V_{int(U)}$ (typ. = 4.0 V) voltage, the LVDI sets the \overline{LVDINT} signal to 1. If the LVDUE bit is 1 at this time, the LVDUF bit in LVDSR is set to 1 and an IRQ0 interrupt request is simultaneously generated.

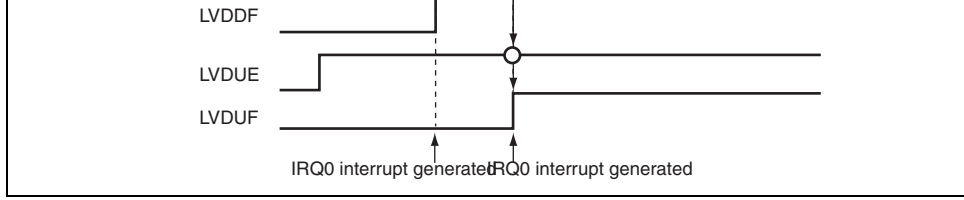


Figure 20.4 Operational Timing of LVDI Circuit

Procedures for Clearing Settings when Using LVDR and LVDI:

To operate or release the low-voltage detection circuit normally, follow the procedure described below. Figure 20.5 shows the timing for the operation and release of the low-voltage detection circuit.

1. To operate the low-voltage detection circuit, set the LVDE bit in LVDCR to 1.
2. Wait for $50\ \mu\text{s}$ (t_{LVDon}) until the reference voltage and the low-voltage-detection power have stabilized by a software timer, etc. Then, clear the LVDDF and LVDUF bits in LVDCR to 0 and set the LVDRE, LVDDE, and LVDUE bits in LVDCR to 1, as required.
3. To release the low-voltage detection circuit, start by clearing all of the LVDRE, LVDDE, LVDUE bits to 0. Then clear the LVDE bit to 0. The LVDE bit must not be cleared at the same timing as the LVDRE, LVDDE, and LVDUE bits because incorrect operation

21.1 When Using Internal Power Supply Step-Down Circuit

Connect the external power supply to the V_{CC} pin, and connect a capacitance of approximately μF between V_{CL} and V_{SS} , as shown in figure 21.1. The internal step-down circuit is made simply by adding this external circuit. In the external circuit interface, the external power voltage connected to V_{CC} and the GND potential connected to V_{SS} are the reference levels. For example, for port input/output levels, the V_{CC} level is the reference for the high level, and the V_{SS} level is that for the low level. The A/D converter analog power supply is not affected by the internal step-down circuit.

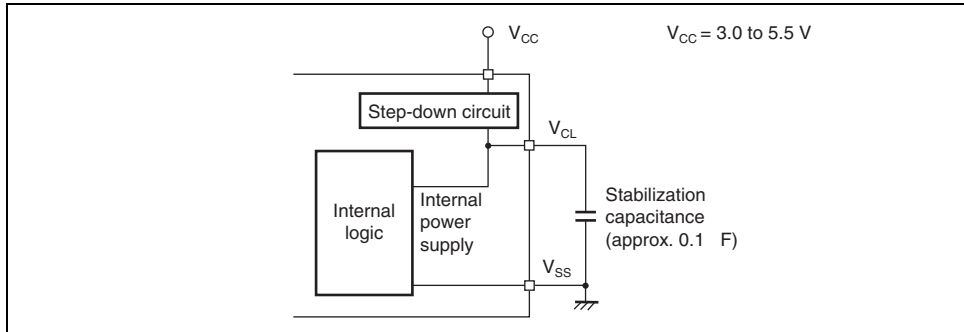


Figure 21.1 Power Supply Connection when Internal Step-Down Circuit is Used

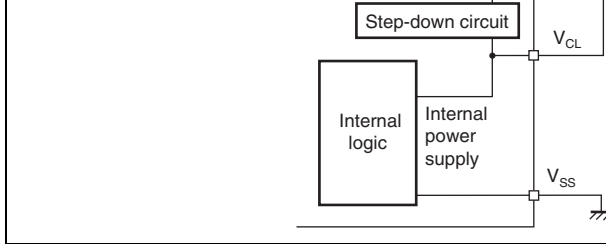


Figure 21.2 Power Supply Connection when Internal Step-Down Circuit is Not

Do not attempt to access reserved addresses.

- When the address is 16-bit wide, the address of the upper byte is given in the list.
- Registers are classified by functional modules.
- The data bus width is indicated.

2. Register bits

- Bit configurations of the registers are described in the same order as the register address.
- Reserved bits are indicated by — in the bit name column.
- No entry in the bit-name column indicates that the whole register is allocated as a control register for holding data.
- 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 special mode for an on-chip peripheral module, refer to the section on that on-chip peripheral module.

—	—	—	H'FFF000 to H'FFF5FF	—	—
Serial mode register_3	SMR_3	8	H'FFF600	SCI3_3	8
Bit rate register_3	BRR_3	8	H'FFF601	SCI3_3	8
Serial control register 3_3	SCR3_3	8	H'FFF602	SCI3_3	8
Transmit data register_3	TDR_3	8	H'FFF603	SCI3_3	8
Serial status register_3	SSR_3	8	H'FFF604	SCI3_3	8
Receive data register_3	RDR_3	8	H'FFF605	SCI3_3	8
—	—	—	H'FFF606, H'FFF607	—	—
Serial mode control register	SMCR_3	8	H'FFF608	SCI3_3	8
—	—	—	H'FFF609 to H'FFF6FF	—	—
Timer control register_0	TCR_0	8	H'FFF700	Timer Z0	8
Timer I/O control register A_0	TIORA_0	8	H'FFF701	Timer Z0	8
Timer I/O control register C_0	TIORC_0	8	H'FFF702	Timer Z0	8
Timer status register_0	TSR_0	8	H'FFF703	Timer Z0	8
Timer interrupt enable register_0	TIER_0	8	H'FFF704	Timer Z0	8
PWM mode output level control register_0	POCR_0	8	H'FFF705	Timer Z0	8
Timer counter_0	TCNT_0	16	H'FFF706	Timer Z0	16
General register A_0	GRA_0	16	H'FFF708	Timer Z0	16
General register B_0	GRB_0	16	H'FFF70A	Timer Z0	16
General register C_0	GRC_0	16	H'FFF70C	Timer Z0	16
General register D_0	GRD_0	16	H'FFF70E	Timer Z0	16

register_1					
Timer counter_1	TCNT_1	16	H'FFF716	Timer Z1	16
General register A_1	GRA_1	16	H'FFF718	Timer Z1	16
General register B_1	GRB_1	16	H'FFF71A	Timer Z1	16
General register C_1	GRC_1	16	H'FFF71C	Timer Z1	16
General register D_1	GRD_1	16	H'FFF71E	Timer Z1	16
Timer start register	TSTR	8	H'FFF720	Timer Z common	8
Timer mode register	TMDR	8	H'FFF721	Timer Z common	8
Timer PWM mode register	TPMR	8	H'FFF722	Timer Z common	8
Timer function control register	TFCR	8	H'FFF723	Timer Z common	8
Timer output master enable register	TOER	8	H'FFF724	Timer Z common	8
Timer output control register	TOCR	8	H'FFF725	Timer Z common	8
—	—	—	H'FFF726, H'FFF727	—	—
Second data register/free running counter data register	RSECDR	8	H'FFF728	RTC	8
Minute data register	RMINDR	8	H'FFF729	RTC	8
Hour data register	RHRDR	8	H'FFF72A	RTC	8
Day-of-week data register	RWKDR	8	H'FFF72B	RTC	8
RTC control register 1	RTCCR1	8	H'FFF72C	RTC	8

—	—	—	H'FFF732 to H'FFF73F	—	—
Serial mode register_2	SMR_2	8	H'FFF740	SCI3_2	8
Bit rate register_2	BRR_2	8	H'FFF741	SCI3_2	8
Serial control register 3_2	SCR3_2	8	H'FFF742	SCI3_2	8
Transmit data register_2	TDR_2	8	H'FFF743	SCI3_2	8
Serial status register_2	SSR_2	8	H'FFF744	SCI3_2	8
Receive data register_2	RDR_2	8	H'FFF745	SCI3_2	8
—	—	—	H'FFF746, H'FFF747	—	—
I ² C bus control register 1	ICCR1	8	H'FFF748	IIC2	8
I ² C bus control register 2	ICCR2	8	H'FFF749	IIC2	8
I ² C bus mode register	ICMR	8	H'FFF74A	IIC2	8
I ² C bus interrupt enable register	ICIER	8	H'FFF74B	IIC2	8
I ² C bus status register	ICSR	8	H'FFF74C	IIC2	8
Slave address register	SAR	8	H'FFF74D	IIC2	8
I ² C bus transmit data register	ICDRT	8	H'FFF74E	IIC2	8
I ² C bus receive data register	ICDRR	8	H'FFF74F	IIC2	8
—	—	—	H'FFF750 to H'FFF75F	—	—
Timer mode register B1	TMB1	8	H'FFF760	Timer B1	8
Timer counter B1	TCB1	8	H'FFF761	Timer B1	8
Timer load register B1	TLB1	8	H'FFF761	Timer B1	8
—	—	—	H'FFF762 to H'FFFF7F	—	—

General register A	GRA	16	H'FFFF88	Timer W	16
General register B	GRB	16	H'FFFF8A	Timer W	16
General register C	GRC	16	H'FFFF8C	Timer W	16
General register D	GRD	16	H'FFFF8E	Timer W	16
Flash memory control register 1	FLMCR1	8	H'FFFF90	ROM	8
Flash memory control register 2	FLMCR2	8	H'FFFF91	ROM	8
Flash memory power control register	FLPWC	8	H'FFFF92	ROM	8
Erase block register 1	EBR1	8	H'FFFF93	ROM	8
—	—	—	H'FFFF94 to H'FFFF9A	—	—
Flash memory enable register	FENR	8	H'FFFF9B	ROM	8
—	—	—	H'FFFF9C to H'FFFF9F	—	—
Timer control register V0	TCRV0	8	H'FFFFA0	Timer V	8
Timer control/status register V	TCSR	8	H'FFFFA1	Timer V	8
Time constant register A	TCORA	8	H'FFFFA2	Timer V	8
Time constant register B	TCORB	8	H'FFFFA3	Timer V	8
Timer counter V	TCNTV	8	H'FFFFA4	Timer V	8
Timer control register V1	TCRV1	8	H'FFFFA5	Timer V	8
—	—	—	H'FFFFA6, H'FFFFA7	—	—
Serial mode register	SMR	8	H'FFFFA8	SCI3	8
Bit rate register	BRR	8	H'FFFFA9	SCI3	8

A/D data register B	ADDRB	16	H'FFFFB2	A/D converter	8
A/D data register C	ADDRC	16	H'FFFFB4	A/D converter	8
A/D data register D	ADDRD	16	H'FFFFB6	A/D converter	8
A/D control/status register	ADCSR	8	H'FFFFB8	A/D converter	8
A/D control register	ADCR	8	H'FFFFB9	A/D converter	8
—	—	—	H'FFFFBA, H'FFFFBB	—	—
PWM data register L	PWDRL	8	H'FFFFBC	14-bit PWM	8
PWM data register U	PWDRU	8	H'FFFFBD	14-bit PWM	8
PWM control register	PWCR	8	H'FFFFBE	14-bit PWM	8
—	—	—	H'FFFFBF	—	—
Timer control/status register WD	TCSRWD	8	H'FFFFC0	WDT* ²	8
Timer counter WD	TCWD	8	H'FFFFC1	WDT* ²	8
Timer mode register WD	TMWD	8	H'FFFFC2	WDT* ²	8
—	—	—	H'FFFFC3	—	—
—	—	—	H'FFFFC4 to H'FFFFC7	—	—

Break data register H	BDRH	8	H'FFFFCC	Address break	8
Break data register L	BDRL	8	H'FFFFCD	Address break	8
—	—	—	H'FFFFCE	—	—
Break address register E	BARE	8	H'FFFFCF	Address break	8
Port pull-up control register 1	PUCR1	8	H'FFFFD0	I/O port	8
Port pull-up control register 5	PUCR5	8	H'FFFFD1	I/O port	8
—	—	—	H'FFFFD2, H'FFFFD3	—	—
Port data register 1	PDR1	8	H'FFFFD4	I/O port	8
Port data register 2	PDR2	8	H'FFFFD5	I/O port	8
Port data register 3	PDR3	8	H'FFFFD6	I/O port	8
—	—	—	H'FFFFD7	—	—
Port data register 5	PDR5	8	H'FFFFD8	I/O port	8
Port data register 6	PDR6	8	H'FFFFD9	I/O port	8
Port data register 7	PDR7	8	H'FFFFDA	I/O port	8
Port data register 8	PDR8	8	H'FFFFDB	I/O port	8
Port data register 9	PDR9	8	H'FFFFDC	I/O port	8
Port data register B	PDRB	8	H'FFFFDD	I/O port	8
—	—	—	H'FFFFDE, H'FFFFDF	—	—
Port mode register 1	PMR1	8	H'FFFFE0	I/O port	8
Port mode register 5	PMR5	8	H'FFFFE1	I/O port	8

Port control register 6	PCR6	8	H'FFFFE9	I/O port	8
Port control register 7	PCR7	8	H'FFFFEA	I/O port	8
Port control register 8	PCR8	8	H'FFFFEB	I/O port	8
Port control register 9	PCR9	8	H'FFFFEC	I/O port	8
—	—	—	H'FFFFED to H'FFFFEF	—	—
System control register 1	SYSCR1	8	H'FFFFF0	Power-down	8
System control register 2	SYSCR2	8	H'FFFFF1	Power-down	8
Interrupt edge select register 1	IEGR1	8	H'FFFFF2	Interrupt	8
Interrupt edge select register 2	IEGR2	8	H'FFFFF3	Interrupt	8
Interrupt enable register 1	IENR1	8	H'FFFFF4	Interrupt	8
Interrupt enable register 2	IENR2	8	H'FFFFF5	Interrupt	8
Interrupt flag register 1	IRR1	8	H'FFFFF6	Interrupt	8
Interrupt flag register 2	IRR2	8	H'FFFFF7	Interrupt	8
Wakeup interrupt flag register	IWPR	8	H'FFFFF8	Interrupt	8
Module standby control register 1	MSTCR1	8	H'FFFFF9	Power-down	8
Module standby control register 2	MSTCR2	8	H'FFFFFA	Power-down	8
—	—	—	H'FFFFFB to H'FFFFFC	—	—

Notes: 1. LVDC: Low-voltage detection circuits (optional)
2. WDT: Watchdog timer

SSR_3	TDRE	RDRF	OER	FER	PER	TEND	MPBR	MPBT
RDR_3	RDR7	RDR6	RDR5	RDR4	RDR3	RDR2	RDR1	RDR0
SMCR_3	—	—	—	—	—	NFEN_3	TXD_3	MSTS3_3
TCR_0	CCLR2	CCLR1	CCLR0	CKEG1	CKEG0	TPSC2	TPSC1	TPSC0
TIORA_0	—	IOB2	IOB1	IOB0	—	IOA2	IOA1	IOA0
TIORC_0	—	IOD2	IOD1	IOD0	—	IOC2	IOC1	IOC0
TSR_0	—	—	—	OVF	IMFD	IMFC	IMFB	IMFA
TIER_0	—	—	—	OVIE	IMIED	IMIEC	IMIEB	IMIEA
POCR_0	—	—	—	—	—	POLD	POLC	POLB
TCNT_0	TCNT0H7	TCNT0H6	TCNT0H5	TCNT0H4	TCNT0H3	TCNT0H2	TCNT0H1	TCNT0H0
	TCNT0L7	TCNT0L6	TCNT0L5	TCNT0L4	TCNT0L3	TCNT0L2	TCNT0L1	TCNT0L0
GRA_0	GRA0H7	GRA0H6	GRA0H5	GRA0H4	GRA0H3	GRA0H2	GRA0H1	GRA0H0
	GRA0L7	GRA0L6	GRA0L5	GRA0L4	GRA0L3	GRA0L2	GRA0L1	GRA0L0
GRB_0	GRB0H7	GRB0H6	GRB0H5	GRB0H4	GRB0H3	GRB0H2	GRB0H1	GRB0H0
	GRB0L7	GRB0L6	GRB0L5	GRB0L4	GRB0L3	GRB0L2	GRB0L1	GRB0L0
GRC_0	GRC0H7	GRC0H6	GRC0H5	GRC0H4	GRC0H3	GRC0H2	GRC0H1	GRC0H0
	GRC0L7	GRC0L6	GRC0L5	GRC0L4	GRC0L3	GRC0L2	GRC0L1	GRC0L0
GRD_0	GRD0H7	GRD0H6	GRD0H5	GRD0H4	GRD0H3	GRD0H2	GRD0H1	GRD0H0
	GRD0L7	GRD0L6	GRD0L5	GRD0L4	GRD0L3	GRD0L2	GRD0L1	GRD0L0
TCR_1	CCLR2	CCLR1	CCLR0	CKEG1	CKEG0	TPSC2	TPSC1	TPSC0
TIORA_1	—	IOB2	IOB1	IOB0	—	IOA2	IOA1	IOA0
TIORC_1	—	IOD2	IOD1	IOD0	—	IOC2	IOC1	IOC0

