To our customers,

Old Company Name in Catalogs and Other Documents

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

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

April 1st, 2010
Renesas Electronics Corporation

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

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

Notice

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

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

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

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

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

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

User's Manual

16

H8/36079 Group, H8/36077 Group

Hardware Manual

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

H8/36079	H8/36079GF,	HD64F36079G
	H8/36079LF,	HD64F36079L
	H8/36078GF,	HD64F36078G
	H8/36078LF,	HD64F36078L
H8/36077	H8/36077GF,	HD64F36077G
	H8/36077LF,	HD64F36077L
	H8/36074GF,	HD64F36074G
	H8/36074LF,	HD64F36074L

Renesas Electronics
www.renesas.com

Rev.3.00 2007.09

document, please confirm the latest product information with a Renesas sales office. Also, please pay
and careful attention to additional and different information to be disclosed by Renesas such as that dis
through our website. (http://www.renesas.com)

5. Renesas has used reasonable care in compiling the information included in this document, but Renesa
assumes no liability whatsoever for any damages incurred as a result of errors or omissions in the infor
included in this document.
6. When using or otherwise relying on the information in this document, you should evaluate the informati
light of the total system before deciding about the applicability of such information to the intended appli
Renesas makes no representations, warranties or guaranties regarding the suitability of its products fo
particular application and specifically disclaims any liability arising out of the application and use of the
information in this document or Renesas products.
7. With the exception of products specified by Renesas as suitable for automobile applications, Renesas
products are not designed, manufactured or tested for applications or otherwise in systems the failure o
malfunction of which may cause a direct threat to human life or create a risk of human injury or which r
especially high quality and reliability such as safety systems, or equipment or systems for transportatio
traffic, healthcare, combustion control, aerospace and aeronautics, nuclear power, or undersea commu
transmission. If you are considering the use of our products for such purposes, please contact a Renes
sales office beforehand. Renesas shall have no liability for damages arising out of the uses set forth ab
8. Notwithstanding the preceding paragraph, you should not use Renesas products for the purposes listed
   (1) artificial life support devices or systems
   (2) surgical implantations
   (3) healthcare intervention (e.g., excision, administration of medication, etc.)
   (4) any other purposes that pose a direct threat to human life

   Renesas shall have no liability for damages arising out of the uses set forth in the above and purchase
   elect to use Renesas products in any of the foregoing applications shall indemnify and hold harmless R
   Technology Corp., its affiliated companies and their officers, directors, and employees against any and
   damages arising out of such applications.
9. You should use the products described herein within the range specified by Renesas, especially with re
to the maximum rating, operating supply voltage range, movement power voltage range, heat radiation
characteristics, installation and other product characteristics. Renesas shall have no liability for malfunc
damages arising out of the use of Renesas products beyond such specified ranges.
10. Although Renesas endeavors to improve the quality and reliability of its products, IC products have spe
characteristics such as the occurrence of failure at a certain rate and malfunctions under certain use
conditions. Please be sure to implement safety measures to guard against the possibility of physical inj
injury or damage caused by fire in the event of the failure of a Renesas product, such as safety design
hardware and software including but not limited to redundancy, fire control and malfunction prevention,
appropriate treatment for aging degradation or any other applicable measures. Among others, since th
evaluation of microcomputer software alone is very difficult, please evaluate the safety of the final prod
system manufactured by you.
11. In case Renesas products listed in this document are detached from the products to which the Renesa
products are attached or affixed, the risk of accident such as swallowing by infants and small children is
high. You should implement safety measures so that Renesas products may not be easily detached fro
products. Renesas shall have no liability for damages arising out of such detachment.
12. This document may not be reproduced or duplicated, in any form, in whole or in part, without prior writt
approval from Renesas.
13. Please contact a Renesas sales office if you have any questions regarding the information contained in
document, Renesas semiconductor products, or if you have any other inquiries.

RENESAS

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

3. Processing before Initialization

Note: When power is first supplied, the product's state is undefined.
The states of internal circuits are undefined until full power is supplied throughou
chip and a low level is input on the reset pin. During the period where the states a
undefined, the register settings and the output state of each pin are also undefined
your system so that it does not malfunction because of processing while it is in th
undefined state. For those products which have a reset function, reset the LSI imm
after the power supply has been turned on.

4. Prohibition of Access to Undefined or Reserved Addresses

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

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

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

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

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

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

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

11. Index

microcomputers.

Objective: This manual was written to explain the hardware functions and electrical characteristics of the H8/36079 Group and H8/36077 Group to the target Refer to the H8/300H Series Software Manual for a detailed description o instruction set.

Notes on reading this manual:

- In order to understand the overall functions of the chip

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

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

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

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

4. The following areas must on no account be accessed.
   H8/36079 Group: H'FFF780 to H'FFFB7F
   H8/36077 Group: H'F780 to H'FB7F
5. In usage with the E7or E8, address breaks can be set as either available to the user or by the E7 or E8. If address breaks are set as being used by the E7, the address break registers must not be accessed.
6. In usage with the E7 or E8, $\overline{\text{NMI}}$ is an input/output pin (open-drain in output mode), P87 are input pins, and P86 is an output pin.
7. Use channel 1 of the SCI3 (P21/RXD, P22/TXD) in on-board programming mode by mode.
8. In usage with the E7or E8, the power supply voltage for H8/36079 Group products m greater than the reset detection voltage of the low voltage detection circuit.

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

User's Manual	
H8S, H8/300 Series Simulator/Debugger User's Manual	REJ10B02
H8S, H8/300 Series High-Performance Embedded Workshop 3, Tutorial	REJ10B00
H8S, H8/300 Series High-Performance Embedded Workshop 3, User's Manual	REJ10B00

Application notes:

Document Title	Documen
H8S, H8/300 Series C/C++ Compiler Package Application Note	REJ05B04
Single Power Supply F-ZTAT™ On-Board Programming	REJ05B05

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 Table 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 EEPMOV 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 (BARE, BARH, BARL)
4.1.4 Break Data Registers (BDRH, BDRL)
4.2 Operation

Section 5 Clock Pulse Generator
5.1 Features
5.2 Register Descriptions
5.2.1 RC Control Register (RCCR)
5.2.2 RC Trimming Data Protect Register (RCTRMDPR)
5.2.3 RC Trimming Data Register (RCTRMDR)
5.2.4 Clock Control/Status Register (CKCSR)
5.3 System Clock Select Operation
5.3.1 Clock Control Operation
5.3.2 Clock Switching Timing
5.4 Trimming of On-Chip Oscillator Frequency
5.5 External Oscillators
5.5.1 Connecting Crystal Resonator
5.5.2 Connecting Ceramic Resonator
5.5.3 Inputting External Clock
5.6 Subclock Oscillator
5.6.1 Connecting 32.768-kHz Crystal Resonator

6.1.2 System Control Register 2 (SYSCR2)
6.1.3 Module Standby Control Register 1 (MSTCR1)
6.1.4 Module Standby Control Register 2 (MSTCR2)
6.2 Mode Transitions and States of LSI
6.2.1 Sleep Mode
6.2.2 Standby Mode
6.2.3 Subsleep Mode
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

9.1.1 Port Mode Register 1 (PMR1)
9.1.2 Port Control Register 1 (PCR1)
9.1.3 Port Data Register 1 (PDR1)
9.1.4 Port Pull-Up Control Register 1 (PUCR1)
9.1.5 Pin Functions
9.2 Port 2
9.2.1 Port Control Register 2 (PCR2)
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

Section 10 Realtime Clock (RTC)
10.1 Features
10.2 Input/Output Pin
10.3 Register Descriptions
10.3.1 Second Data Register/Free Running Counter Data Register (RSECDR)
10.3.2 Minute Data Register (RMINDR)
10.3.3 Hour Data Register (RHRDR)
10.3.4 Day-of-Week Data Register (RWKDR)
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

12.5 Timer V Application Examples
12.5.1 Pulse Output with Arbitrary Duty Cycle
12.5.2 Pulse Output with Arbitrary Pulse Width and Delay from TRGV Input
12.6 Usage Notes

Section 13 Timer Z
13.1 Features
13.2 Input/Output Pins
13.3 Register Descriptions
13.3.1 Timer Start Register (TSTR)
13.3.2 Timer Mode Register (TMDR)
13.3.3 Timer PWM Mode Register (TPMR)
13.3.4 Timer Function Control Register (TFCR)
13.3.5 Timer Output Master Enable Register (TOER)
13.3.6 Timer Output Control Register (TOCR)
13.3.7 Timer Counter (TCNT)
13.3.8 General Registers A, B, C, and D (GRA, GRB, GRC, and GRD)
13.3.9 Timer Control Register (TCR)
13.3.10 Timer I/O Control Register (TIORA and TIORC)
13.3.11 Timer Status Register (TSR)
13.3.12 Timer Interrupt Enable Register (TIER)
13.3.13 PWM Mode Output Level Control Register (POCR)
13.3.14 Interface with CPU
13.4 Operation
13.4.1 Counter Operation
13.4.2 Waveform Output by Compare Match
13.4.3 Input Capture Function
13.4.4 Synchronous Operation
13.4.5 PWM Mode

14.2 Register Descriptions
14.2.1 Timer Control/Status Register WD (TCSRWD)
14.2.2 Timer Counter WD (TCWD)
14.2.3 Timer Mode Register WD (TMWD)
14.3 Operation

Section 15 14-Bit PWM
15.1 Features
15.2 Input/Output Pin
15.3 Register Descriptions
15.3.1 PWM Control Register (PWCR)
15.3.2 PWM Data Registers U and L (PWDRU, PWDRL)
15.4 Operation

Section 16 Serial Communication Interface 3 (SCI3)
16.1 Features
16.2 Input/Output Pins
16.3 Register Descriptions
16.3.1 Receive Shift Register (RSR)
16.3.2 Receive Data Register (RDR)
16.3.3 Transmit Shift Register (TSR)
16.3.4 Transmit Data Register (TDR)
16.3.5 Serial Mode Register (SMR)
16.3.6 Serial Control Register 3 (SCR3)
16.3.7 Serial Status Register (SSR)
16.3.8 Bit Rate Register (BRR)
16.4 Operation in Asynchronous Mode
16.4.1 Clock
16.4.2 SCI3 Initialization

16.7 Interrupts
16.8 Usage Notes
16.8.1 Break Detection and Processing
16.8.2 Mark State and Break Sending
16.8.3 Receive Error Flags and Transmit Operations (Clock Synchronous Mode Only)
16.8.4 Receive Data Sampling Timing and Reception Margin in Asynchronous Mode

Section 17 I^2C Bus Interface 2 (IIC2)
17.1 Features
17.2 Input/Output Pins
17.3 Register Descriptions
17.3.1 I^2C Bus Control Register 1 (ICCR1)
17.3.2 I^2C Bus Control Register 2 (ICCR2)
17.3.3 I^2C Bus Mode Register (ICMR)
17.3.4 I^2C Bus Interrupt Enable Register (ICIER)
17.3.5 I^2C Bus Status Register (ICSR)
17.3.6 Slave Address Register (SAR)
17.3.7 I^2C Bus Transmit Data Register (ICDRT)
17.3.8 I^2C Bus Receive Data Register (ICDRR)
17.3.9 I^2C Bus Shift Register (ICDRS)
17.4 Operation
17.4.1 I^2C Bus Format
17.4.2 Master Transmit Operation
17.4.3 Master Receive Operation
17.4.4 Slave Transmit Operation
17.4.5 Slave Receive Operation
17.4.6 Clock Synchronous Serial Format

18.1 Features
18.2 Input/Output Pins
18.3 Register Descriptions
18.3.1 A/D Data Registers A to D (ADDRA to ADDRD)
18.3.2 A/D Control/Status Register (ADCSR)
18.3.3 A/D Control Register (ADCR)
18.4 Operation
18.4.1 Single Mode
18.4.2 Scan Mode
18.4.3 Input Sampling and A/D Conversion Time
18.4.4 External Trigger Input Timing
18.5 A/D Conversion Accuracy Definitions
18.6 Usage Notes
18.6.1 Permissible Signal Source Impedance
18.6.2 Influences on Absolute Accuracy

Section 19 Band-Gap Circuit, Power-On Reset, and Low-Voltage Detection Circuits
19.1 Features
19.2 Register Descriptions
19.2.1 Low-Voltage-Detection Control Register (LVDCR)
19.2.2 Low-Voltage-Detection Status Register (LVDSR)
19.2.3 Reset Source Decision Register (LVDRF)
19.3 Operations
19.3.1 Power-On Reset Circuit
19.3.2 Low-Voltage Detection Circuit
19.3.3 Deciding Reset Source

22.1 Absolute Maximum Ratings
22.2 Electrical Characteristics (F-ZTAT™ 5.0-V Version)
22.2.1 Power Supply Voltage and Operating Ranges
22.2.2 DC Characteristics
22.2.3 AC Characteristics
22.2.4 A/D Converter Characteristics
22.2.5 Watchdog Timer Characteristics
22.2.6 Flash Memory Characteristics
22.2.7 Power-Supply-Voltage Detection Circuit Characteristics
22.2.8 LVDI External Input Voltage Detection Circuit Characteristics
22.2.9 Power-On Reset Circuit Characteristics
22.3 Electrical Characteristics (F-ZTAT™ 3.3-V Version)
22.3.1 Power Supply Voltage and Operating Ranges
22.3.2 DC Characteristics
22.3.3 AC Characteristics
22.3.4 A/D Converter Characteristics
22.3.5 Watchdog Timer Characteristics
22.3.6 Flash Memory Characteristics
22.3.7 Power-Supply-Voltage Detection Circuit Characteristics
22.3.8 LVDI External Input Voltage Detection Circuit Characteristics
22.3.9 Power-On Reset Circuit Characteristics
22.4 Operation Timing
22.5 Output Load Condition