GRB1L7	GRB1L6	GRB1L5	GRB1L4	GRB1L3	GRB1L2	GRB1L1	GRB1L0	
GRC_1	GRC1H7	GRC1H6	GRC1H5	GRC1H4	GRC1H3	GRC1H2	GRC1H1	GRC1H0
	GRC1L7	GRC1L6	GRC1L5	GRC1L4	GRC1L3	GRC1L2	GRC1L1	GRC1L0
GRD_1	GRD1H7	GRD1H6	GRD1H5	GRD1H4	GRD1H3	GRD1H2	GRD1H1	GRD1H0
	GRD1L7	GRD1L6	GRD1L5	GRD1L4	GRD1L3	GRD1L2	GRD1L1	GRD1L0
TSTR	—	—	—	—	—	—	STR1	STRO
TMDR	BFD1	BFC1	BFD0	BFC0	—	—	—	SYNC
TPMR	—	PWMD1	PWMC1	PWMB1	—	PWMD0	PWMC0	PWMB0
TFCR	—	STCLK	ADEG	ADTRG	OLS1	OLS0	CMD1	CMD0
TOER	ED1	EC1	EB1	EA1	ED0	EC0	EB0	EA0
TOCR	TOD1	TOC1	TOB1	TOA1	TOD0	TOC0	TOB0	TOA0
RSECDR	BSY	SC12	SC11	SC10	SC03	SC02	SC01	SC00
RMINDR	BSY	MN12	MN11	MN10	MN03	MN02	MN01	MN00
RHRDR	BSY	—	HR11	HR10	HR03	HR02	HR01	HR00
RWKDR	BSY	—	—	—	—	WK2	WK1	WK0
RTCCR1	RUN	12/24	PM	RST	INT	—	—	—
RTCCR2	—	—	FOIE	WKIE	DYIE	HRIE	MNIE	SEIE
RTCCSR	—	RCS6	RCS5	—	RCS3	RCS2	RCS1	RCS0
LVDCR	LVDE	—	—	—	LVDSSEL	LVDRE	LVDDE	LVDUE
LVDSR	—	—	—	—	—	—	LVDDF	LVDUF

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
TCB1	TCB17	TCB16	TCB15	TCB14	TCB13	TCB12	TCB11	TCB10
TLB1	TLB17	TLB16	TLB15	TLB14	TLB13	TLB12	TLB11	TLB10
TMRW	CTS	—	BUFEB	BUFEA	—	PWMD	PWMC	PWMB
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
GRC	GRC15	GRC14	GRC13	GRC12	GRC11	GRC10	GRC9	GRC8
	GRC7	GRC6	GRC5	GRC4	GRC3	GRC2	GRC1	GRC0

TCRV0	CMIEB	CMIEA	OVIE	OCIE1	OCIE0	OS2	OS1	OS0
TCSR	CMFB	CFMA	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
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
ADDRA	AD9	AD8	AD7	AD6	AD5	AD4	AD3	AD2
	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	—	—	—	—	—	—	—

BARH	BARH7	BARH6	BARH5	BARH4	BARH3	BARH2	BARH1	BARH0
BARL	BARL7	BARL6	BARL5	BARL4	BARL3	BARL2	BARL1	BARL0
BDRH	BDRH7	BDRH6	BDRH5	BDRH4	BDRH3	BDRH2	BDRH1	BDRH0
BDRL	BDRL7	BDRL6	BDRL5	BDRL4	BDRL3	BDRL2	BDRL1	BDRL0
BARE	BARE7	BARE6	BARE5	BARE4	BARE3	BARE2	BARE1	BARE0
PUCR1	PUCR17	PUCR16	PUCR15	PUCR14	—	PUCR12	PUCR11	PUCR10
PUCR5	—	—	PUCR55	PUCR54	PUCR53	PUCR52	PUCR51	PUCR50
PDR1	P17	P16	P15	P14	—	P12	P11	P10
PDR2	—	—	—	P24	P23	P22	P21	P20
PDR3	P37	P36	P35	P34	P33	P32	P31	P30
PDR5	P57	P56	P55	P54	P53	P52	P51	P50
PDR6	P67	P66	P65	P64	P63	P62	P61	P60
PDR7	P77	P76	P75	P74	—	P72	P71	P70
PDR8	P87	P86	P85	P84	P83	P82	P81	P80
PDR9	P97	P96	P95	P94	P93	P92	P91	P90
PDRB	PB7	PB6	PB5	PB4	PB3	PB2	PB1	PB0
PMR1	IRQ3	IRQ2	IRQ1	IRQ0	TXD2	PWM	TXD	TMOW
PMR5	POF57	POF56	WKP5	WKP4	WKP3	WKP2	WKP1	WKP0
PMR3	—	—	—	POF24	POF23	—	—	—
PCR1	PCR17	PCR16	PCR15	PCR14	—	PCR12	PCR11	PCR10
PCR2	—	—	—	PCR24	PCR23	PCR22	PCR21	PCR20
PCR3	PCR37	PCR36	PCR35	PCR34	PCR33	PCR32	PCR31	PCR30
PCR5	PCR57	PCR56	PCR55	PCR54	PCR53	PCR52	PCR51	PCR50

IENR2	—	—	WPLEG3	WPLEG4	WPLEG5	WPLEG2	WPLEG1	WPLEG0
IENR1	IENDT	IENTA	IENWP	—	IEN3	IEN2	IEN1	IEN0
IENR2	—	—	IENRB1	—	—	—	—	—
IRR1	IRRDT	IRRTA	—	—	IRRI3	IRRI2	IRRI1	IRRI0
IRR2	—	—	IRRTB1	—	—	—	—	—
IWPR	—	—	IWPF5	IWPF4	IWPF3	IWPF2	IWPF1	IWPF0
MSTCR1	—	MSTIIC	MSTS3	MSTAD	MSTWD	MSTTW	MSTTV	MSTTA
MSTCR2	MSTS3_2	—	—	MSTTB1	—	—	MSTTZ	MSTPWM

- Notes: 1. LVDC: Low-voltage detection circuits (optional)
2. WDT: Watchdog timer

SMCR_3	Initialized	—	—	—	—	—
TCR_0	Initialized	—	—	—	—	Tim
TIORA_0	Initialized	—	—	—	—	
TIORC_0	Initialized	—	—	—	—	
TSR_0	Initialized	—	—	—	—	
TIER_0	Initialized	—	—	—	—	
POCR_0	Initialized	—	—	—	—	
TCNT_0	Initialized	—	—	—	—	
GRA_0	Initialized	—	—	—	—	
GRB_0	Initialized	—	—	—	—	
GRC_0	Initialized	—	—	—	—	
GRD_0	Initialized	—	—	—	—	
TCR_1	Initialized	—	—	—	—	Tim
TIORA_1	Initialized	—	—	—	—	
TIORC_1	Initialized	—	—	—	—	
TSR_1	Initialized	—	—	—	—	
TIER_1	Initialized	—	—	—	—	
POCR_1	Initialized	—	—	—	—	
TCNT_1	Initialized	—	—	—	—	
GRA_1	Initialized	—	—	—	—	
GRB_1	Initialized	—	—	—	—	
GRC_1	Initialized	—	—	—	—	
GRD_1	Initialized	—	—	—	—	

RHRDR	Initialized	—	—	—	—	—	
RWKDR	—	—	—	—	—	—	
RTCCR1	—	—	—	—	—	—	
RTCCR2	—	—	—	—	—	—	
RTCCSR	Initialized	—	—	—	—	—	
LVDCR	Initialized	—	—	—	—	—	LVD
LVDSR	Initialized	—	—	—	—	—	(opti
SMR_2	Initialized	—	—	Initialized	Initialized	Initialized	SCI3
BRR_2	Initialized	—	—	Initialized	Initialized	Initialized	
SCR3_2	Initialized	—	—	Initialized	Initialized	Initialized	
TDR_2	Initialized	—	—	Initialized	Initialized	Initialized	
SSR_2	Initialized	—	—	Initialized	Initialized	Initialized	
RDR_2	Initialized	—	—	Initialized	Initialized	Initialized	
ICCR1	Initialized	—	—	—	—	—	IIC2
ICCR2	Initialized	—	—	—	—	—	
ICMR	Initialized	—	—	—	—	—	
ICIER	Initialized	—	—	—	—	—	
ICSR	Initialized	—	—	—	—	—	
SAR	Initialized	—	—	—	—	—	
ICDRT	Initialized	—	—	—	—	—	
ICDRR	Initialized	—	—	—	—	—	
TMB1	Initialized	—	—	—	—	—	Time
TCB1	Initialized	—	—	—	—	—	
TLB1	Initialized	—	—	—	—	—	

GRB	Initialized	—	—	—	—	—	
GRC	Initialized	—	—	—	—	—	
GRD	Initialized	—	—	—	—	—	
FLMCR1	Initialized	—	—	Initialized	Initialized	Initialized	RO
FLMCR2	Initialized	—	—	—	—	—	
FLPWCR	Initialized	—	—	—	—	—	
EBR1	Initialized	—	—	Initialized	Initialized	Initialized	
FENR	Initialized	—	—	—	—	—	
TCRV0	Initialized	—	—	Initialized	Initialized	Initialized	Tim
TCSRV	Initialized	—	—	Initialized	Initialized	Initialized	
TCORA	Initialized	—	—	Initialized	Initialized	Initialized	
TCORB	Initialized	—	—	Initialized	Initialized	Initialized	
TCNTV	Initialized	—	—	Initialized	Initialized	Initialized	
TCRV1	Initialized	—	—	Initialized	Initialized	Initialized	
SMR	Initialized	—	—	Initialized	Initialized	Initialized	SCI
BRR	Initialized	—	—	Initialized	Initialized	Initialized	
SCR3	Initialized	—	—	Initialized	Initialized	Initialized	
TDR	Initialized	—	—	Initialized	Initialized	Initialized	
SSR	Initialized	—	—	Initialized	Initialized	Initialized	
RDR	Initialized	—	—	Initialized	Initialized	Initialized	
ADDRA	Initialized	—	—	Initialized	Initialized	Initialized	A/D
ADDRB	Initialized	—	—	Initialized	Initialized	Initialized	
ADDRC	Initialized	—	—	Initialized	Initialized	Initialized	

TMWD	Initialized	—	—	—	—	—
ABRKCR	Initialized	—	—	—	—	—
ABRKSR	Initialized	—	—	—	—	—
BARH	Initialized	—	—	—	—	—
BARL	Initialized	—	—	—	—	—
BDRH	Initialized	—	—	—	—	—
BDRL	Initialized	—	—	—	—	—
BARE	Initialized	—	—	—	—	—
PUCR1	Initialized	—	—	—	—	—
PUCR5	Initialized	—	—	—	—	—
PDR1	Initialized	—	—	—	—	—
PDR2	Initialized	—	—	—	—	—
PDR3	Initialized	—	—	—	—	—
PDR5	Initialized	—	—	—	—	—
PDR6	Initialized	—	—	—	—	—
PDR7	Initialized	—	—	—	—	—
PDR8	Initialized	—	—	—	—	—
PDR9	Initialized	—	—	—	—	—
PDRB	Initialized	—	—	—	—	—
PMR1	Initialized	—	—	—	—	—
PMR5	Initialized	—	—	—	—	—
PMR3	Initialized	—	—	—	—	—
PCR1	Initialized	—	—	—	—	—

SYSCR1	Initialized	—	—	—	—	—	
SYSCR2	Initialized	—	—	—	—	—	
IEGR1	Initialized	—	—	—	—	—	Inte
IEGR2	Initialized	—	—	—	—	—	
IENR1	Initialized	—	—	—	—	—	
IENR2	Initialized	—	—	—	—	—	
IRR1	Initialized	—	—	—	—	—	
IRR2	Initialized	—	—	—	—	—	
IWPR	Initialized	—	—	—	—	—	
MSTCR1	Initialized	—	—	—	—	—	Pov
MSTCR2	Initialized	—	—	—	—	—	

Notes: — is not initialized

1. LVDC: Low-voltage detection circuits (optional)
2. WDT: Watchdog timer

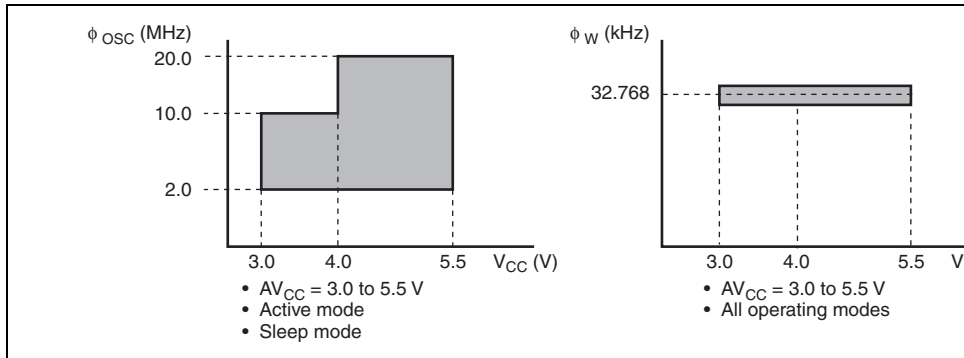
	and X1		
	Port B		-0.3 to $AV_{CC} + 0.3$ V
	X1		-0.3 to 4.3 V
Operating temperature	T_{opr}	-20 to +75	°C
Storage temperature	T_{stg}	-55 to +125	°C

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

23.2 Electrical Characteristics (F-ZTAT™ Version)

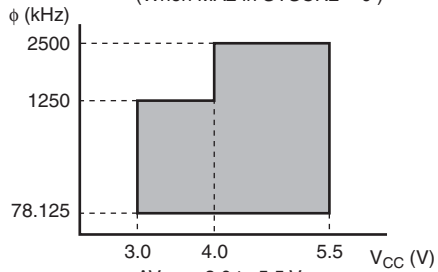
23.2.1 Power Supply Voltage and Operating Ranges

Power Supply Voltage and Oscillation Frequency Range



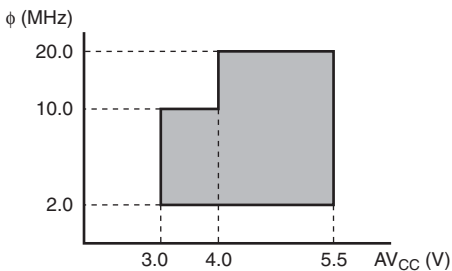
- Active mode
- Sleep mode
(When MA2 in SYSCR2 = 0)

- Subactive mode
- Subsleep mode

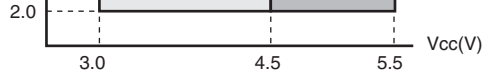


- AV_{CC} = 3.0 to 5.5 V
- Active mode
- Sleep mode
(When MA2 in SYSCR2 = 1)

Analog Power Supply Voltage and A/D Converter Accuracy Guarantee Range



- V_{CC} = 3.0 to 5.5 V
- Active mode
- Sleep mode



- Operation guarantee range
- Operation guarantee range except A/D conversion accuracy

ADTRG, TMRIV, TMCIV, FTIOA0 to FTIOD0, FTIOA1 to FTIOD1, FTIOA to FTIOD, SCK3, SCK3_2, SCK3_3, TRGV, FTCI, TMIB1		$V_{CC} \times 0.9$	—	$V_{CC} + 0.3$	
RXD, RXD_2, RXD_3, SCL, SDA, P10 to P12, P14 to P17, P20 to P24, P30 to P37, P50 to P57, P60 to P67, P70 to P72, P74 to P77, P80 to P87, P90 to P97	$V_{CC} = 4.0 \text{ to } 5.5 \text{ V}$	$V_{CC} \times 0.7$	—	$V_{CC} + 0.3$	V
		$V_{CC} \times 0.8$	—	$V_{CC} + 0.3$	
PB0 to PB7	$V_{CC} = 4.0 \text{ to } 5.5 \text{ V}$	$V_{CC} \times 0.7$	—	$AV_{CC} + 0.3$	V
		$V_{CC} \times 0.8$	—	$AV_{CC} + 0.3$	
OSC1	$V_{CC} = 4.0 \text{ to } 5.5 \text{ V}$	$V_{CC} - 0.5$	—	$V_{CC} + 0.3$	V
		$V_{CC} - 0.3$	—	$V_{CC} + 0.3$	

Note: Connect the TEST pin to Vss.