Appendix A Instruction Set
A.1 Instruction List
A.2 Operation Code Map
A.3 Number of Execution States
A.4 Combinations of Instructions and Addressing Modes

Figure 2.3 Usage of General Registers
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 Generator

Figure 5.1 Block Diagram of Clock Pulse Generator
Figure 5.2 State Transition of System Clock
Figure 5.3 Flowchart of Clock Switching with Backup Function Enabled
Figure 5.4 Flowchart of Clock Switching with Backup Function Disabled (1) (From On-Chip Oscillator Clock to External Clock)

Figure 5.13 Equivalent Circuit of Crystal Resonator
Figure 5.14 Example of Connection to Ceramic Resonator
Figure 5.15 Example of External Clock Input
Figure 5.16 Block Diagram of Subclock Oscillator
Figure 5.17 Typical Connection to 32.768-kHz Crystal Resonator
Figure 5.18 Equivalent Circuit of 32.768-kHz Crystal Resonator
Figure 5.19 Pin Connection when not Using Subclock
Figure 5.20 Example of Incorrect Board Design

Section 6 Power-Down Modes

Figure 6.1 Mode Transition Diagram

Section 7 ROM

Figure 7.1 Flash Memory Block Configuration (1) (H8/36079GF and H8/36079LF)
Figure 7.1 Flash Memory Block Configuration (2) (H8/36078GF and H8/36078LF)
Figure 7.1 Flash Memory Block Configuration (3) (H8/36077GF and H8/36077LF)
Figure 7.1 Flash Memory Block Configuration (4) (H8/36074GF and H8/36074LF)
Figure 7.2 Programming/Erasing Flowchart Example in User Program Mode
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 B Pin Configuration
Figure 9.9 Port C Pin Configuration

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 12.8 Clear Timing by TMRIV Input
Figure 12.9 Pulse Output Example
Figure 12.10 Example of Pulse Output Synchronized to TRGV Input
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

Section 13 Timer Z

Figure 13.1 Timer Z Block Diagram
Figure 13.2 Timer Z (Channel 0) Block Diagram
Figure 13.3 Timer Z (Channel 1) Block Diagram
Figure 13.4 Example of Outputs in Reset Synchronous PWM Mode and Complementary PWM Mode
Figure 13.5 Accessing Operation of 16-Bit Register (between CPU and TCNT (16 bits))
Figure 13.6 Accessing Operation of 8-Bit Register (between CPU and TSTR (8 bits))
Figure 13.7 Example of Counter Operation Setting Procedure
Figure 13.8 Free-Running Counter Operation
Figure 13.9 Periodic Counter Operation
Figure 13.10 Count Timing at Internal Clock Operation
Figure 13.11 Count Timing at External Clock Operation (Both Edges Detected)
Figure 13.12 Example of Setting Procedure for Waveform Output by Compare Match
Figure 13.13 Example of 0 Output/1 Output Operation

Figure 13.25 Example of PWM Mode Operation (4) .....
Figure 13.26 Example of Reset Synchronous PWM Mode Setting Procedure.....
Figure 13.27 Example of Reset Synchronous PWM Mode Operation
(OLS0 = OLS1 = 1) .....
Figure 13.28 Example of Reset Synchronous PWM Mode Operation
(OLS0 = OLS1 = 0) .....
Figure 13.29 Example of Complementary PWM Mode Setting Procedure.....
Figure 13.30 Canceling Procedure of Complementary PWM Mode .....
Figure 13.31 Example of Complementary PWM Mode Operation (1).....
Figure 13.32 (1) Example of Complementary PWM Mode Operation
(TPSC2 = TPSC1 = TPSC0 = 0) (2).....
Figure 13.32 (2) Example of Complementary PWM Mode Operation
(TPSC2 = TPSC1 = TPSC0 ≠ 0) (3).....
Figure 13.33 Timing of Overshooting .....
Figure 13.34 Timing of Undershooting .....
Figure 13.35 Compare Match Buffer Operation.....
Figure 13.36 Input Capture Buffer Operation.....
Figure 13.37 Example of Buffer Operation Setting Procedure.....
Figure 13.38 Example of Buffer Operation (1)
(Buffer Operation for Output Compare Register).....
Figure 13.39 Example of Compare Match Timing for Buffer Operation .....
Figure 13.40 Example of Buffer Operation (2)
(Buffer Operation for Input Capture Register) .....
Figure 13.41 Input Capture Timing of Buffer Operation.....
Figure 13.42 Buffer Operation (3)
(Buffer Operation in Complementary PWM Mode CMD1 = CMD0 = 1) ..
Figure 13.43 Buffer Operation (4)
(Buffer Operation in Complementary PWM Mode CMD1 = CMD0 = 1) ..
Figure 13.44 Example of Output Disable Timing of Timer Z by Writing to TOER .....

Figure 13.56 Contention between GR Read and Input Capture
Figure 13.57 Contention between Count Clearing and Increment Operations by Input Capture
Figure 13.58 Contention between GR Write and Input Capture
Figure 13.59 When Compare Match and Bit Manipulation Instruction to TOCR Occur at the Same Timing

Section 14 Watchdog Timer
Figure 14.1 Block Diagram of Watchdog Timer
Figure 14.2 Watchdog Timer Operation Example

Section 15 14-BIt PWM
Figure 15.1 Block Diagram of 14-Bit PWM
Figure 15.2 Waveform Output by 14-Bit PWM

Section 16 Serial Communication Interface 3 (SCI3)
Figure 16.1 Block Diagram of SCI3
Figure 16.2 Data Format in Asynchronous Communication
Figure 16.3 Relationship between Output Clock and Transfer Data Phase (Asynchronous Mode) (Example with 8-Bit Data, Parity, Two Stop Bits)
Figure 16.4 Sample SCI3 Initialization Flowchart
Figure 16.5 Example of SCI3 Transmission in Asynchronous Mode (8-Bit Data, Parity, One Stop Bit)
Figure 16.6 Sample Serial Transmission Data Flowchart (Asynchronous Mode)
Figure 16.7 Example of SCI3 Reception in Asynchronous Mode (8-Bit Data, Parity, One Stop Bit)
Figure 16.8 Sample Serial Reception Data Flowchart (Asynchronous Mode) (1)
Figure 16.8 Sample Serial Reception Data Flowchart (Asynchronous Mode) (2)
Figure 16.9 Data Format in Clock Synchronous Communication
Figure 16.10 Example of SCI3 Transmission in Clock Synchronous Mode
Figure 16.11 Sample Serial Transmission Flowchart (Clock Synchronous Mode)

Figure 16.19 Receive Data Sampling Timing in Asynchronous Mode .....

Section 17 I²C Bus Interface 2 (IIC2)

Figure 17.1 Block Diagram of I²C Bus Interface 2.....
Figure 17.2 External Circuit Connections of I/O Pins .....
Figure 17.3 I²C Bus Formats .....
Figure 17.4 I²C Bus Timing.....
Figure 17.5 Master Transmit Mode Operation Timing (1).....
Figure 17.6 Master Transmit Mode Operation Timing (2).....
Figure 17.7 Master Receive Mode Operation Timing (1) .....
Figure 17.8 Master Receive Mode Operation Timing (2) .....
Figure 17.9 Slave Transmit Mode Operation Timing (1) .....
Figure 17.10 Slave Transmit Mode Operation Timing (2) .....
Figure 17.11 Slave Receive Mode Operation Timing (1).....
Figure 17.12 Slave Receive Mode Operation Timing (2).....
Figure 17.13 Clock Synchronous Serial Transfer Format .....
Figure 17.14 Transmit Mode Operation Timing.....
Figure 17.15 Receive Mode Operation Timing .....
Figure 17.16 Block Diagram of Noise Filter .....
Figure 17.17 Sample Flowchart for Master Transmit Mode.....
Figure 17.18 Sample Flowchart for Master Receive Mode .....
Figure 17.19 Sample Flowchart for Slave Transmit Mode.....
Figure 17.20 Sample Flowchart for Slave Receive Mode .....
Figure 17.21 The Timing of the Bit Synchronous Circuit .....

Section 18 A/D Converter

Figure 18.1 Block Diagram of A/D Converter .....
Figure 18.2 A/D Conversion Timing.....
Figure 18.3 External Trigger Input Timing .....
Figure 18.4 A/D Conversion Accuracy Definitions (1).....

(When Compared Voltage is Input through ExtU and ExtD Pins) .....
Figure 19.7 Timing of Setting Bits in Reset Source Decision Register .....

Section 20 Power Supply Circuit
Figure 20.1 Power Supply Connection of 5.0-V-Specification Microcontrollers .....
Figure 20.2 Power Supply Connection of 3.3-V-Specification Microcontrollers .....

Section 22 Electrical Characteristics
Figure 22.1 System Clock Input Timing .....
Figure 22.2 $\overline{\text{RES}}$ Low Width Timing .....
Figure 22.3 Input Timing .....
Figure 22.4 I^2C Bus Interface Input/Output Timing .....
Figure 22.5 SCK3 Input Clock Timing .....
Figure 22.6 SCI Input/Output Timing in Clock Synchronous Mode .....
Figure 22.7 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 2 Block Diagram (P11) .....
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.26 Port B Block Diagram (PC0) .....
Figure D.1 FP-64K Package Dimensions .....
Figure D.2 FP-64A 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 Generator
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 due to 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 Z
Table 13.1 Timer Z Functions
Table 13.2 Pin Configuration
Table 13.3 Initial Output Level of FTIOB0 Pin
Table 13.4 Output Pins in Reset Synchronous PWM Mode
Table 13.5 Register Settings in Reset Synchronous PWM Mode
Table 13.6 Output Pins in Complementary PWM Mode
Table 13.7 Register Settings in Complementary PWM Mode
Table 13.8 Register Combinations in Buffer Operation

Section 15 14-Bit PWM
Table 15.1 Pin Configuration

Section 16 Serial Communication Interface 3 (SCI3)
Table 16.1 Channel Configuration
Table 16.2 Pin Configuration
Table 16.3 Examples of BRR Settings for Various Bit Rates (Asynchronous Mode) (1
Table 16.3 Examples of BRR Settings for Various Bit Rates (Asynchronous Mode) (2
Table 16.3 Examples of BRR Settings for Various Bit Rates (Asynchronous Mode) (3
Table 16.4 Maximum Bit Rate for Each Frequency (Asynchronous Mode)
Table 16.5 Examples of BRR Settings for Various Bit Rates (Clock Synchronous Mode) (1)
Table 16.5 Examples of BRR Settings for Various Bit Rates (Clock Synchronous Mode) (2)

Table 18.3 A/D Conversion Time (Single Mode)

Section 19 Band-Gap Circuit, Power-On Reset, and Low-Voltage Detection Circu

Table 19.1 LVDCR Settings and Select Functions

Table 19.2 Deciding Reset Source

Section 22 Electrical Characteristics

Table 22.1 Absolute Maximum Ratings

Table 22.2 DC Characteristics (1)

Table 22.2 DC Characteristics (2)

Table 22.3 AC Characteristics

Table 22.4 I^2C Bus Interface Timing

Table 22.5 Serial Communication Interface (SCI) Timing

Table 22.6 A/D Converter Characteristics

Table 22.7 Watchdog Timer Characteristics

Table 22.8 Flash Memory Characteristics

Table 22.9 Power-Supply-Voltage Detection Circuit Characteristics

Table 22.10 LVDI External Input Voltage Detection Circuit Characteristics

Table 22.11 Power-On Reset Circuit Characteristics

Table 22.12 DC Characteristics (1)

Table 22.12 DC Characteristics (2)

Table 22.13 AC Characteristics

Table 22.14 I^2C Bus Interface Timing

Table 22.15 Serial Communication Interface (SCI) Timing

Table 22.16 A/D Converter Characteristics

Table 22.17 Watchdog Timer Characteristics

Table 22.18 Flash Memory Characteristics

Table 22.19 Power-Supply-Voltage Detection Circuit Characteristics

Table 22.20 LVDI External Input Voltage Detection Circuit Characteristics

Table 22.21 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 Z (16-bit timer)
— 14-bit PWM
— Watchdog timer
— SCI (asynchronous or clock synchronous serial communication interface) × 2 cha
— I^2C bus interface (conforms to the I^2C bus interface format that is advocated by P
Electronics)
— 10-bit A/D converter
— POR/LVD (power-on reset & low-voltage detection circuit)
— On-chip oscillator

		3.3-V model	H8/36074LF	HD64F36074L		

- Operating voltage and maximum operating frequency

Product Classification				Product Model	Operating Voltage Range	Maximum Operating Frequency
Flash memory version (F-ZTAT™ version)	H8/36079 Group	5.0-V model	H8/36079GF	HD64F36079G	4.5 V to 5.5 V	20.0 MHz
		3.3-V model	H8/36079LF	HD64F36079L	3.0 V to 3.6 V	16.0 MHz
		5.0-V model	H8/36078GF	HD64F36078G	4.5 V to 5.5 V	20.0 MHz
		3.3-V model	H8/36078LF	HD64F36078L	3.0 V to 3.6 V	16.0 MHz
	H8/36077 Group	5.0-V model	H8/36077GF	HD64F36077G	4.5 V to 5.5 V	20.0 MHz
		3.3-V model	H8/36077LF	HD64F36077L	3.0 V to 3.6 V	16.0 MHz
		5.0-V model	H8/36074GF	HD64F36074G	4.5 V to 5.5 V	20.0 MHz
		3.3-V model	H8/36074LF	HD64F36074L	3.0 V to 3.6 V	16.0 MHz

	3.3-V model	H8/36077LF	HD64F36077L
	5.0-V model	H8/36074GF	HD64F36074G
	3.3-V model	H8/36074LF	HD64F36074L

- General I/O ports
  - I/O pins: 47 I/O pins, including 8 for large currents (I_{OL} = 20 mA, @V_{OL} = 1.5 V)
  - Input-only pins: 8 (also used for analog input)
- I^2C bus interface (conforms to the I^2C bus interface format put forward by Philips El
- Supports various power-down modes

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

- Compact package

Package	Code	Body Size	Pin Pitch
LQFP-64	FP-64K	10.0 × 10.0 mm	0.5 mm
QFP-64	FP-64A	14.0 × 14.0 mm	0.8 mm

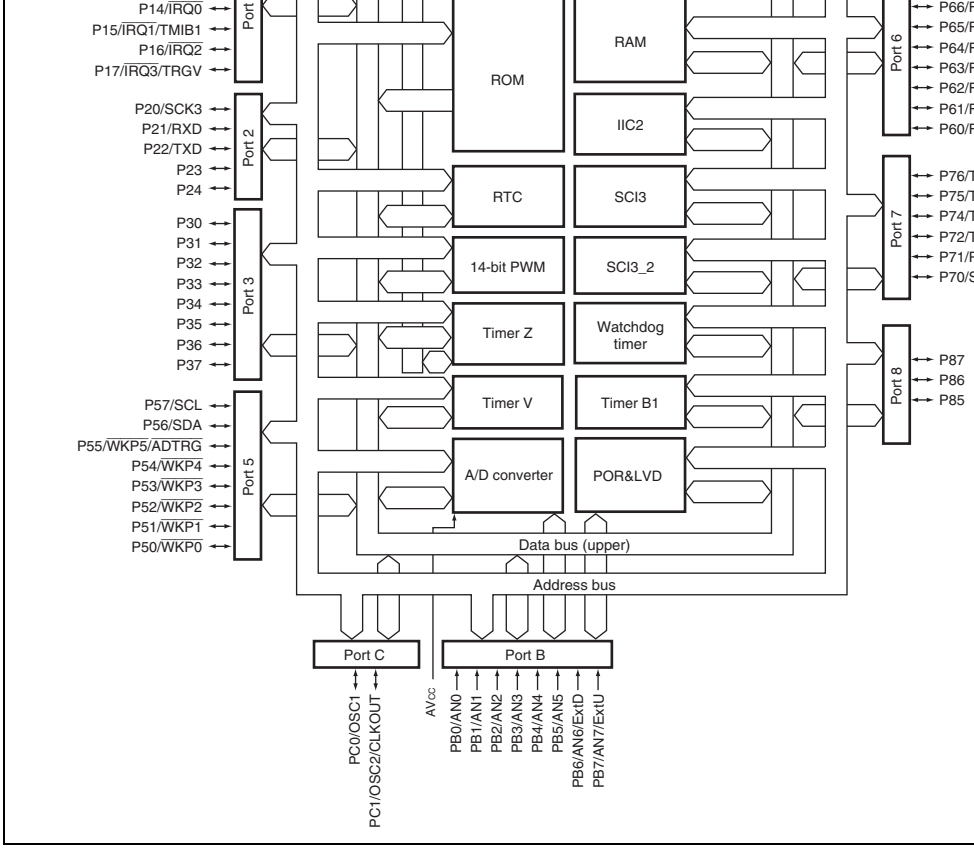


Figure 1.1 Block Diagram of H8/36079 Group and H8/36077 Group

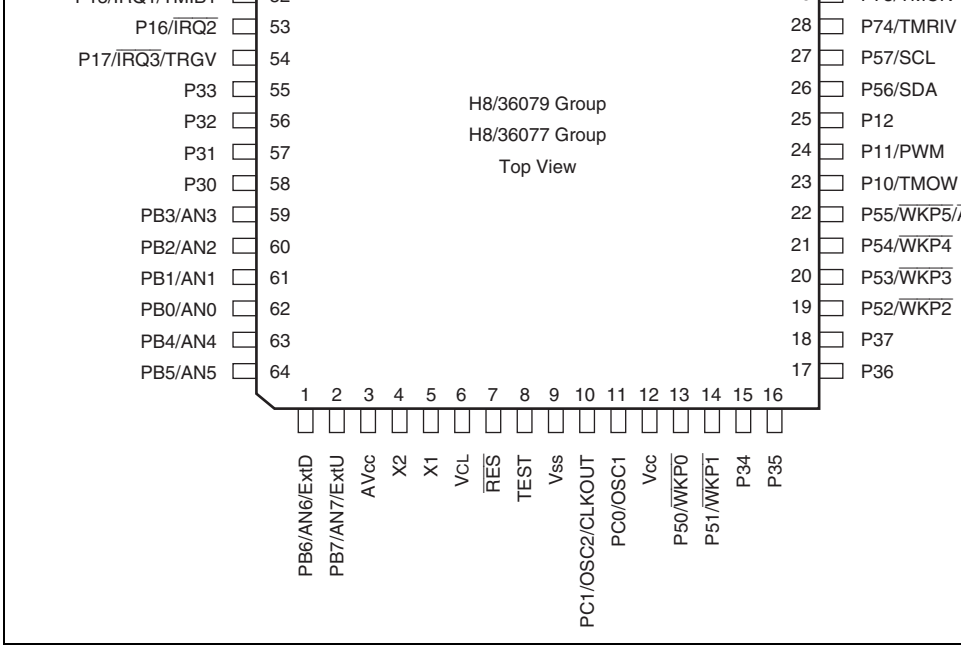


Figure 1.2 Pin Arrangement of H8/36079 Group and H8/36077 Group (FP-64K, FP-64A)

	AV_{CC}	3	Input	Analog power supply pin for the A/D co When the A/D converter is not used, co this pin to the system power supply.
	V_{CL}	6	Input	See section 20, Power Supply Circuit, f typical connection.
Clock pins	OSC1	11	Input	These pins connect with crystal or cera resonator for the system clock, or can b to input an external clock. When the on oscillator is used, the system clock can output on OSC2 pin. See section 5, Clock Pulse Generator, typical connection.
	OSC2/ CLKOUT	10	Output	
	X1	5	Input	These pins connect with a 32.768 kHz resonator for the subclock. See section Pulse Generator, for a typical connectio
	X2	4	Output	
System control	$\overline{RES}$	7	Input	Reset pin. The pull-up resistor (typ. 150 incorporated. When driven low, this LS
	TEST	8	Input	Test pin. Connect this pin to Vss.
Interrupt pins	$\overline{NMI}$	35	Input	Non-maskable interrupt request input p sure to pull up by a resistor.
	$\overline{IRQ0}$ to $\overline{IRQ3}$	51 to 54	Input	External interrupt request input pins. Ca the rising or falling edge.
	$\overline{WKP0}$ to $\overline{WKP5}$	13, 14, 19 to 22	Input	External interrupt request input pins. Ca the rising or falling edge.
RTC	TMOW	23	Output	This is an output pin for a divided clock
Timer B1	TMIB1	52	Input	External event input pin

	FTIOB0	34	I/O	Output compare output/input capture input/PWM output pin
	FTIOC0	33	I/O	Output compare output/input capture input/PWM sync output pin (at a rese complementary PWM mode)
	FTIOD0	32	I/O	Output compare output/input capture input/PWM output pin
	FTIOA1	37	I/O	Output compare output/input capture input/PWM output pin (at a reset, complementary PWM mode)
	FTIOB1 to FTIOD1	38 to 40	I/O	Output compare output/input capture input/PWM output pin
14-bit PWM	PWM	24	Output	14-bit PWM square wave output pin
I^2C bus interface 2 (IIC2)	SDA	26	I/O	I^2C data I/O pin. Can directly drive a NMOS open-drain output. When usi external pull-up resistor is required.
	SCL	27	I/O	I^2C clock I/O pin. Can directly drive a NMOS open-drain output. When usi external pull-up resistor is required.
Serial communication interface 3 (SCI3)	TXD, TXD_2	46, 50	Output	Transmit data output pin
	RXD, RXD_2	45, 49	Input	Receive data input pin
	SCK3, SCK3_2	44, 48	I/O	Clock I/O pin

	P17 to P14, P12 to P10	54 to 51, 25 to 23	I/O	7-bit I/O port
	P24 to P20	31, 47 to 44	I/O	5-bit I/O port
	P37 to P30	18 to 15, 55 to 58	I/O	8-bit I/O port
	P57 to P50	27, 26 22 to 19 14, 13	I/O	8-bit I/O port
	P67 to P60	40 to 37 32 to 34, 36	I/O	8-bit I/O port
	P76 to P74, P72 to P70	30 to 28, 50 to 48	I/O	6-bit I/O port
	P87 to P85	43 to 41	I/O	3-bit I/O port
Low-voltage detection circuit	ExtU, ExtD	2, 1	Input	This pin is used to externally input the detection voltage for the low-voltage circuit.

— 32-bit transfer and arithmetic and logic instructions are added
— Signed multiply and divide instructions are added.

- General-register architecture
  — Sixteen 16-bit general registers also usable as sixteen 8-bit registers and eight 16 registers, or eight 32-bit registers
- Sixty-two basic instructions
  — 8/16/32-bit data transfer and arithmetic and logic instructions
  — Multiply and divide instructions
  — Powerful bit-manipulation instructions
- Eight addressing modes
  — Register direct [Rn]
  — Register indirect [@ERn]
  — Register indirect with displacement [@(d:16,ERn) or @(d:24,ERn)]
  — Register indirect with post-increment or pre-decrement [@ERn+ or @–ERn]
  — Absolute address [@aa:8, @aa:16, @aa:24]
  — Immediate [#xx:8, #xx:16, or #xx:32]
  — Program-counter relative [@(d:8,PC) or @(d:16,PC)]
  — Memory indirect [@@aa:8]
- Address space
  — H8/36079 Group: 16 Mbytes
  — H8/36077 Group: 64 Kbytes

H8/36077 Group: Normal mode

- Power-down state
  - Transition to power-down state by SLEEP instruction

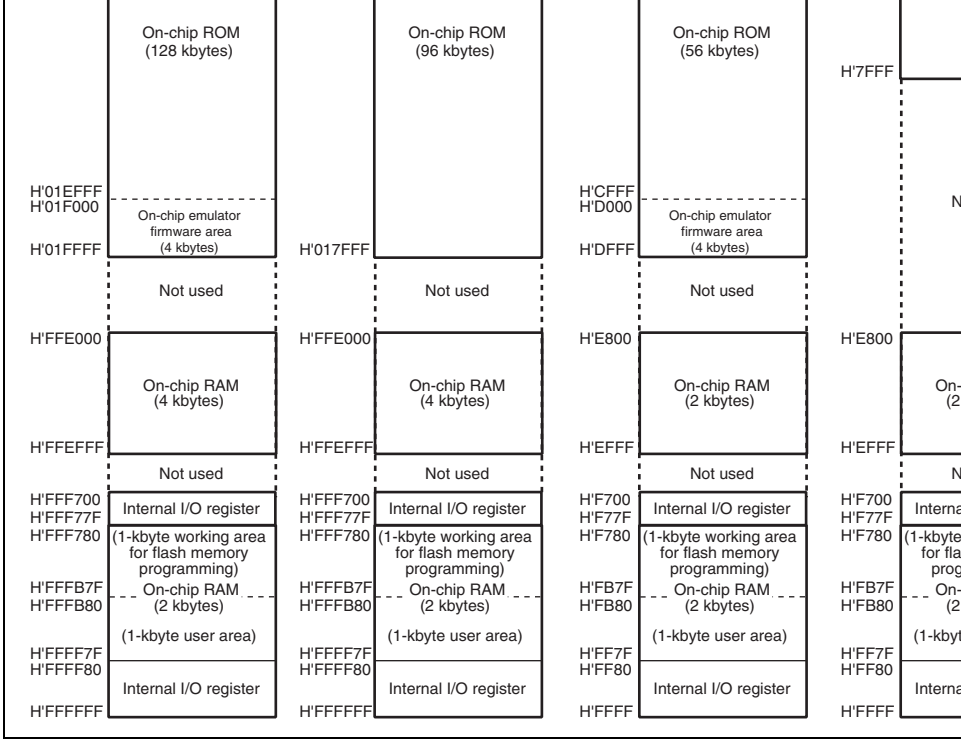
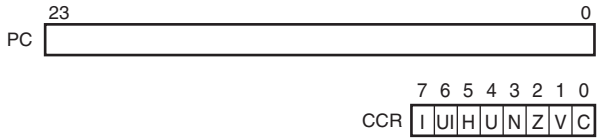


Figure 2.1 Memory Map

ER3	E3	R3H	R3L
ER4	E4	R4H	R4L
ER5	E5	R5H	R5L
ER6	E6	R6H	R6L
ER7	E7	R7H	R7L

(SP)

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 are divided into 8-bit registers designated by the letters RH (R0H to R7H
(R0L to R7L). These registers are functionally equivalent, providing a maximum of sixt
registers.