SCK3_2,
 SCK3_3, TRGV,
 FTCl, TMIB1

RXD, RXD_2, RXD_3, SCL, SDA, P10 to P12, P14 to P17, P20 to P24, P30 to P37, P50 to P57, P60 to P67, P70 to P72, P74 to P77, P80 to P87, P90 to P97	$V_{cc} = 4.0 \text{ to } 5.5 \text{ V}$	-0.3	—	$V_{cc} \times 0.3$	V
		-0.3	—	$V_{cc} \times 0.2$	
PB0 to PB7	$V_{cc} = 4.0 \text{ to } 5.5 \text{ V}$	-0.3	—	$V_{cc} \times 0.3$	V
		-0.3	—	$V_{cc} \times 0.2$	
OSC1	$V_{cc} = 4.0 \text{ to } 5.5 \text{ V}$	-0.3	—	0.5	V
		-0.3	—	0.3	

		P56, P57	$4.0\text{ V} \leq V_{CC} \leq 5.5\text{ V}$ $-I_{OH} = 0.1\text{ mA}$	$V_{CC} - 2.5$	—	—	V
			$3.0\text{ V} \leq V_{CC} < 4.0\text{ V}$ $-I_{OH} = 0.1\text{ mA}$	$V_{CC} - 2.0$	—	—	
Output low voltage	V_{OL}	P10 to P12, P14 to P17, P20 to P24, P30 to P37, P50 to P57, P70 to P72, P74 to P77, P85 to P87, P90 to P97	$V_{CC} = 4.0\text{ to }5.5\text{ V}$ $I_{OL} = 1.6\text{ mA}$ $I_{OL} = 0.4\text{ mA}$	—	—	0.6	V
		P60 to P67, P80 to P84	$V_{CC} = 4.0\text{ to }5.5\text{ V}$ $I_{OL} = 20.0\text{ mA}$ $V_{CC} = 4.0\text{ to }5.5\text{ V}$ $I_{OL} = 10.0\text{ mA}$ $V_{CC} = 4.0\text{ to }5.5\text{ V}$ $I_{OL} = 1.6\text{ mA}$ $I_{OL} = 0.4\text{ mA}$	—	—	1.5	V
						1.0	
						0.4	
						0.4	
		SCL, SDA	$V_{CC} = 4.0\text{ to }5.5\text{ V}$ $I_{OL} = 6.0\text{ mA}$ $I_{OL} = 3.0\text{ mA}$	—	—	0.6	
						0.4	

RXD_2, SCK3_2,
 RXD_3, SCK3_3,
 SCL, SDA,
 TMIB1, FTCl

		P10 to P12, P14 to P17, P20 to P24, P30 to P37, P50 to P57, P60 to P67, P70 to P72, P74 to P77, P80 to P87, P90 to P97	$V_{IN} = 0.5 \text{ V}$ or higher ($V_{CC} - 0.5 \text{ V}$)	—	—	1.0	μA
		PB0 to PB7	$V_{IN} = 0.5 \text{ V}$ or higher ($AV_{CC} - 0.5 \text{ V}$)	—	—	1.0	μA
Pull-up MOS current	$-I_p$	P10 to P12, P14 to P17,	$V_{CC} = 5.0 \text{ V}$, $V_{IN} = 0.0 \text{ V}$	50.0	—	300.0	μA
		P50 to P55	$V_{CC} = 3.0 \text{ V}$, $V_{IN} = 0.0 \text{ V}$	—	60.0	—	
Pull-up MOS resistance	R_{RES}	RES		—	150	—	$\text{k}\Omega$
Input capaci- tance	C_{in}	All input pins except power supply pins	$f = 1 \text{ MHz}$, $V_{IN} = 0.0 \text{ V}$, $T_a = 25^\circ\text{C}$	—	—	15.0	pF

			Active mode 2	—	1.3	—		
			$V_{CC} = 3.0\text{ V}$, $f_{OSC} = 10\text{ MHz}$					
Sleep mode supply current	I_{SLEEP1}	V_{CC}	Sleep mode 1	—	18.0	23.0	mA	*
			$V_{CC} = 5.0\text{ V}$, $f_{OSC} = 20\text{ MHz}$					
	I_{SLEEP2}	V_{CC}	Sleep mode 1	—	8.0	—		*
			$V_{CC} = 3.0\text{ V}$, $f_{OSC} = 10\text{ MHz}$					
Subactive mode supply current	I_{SUB}	V_{CC}	Sleep mode 2	—	2.1	3.1	mA	*
			$V_{CC} = 5.0\text{ V}$, $f_{OSC} = 20\text{ MHz}$					
Subactive mode supply current	I_{SUB}	V_{CC}	Sleep mode 2	—	1.2	—		*
			$V_{CC} = 3.0\text{ V}$, $f_{OSC} = 10\text{ MHz}$					
Subactive mode supply current	I_{SUB}	V_{CC}	$V_{CC} = 3.0\text{ V}$ 32-kHz crystal resonator ($\phi_{SUB} = \phi_W/2$)	—	35.0	70.0	μA	*
			$V_{CC} = 3.0\text{ V}$ 32-kHz crystal resonator ($\phi_{SUB} = \phi_W/8$)	—	25.0	—		*
Subsleep mode supply current	I_{SUBSP}	V_{CC}	$V_{CC} = 3.0\text{ V}$ 32-kHz crystal resonator ($\phi_{SUB} = \phi_W/2$)	—	25.0	50.0	μA	*

Note: * Pin states during supply current measurement are given below (excluding CPU pull-up MOS transistors and output buffers).

Mode	$\overline{\text{RES}}$ Pin	Internal State	Other Pins	Oscillator Pins
Active mode 1	V_{CC}	Operates	V_{CC}	Main clock: ceramic or crystal
Active mode 2		Operates ($\phi_{osc}/64$)		Subclock: Pin X1 = V_{SS}
Sleep mode 1	V_{CC}	Only timers operate	V_{CC}	
Sleep mode 2		Only timers operate ($\phi_{osc}/64$)		
Subactive mode	V_{CC}	Operates	V_{CC}	Main clock: ceramic or crystal
Subsleep mode	V_{CC}	Only timers operate	V_{CC}	Subclock: crystal resonator
Standby mode	V_{CC}	CPU and timers both stop	V_{CC}	Main clock: ceramic or crystal Subclock: Pin X1 = V_{SS}

		Output pins except port 6, P80 to P84, SCL, and SDA		—	—	0.5	
		Port 6, P80 to P84		—	—	10.0	
		SCL, SDA		—	—	6.0	
Allowable output low current (total)	ΣI_{OL}	Output pins except port 6, P80 to P84, SCL, and SDA	$V_{CC} = 4.0 \text{ to } 5.5 \text{ V}$	—	—	40.0	
		Port 6, P80 to P84, SCL, and SDA		—	—	80.0	
		Output pins except port 6, P80 to P84, SCL, and SDA		—	—	20.0	
		Port 6, P80 to P84, SCL, and SDA		—	—	40.0	
Allowable output high current (per pin)	$ -I_{OH} $	All output pins	$V_{CC} = 4.0 \text{ to } 5.5 \text{ V}$	—	—	2.0	
				—	—	0.2	
Allowable output high current (total)	$ -\Sigma I_{OH} $	All output pins	$V_{CC} = 4.0 \text{ to } 5.5 \text{ V}$	—	—	30.0	
				—	—	8.0	

System clock (ϕ) cycle time	t_{cyc}			1	—	64	t_{OSC}
				—	—	12.8	μs
Subclock oscillation frequency	f_W	X1, X2		—	32.768	—	kHz
Watch clock (ϕ_W) cycle time	t_W	X1, X2		—	30.5	—	μs
Subclock (ϕ_{SUB}) cycle time	t_{subcyc}			2	—	8	t_W
Instruction cycle time				2	—	—	t_{cyc} t_{subcyc}
Oscillation stabilization time (crystal resonator)	t_{rc}	OSC1, OSC2		—	—	10.0	ms
Oscillation stabilization time (ceramic resonator)	t_{rc}	OSC1, OSC2		—	—	5.0	ms
Oscillation stabilization time	t_{rcx}	X1, X2		—	—	2.0	s
External clock high width	t_{CPH}	OSC1	$V_{CC} = 4.0$ to 5.5 V	20.0	—	—	ns
				40.0	—	—	
External clock low width	t_{CPL}	OSC1	$V_{CC} = 4.0$ to 5.5 V	20.0	—	—	ns
				40.0	—	—	
External clock rise time	t_{CPr}	OSC1	$V_{CC} = 4.0$ to 5.5 V	—	—	10.0	ns
				—	—	15.0	
External clock fall time	t_{CPl}	OSC1	$V_{CC} = 4.0$ to 5.5 V	—	—	10.0	ns
				—	—	15.0	

WKP0 to
 $\overline{\text{WKP5}}$,
 TMCIV,
 TMRIV,
 TRGV,
 $\overline{\text{ADTRG}}$,
 FTIOA0 to
 FTIOD0,
 FTIOA1 to
 FTIOD1,
 FTIOA to
 FTIOD, FTCI

Input pin low width	t_{IL}	$\overline{\text{NMI}}$, TMBI1, $\overline{\text{IRQ0}}$ to $\overline{\text{IRQ3}}$, $\overline{\text{WKP0}}$ to $\overline{\text{WKP5}}$, TMCIV, TMRIV, TRGV, $\overline{\text{ADTRG}}$, FTIOA0 to FTIOD0, FTIOA1 to FTIOD1, FTIOA to FTIOD, FTCI	2	—	—	t_{cyc} t_{subcyc}
---------------------	-----------------	---	---	---	---	---

- Notes: 1. When an external clock is input, the minimum system clock oscillation frequency is 1.0 MHz.
2. Determined by the MA2, MA1, MA0, SA1, and SA0 bits in the system control register (SYSCR2).

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	—	—	250
		5.5 V	—	—	300

Input clock pulse width	t_{SCKW}	SCK3		0.4	—	0.6	t_{Seyc}
Transmit data delay time (clocked synchronous)	t_{TXD}	TXD	$V_{CC} = 4.0 \text{ to } 5.5 \text{ V}$	—	—	1	t_{cyc}
				—	—	1	
Receive data setup time (clocked synchronous)	t_{RXS}	RXD	$V_{CC} = 4.0 \text{ to } 5.5 \text{ V}$	50.0	—	—	ns
				100.0	—	—	
Receive data hold time (clocked synchronous)	t_{RXH}	RXD	$V_{CC} = 4.0 \text{ to } 5.5 \text{ V}$	50.0	—	—	ns
				100.0	—	—	

		AN7					
Analog power supply current	AI_{OPE}	AV_{CC}	$AV_{CC} = 5.0\text{ V}$ $f_{OSC} = 20\text{ MHz}$	—	—	2.0	mA
	AI_{STOP1}	AV_{CC}		—	50	—	μA
	AI_{STOP2}	AV_{CC}		—	—	5.0	μA
Analog input capacitance	C_{AIN}	AN0 to AN7		—	—	30.0	pF
Allowable signal source impedance	R_{AIN}	AN0 to AN7		—	—	5.0	k Ω
Resolution (data length)				10	10	10	Bit
Conversion time (single mode)			$AV_{CC} = 3.0\text{ to }5.5\text{ V}$	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)			$AV_{CC} = 4.0\text{ to }5.5\text{ V}$	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. I_{STOP1} is the current in active and sleep modes while the A/D converter is idle.
3. I_{STOP2} is the current at reset and in standby, subactive, and subsleep modes while the A/D converter is idle.

23.2.5 Watchdog Timer Characteristics

Table 23.7 Watchdog Timer Characteristics

$V_{CC} = 3.0$ to 5.5 V, $V_{SS} = 0.0$ V, $T_a = -20$ to $+75^\circ\text{C}$, unless otherwise indicated.

Item	Symbol	Applicable Pins	Test Condition	Values			Unit	Notes
				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 generated when the internal oscillator is selected.

Reprogramming count		N_{WEC}	1000	10000	—	
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	$1 \leq n \leq 6$	28	30	32
		z2	$7 \leq n \leq 1000$	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		

Wait time after dummy write* ¹	ϵ	2	—	—
Wait time after EV bit clear* ¹	η	4	—	—
Wait time after SWE bit clear* ¹	θ	100	—	—
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 128 bytes. (Indicates the total time for which the P bit in memory control register 1 (FLMCR1) is set. The program-verify time is not included.)
 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 included.)
 4. Maximum programming time (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 z1 and z3, so that it does not exceed the maximum programming time (t_p (max.)). The programming count (n) should be changed as follows according to the actual set value of the programming count (n).

Programming count (n)

$$1 \leq n \leq 6 \quad z1 = 30 \mu\text{s}$$

$$7 \leq n \leq 1000 \quad z2 = 200 \mu\text{s}$$

6. Maximum erase time (t_E (max.)) = wait time after E bit setting (z) × maximum erase count (N)
7. Set the maximum erase count (N) according to the actual set value of (z), so that the maximum erase time (t_E (max.)) does not exceed the maximum erase time (t_E (max.)).

Reset detection voltage 1* ¹	Vreset1	LVDSSEL = 0	—	2.3	2.7
Reset detection voltage 2* ²	Vreset2	LVDSSEL = 1	3.0	3.6	4.2
Lower-limit voltage of LVDR operation* ³	V _{LVDRmin}	—	1.0	—	—
LVD stabilization time	t _{LVDRON}	—	50	—	—
Supply current in standby mode	I _{STBY}	LVDE = 1, Vcc = 5.0 V, When a 32- kHz crystal resonator is not used	—	—	350

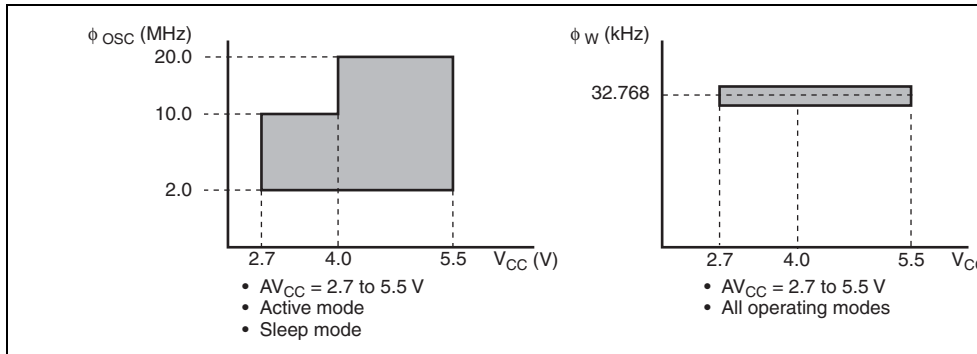
- Notes:
1. This voltage should be used when the falling and rising voltage detection function is used.
 2. Select the low-voltage reset 2 when only the low-voltage detection reset is used.
 3. When the power-supply voltage (Vcc) falls below V_{LVDRmin} = 1.0 V and then rises, a reset may not occur.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

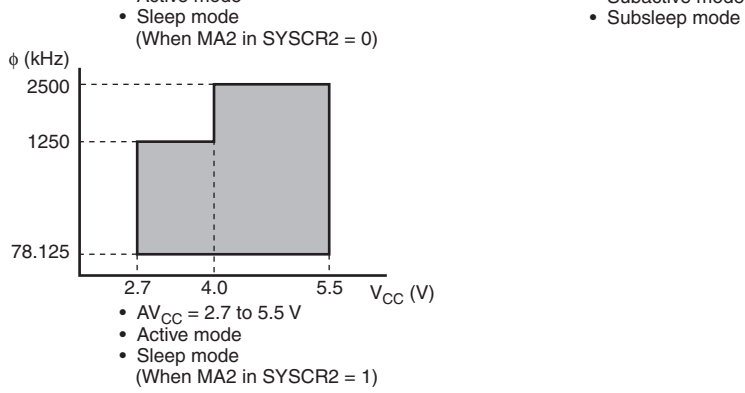
charge of the $\overline{\text{RES}}$ pin is removed completely. In order to remove charge of the pin, it is recommended that the diode be placed in the V_{CC} side. If the power-supply voltage (V_{CC}) rises from the point over 100 mV, a power-on reset may not occur.