The usage of each register can be selected independently.

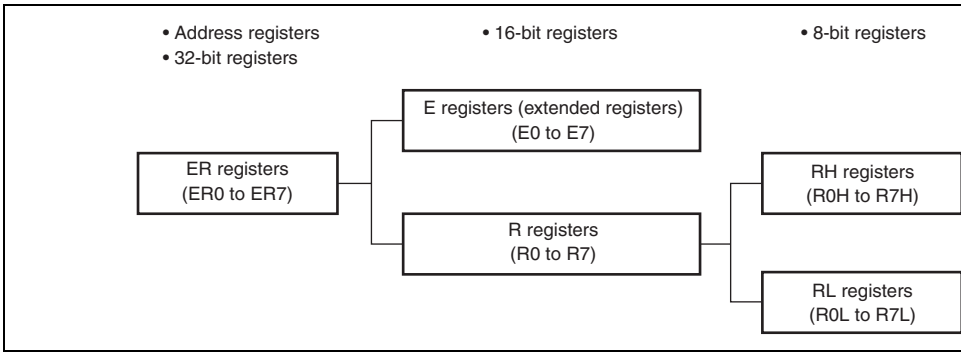


Figure 2.3 Usage of General Registers

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

Figure 2.4 Relationship between Stack Pointer and Stack Area

2.2.2 Program Counter (PC)

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

2.2.3 Condition-Code Register (CCR)

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

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

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

				When the ADD.B, ADDX.B, SUB.B, SUBX.B, or NEG.B instruction is executed, this flag is s there is a carry or borrow at bit 3, and cleared otherwise. When the ADD.W, SUB.W, CMP.W NEG.W instruction is executed, the H flag is s there is a carry or borrow at bit 11, and cleare otherwise. When the ADD.L, SUB.L, CMP.L, instruction is executed, the H flag is set to 1 if carry or borrow at bit 27, and cleared to 0 oth
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 sign bit.
2	Z	Undefined	R/W	Zero Flag Set to 1 to indicate zero data, and cleared to indicate non-zero data.
1	V	Undefined	R/W	Overflow Flag Set to 1 when an arithmetic overflow occurs, cleared to 0 at other times.
0	C	Undefined	R/W	Carry Flag Set to 1 when a carry occurs, and cleared to otherwise. Used by: • Add instructions, to indicate a carry • Subtract instructions, to indicate a borrow • Shift and rotate instructions, to indicate a The carry flag is also used as a bit accumulat manipulation instructions.

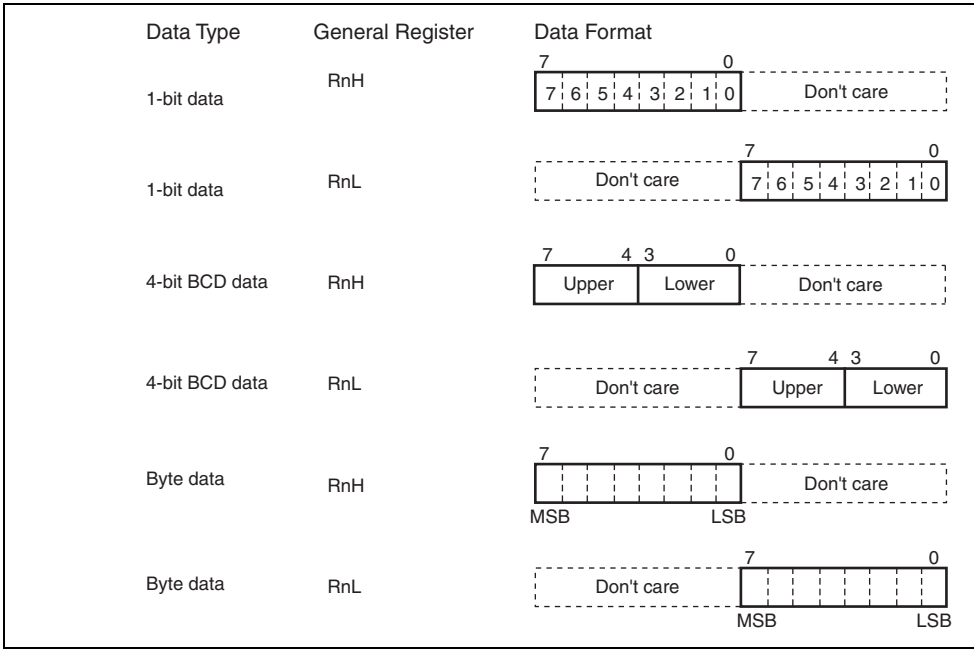
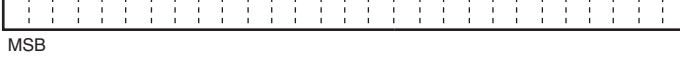


Figure 2.5 General Register Data Formats (1)



[Legend]

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

Figure 2.5 General Register Data Formats (2)

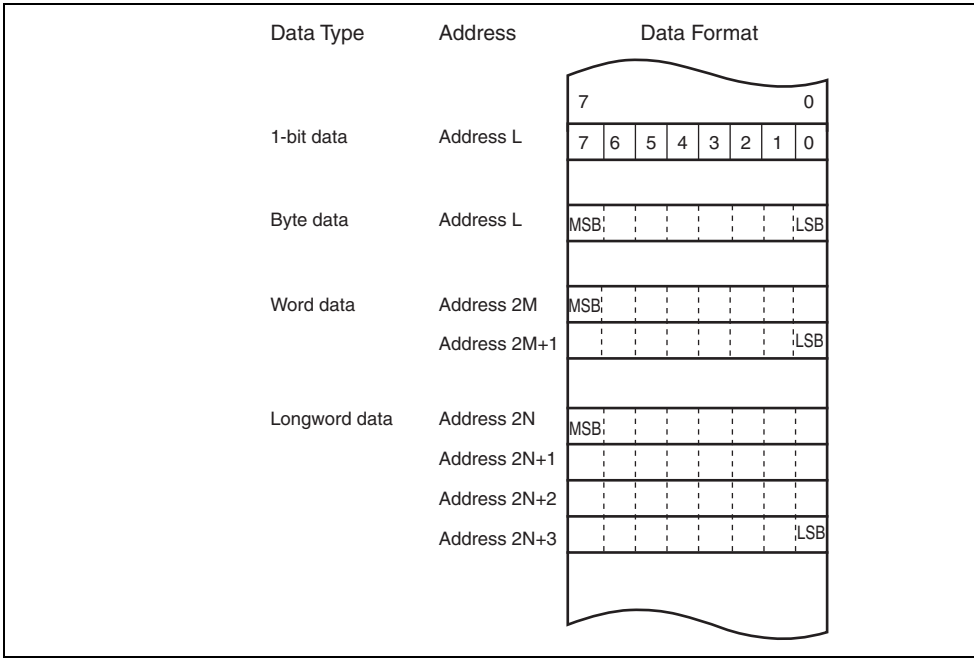


Figure 2.6 Memory Data Formats

Rs	General register (source)*
Rn	General register*
ERn	General register (32-bit register or address register)
(EAd)	Destination operand
(EAs)	Source operand
CCR	Condition-code register
N	N (negative) flag in CCR
Z	Z (zero) flag in CCR
V	V (overflow) flag in CCR
C	C (carry) flag in CCR
PC	Program counter
SP	Stack pointer
#IMM	Immediate data
disp	Displacement
+	Addition
–	Subtraction
×	Multiplication
÷	Division
∧	Logical AND
∨	Logical OR
⊕	Logical XOR
→	Move
~	NOT (logical complement)
:3/:8/:16/:24	3-, 8-, 16-, or 24-bit length

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

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

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

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

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

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

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

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

Table 2.5 Shift Instructions

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

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

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

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

		carry flag.
BILD	B	~(<bit-No.> of <EAd>) → C Transfers the inverse of a specified bit in a general register or operand to the carry flag. The bit number is specified by 3-bit immediate data.
BST	B	C → (<bit-No.> of <EAd>) Transfers the carry flag value to a specified bit in a general reg memory operand.
BIST	B	~C → (<bit-No.> of <EAd>) Transfers the inverse of the carry flag value to a specified bit i 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 ⊕ V = 0
		BLT	Less than	N ⊕ V = 1
		BGT	Greater than	Z∨(N ⊕ V) = 0
		BLE	Less or equal	Z∨(N ⊕ V) = 1
JMP	—	Branches unconditionally to a specified address.		
BSR	—	Branches to a subroutine at a specified address.		
JSR	—	Branches to a subroutine at a specified address.		
RTS	—	Returns from a subroutine		

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

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

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

else next;

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

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

Some instructions have two operation fields.

- Register Field

  Specifies a general register. Address registers are specified by 3 bits, and data registe
  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. A
  address or displacement is treated as a 32-bit data in which the first 8 bits are 0 (H'00

- Condition Field

  Specifies the branching condition of Bcc instructions.

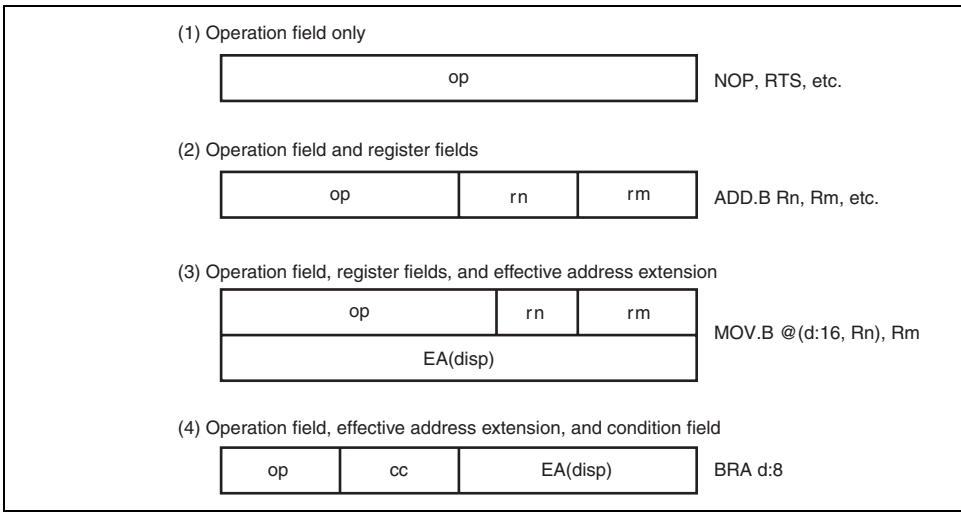


Figure 2.7 Instruction Formats

Bit-manipulation instructions use register direct, register indirect, or the absolute addressi
(@aa:8) to specify an operand, and register direct (BSET, BCLR, BNOT, and BTST instr
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

(1) Register Direct—Rn

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

(a) Register indirect with post-increment—@ERn+

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

(b) Register indirect with pre-decrement—@-ERn

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

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

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

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

Table 2.11 shows the access ranges of absolute addresses according to the modes of ope each product group.

The instruction contains 8-bit (#xx:8), 16-bit (#xx:16), or 32-bit (#xx:32) immediate data
operand.

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

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

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

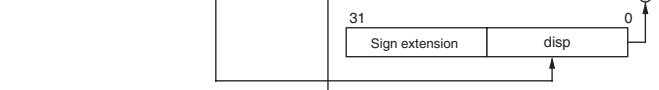
(8) Memory Indirect—@@aa:8

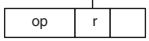
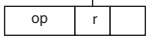
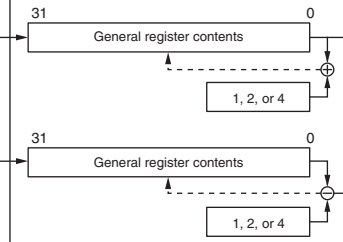
This mode can be used by the JMP and JSR instructions. The instruction code contains a
absolute address specifying a memory operand. This memory operand contains a branch
The memory operand is accessed by longword access. The first byte of the memory opera
ignored, generating a 24-bit branch address.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

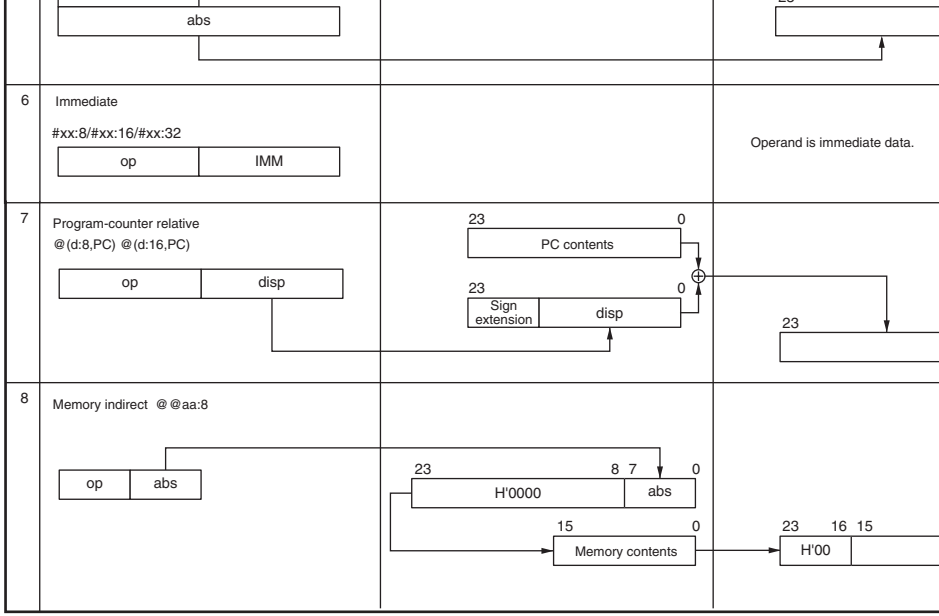
Note that the first part of the address range is also the exception vector area.