23.3 Electrical Characteristics (Masked ROM Version)

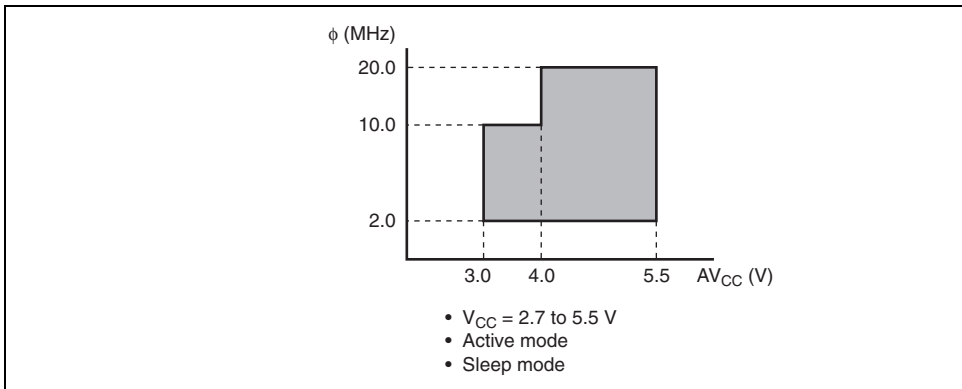
23.3.1 Power Supply Voltage and Operating Ranges

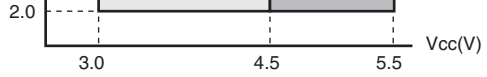
Power Supply Voltage and Oscillation Frequency Range





Analog Power Supply Voltage and A/D Converter Accuracy Guarantee Range





- Operation guarantee range
- Operation guarantee range except A/D conversion accuracy

ADTRG, TMRIV, TMCIV, FTIOA0 to FTIOD0, FTIOA1 to FTIOD1, FTIOA to FTIOD, SCK3, SCK3_2, SCK3_3, TRGV, FTCI, TMIB1		$V_{cc} \times 0.9$	—	$V_{cc} + 0.3$	
RXD, RXD_2, RXD_3, SCL, SDA, P10 to P12, P14 to P17, P20 to P24, P30 to P37	$V_{cc} = 4.0 \text{ to } 5.5 \text{ V}$	$V_{cc} \times 0.7$	—	$V_{cc} + 0.3$	V
P50 to P57, P60 to P67, P70 to P72, P74 to P77, P80 to P87, P90 to P97		$V_{cc} \times 0.8$	—	$V_{cc} + 0.3$	
PB0 to PB7	$V_{cc} = 4.0 \text{ to } 5.5 \text{ V}$	$V_{cc} \times 0.7$	—	$AV_{cc} + 0.3$	V
		$V_{cc} \times 0.8$	—	$AV_{cc} + 0.3$	
OSC1	$V_{cc} = 4.0 \text{ to } 5.5 \text{ V}$	$V_{cc} - 0.5$	—	$V_{cc} + 0.3$	V
		$V_{cc} - 0.3$	—	$V_{cc} + 0.3$	

Note: Connect the TEST pin to Vss.

CSK3_2,
 SCK3_3, TRGV,
 FTCl, TMIB1

RXD, RXD_2, RXD_3, SCL, SDA, P10 to P12, P14 to P17, P20 to P24, P30 to P37, P50 to P57, P60 to P67,. P70 to P72, P74 to P77, P80 to P87, P90 to P97	$V_{cc} = 4.0 \text{ to } 5.5 \text{ V}$	-0.3	—	$V_{cc} \times 0.3$	V
		-0.3	—	$V_{cc} \times 0.2$	
PB0 to PB7	$V_{cc} = 4.0 \text{ to } 5.5 \text{ V}$	-0.3	—	$V_{cc} \times 0.3$	V
		-0.3	—	$V_{cc} \times 0.2$	
OSC1	$V_{cc} = 4.0 \text{ to } 5.5 \text{ V}$	-0.3	—	0.5	V
		-0.3	—	0.3	

		P56, P57	$4.0\text{ V} \leq V_{CC} \leq 5.5\text{ V}$ $-I_{OH} = 0.1\text{ mA}$	$V_{CC} - 2.5$	—	—	V
			$2.0\text{ V} \leq V_{CC} < 4.0\text{ V}$ $-I_{OH} = 0.1\text{ mA}$	$V_{CC} - 2.0$	—	—	
Output low voltage	V_{OL}	P10 to P12, P14 to P17, P20 to P24, P30 to P37,	$V_{CC} = 4.0\text{ to }5.5\text{ V}$ $I_{OL} = 1.6\text{ mA}$	—	—	0.6	V
		P50 to P57, P70 to P72, P74 to P77, P85 to P87, P90 to P97	$I_{OL} = 0.4\text{ mA}$	—	—	0.4	
		P60 to P67, P80 to P84	$V_{CC} = 4.0\text{ to }5.5\text{ V}$ $I_{OL} = 20.0\text{ mA}$	—	—	1.5	V
			$V_{CC} = 4.0\text{ to }5.5\text{ V}$ $I_{OL} = 10.0\text{ mA}$	—	—	1.0	
			$V_{CC} = 4.0\text{ to }5.5\text{ V}$ $I_{OL} = 1.6\text{ mA}$	—	—	0.4	
			$I_{OL} = 0.4\text{ mA}$	—	—	0.4	
		SCL, SDA	$V_{CC} = 4.0\text{ to }5.5\text{ V}$ $I_{OL} = 6.0\text{ mA}$	—	—	0.6	V
			$I_{OL} = 3.0\text{ mA}$	—	—	0.4	

RXD_2, SCK3_2,
 RXD_3, SCK3_3,
 SCL, SDA,
 TMIB1, FTCl

		P10 to P12, P14 to P17, P20 to P24, P30 to P37, P50 to P57, P60 to P67, P70 to P72, P74 to P77, P80 to P87, P90 to P97	$V_{IN} = 0.5 \text{ V}$ or higher ($V_{CC} - 0.5 \text{ V}$)	—	—	1.0	μA
		PB0 to PB7	$V_{IN} = 0.5 \text{ V}$ or higher ($AV_{CC} - 0.5 \text{ V}$)	—	—	1.0	μA
Pull-up MOS current	$-I_p$	P10 to P12, P14 to P17,	$V_{CC} = 5.0 \text{ V}$, $V_{IN} = 0.0 \text{ V}$	50.0	—	300.0	μA
		P50 to P55	$V_{CC} = 3.0 \text{ V}$, $V_{IN} = 0.0 \text{ V}$	—	60.0	—	
Pull-up MOS resistance	R_{RES}	\overline{RES}		—	150	—	$\text{k}\Omega$
Input capaci- tance	C_{in}	All input pins except power supply pins	$f = 1 \text{ MHz}$, $V_{IN} = 0.0 \text{ V}$, $T_a = 25^\circ\text{C}$	—	—	15.0	pF

			Active mode 2	—	1.3	—	
			$V_{CC} = 3.0\text{ V}$, $f_{OSC} = 10\text{ MHz}$				
Sleep mode supply current	I_{SLEEP1}	V_{CC}	Sleep mode 1	—	18.0	23.0	mA
			$V_{CC} = 5.0\text{ V}$, $f_{OSC} = 20\text{ MHz}$				
	I_{SLEEP2}	V_{CC}	Sleep mode 1	—	8.0	—	
			$V_{CC} = 3.0\text{ V}$, $f_{OSC} = 10\text{ MHz}$				
Subactive mode supply current	I_{SUB}	V_{CC}	Sleep mode 2	—	2.1	3.1	mA
			$V_{CC} = 5.0\text{ V}$, $f_{OSC} = 20\text{ MHz}$				
	I_{SUB}	V_{CC}	Sleep mode 2	—	1.2	—	
			$V_{CC} = 3.0\text{ V}$, $f_{OSC} = 10\text{ MHz}$				
Subactive mode supply current	I_{SUB}	V_{CC}	$V_{CC} = 3.0\text{ V}$ 32-kHz crystal resonator ($\phi_{SUB} = \phi_W/2$)	—	35.0	70.0	μA
			$V_{CC} = 3.0\text{ V}$ 32-kHz crystal resonator ($\phi_{SUB} = \phi_W/8$)	—	25.0	—	
Subsleep mode supply current	I_{SUBSP}	V_{CC}	$V_{CC} = 3.0\text{ V}$ 32-kHz crystal resonator ($\phi_{SUB} = \phi_W/2$)	—	25.0	50.0	μA

Mode	RES Pin	Internal State	Other Pins	Oscillator Pin
Active mode 1	V_{CC}	Operates	V_{CC}	Main clock: ceramic or cry- resonator
Active mode 2		Operates ($\phi_{OSC}/64$)		Subclock: Pin X1 = V_{SS}
Sleep mode 1	V_{CC}	Only timers operate	V_{CC}	
Sleep mode 2		Only timers operate ($\phi_{OSC}/64$)		
Subactive mode	V_{CC}	Operates	V_{CC}	Main clock: ceramic or cry- resonator
Subsleep mode	V_{CC}	Only timers operate	V_{CC}	Subclock: crystal resonat
Standby mode	V_{CC}	CPU and timers both stop	V_{CC}	Main clock: ceramic or cry- resonator Subclock: Pin X1 = V_{SS}

		Output pins except port 6, P80 to P84, SCL, and SDA		—	—	0.5
		Port 6, P80 to P84		—	—	10.0
		SCL, SDA		—	—	6.0
Allowable output low current (total)	ΣI_{OL}	Output pins except port 6, P80 to P84, SCL, and SDA	$V_{CC} = 4.0 \text{ to } 5.5 \text{ V}$	—	—	40.0
		Port 6, P80 to P84, SCL, and SDA		—	—	80.0
		Output pins except port 6, P80 to P84, SCL, and SDA		—	—	20.0
		Port 6, P80 to P84, SCL, and SDA		—	—	40.0
Allowable output high current (per pin)	$ -I_{OH} $	All output pins	$V_{CC} = 4.0 \text{ to } 5.5 \text{ V}$	—	—	2.0
				—	—	0.2
Allowable output high current (total)	$ -\Sigma I_{OH} $	All output pins	$V_{CC} = 4.0 \text{ to } 5.5 \text{ V}$	—	—	30.0
				—	—	8.0

System clock (ϕ) cycle time	t_{cyc}			1	—	64	t_{osc}	*2
				—	—	12.8	μs	
Subclock oscillation frequency	f_W	X1, X2		—	32.768	—	kHz	
Watch clock (ϕ_W) cycle time	t_W	X1, X2		—	30.5	—	μs	
Subclock (ϕ_{SUB}) cycle time	t_{subcyc}			2	—	8	t_W	*2
Instruction cycle time				2	—	—	t_{cyc} t_{subcyc}	
Oscillation stabilization time (crystal resonator)	t_{rc}	OSC1, OSC2		—	—	10.0	ms	
Oscillation stabilization time (ceramic resonator)	t_{rc}	OSC1, OSC2		—	—	5.0	ms	
Oscillation stabilization time	t_{rcx}	X1, X2		—	—	2.0	s	
External clock high width	t_{CPH}	OSC1	$V_{CC} = 4.0$ to 5.5 V	205.0	—	—	ns	F
				40.0	—	—		
External clock low width	t_{CPL}	OSC1	$V_{CC} = 4.0$ to 5.5 V	20.0	—	—	ns	
				40.0	—	—		
External clock rise time	t_{CPr}	OSC1	$V_{CC} = 4.0$ to 5.5 V	—	—	10.0	ns	
				—	—	15.0		
External clock fall time	t_{CPf}	OSC1	$V_{CC} = 4.0$ to 5.5 V	—	—	10.0	ns	
				—	—	15.0		

WKP0 to
 WKP5,
 TMCIV,
 TMRIV,
 TRGV,
 ADTRG,
 FTIOA0 to
 FTIOD0,
 FTIOA1 to
 FTIOD1,
 FTIOA to
 FTIOD, FTCI

Input pin low width	t_{IL}	NMI, TMBI1, IRQ0 to IRQ3, WKP0 to WKP5, TMCIV, TMRIV, TRGV, ADTRG, FTIOA0 to FTIOD0, FTIOA1 to FTIOD1, FTIOA to FTIOD, FTCI	2	—	—	t_{cyc} t_{subcyc}
---------------------	----------	---	---	---	---	---------------------------

- Notes: 1. When an external clock is input, the minimum system clock oscillation frequency is 1.0 MHz.
2. Determined by the MA2, MA1, MA0, SA1, and SA0 bits in the system control register (SYSCR2).

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_{SI}	$V_{CC} = 4.0$	—	—	250
		to 5.5 V	—	—	300

width

Transmit data delay time (clocked synchronous)	t_{TXD}	TXD	$V_{CC} = 4.0 \text{ to } 5.5 \text{ V}$	—	—	1	t_{cyc}	F
				—	—	1		
Receive data setup time (clocked synchronous)	t_{RXS}	RXD	$V_{CC} = 4.0 \text{ to } 5.5 \text{ V}$	50.0	—	—	ns	
				100.0	—	—		
Receive data hold time (clocked synchronous)	t_{RXH}	RXD	$V_{CC} = 4.0 \text{ to } 5.5 \text{ V}$	50.0	—	—	ns	
				100.0	—	—		

		AN7	0.3				
Analog power supply current	I_{OPE}	AV_{CC}	$AV_{CC} = 5.0\text{ V}$ $f_{OSC} = 20\text{ MHz}$	—	—	2.0	mA
	I_{STOP1}	AV_{CC}		—	50	—	μA
	I_{STOP2}	AV_{CC}		—	—	5.0	μA
Analog input capacitance	C_{AIN}	AN0 to AN7		—	—	30.0	pF
Allowable signal source impedance	R_{AIN}	AN0 to AN7		—	—	5.0	k Ω
Resolution (data length)				10	10	10	Bit
Conversion time (single mode)			$AV_{CC} = 2.7\text{ to }5.5\text{ V}$	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)			$AV_{CC} = 4.0\text{ to }5.5\text{ V}$	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. I_{STOP1} is the current in active and sleep modes while the A/D converter is idle.
3. I_{STOP2} is the current at reset and in standby, subactive, and subsleep modes while the A/D converter is idle.

23.3.5 Watchdog Timer Characteristics

Table 23.16 Watchdog Timer Characteristics

$V_{CC} = 2.7$ to 5.5 V, $V_{SS} = 0.0$ V, $T_a = -20$ to $+75^\circ\text{C}$, unless otherwise indicated.

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 generated when the internal oscillator is selected.

Reset detection voltage 1* ¹	Vreset1	LVDSSEL = 0	—	2.3	2.7
Reset detection voltage 2* ²	Vreset2	LVDSSEL = 1	3.0	3.6	4.2
Lower-limit voltage of LVDR operation* ³	$V_{LVDRmin}$	—	1.0	—	—
LVD stabilization time	t_{LVDRON}	—	50	—	—
Supply current in standby mode	I_{STBY}	LVDE = 1, Vcc = 5.0 V, When a 32- kHz crystal resonator is not used	—	—	350

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

charge of the $\overline{\text{RES}}$ 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-voltage (Vcc) rises from the point over 100 mV, a power-on reset may not occur.