2.5.2 Effective Address Calculation

Table 2.12 indicates how effective addresses are calculated in each addressing mode. Fo
operation in normal mode, the upper 8 bits of the effective address are ignored in order t
a 16-bit effective address. For operation in advanced mode, the 24-bit result of effective
calculation is generated as the address.



4	Register indirect with post-increment or pre-decrement •Register indirect with post-increment @ERn+  •Register indirect with pre-decrement @-ERn 	 The value to be added or subtracted is 1 when the operand is byte size, 2 for word size, and 4 for longword size.



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

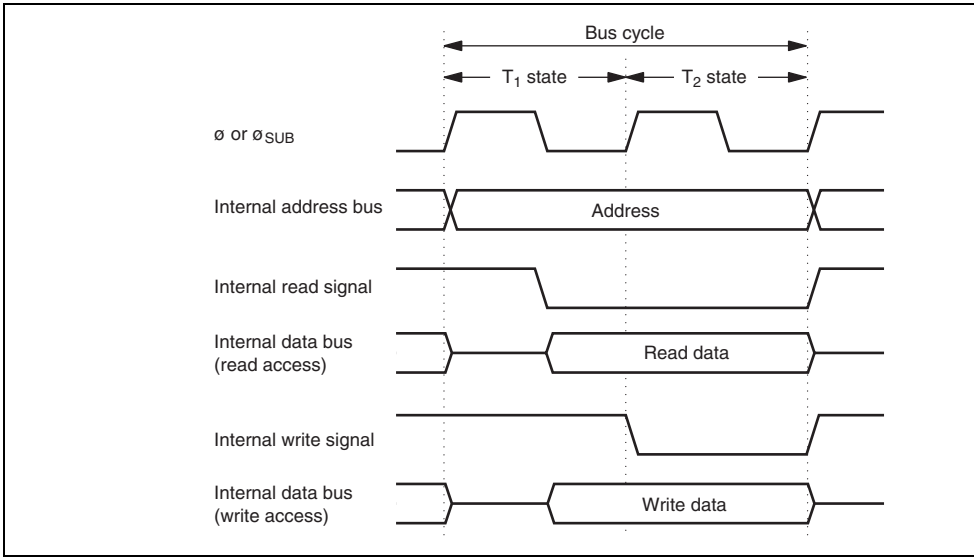


Figure 2.9 On-Chip Memory Access Cycle

module.

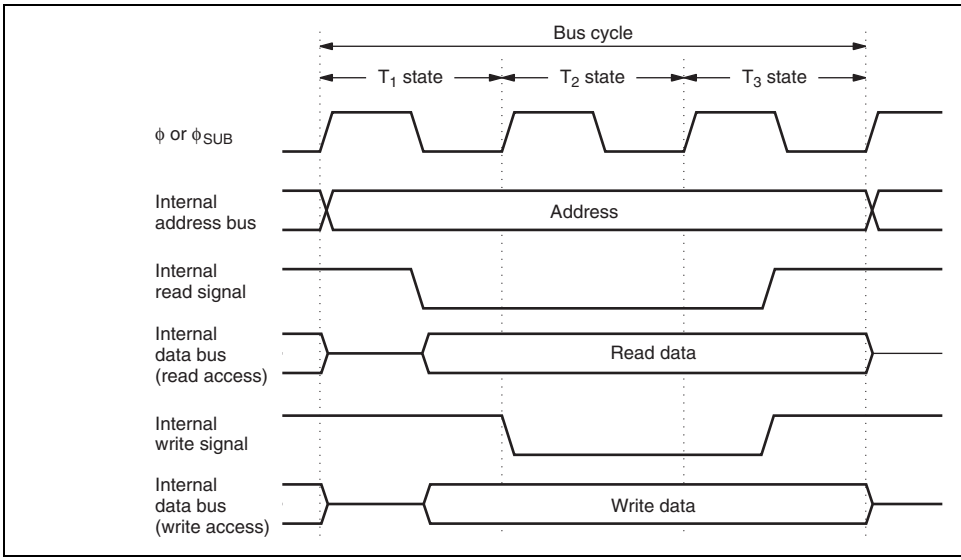


Figure 2.10 On-Chip Peripheral Module Access Cycle (3-State Access)

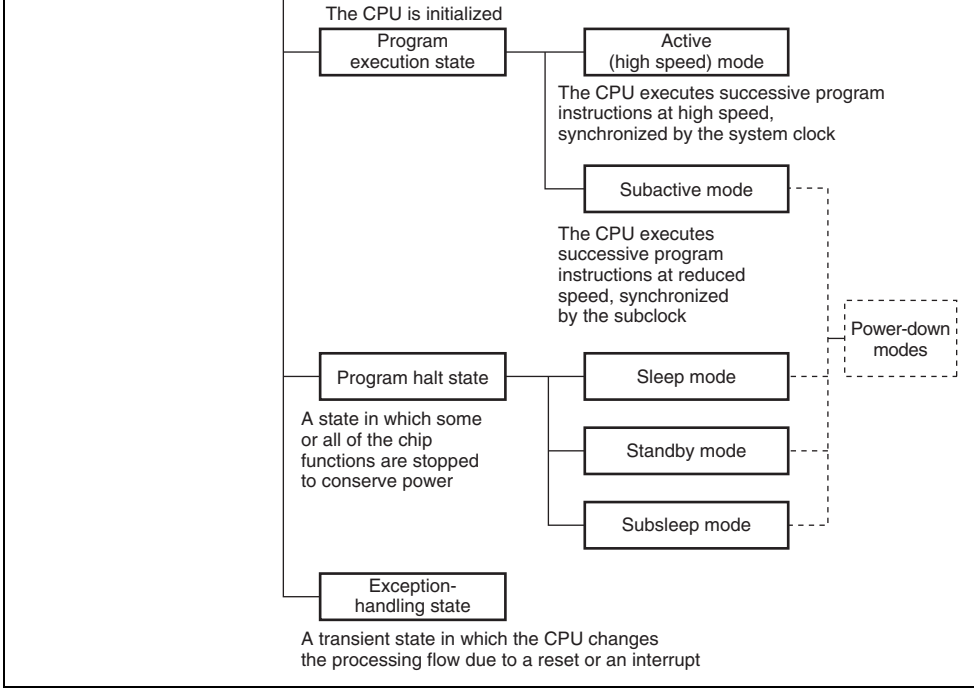


Figure 2.11 CPU Operation States

Figure 2.12 State Transitions

2.8 Usage Notes

2.8.1 Notes on Data Access to Empty Areas

The address space of this LSI includes empty areas in addition to the ROM, RAM, and
I/O registers areas available to the user. When data is transferred from CPU to empty ar
transferred data will be lost. This action may also cause the CPU to malfunction. When
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 number of bytes indicated by
starting at the address indicated by R5 to the address indicated by R6. In products for no
mode operation, set R4L and R6 so that the final address at the destination for transfer (
R6 + R4L) does not exceed H'FFFF (the value of R6 must not change from H'FFFF to H
during execution). In products for advanced-mode operation, set R4L and R6 so that the
address at the destination for transfer (value of R6 + R4L) does not exceed H'FFFFFF (t
of R6 must not change from H'FFFFFF to H'000000 during execution).

Example 1: Bit manipulation for the timer load register and timer counter (Applicable for timer B1 in the H8/36079 Group and H8/36077 Group.)

Figure 2.13 shows an example of a timer in which two timer registers are assigned to the address. When a bit-manipulation instruction accesses the timer load register and timer co a reloadable timer, since these two registers share the same address, the following operati 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 tim register. As a result, bits other than the intended bit in the timer counter may be modified modified value may be written to the timer load register.

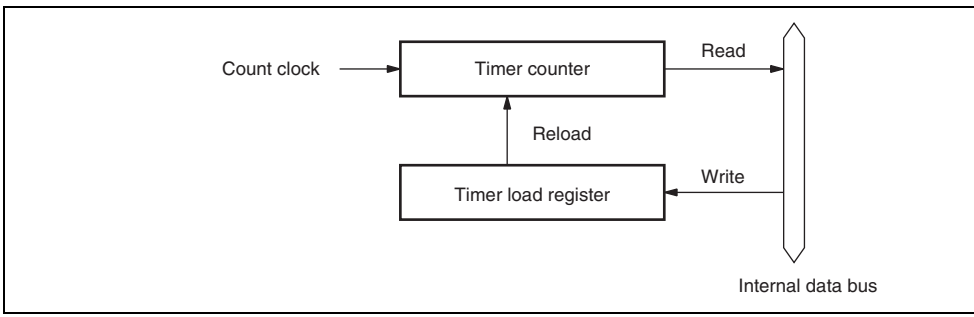


Figure 2.13 Example of Timer Configuration with Two Registers Allocated to Same Address

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

- Description on operation

1. When the BSET instruction is executed, first the CPU reads port 5.
   Since P57 and P56 are input pins, the CPU reads the pin states (low-level and high-l input).
   P55 to P50 are output pins, so the CPU reads the value in PDR5. In this example PD value of H'80, but the value read by the CPU is H'40.
2. Next, the CPU sets bit 0 of the read data to 1, changing the PDR5 data to H'41.
3. Finally, the CPU writes H'41 to PDR5, completing execution of BSET 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
RAM0	1	0	0	0	0	0	0

- BSET instruction executed

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

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

- After executing BSET instruction

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

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

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

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

- BCLR instruction executed

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

The BCLR instruction is executed for PCR5.

- After executing BCLR instruction

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

- Description on operation

1. When the BCLR instruction is executed, first the CPU reads PCR5. Since PCR5 is a register, the CPU reads a value of H'FF, even though the PCR5 value is actually H'3
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. Ho bits 7 and 6 in PCR5 change to 1, so that P57 and P56 change from input pins to outp To prevent this problem, store a copy of the PDR5 data in a work area in memory an manipulate data of the bit in the work area, then write this data to PDR5.

PDR5	1	0	0	0	0	0	0
RAM0	0	0	1	1	1	1	1

- BCLR instruction executed

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

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

- After executing BCLR instruction

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

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

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

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

- Interrupts

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

—	Reserved for system use	1 to 6	H'0002 to H'000D	H'000004 to H'00001B
External interrupt pin	NMI	7	H'000E to H'000F	H'00001C to H'00001F
CPU	Trap instruction #0	8	H'0010 to H'0011	H'000020 to H'000023
	Trap instruction #1	9	H'0012 to H'0013	H'000024 to H'000027
	Trap instruction #2	10	H'0014 to H'0015	H'000028 to H'00002B
	Trap instruction #3	11	H'0016 to H'0017	H'00002C to H'00002F
Address break	Break conditions satisfied	12	H'0018 to H'0019	H'000030 to H'000033
CPU	Direct transition by executing the SLEEP instruction	13	H'001A to H'001B	H'000034 to H'000037
External interrupt pin	IRQ0 Low-voltage detection interrupt	14	H'001C to H'001D	H'000038 to H'00003B
	IRQ1	15	H'001E to H'001F	H'00003C to H'00003F
	IRQ2	16	H'0020 to H'0021	H'000040 to H'000043
	IRQ3	17	H'0022 to H'0023	H'000044 to H'000047
	WKP	18	H'0024 to H'0025	H'000048 to H'00004B
RTC	Overflow	19	H'0026 to H'0027	H'00004C to H'0004F

	SCI3 receive error			
IIC2	Transmit data empty Transmit end Receive data full Arbitration lost/Overrun error NACK detection Stop conditions detected	24	H'0030 to H'0031	H'000060 to H'00006
A/D converter	A/D conversion end	25	H'0032 to H'0033	H'000064 to H'00006
Timer Z	Compare match/input capture A0 to D0 Timer Z overflow	26	H'0034 to H'0035	H'000068 to H'00006
	Compare match/input capture A1 to D1 Timer Z overflow Timer Z underflow	27	H'0036 to H'0037	H'00006C to H'00006
—	Reserved for system use	28	H'0038 to H'0039	H'000070 to H'00007
Timer B1	Timer B1 overflow	29	H'003A to H'003B	H'000074 to H'00007
—	Reserved for system use	30, 31	H'003C to H'003F	H'000078 to H'00007
SCI3_2	Receive data full Transmit data empty Transmit end Receive error	32	H'0040 to H'0041	H'000080 to H'00008
—	Reserved for system use	33	H'0042 to H'0043	H'000084 to H'00008
Clock source switching	Clock source switching (from external clock to on-chip oscillator)	34	H'0044 to H'0045	H'000088 to H'00008

Note: * A low-voltage detection interrupt is enabled only in the product with an on-chi on reset and low-voltage detection circuit.

- Wakeup interrupt flag register (IWPR)

3.2.1 Interrupt Edge Select Register 1 (IEGR1)

IEGR1 selects the direction of an edge that generates interrupt requests of pins $\overline{\text{NMI}}$ and $\overline{\text{IRQ0}}$.