23.4 Operation Timing

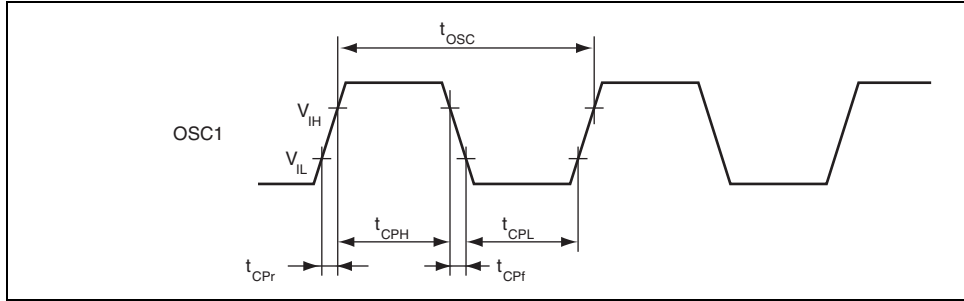


Figure 23.1 System Clock Input Timing

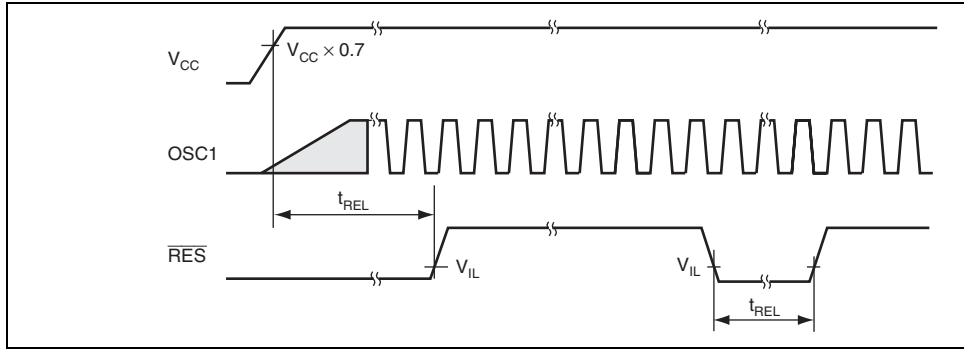
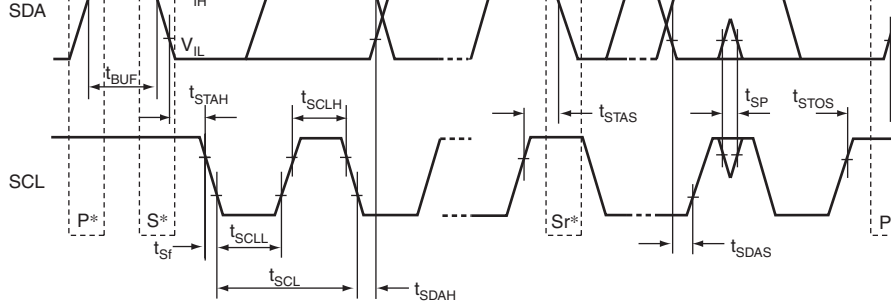


Figure 23.2 $\overline{\text{RES}}$ Low Width Timing



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

Figure 23.4 I²C Bus Interface Input/Output Timing

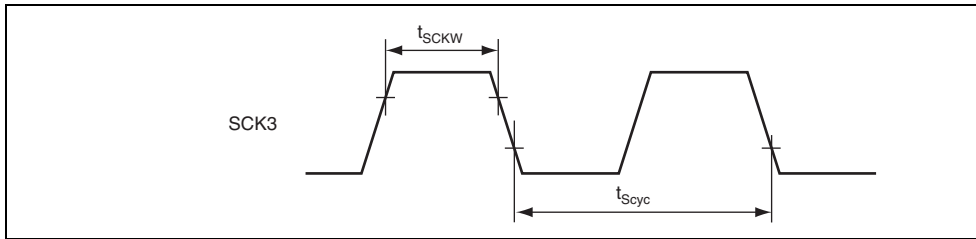
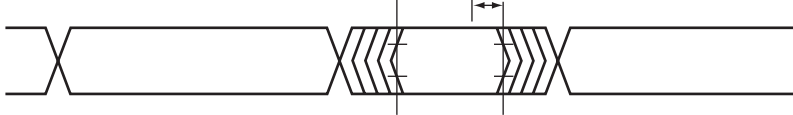


Figure 23.5 SCK3 Input Clock Timing

RXD
(receive data)



Note: * Output timing reference levels
Output high: $V_{OH} = 2.0\text{ V}$
Output low: $V_{OL} = 0.8\text{ V}$
Load conditions are shown in figure 23.7.

Figure 23.6 SCI Input/Output Timing in Clocked Synchronous Mode

23.5 Output Load Condition

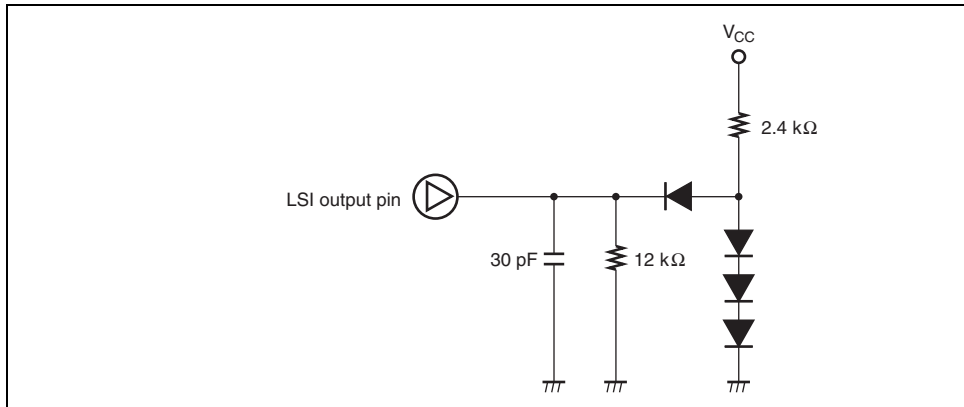


Figure 23.7 Output Load Circuit

Rd	General destination register
Rs	General source register
Rn	General register
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 transfer 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

0	Cleared to 0
1	Set to 1
—	Not affected by execution of the instruction
Δ	Varies depending on conditions, described in notes

Note: General registers include 8-bit registers (R0H to R7H and R0L to R7L) and 16-bit registers (R0 to R7 and E0 to E7).

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				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	—
MOVFP	MOVFP @aa:16, Rd	B				4					Cannot be used in this LSI	Cannot be used in this LSI					
MOVTP	MOVTP 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)	↕	↕	↕	

EXTS	EXTS.W Rd	W	2											(<bit 7> of Rd16) → (<bits 15 to 8> of Rd16)	—	—	↕	↕	0
	EXTS.L ERd	L	2											(<bit 15> of ERd32) → (<bits 31 to 16> of ERd32)	—	—	↕	↕	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													Rd8Rs8 → Rd8	—	—	↕	↕	0	—
	OR.W #xx:16, Rd	W	4													Rd16#xx:16 → Rd16	—	—	↕	↕	0	—
	OR.W Rs, Rd	W	2													Rd16Rs16 → 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				$C \rightarrow (\#xx:3 \text{ of } @ERd24)$	—	—	—	—
	BST #xx:3, @aa:8	B				4		$C \rightarrow (\#xx:3 \text{ of } @aa:8)$	—	—	—	—
BIST	BIST #xx:3, Rd	B	2					$\sim C \rightarrow (\#xx:3 \text{ of } Rd8)$	—	—	—	—
	BIST #xx:3, @ERd	B		4				$\sim C \rightarrow (\#xx:3 \text{ of } @ERd24)$	—	—	—	—
	BIST #xx:3, @aa:8	B				4		$\sim C \rightarrow (\#xx:3 \text{ of } @aa:8)$	—	—	—	—
BAND	BAND #xx:3, Rd	B	2					$C \wedge (\#xx:3 \text{ of } Rd8) \rightarrow C$	—	—	—	—
	BAND #xx:3, @ERd	B		4				$C \wedge (\#xx:3 \text{ of } @ERd24) \rightarrow C$	—	—	—	—
	BAND #xx:3, @aa:8	B				4		$C \wedge (\#xx:3 \text{ of } @aa:8) \rightarrow C$	—	—	—	—
BIAND	BIAND #xx:3, Rd	B	2					$C \wedge \sim (\#xx:3 \text{ of } Rd8) \rightarrow C$	—	—	—	—
	BIAND #xx:3, @ERd	B		4				$C \wedge \sim (\#xx:3 \text{ of } @ERd24) \rightarrow C$	—	—	—	—
	BIAND #xx:3, @aa:8	B				4		$C \wedge \sim (\#xx:3 \text{ of } @aa:8) \rightarrow C$	—	—	—	—
BOR	BOR #xx:3, Rd	B	2					$C \vee (\#xx:3 \text{ of } Rd8) \rightarrow C$	—	—	—	—
	BOR #xx:3, @ERd	B		4				$C \vee (\#xx:3 \text{ of } @ERd24) \rightarrow C$	—	—	—	—
	BOR #xx:3, @aa:8	B				4		$C \vee (\#xx:3 \text{ of } @aa:8) \rightarrow C$	—	—	—	—
BIOR	BIOR #xx:3, Rd	B	2					$C \vee \sim (\#xx:3 \text{ of } Rd8) \rightarrow C$	—	—	—	—
	BIOR #xx:3, @ERd	B		4				$C \vee \sim (\#xx:3 \text{ of } @ERd24) \rightarrow C$	—	—	—	—
	BIOR #xx:3, @aa:8	B				4		$C \vee \sim (\#xx:3 \text{ of } @aa:8) \rightarrow C$	—	—	—	—
BXOR	BXOR #xx:3, Rd	B	2					$C \oplus (\#xx:3 \text{ of } Rd8) \rightarrow C$	—	—	—	—
	BXOR #xx:3, @ERd	B		4				$C \oplus (\#xx:3 \text{ of } @ERd24) \rightarrow C$	—	—	—	—
	BXOR #xx:3, @aa:8	B				4		$C \oplus (\#xx:3 \text{ of } @aa:8) \rightarrow C$	—	—	—	—
BIXOR	BIXOR #xx:3, Rd	B	2					$C \oplus \sim (\#xx:3 \text{ of } Rd8) \rightarrow C$	—	—	—	—
	BIXOR #xx:3, @ERd	B		4				$C \oplus \sim (\#xx:3 \text{ of } @ERd24) \rightarrow C$	—	—	—	—
	BIXOR #xx:3, @aa:8	B				4		$C \oplus \sim (\#xx:3 \text{ of } @aa:8) \rightarrow C$	—	—	—	—

	JSR @n	—					2										PC → @-SP PC ← ERn						
JSR	JSR @aa:24	—								4							PC → @-SP PC ← aa:24						
	JSR @@aa:8	—											2				PC → @-SP PC ← @aa:8						
	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	—	—	—	—	—	—

Instruction code: 1st byte 2nd byte

AH	AL	BH	BL
----	----	----	----

AL	0	1	2	3	4	5	6	7	8	9	A	B	C
AH	NOP	Table A.2 (2)	STC	LDC	ORC	XORC	ANDC	LDC	ADD		Table A.2 (2)	Table A.2 (2)	Table A.2 (2)
	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)	Table A.2 (2)
2	MOV.B												
3	MOV												
4	BRA	BRN	BHI	BLS	BCC	BCS	BNE	BEQ	BVC	BVS	BPL	BMI	BGE
5	MULXU	DIVXU	MULXU	DIVXU	RTS	BSR	RTE	TRAPA	Table A.2 (2)		JMP		BSR
6	BSET	BNOT	BCLR	BTST	OR	XOR	AND	BST	Table A.2 (2)				MOV
7					BOR	BXOR	BAND	BISL	Table A.2 (2)	MOV	Table A.2 (2)	Table A.2 (2)	EEMOV
					BIOR	BIXOR	BIAND	BILD					
8	ADD												
9	ADDX												
A	CMP												
B	SUBX												
C	OR												
D	XOR												
E	AND												

Instruction code:

1st byte		2nd byte	
AH	AL	BH	BL

BH AH/AL	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					



CL AH ALBH BLCH	0		1		2		3		4		5		6		7		8		9		A		B							
	MULXS	MULXS	DIVXS	DIVXS	MULXS	MULXS	DIVXS	DIVXS	OR	XOR	AND	BOR	BOR	BOR	BOR	BOR	BOR	BOR	BOR	BOR	LDC	STC	LDC	STC						
01406																														
01C05	MULXS																													
01D05			DIVXS																											
01F06																														
7C06*1																														
7C07*1																														
7D06*1	BSET	BNOT		BCLR																										
7D07*1	BSET		BNOT		BCLR																									
7Eaa6*2																														
7Eaa7*2																														
7Faa6*2	BSET	BNOT		BCLR																										
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:

$$I = L = 2, \quad J = K = M = N = 0$$

From table A.3:

$$S_i = 2, \quad S_L = 2$$

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 ROM. On-chip RAM is used for stack area.

JSR @@ 30

From table A.4:

$$I = 2, \quad J = K = 1, \quad L = M = N = 0$$

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 22.1, 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	

	BCC d:16(BHS d:16)	2	
	BCS d:16(BLO d:16)	2	
	BNE d:16	2	
	BEQ d:16	2	
	BVC d:16	2	
	BVS d:16	2	
	BPL d:16	2	
	BMI d:16	2	
	BGE d:16	2	
	BLT d:16	2	
	BGT d:16	2	
	BLE d:16	2	
BCLR	BCLR #xx:3, Rd	1	
	BCLR #xx:3, @ERd	2	2
	BCLR #xx:3, @aa:8	2	2
	BCLR 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, @ERd	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, @EHd	2	1
	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	
DIVXS	DIVXS.B Rs, Rd	2	
	DIVXS.W Rs, ERd	2	
DIVXU	DIVXU.B Rs, Rd	1	
	DIVXU.W Rs, ERd	1	
EEPMOV	EEPMOV.B	2	$2n+2^{*1}$
	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:24	2		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 @ERS+, Rd	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
MOVFP	MOVFP @aa:16, Rd* ²	2	1
MOVTP	MOVTP Rs, @aa:16* ²	2	1

NOP	NOP	1	
NOT	NOT.B Rd	1	
	NOT.W Rd	1	
	NOT.L ERd	1	
OR	OR.B #xx:8, Rd	1	
	OR.B Rs, Rd	1	
	OR.W #xx:16, Rd	2	
	OR.W Rs, Rd	1	
	OR.L #xx:32, ERd	3	
	OR.L ERs, ERd	2	
ORC	ORC #xx:8, CCR	1	
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	

	SHAL.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 #xx:32, ERd	3
	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	POP, PUSH	—	—	—	—	—	—	—	—	—	—	—	—
	MOVFP, MOVTP	—	—	—	—	—	—	—	—	—	—	—	—
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	—	—	—	—	—	—	—	—	—	—
Logical operations	EXTU, EXTS	—	WL	—	—	—	—	—	—	—	—	—	—
	AND, OR, XOR	—	BWL	—	—	—	—	—	—	—	—	—	—
Shift operations	NOT	—	BWL	—	—	—	—	—	—	—	—	—	—
	Bit manipulations	—	BWL	—	—	—	—	—	—	—	—	—	—
Branching instructions	BCC, BSR	—	B	B	—	—	—	B	—	—	—	—	—
	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	—	—	—	—	—	—	—	—	—	—	—
Block data transfer instructions	NOP	—	—	—	—	—	—	—	—	—	—	—	—
		—	—	—	—	—	—	—	—	—	—	—	—

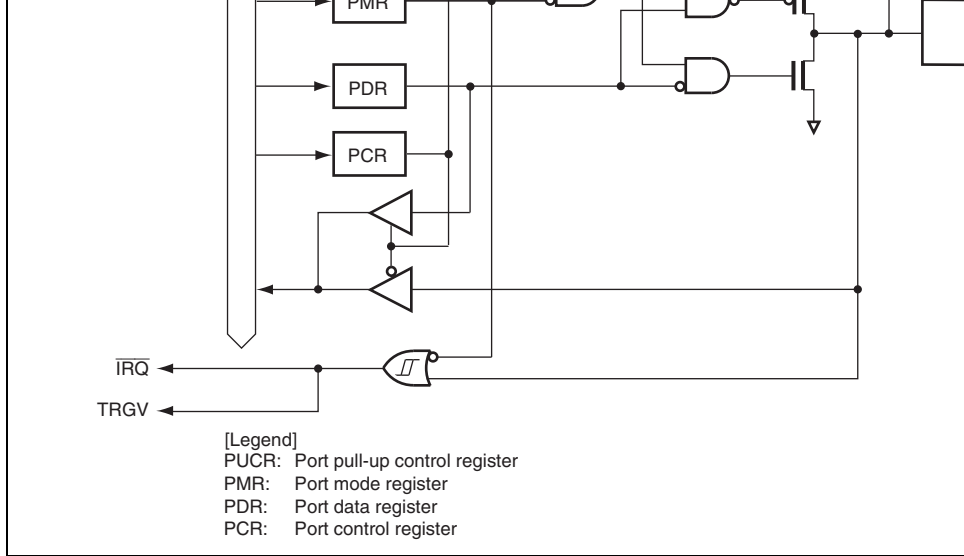
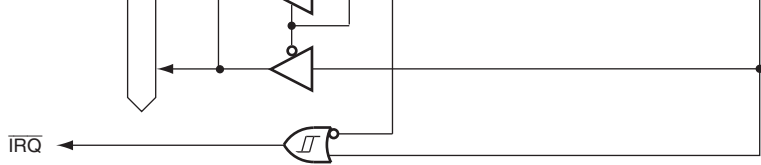


Figure B.1 Port 1 Block Diagram (P17)



[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, P16)

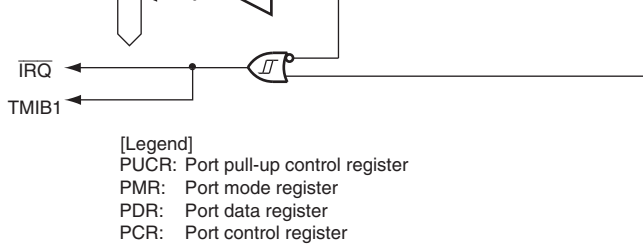


Figure B.3 Port 1 Block Diagram (P15)