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

				1: Rising edge of $\overline{\text{WKP5}}$($\overline{\text{ADTRG}}$) pin input is d
4	WPEG4	0	R/W	WKP4 Edge Select
				0: Falling edge of $\overline{\text{WKP4}}$ pin input is detected
				1: Rising edge of $\overline{\text{WKP4}}$ pin input is detected
3	WPEG3	0	R/W	WKP3 Edge Select
				0: Falling edge of $\overline{\text{WKP3}}$ pin input is detected
				1: Rising edge of $\overline{\text{WKP3}}$ pin input is detected
2	WPEG2	0	R/W	WKP2 Edge Select
				0: Falling edge of $\overline{\text{WKP2}}$ pin input is detected
				1: Rising edge of $\overline{\text{WKP2}}$ pin input is detected
1	WPEG1	0	R/W	WKP1Edge Select
				0: Falling edge of $\overline{\text{WKP1}}$ pin input is detected
				1: Rising edge of $\overline{\text{WKP1}}$ pin input is detected
0	WPEG0	0	R/W	WKP0 Edge Select
				0: Falling edge of $\overline{\text{WKP0}}$ pin input is detected
				1: Rising edge of $\overline{\text{WKP0}}$ pin input is detected

				When this bit is set to 1, RTC interrupt requests enabled.
5	IENWP	0	R/W	Wakeup Interrupt Enable This bit is an enable bit, which is common to the $\overline{WKP5}$ to $\overline{WKP0}$. When the bit is set to 1, interru requests are enabled.
4	—	1	—	Reserved This bit is always read as 1.
3	IEN3	0	R/W	IRQ3 Interrupt Enable When this bit is set to 1, interrupt requests of the are enabled.
2	IEN2	0	R/W	IRQ2 Interrupt Enable When this bit is set to 1, interrupt requests of the are enabled.
1	IEN1	0	R/W	IRQ1 Interrupt Enable When this bit is set to 1, interrupt requests of the are enabled.
0	IEN0	0	R/W	IRQ0 Interrupt Enable When this bit is set to 1, interrupt requests of the 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 cl operations are performed while I = 0, and as a result a conflict arises between the clear in and an interrupt request, exception handling for the interrupt will be executed after the cl instruction has been executed.

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

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

3.2.5 Interrupt Flag Register 1 (IRR1)

IRR1 is a status flag register for direct transition interrupts, RTC interrupts, and $\overline{\text{IRQ3}}$ to 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 S instruction while DTON in SYSCR2 is set to 1. [Clearing condition] When IRRDT is cleared by writing 0

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

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

3.2.7 Wakeup Interrupt Flag Register (IWPR)

IWPR is a status flag register for $\overline{\text{WKP5}}$ to $\overline{\text{WKP0}}$ interrupt requests.

Bit	Bit Name	Initial Value	R/W	Description
7, 6	—	All 1	—	Reserved
				These bits are always read as 1.
5	IWPF5	0	R/W	WKP5 Interrupt Request Flag
				[Setting condition]
				When $\overline{\text{WKP5}}$ pin is designated for interrupt inpu 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 $\overline{\text{WKP4}}$ pin is designated for interrupt inpu designated signal edge is detected.
				[Clearing condition]
				When IWPF4 is cleared by writing 0.

				When $\overline{\text{WKP2}}$ pin is designated for interrupt input 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 $\overline{\text{WKP1}}$ pin is designated for interrupt input designated signal edge is detected. [Clearing condition] When IWPF1 is cleared by writing 0.
0	IWPF0	0	R/W	WKP0 Interrupt Request Flag [Setting condition] When $\overline{\text{WKP0}}$ pin is designated for interrupt input designated signal edge is detected. [Clearing condition] When IWPF0 is cleared by writing 0.

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, the data in that addres the program counter (PC) as the start address, and program execution starts from tha The reset exception handling vector address is H'0000 to H'0001 for normal mode op and H'000000 to H'000003 in advanced mode operation.

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

(2) IRQ3 to IRQ0 Interrupts

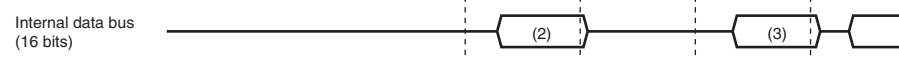
IRQ3 to IRQ0 interrupts are requested by input signals to pins $\overline{\text{IRQ3}}$ to $\overline{\text{IRQ0}}$. These f
interrupts are given different vector addresses, and are detected individually by either
edge sensing or falling edge sensing, depending on the settings of bits IEG3 to IEG0 i

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

(3) WKP5 to WKP0 Interrupts

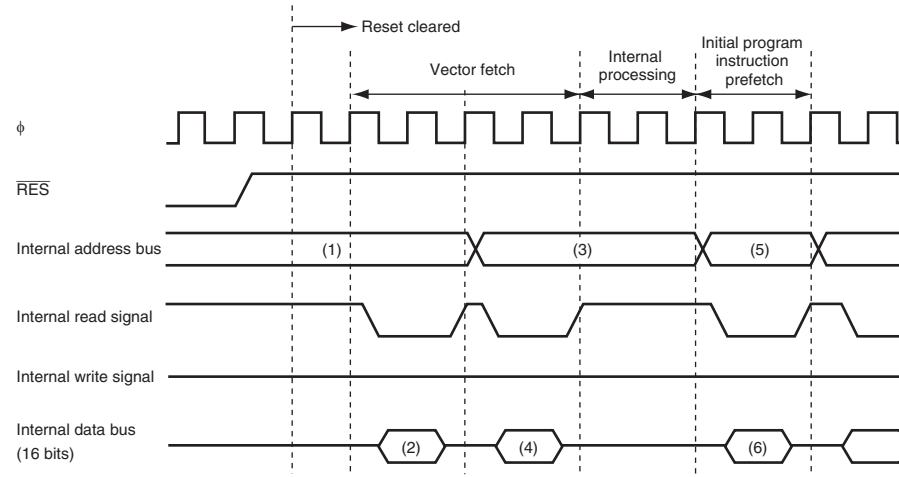
WKP5 to WKP0 interrupts are requested by input signals to pins $\overline{\text{WKP5}}$ to $\overline{\text{WKP0}}$. Th
interrupts have the same vector addresses, and are detected individually by either risin
sensing or falling edge sensing, depending on the settings of bits WPEG5 to WPEG0
IEGR2.

When pins $\overline{\text{WKP5}}$ to $\overline{\text{WKP0}}$ are designated for interrupt input in PMR5 and the desig
signal edge is input, the corresponding bit in IWPR is set to 1, requesting the CPU of
interrupt. These interrupts can be masked by setting bit IENWP in IENR1.



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

Advanced mode operation



(1), (3) Reset exception handling vector address ((1) = H'000000 (3) = H'000002)
(2), (4) Start address (contents of reset exception handling vector address)
(5) Start address
(6) Initial program instruction

Figure 3.1 Reset Sequence

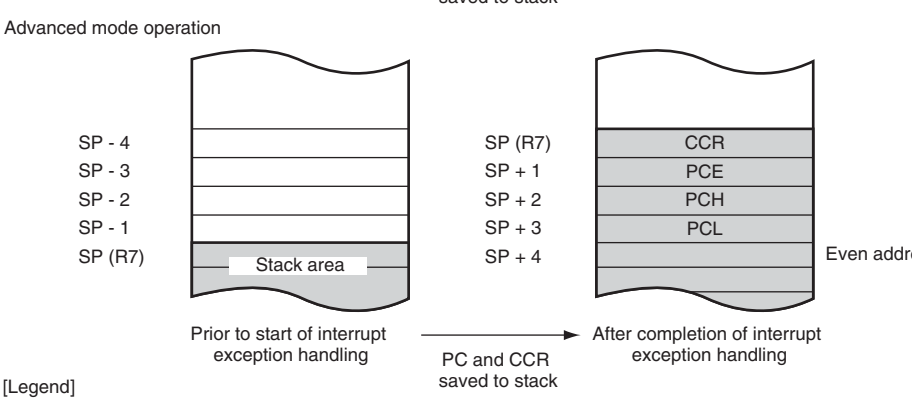
3.4.3 Interrupt Handling Sequence

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

Figure 3.3 shows a typical interrupt sequence where the program area is in the on-chip R the stack area is in the on-chip RAM.



[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
2. Register contents must always be saved and restored in word units, starting from an even-numbere
* Ignored when returning from the interrupt handling routine.

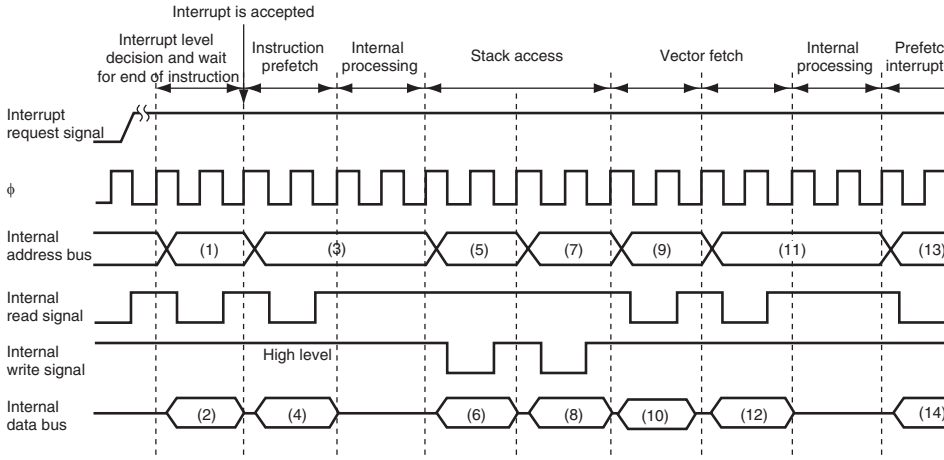
Figure 3.2 Stack Status after Exception Handling

Vector fetch	2 (1)
Instruction fetch	4
Internal processing	4

Notes: 1. Not including EEPMOV instruction.
2. The value in parentheses is the number of states in advance mode.

(1) Instruction prefetch address (Instruction is not executed. Address is saved as PC contents, becoming return address.)
(2), (4) Instruction code (Instruction is not executed.)
(3) Instruction prefetch address (Instruction is not executed.)
(5) SP - 2
(6) SP - 4
(7) CCR
(8) Vector address
(9) Start address of interrupt handling routine (contents of ve
(10) First instruction of interrupt handling routine.

Advanced mode operation



(1) Instruction prefetch address (Instruction is not executed. Address is saved as PC contents, becoming return address.)
(2), (4) Instruction code (Instruction is not executed.)
(3) Instruction prefetch address (Instruction is not executed.)
(5) SP - 2
(7) SP - 4
(6), (8) Saved PC and CCR
(9), (11) Vector address
(10), (12) Start address of interrupt handling routine (contents o
(13) Start address of interrupt handling routine ((13), (10),
(14) First instruction of interrupt handling routine.

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. Acce
stack always takes place in word size, so the stack pointer (SP: R7) should never indicate
address. Use PUSH Rn (MOV.W Rn, @–SP) or POP Rn (MOV.W @SP+, Rn) to save or
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, $\overline{\text{I}}$
$\overline{\text{IRQ0}}$, 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
After accessing the port mode register, execute at least one instruction (e.g., NOP), then c
interrupt request flag from 1 to 0.

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

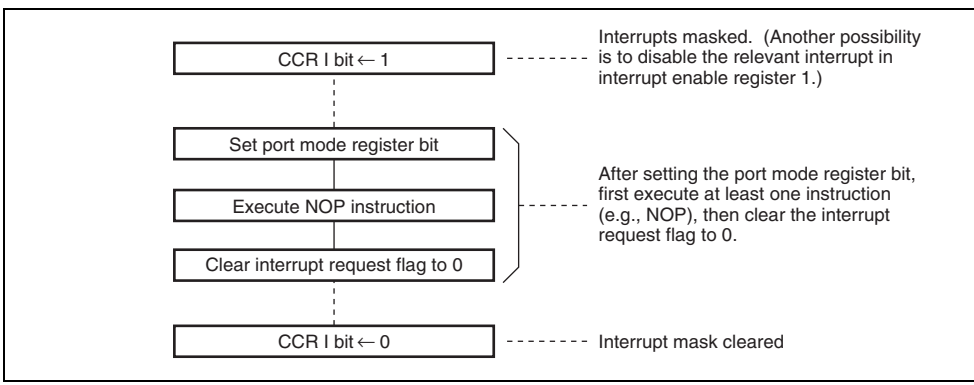
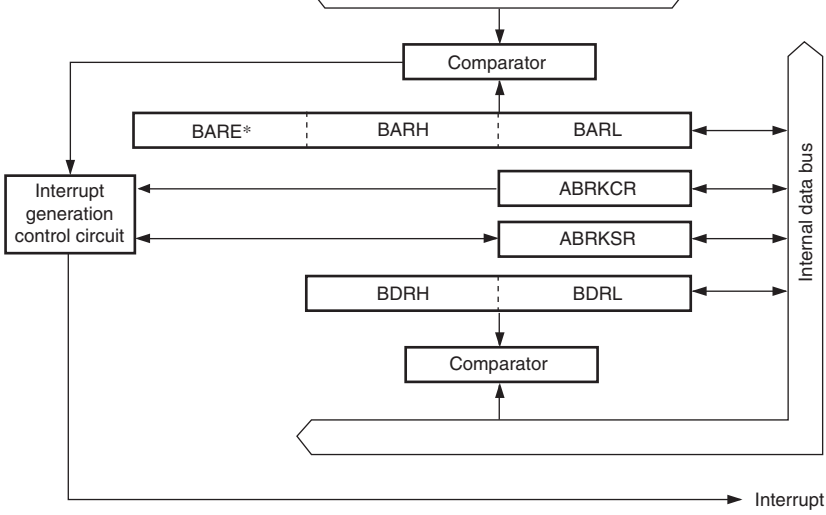


Figure 3.4 Port Mode Register Setting and Interrupt Request Flag Clearing Pro



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

Note: * Only provided for microcontrollers that support advanced mode.