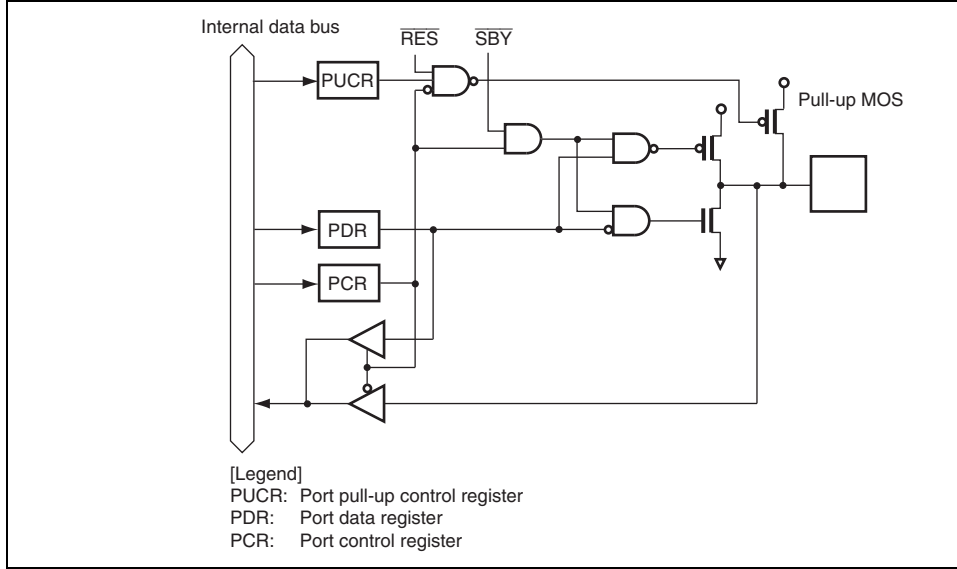


Figure B.4 Port 1 Block Diagram (P12)

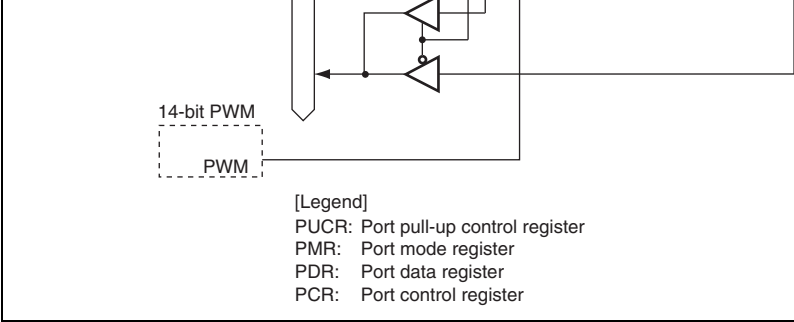


Figure B.5 Port 1 Block Diagram (P1)

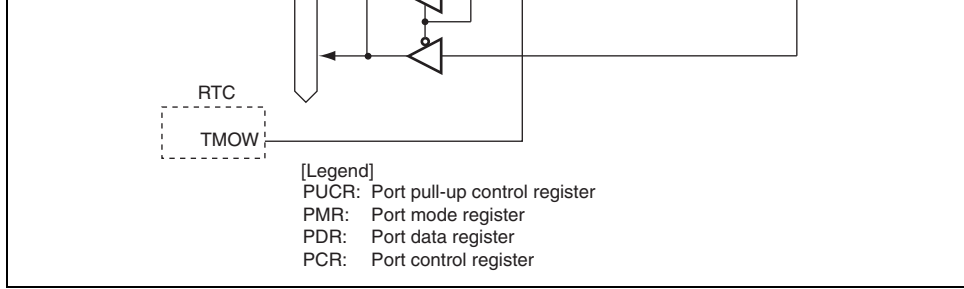


Figure B.6 Port 1 Block Diagram (P10)

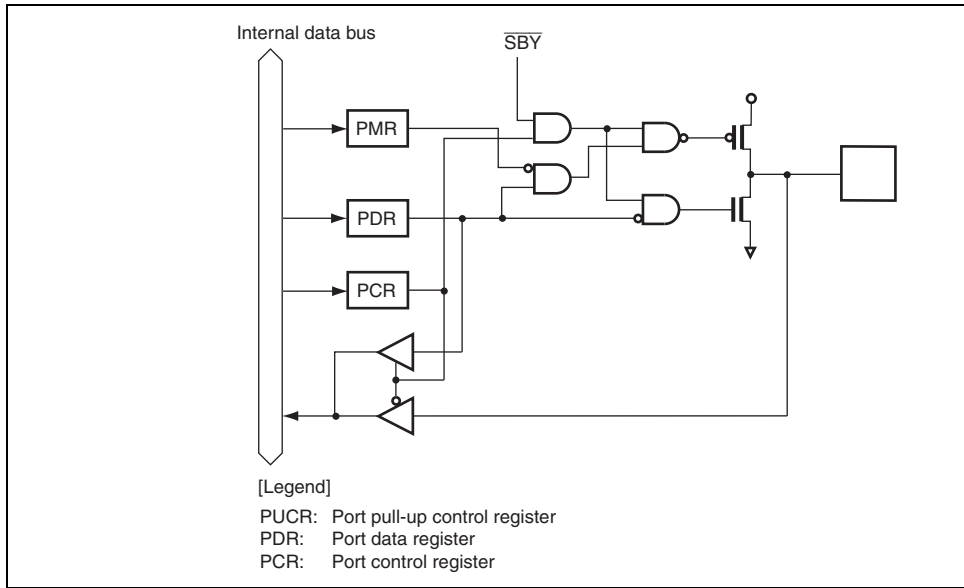


Figure B.7 Port 2 Block Diagram (P24, P23)

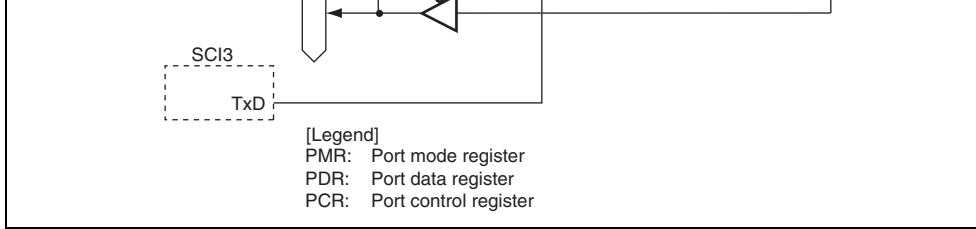


Figure B.8 Port 2 Block Diagram (P22)

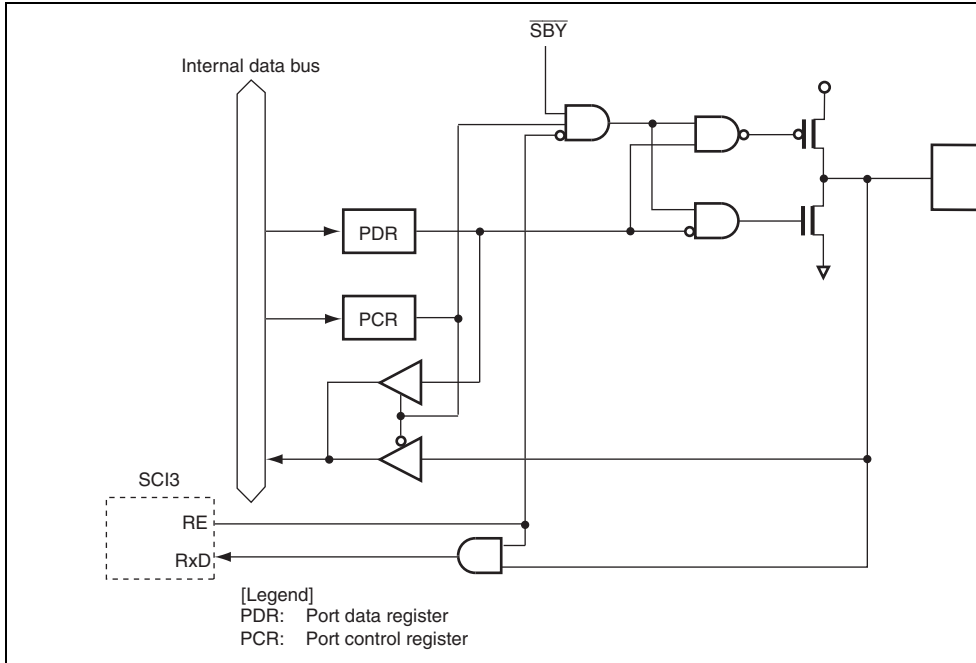


Figure B.9 Port 2 Block Diagram (P21)

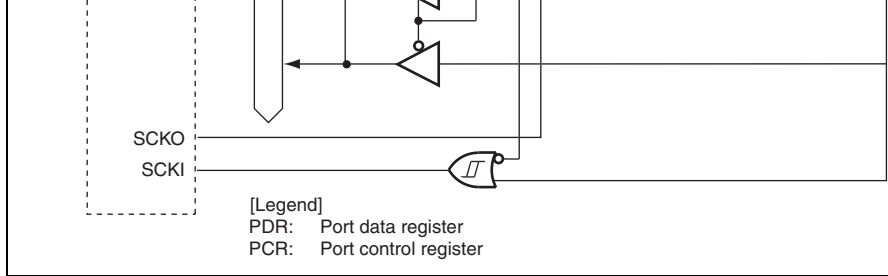


Figure B.10 Port 2 Block Diagram (P20)

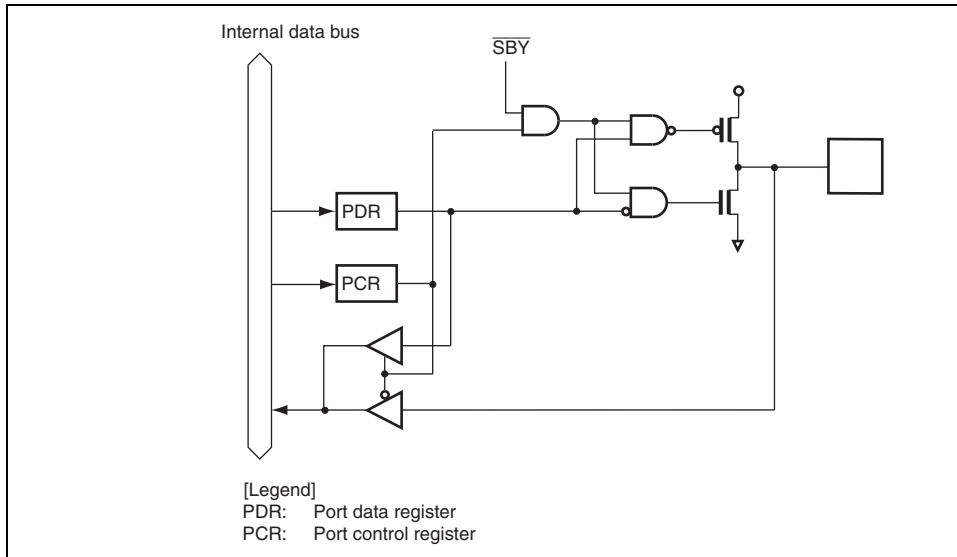


Figure B.11 Port 3 Block Diagram (P37 to P30)

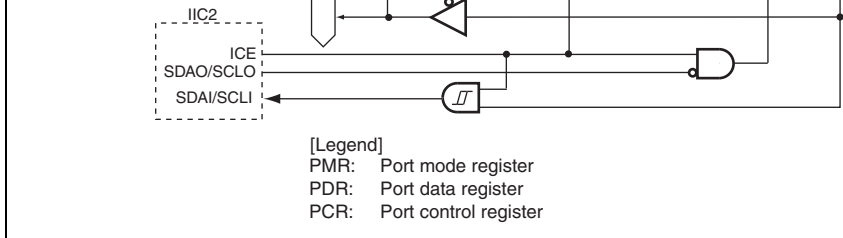


Figure B.12 Port 5 Block Diagram (P57, P56)

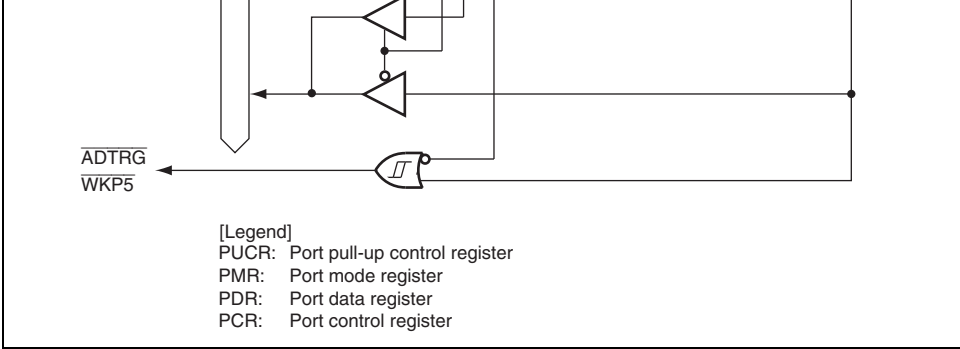


Figure B.13 Port 5 Block Diagram (P55)

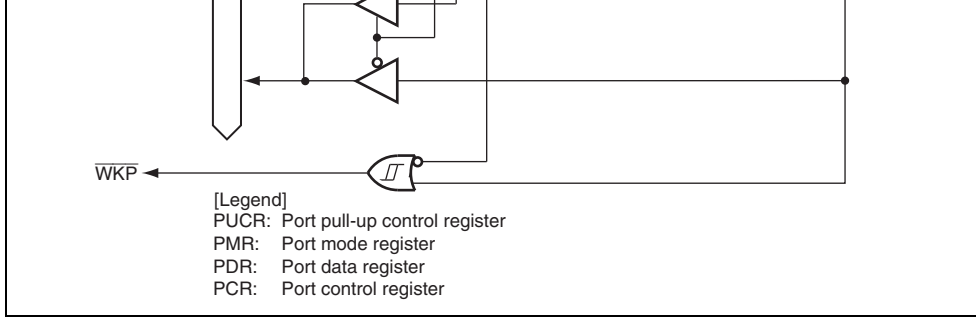


Figure B.14 Port 5 Block Diagram (P54 to P50)

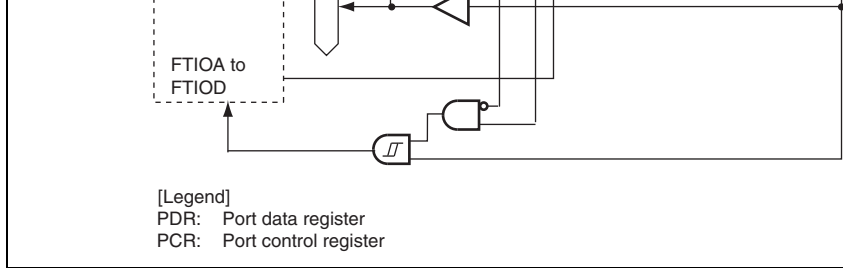


Figure B.15 Port 6 Block Diagram (P67 to P60)

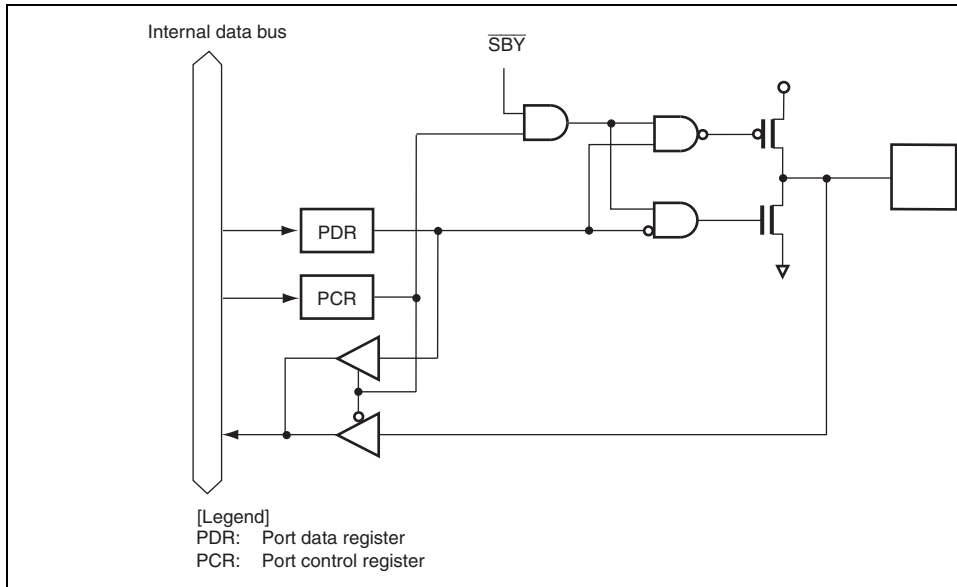


Figure B.16 Port 7 Block Diagram (P77)

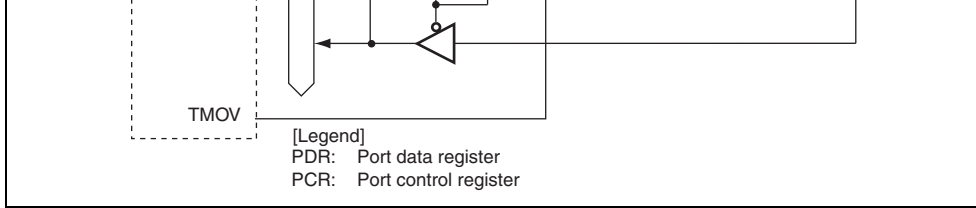


Figure B.17 Port 7 Block Diagram (P76)

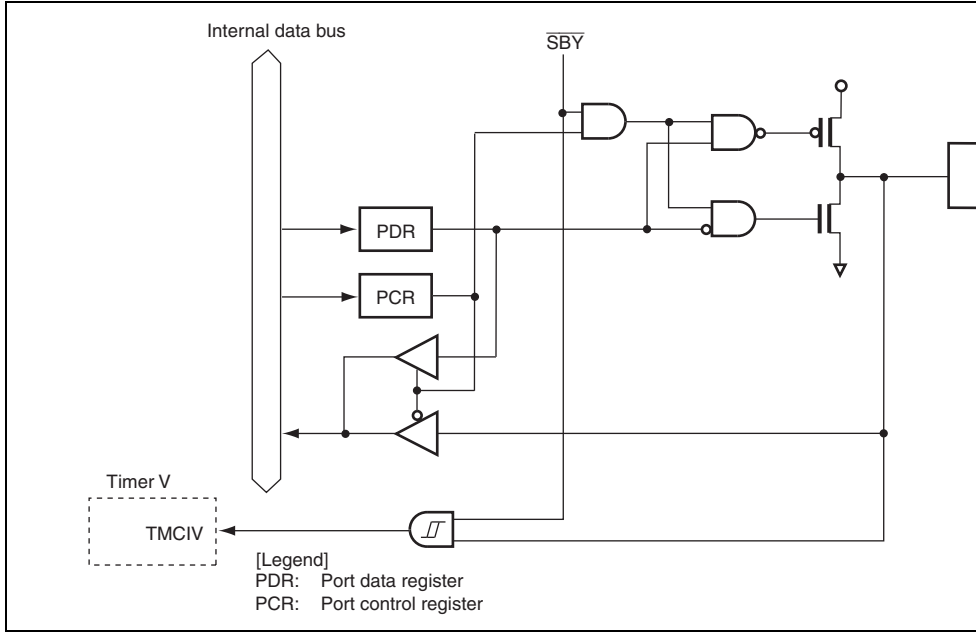


Figure B.18 Port 7 Block Diagram (P75)

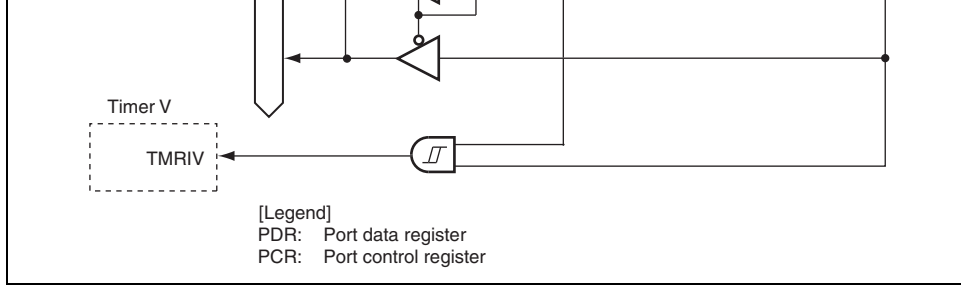


Figure B.19 Port 7 Block Diagram (P74)

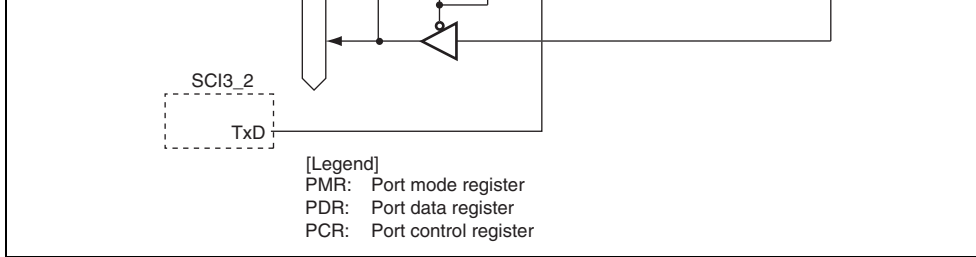


Figure B.20 Port 7 Block Diagram (P72)

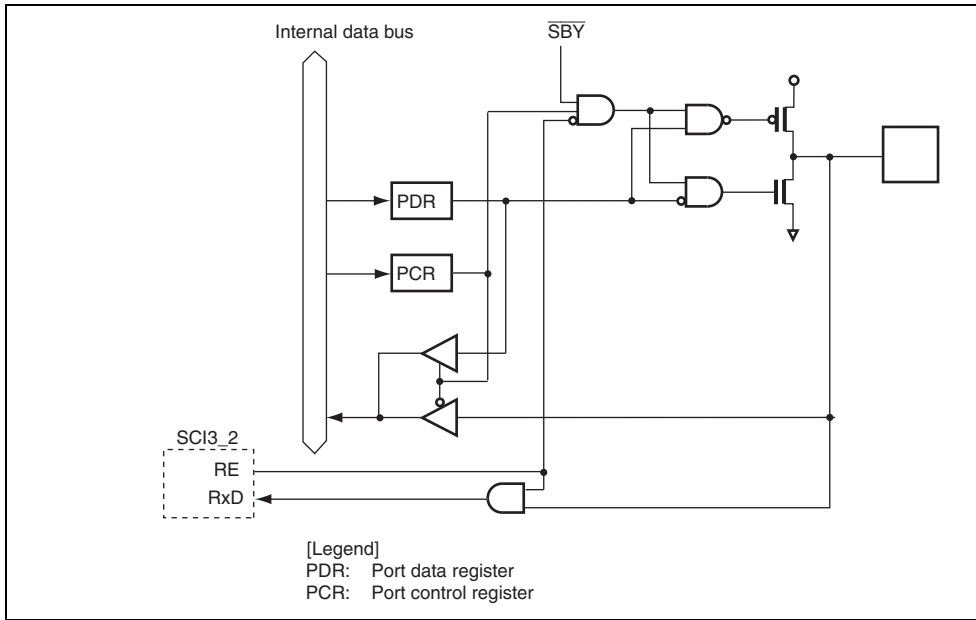


Figure B.21 Port 7 Block Diagram (P71)

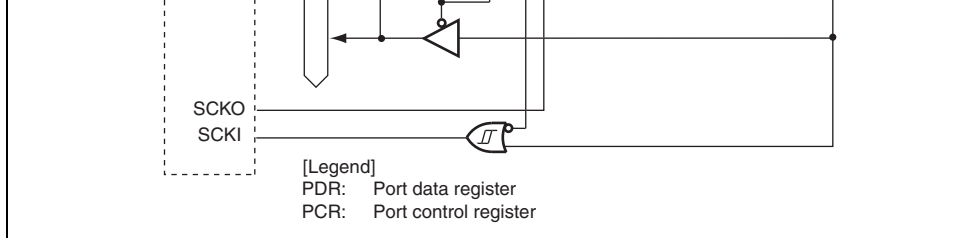


Figure B.22 Port 7 Block Diagram (P70)

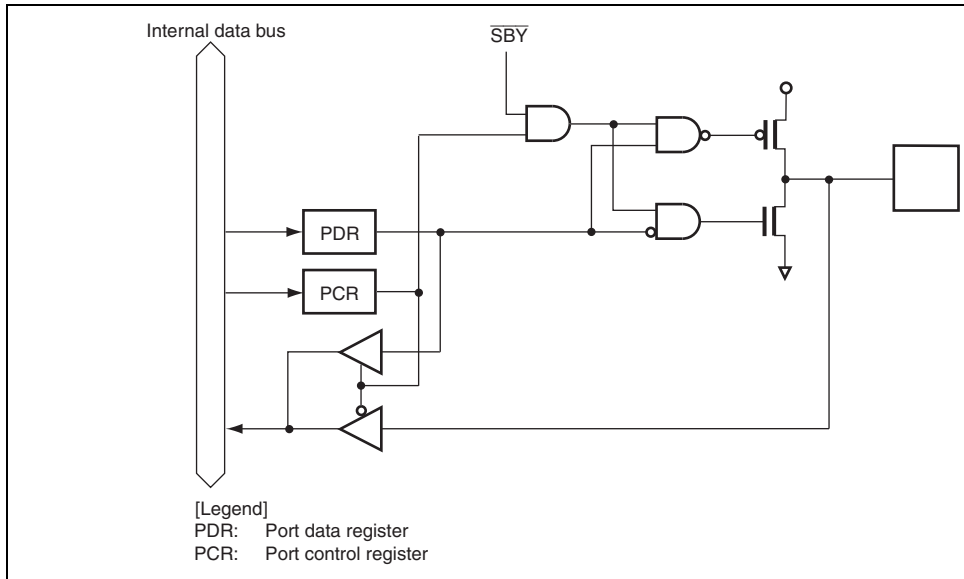


Figure B.23 Port 8 Block Diagram (P87 to P85)

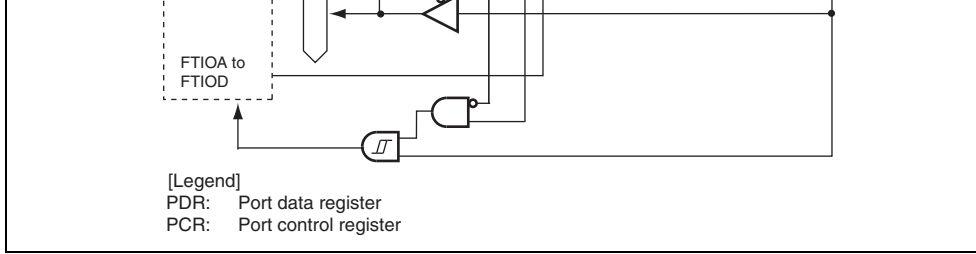


Figure B.24 Port 8 Block Diagram (P84 to P81)

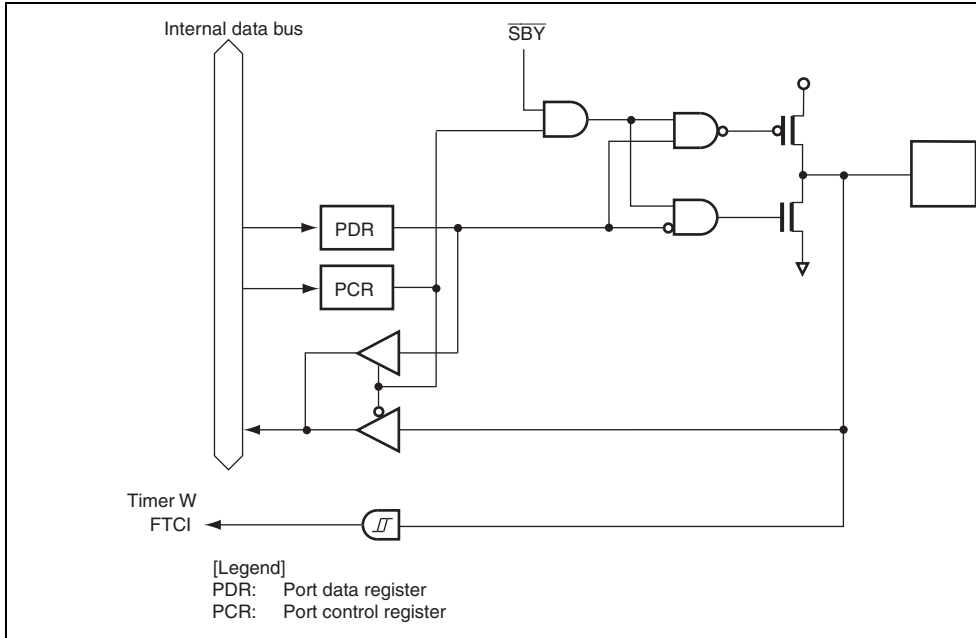


Figure B.25 Port8 Block Diagram (P80)



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

Figure B.26 Port 9 Block Diagram (P97 to P93)

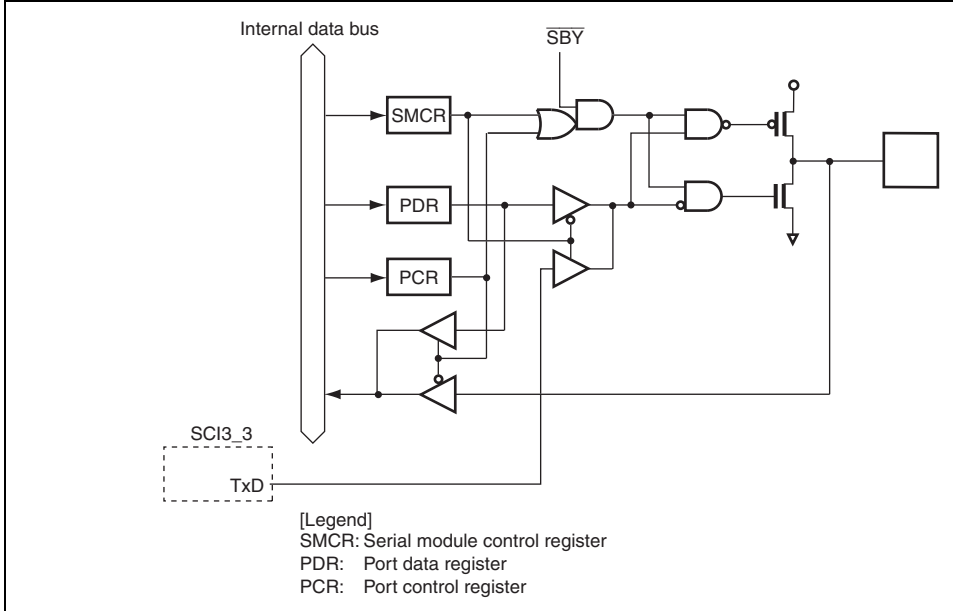


Figure B.27 Port 9 Block Diagram (P92)

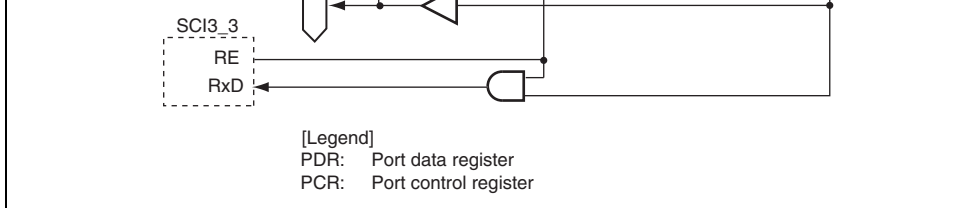


Figure B.28 Port 9 Block Diagram (P91)

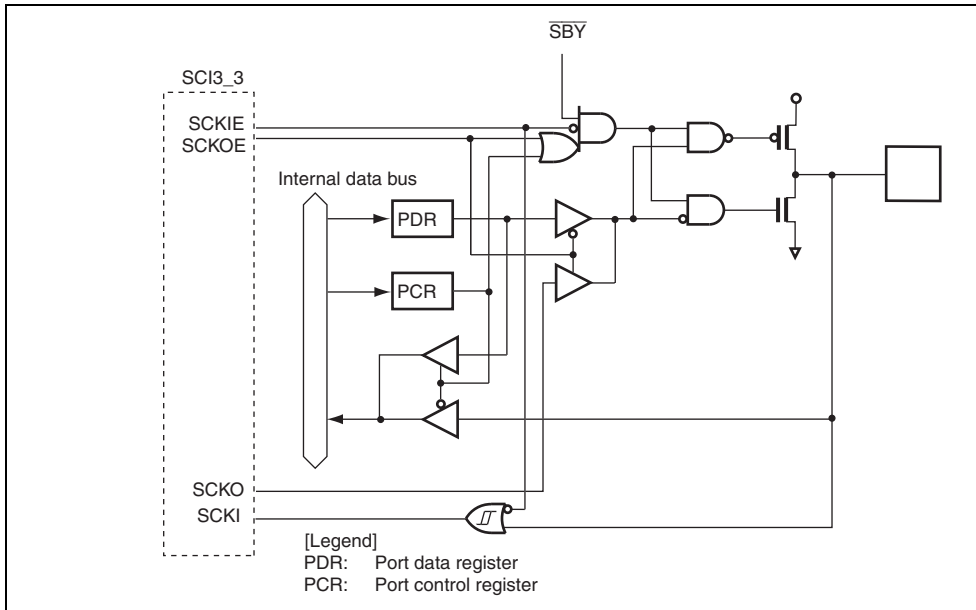


Figure B.29 Port 9 Block Diagram (P90)