Figure 4.1 Block Diagram of Address Break

ABRKCR sets address break conditions.

Bit	Bit Name	Initial Value	R/W	Description
7	RTINTE	1	R/W	RTE Interrupt Enable When this bit is 0, the interrupt immediately after executing RTE is masked and then one instructi be executed. When this bit is 1, the interrupt is n masked.
6	CSEL1	0	R/W	Condition Select 1 and 0 These bits set address break conditions. 00: Instruction execution cycle 01: CPU data read cycle 10: CPU data write cycle 11: CPU data read/write cycle
5	CSEL0	0	R/W	

				1XX: Reserved (setting prohibited) Advanced mode 000: Compares 24-bit addresses 001: Compares the higher-order 20 bits of the a 010: Compares the higher-order 16 bits of the a 011: Compares the higher-order 12 bits of the a 1XX: Reserved (setting prohibited)
1	DCMP1	0	R/W	Data Compare Condition Select 1 and 0
0	DCMP0	0	R/W	These bits set the comparison condition betwee set in BDR and the internal data bus. 00: No data comparison 01: Compares lower 8-bit data between BDRL a bus 10: Compares upper 8-bit data between BDRH bus 11: Compares 16-bit data between BDR and da

[Legend] X: Don't care.

RAM space	Upper 8 bits	Lower 8 bits	Upper 8 bits	Upper
I/O register with 8-bit data bus width	Upper 8 bits	Upper 8 bits	Upper 8 bits	Upper
I/O register with 16-bit data bus width	Upper 8 bits	Lower 8 bits	—	—

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

4.1.3 Break Address Registers (BARE, BARH, BARL)

Settings of the address where an address-break interrupt is to be generated are made in t address registers (BAR: BARE*, BARH and BARL). For microcontrollers that support mode operation, BAR is a 16-bit readable/writable register with initial value H'FFFF. Fo microcontrollers that support advanced-mode operation, BAR is a 24-bit readable/writab with the initial value H'FFFFFF. When setting an instruction execution cycle as the addr condition, set BAR to the address of the first byte of the instruction.

Note: * BARE is only provided for microcontrollers that support advanced-mode ope

4.1.4 Break Data Registers (BDRH, BDRL)

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

When the address break is specified in instruction execution cycle

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

Program

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

Underline indicates the address to be stacked.

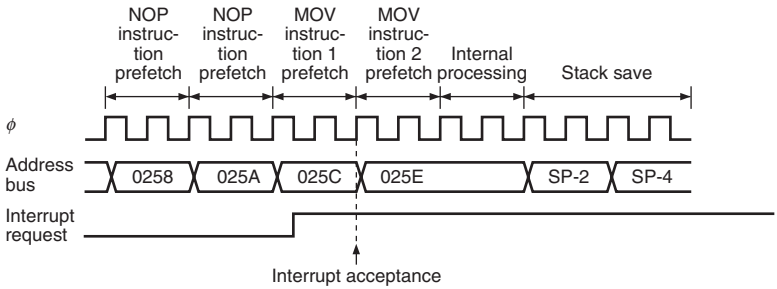


Figure 4.2 Address Break Interrupt Operation Example (1)

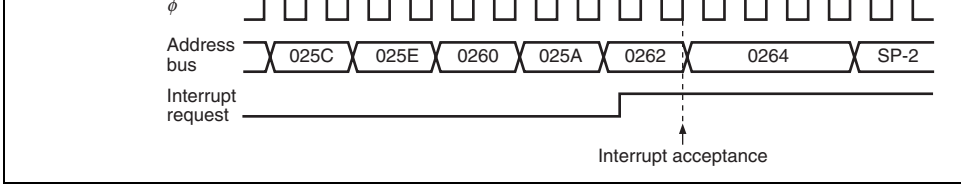


Figure 4.2 Address Break Interrupt Operation Example (2)

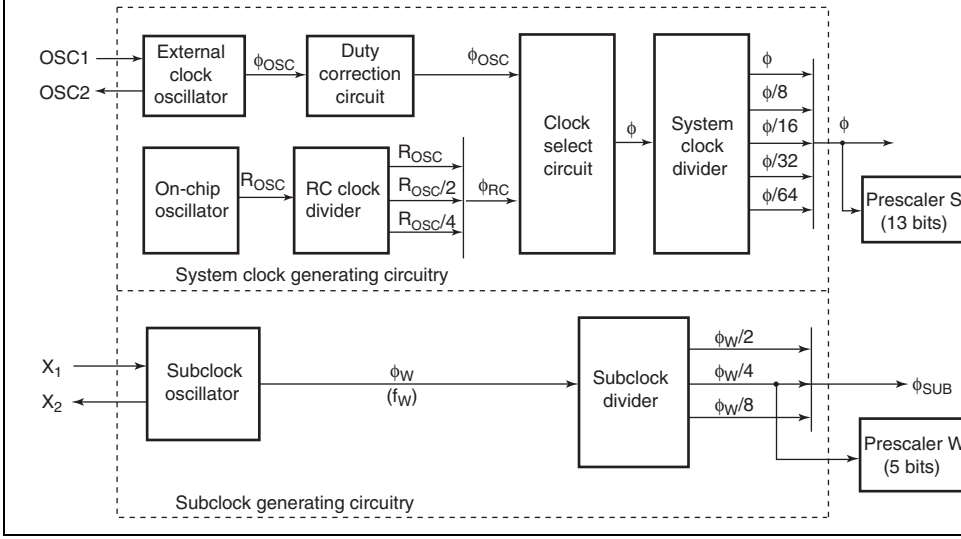


Figure 5.1 Block Diagram of Clock Pulse Generator

The system clock (φ) and subclock (ϕ_{SUB}) are basic clocks on which the CPU and on-chip peripheral modules operate. The system clock is divided into from φ/2 to φ/8192 by pres The subclock is divided into from $\phi_w/8$ to $\phi_w/128$ by prescaler W. These divided clocks a supplied to respective peripheral modules.

Since the initial frequency of the on-chip oscillator in the flash memory version is wit
range of two frequencies shown above, it is normally unnecessary to trim the frequenc
however, still possible to adjust it by rewriting the trimming registers.

- Backup of the external oscillation halt

  This system detects the external oscillator halt. If detected, the system clock source is
  automatically switched to the on-chip oscillator clock.
- Interrupt can be requested to the CPU when the system clock is switched from the ext
  clock to the on-chip oscillator clock.

5.2 Register Descriptions

The CPG has the following registers.

- RC control register (RCCR)
- RC trimming data protect register (RCTRMDPR)
- RC trimming data register (RCTRMDR)
- Clock control/status register (CKCSR)

				0: 16 MHz 1: 20 MHz
5	VCLSEL	0	R/W	Power Supply Select for On-chip Oscillator 0: Selects VBGR 1: Selects VCL When the VCL power is selected, the accuracy chip oscillator frequency cannot be guaranteed
4 to 2	—	All 0	—	Reserved These bits are always read as 0.
1	RCPSC1	1	R/W	Division Ratio Select for On-chip Oscillator
0	RCPSC0	0	R/W	The division ratio of R_{osc} changes right after rev bit. These bits can be written to only when the CKS CKCSR is 0. 0x: R_{osc} (not divided) 10: $R_{osc}/2$ 11: $R_{osc}/4$

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

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

5.2.3 RC Trimming Data Register (RCTRMDR)

RCTRMDR stores the trimming data of the on-chip oscillator frequency (FSEL = 1, 20

Bit	Bit Name	Initial Value	R/W	Description
7	TRMD7	(0)*	R/W	Trimming Data (FSEL = 1, 20 MHz)
6	TRMD6	(0)*	R/W	The trimming data is loaded from the flash mem
5	TRMD5	(0)*	R/W	register right after a reset.
4	TRMD4	(0)*	R/W	The on-chip oscillator clock (FSEL = 1, 20 MHz
3	TRMD3	(0)*	R/W	trimmed by changing these bits.
2	TRMD2	(0)*	R/W	The frequency of the on-chip oscillator clock ch
1	TRMD1	(0)*	R/W	right after writing these bits. These bits are initi H'00.
0	TRMD0	(0)*	R/W	Changes in frequency are shown below (bit TR sign bit).
				(Min.) H'80 ← … ← H'FF ← … ← H'00 → … → → H'7F (Max.)

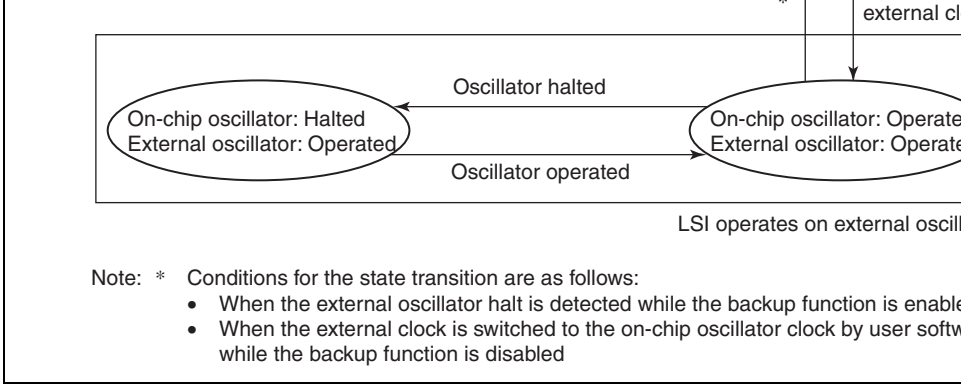
Note: * These values are initialized to the trimming data loaded from the flash memor

1	0	CLKOUT	I/O	
0	1	(Open)	OSC1 (extern clock input)	
1	1	OSC2	OSC1	

5	OSCBAKE	0	R/W	External Clock Backup Enable 0: External clock backup disabled 1: External clock backup enabled The external oscillation detecting circuit is enable this bit is 1. When the external oscillator halt is d while this LSI operates on the external input sign system clock source is automatically switched to chip oscillator regardless of the value of bit 4 in t register. Note: The external oscillation detecting circuit o on the on-chip oscillator clock. When this to 1, do not set the on-chip oscillator to th mode by the RCSTP bit in RCCR.

				This bit is used to switch the on-chip oscillator c external clock. While this LSI is operating on the oscillator clock, setting this bit to 1 switches the clocks to the external clock. [Setting condition] • When 1 is written to this bit while CKSWIF = [Clearing conditions] • When 0 is written to this bit • When the external oscillator halt is detected OSCBAKE = 1
3	CKSWIE	0	R/W	Clock Switching Interrupt Enable Setting this bit to 1 enables the clock switching i request.
2	CKSWIF	0	R/W	Clock Switching Interrupt Request Flag [Setting condition] • When the external clock is switched to the o oscillator clock as the system clock source [Clearing condition] • When writing 0 after reading as 1

0	CKSTA	0	R	LSI Operating Clock Status 0: This LSI operates on the on-chip oscillator clo 1: This LSI operates on the external clock.

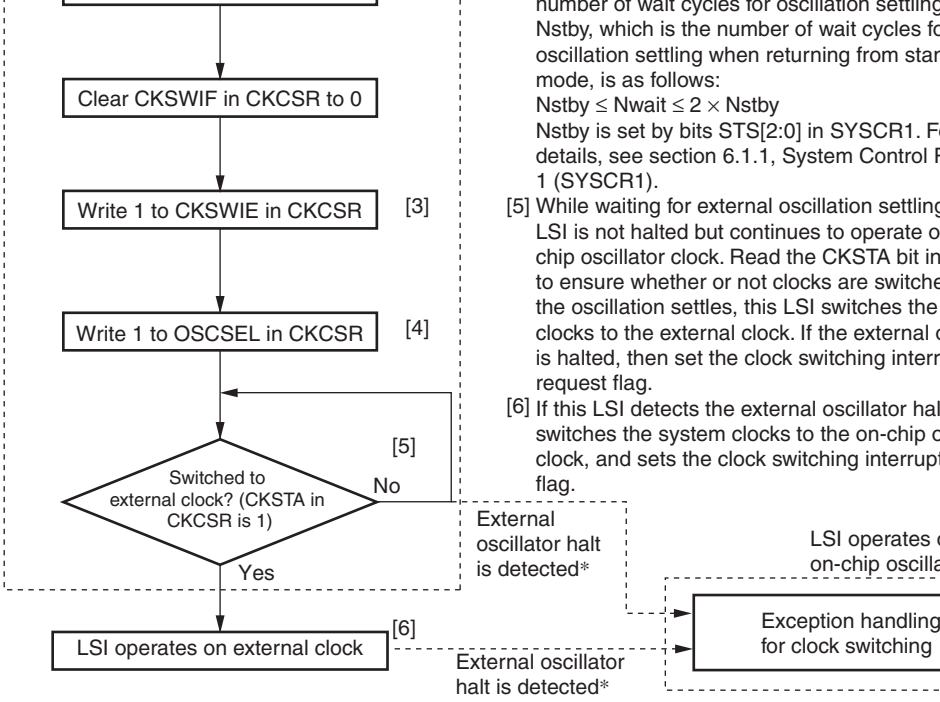


Note: * Conditions for the state transition are as follows:
- When the external oscillator halt is detected while the backup function is enable
- When the external clock is switched to the on-chip oscillator clock by user softw while the backup function is disabled

Figure 5.2 State Transition of System Clock

5.3.1 Clock Control Operation

The LSI system clock is generated by the on-chip oscillator clock after a reset. The syste sources are switched from the on-chip oscillator to the external clock by the user softwa 5.3 shows the flowchart to switch clocks with the external clock backup function enable 5.4 and 5.5 show the flowcharts to switch clocks with the external clock backup functio



number of wait cycles for oscillation settling
Nstby, which is the number of wait cycles fo
oscillation settling when returning from star
mode, is as follows:
Nstby ≤ Nwait ≤ 2 × Nstby
Nstby is set by bits STS[2:0] in SYSCR1. Fo
details, see section 6.1.1, System Control R
1 (SYSCR1).