Figure B.30 Port B Block Diagram (PB7 to PB0)

B.2 Port States in Each Operating Mode

Port	Reset	Sleep	Subsleep	Standby	Subactive	Active
P17 to P14, P12 to P10	High impedance	Retained	Retained	High impedance*	Functioning	Functioning
P24 to P20	High impedance	Retained	Retained	High impedance	Functioning	Functioning
P37 to P30	High impedance	Retained	Retained	High impedance	Functioning	Functioning
P57 to P50	High impedance	Retained	Retained	High impedance*	Functioning	Functioning
P67 to P60	High impedance	Retained	Retained	High impedance	Functioning	Functioning
P76 to P74, P72 to P70	High impedance	Retained	Retained	High impedance	Functioning	Functioning
P87 to P80	High impedance	Retained	Retained	High impedance	Functioning	Functioning
P97 to P90	High impedance	Retained	Retained	High impedance	Functioning	Functioning
PB7 to PB0	High impedance	High impedance	High impedance	High impedance	High impedance	High impedance

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

		Standard product	HD64336049H	HD64336049(***)H
H8/36048	Masked ROM version	Product with POR & LVDC	HD64336048GH	HD64336048(***)GH
		Standard product	HD64336048H	HD64336048(***)H
H8/36047	Masked ROM version	Product with POR & LVDC	HD64336047GH	HD64336047(***)GH
		Standard product	HD64336047H	HD64336047(***)H

[Legend]

(***) : ROM code

POR & LVDC: Power-on reset and low-voltage detection circuits

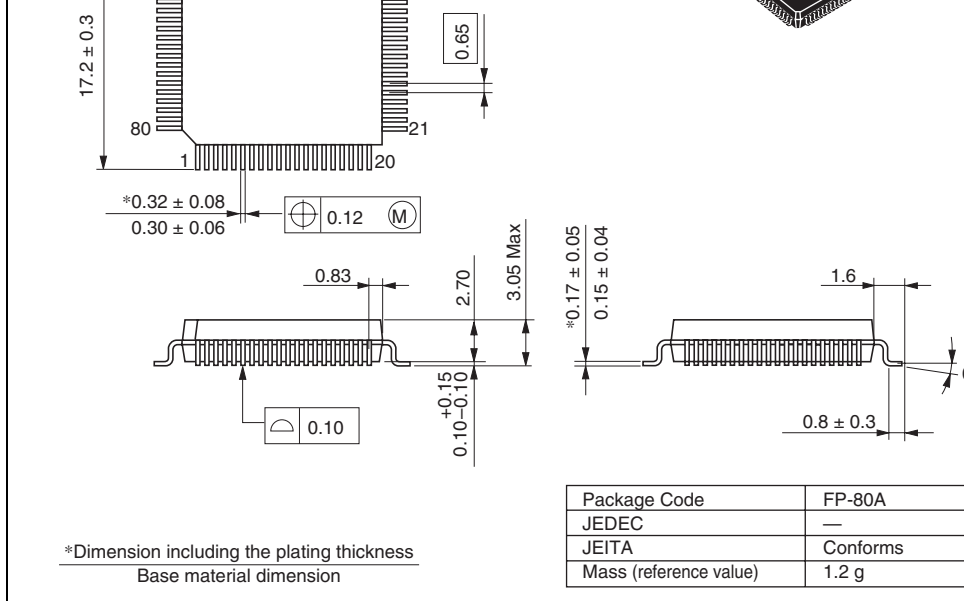


Figure D.1 FP-80A Package Dimensions

3. Area H'FFF780 to H'FFFB7F must on no a accessed.
4. When the E7 or E8 is used, address break set as either available to the user or for use E7 or E8. If address breaks are set as being the E7 or E8, the address break control register must not be accessed.
5. When the E7 or E8 is used, NMI is an input pin (open-drain in output mode), P85 and P86 input pins, and P86 is an output pin.

6.1.1 System Control Register 1 (SYSCR1) 76

Amended

Bit	Bit Name	Description
3	NESEL	Noise Elimination Sampling Frequency Select This bit selects the sampling frequency of the oscillator clear when the watch clock signal is sampled. When $\phi_{osc} = 4$ to 2, clear NESEL to 0.

Section 8 RAM

109 Added

Note: * When the E7 or E8 is used, area H'FFF
H'FFFB7F must not be accessed.

9.7.3 Pin Functions

139 Amended

- P84/FTIOD pin

Register	TMRW
Bit Name	PWMD

9.7.3 Pin Functions

140 Amended

- P83/FTIOC pin

Register	TMRW
Bit Name	PWMC

9.7.3 Pin Functions

140 Amended

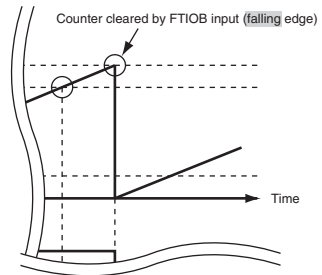
- P82/FTIOB pin

Register	TMRW
Bit Name	PWMB

....The TCNT counters cannot be accessed in units; they must always be accessed as a 16-bit value. The TCNT is initialized to H'0000.

Figure 14.17 Example of Input Capture Operation

Amended



Added

Figure 14.20 shows an example of synchronous operation. In this example, set for the channel counter clearing source. In addition, the same clock has been set as the counter input clock for channel 0 and channel 1. Two-phase PWM waveforms are....

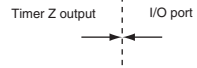
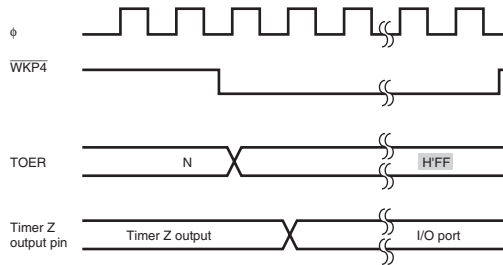


Figure 14.45 Example of Output Disable Timing of Timer Z by External Trigger

277 Amended



15.2.1 Timer Control/Status Register WD (TCSRWD)

290 Amended

Bit	Bit Name	Description
4	TCSRWE	Timer Control/Status Register Write Enable The WDON and WRST bits can be written when the TCSRWE is set to 1. When writing data to this bit, the value for bit 5 must be 0.

general call address or r
 byte slave address, nex
 detection of start conditi
 accords with the address
 SAR

18.7 Usage Note	375	Added
19.3.1 A/D Data Registers A to D (ADDRA to ADDR D)	380	Amended The temporary register contents are transferred from the ADDR when the upper byte data is read. Therefore, byte access to ADDR should be done by reading the upper byte first then the lower one. Access to ADDR in the lower byte address range is also possible. ADDR is initialized to 0.

Figure 20.1 Block Diagram of Power-On Reset Circuit and Low-Voltage Detection Circuit	392	Amended
---	-----	---------

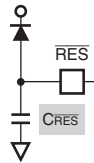


Table 23.2 DC Characteristics (1)	431,	Amended	
Table 23.11 DC Characteristics (1)	450		
	Mode	$\overline{\text{RES}}$ Pin	Internal State
	Active mode 1	V_{CC}	Operates
	Active mode 2		Operates ($\phi_{\text{osc}}/64$)
	Sleep mode 1	V_{CC}	Only timers operate
	Sleep mode 2		Only timers operate

Mnemonic		Operand Size	Condition Code						No. of States ^{*1}	
			I	H	N	Z	V	C	Normal	Advanced
DAA	DAA Rd	B	—	*	↓	↓	*	↓		2

Address break	63
Addressing modes.....	28
Arithmetic operations instructions.....	20
Asynchronous mode	318
Auto-reload timer operation	164

B

Bit manipulation instructions.....	23
Bit rate	309
Bit synchronous circuit.....	374
Block data transfer instructions	26
Boot mode	95
Boot program.....	95
Branch instructions	25
Break.....	339
Buffer operation.....	268

C

Clock pulse generators.....	69
Clock synchronous serial format	366
Clocked synchronous mode	326
Combinations of instructions and addressing modes.....	492
Complementary PWM mode	258
Condition field.....	27
Condition-code register (CCR).....	13
CPU	9

Erase/erase-verify	
Erasing units	
Error protection.....	
Event counter operation	
Exception handling	

F

Flash memory	
Framing error	

G

General registers	
-------------------------	--

H

Hardware protection.....	
--------------------------	--

I

I/O Port block diagrams	
I/O ports	
I ² C Bus format	
I ² C bus interface 2 (IIC2).....	
Immediate	
Initial setting procedure	
Input capture function	
Instruction list	

Large current ports.....	2
Logic operations instructions.....	22
low-voltage detection circuit	391
LVDI	398
LVDI (interrupt by low voltage detect) circuit.....	398
LVDR.....	397
LVDR (reset by low voltage detect) circuit.....	397

M

Mark state	339
Memory indirect	30
Memory map	10
Module standby function	87
Multiprocessor communication function.....	332

N

NMI interrupt.....	55
Noise canceler	368
Noise canceller	301
Number of execution states	481

Pin arrangement	
Power-down modes.....	
Power-down states	
power-on reset.....	
Power-on reset circuit	
Prescaler S	
Prescaler W	
Product code lineup	
Program counter (PC)	
Program/program-verify	
Program-counter relative	
Programmer mode.....	
Programming units.....	
Programming/erasing in user progr mode	
PWM mode.....	
PWM operation.....	

R

Realtime clock (RTC).....	
Register addresses.....	
Register bits	
Register direct.....	
Register field.....	
Register indirect.....	

ADDRA.....	380, 408, 414, 419	MSTCR1.....	79, 410
ADDRB.....	380, 408, 414, 419	MSTCR2.....	80, 410
ADDRC.....	380, 408, 414, 419	PCR1.....	113, 410
ADDRD.....	380, 408, 414, 420	PCR2.....	117, 410
BARE.....	66, 409, 415, 420	PCR3.....	120, 410
BARH.....	66, 409, 415, 420	PCR5.....	125, 410
BARL.....	66, 409, 415, 420	PCR6.....	129, 410
BDRH.....	66, 409, 415, 420	PCR7.....	134, 410
BDRL.....	66, 409, 415, 420	PCR8.....	138, 410
BRR.....	309, 407, 414, 419	PCR9.....	142, 410
EBR1.....	93, 407, 414, 419	PDR1.....	113, 409
FENR.....	94, 407, 414, 419	PDR2.....	117, 409
FLMCR1.....	91, 407, 414, 419	PDR3.....	121, 409
FLMCR2.....	92, 407, 414, 419	PDR5.....	125, 409
FLPWCRC.....	94, 407, 414, 419	PDR6.....	130, 409
GRA.....	193, 228, 404, 407, 411, 413, 417, 419	PDR7.....	135, 409
GRB.....	193, 228, 404, 407, 411, 413, 417, 419	PDR8.....	138, 409
GRC.....	193, 228, 404, 407, 411, 413, 417, 419	PDR9.....	142, 409
GRD.....	193, 228, 404, 407, 411, 414, 417, 419	PDRB.....	145, 409
ICCR1.....	344, 406, 413, 418	PMR1.....	112, 409
ICCR2.....	347, 406, 413, 418	PMR3.....	118, 410
ICDRR.....	356, 406, 413, 418	PMR5.....	124, 409
ICDRS.....	356	POCR.....	235, 409
ICDRT.....	356, 406, 413, 418	PUCR1.....	114, 409
ICIER.....	350, 406, 413, 418	PUCR5.....	126, 409
		PWCR.....	296, 409
		PWDRL.....	297, 409
		PWDRU.....	297, 409
		RDR.....	303, 409

SMR.....	304, 407, 414, 419
SSR.....	307, 408, 414, 419
SYSCR1	76, 410, 416, 421
SYSCR2	77, 410, 416, 421
TCB1	163, 406, 413, 418
TCNT.....	192, 228, 404, 407,411, 413, 417, 419
TCNTV.....	167, 407, 414, 419
TCORA	168, 407, 414, 419
TCORB.....	168, 407, 414, 419
TCR	229, 404, 411, 417
TCRV0	169, 407, 414, 419
TCRV1	172, 407, 414, 419
TCRW	186, 407, 413, 419
TCSRV	170, 407, 414, 419
TCSRWD	290, 408, 415, 420
TCWD	292, 408, 415, 420
TDR.....	304, 408, 414, 419
TFCR.....	223, 405, 412, 418
TIER	234, 404, 411, 417
TIERW	187, 407, 413, 419
TIOR0.....	189, 407, 413, 419
TIOR1.....	191, 407, 413, 419
TIORA	230, 404, 411, 417
TIORC	231, 404, 411, 417
TLB1	163, 406, 413, 418
TMB1	162, 406, 413, 418
TMDR	221, 405, 412, 418
TMRW.....	185, 407, 413, 419

S

Sample-and-hold circuit.....
Scan mode.....
Serial communication interface 3 (SCI3)
Shift instructions
Single mode
Slave address
Sleep mode.....
Software protection.....
Stack pointer (SP)
Stack status
Standby mode
Start condition.....
Stop condition
Subactive mode.....
Subclock generator
Subsleep mode
Synchronous operation.....
System clock generator.....
System control instructions.....

T

Timer B1
Timer V.....
Timer W.....
Timer Z.....
Transfer rate.....

**Renesas 16-Bit Single-Chip Microcomputer
Hardware Manual
H8/36049 Group**

Publication Date: Rev.1.00, Aug. 28, 2003
Rev.3.00, Mar. 15, 2006

Published by: Sales Strategic Planning Div.
Renesas Technology Corp.

Edited by: Customer Support Department
Global Strategic Communication Div.
Renesas Solutions Corp.

© 2006. Renesas Technology Corp., All rights reserved. Printed in Japan.



RENESAS SALES OFFICES

<http://www.renesas.com>

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, AZIA Center, 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



H8/36049 Group Hardware Manual



Renesas Electronics Corporation

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

REJ09B0060-0300

X-ON Electronics

Largest Supplier of Electrical and Electronic Components

Click to view similar products for [16-bit Microcontrollers - MCU category](#):

Click to view products by [Renesas manufacturer](#):

Other Similar products are found below :

[M30302FCFP#U3](#) [MB90F036APMC-GSE1](#) [MB90F428GCPFR-GSE1](#) [MB96F683RBPMC-GSAE1](#) [R5F10MMGDFB#30](#)
[R5F111PGGFB#30](#) [R5F117BCGNA#20](#) [DF3026XBL25V](#) [DF36014GFTV](#) [DF36014GFXV](#) [DF36034GFPV](#) [R5F11B7EANA#U0](#)
[R5F21172DSP#U0](#) [MB90092PF-G-BNDE1](#) [MB90F335APMC1-G-SPE1](#) [MB90F345CAPFR-GSE1](#) [MB90F568PMCR-GE1](#)
[MB96F395RSAPMC-GSE2](#) [DF36024GFXV](#) [UPD78F1018F1-BA4-A](#) [MB96F018RBPMC-GSE1](#) [MB90F867ASPFR-GE1](#)
[M30290FCHP#U3A](#) [DF2239FA20IV](#) [R5F117BCGFP#30](#) [LC88F58B0AU-SQFPH](#) [MB90F548GPF-GE1](#) [MB90214PF-GT-310-BND-AE1](#)
[MB90F342CESPQC-GSE2](#) [MB90F428GAPF-GSE1](#) [ML62Q504H-NNNTBWBX](#) [S912ZVH128F2VLL](#) [UPD78F1500AGK-GAK-AX](#)
[HD64F3337SF16V](#) [MB90F428GCPF-GSE1](#) [MB90F342ESPMC-G-JNE1](#) [MB90022PF-GS-358E1](#) [MB96F395RWAPMC-GSE2](#)
[MB96395RSAPMC-GS-110E2](#) [MB90F883CSPMC-GE1](#) [S912ZVHY64F1CLL](#) [S912ZVHY64F1VLQ](#) [ST10F280](#) [MB96F338RSAPMCR-](#)
[GK5E2](#) [CY90096PF-G-002-BND-ERE1](#) [ML62Q1569-NNNGAZ0AX](#) [ML62Q1739-NNNGAZ0AX](#) [ML62Q1749-NNNGAZ0AX](#)
[ML62Q1579-NNNGAZ0AX](#) [ML62Q1559-NNNGAZ0AX](#)