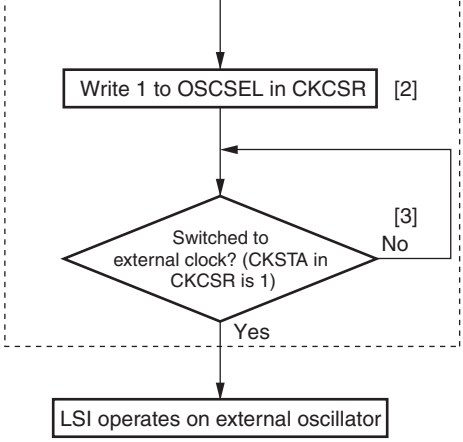
[5] While waiting for external oscillation settling
LSI is not halted but continues to operate o
chip oscillator clock. Read the CKSTA bit in
to ensure whether or not clocks are switche
the oscillation settles, this LSI switches the
clocks to the external clock. If the external c
is halted, then set the clock switching interr
request flag.

[6] If this LSI detects the external oscillator hal
switches the system clocks to the on-chip c
clock, and sets the clock switching interrupt
flag.

LSI operates o
on-chip oscilla

Note: * To prevent the LSI from malfunctioning at the external oscillation halt, switching the clock so
with the watchdog timer is highly recommended.

Figure 5.3 Flowchart of Clock Switching with Backup Function Enabled



Nstby ≤ Nwait ≤ 2 × Nstby
Nstby is set by bits STS[2:0] in SYSCR
details, see section 6.1.1, System Cont
Register 1 (SYSCR1).

[3] While the system is waiting for the exte
oscillation settling, this LSI is not halte
continues to operate on the on-chip os
clock. Read the value of the CKSTA bit
CKCSR to ensure that the system cloc
switched.

Figure 5.4 Flowchart of Clock Switching with Backup Function Disabled (
(From On-Chip Oscillator Clock to External Clock)

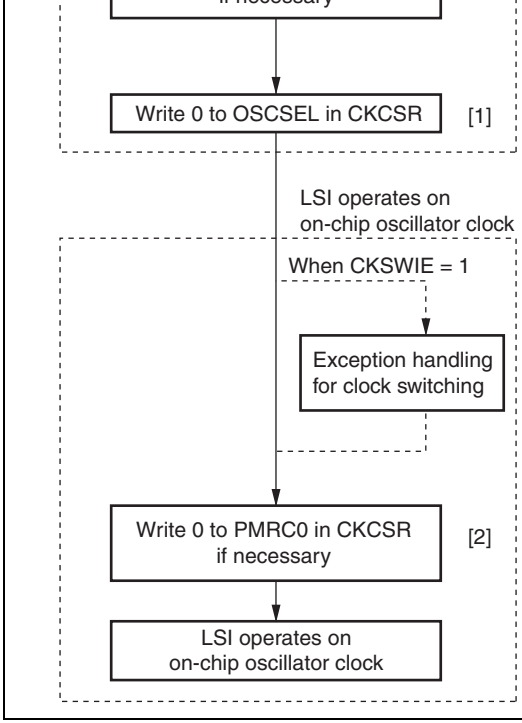
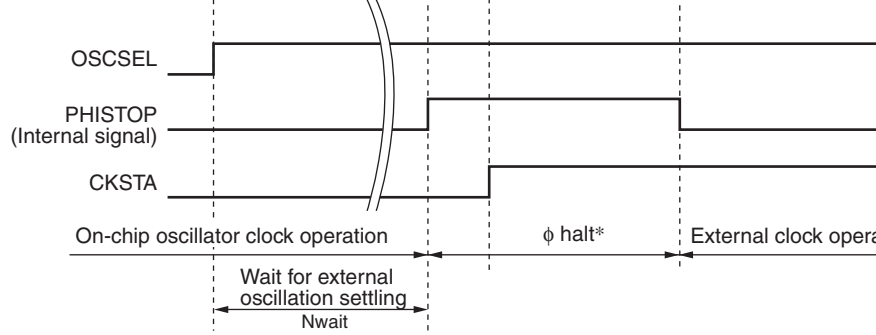


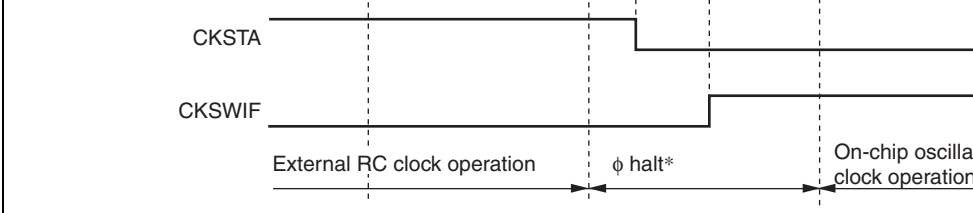
Figure 5.5 Flowchart of Clock Switching with Backup Function Disabled (2
(From External Clock to On-Chip Oscillator Clock)



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

Note: * The ϕ halt duration is the duration from the timing when the ϕ clock stops to the f
rising edge of the ϕ_{OSC} clock after seven clock cycles of the ϕ_{RC} clock have elap

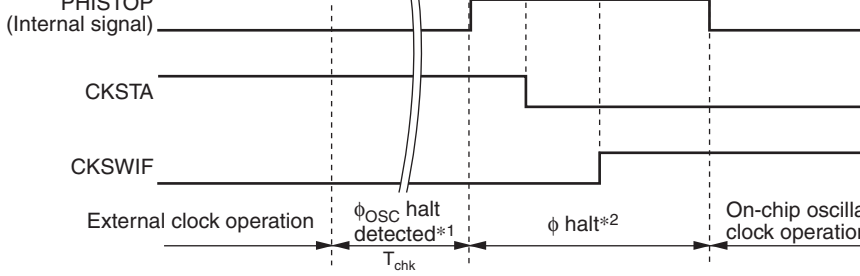
Figure 5.6 Timing Chart of Switching from On-Chip Oscillator Clock to Extern



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

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

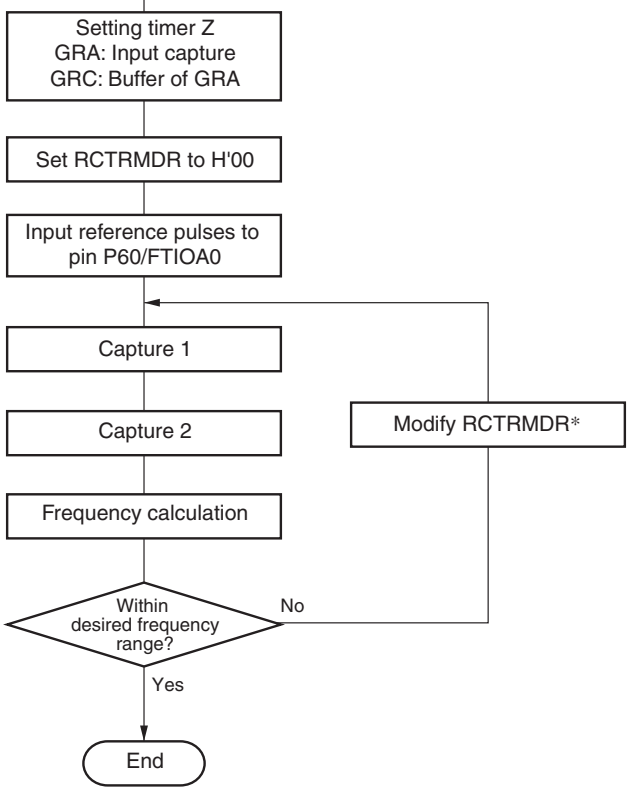
Figure 5.7 Timing Chart to Switch from External Clock to On-Chip Oscillator



[Legend]
ϕOSC: External clock
ϕRC: On-chip oscillator clock
ϕ: System clock
OSCHLT: Bit 1 in CKCSR
PHISTOP: System clock stop control signal
CKSTA: Bit 0 in CKCSR
CKSWIF: Bit 2 in CKCSR

Notes: 1. $44 \times \phi_{RC} \leq T_{chk} \leq 48 \times \phi_{RC}$
2. The ϕ halt duration is the duration from the timing when the ϕ clock stops to the seventh rising edge of the ϕ_{RC} clock.

Figure 5.8 External Oscillation Backup Timing



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

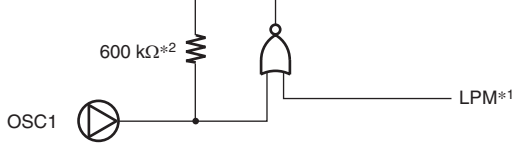
Figure 5.9 Example of Trimming Flow for On-Chip Oscillator Clock

Figure 5.10 Timing Chart of Trimming of On-Chip Oscillator Frequency

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

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

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



Notes: 1. LPM: Power-down mode (standby mode, subactive mode, or subsleep mode)
2. Values here are reference values.

Figure 5.11 Block Diagram of External Clock Oscillator

5.5.1 Connecting Crystal Resonator

Figure 5.12 shows an example of connecting a crystal resonator. An AT-cut parallel-reso
crystal resonator should be used. Figure 5.13 shows the equivalent circuit of a crystal reso
resonator having the characteristics given in table 5.1 should be used.

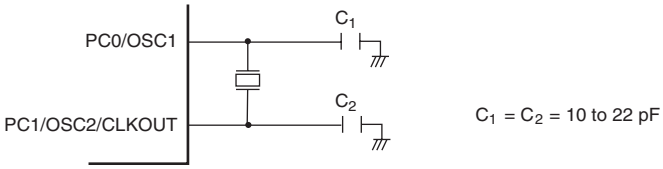


Figure 5.12 Example of Connection to Crystal Resonator

C_o (Max.)	70 pF

5.5.2 Connecting Ceramic Resonator

Figure 5.14 shows an example of connecting a ceramic resonator.

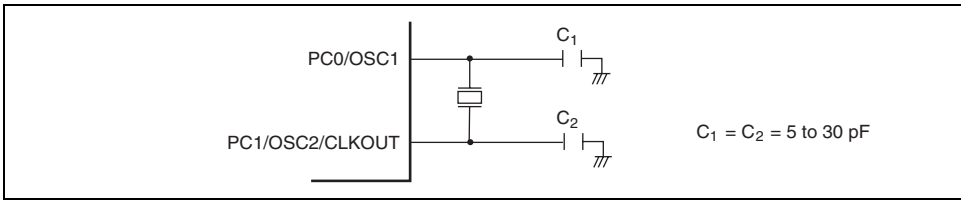


Figure 5.14 Example of Connection to Ceramic Resonator

5.5.3 Inputting External Clock

To use the external clock, input the external clock on pin OSC1. Figure 5.15 shows an e
connection. The duty cycle of the external clock signal must range from 45 to 55%.

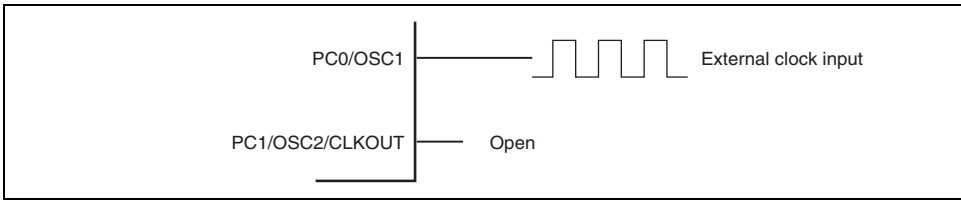


Figure 5.15 Example of External Clock Input

Figure 5.16 Block Diagram of Subclock Oscillator

5.6.1 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 5.17. Figure 5.18 shows the equivalent circuit of the 32.768 crystal resonator.

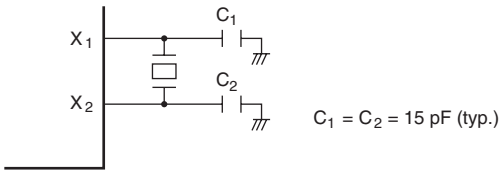
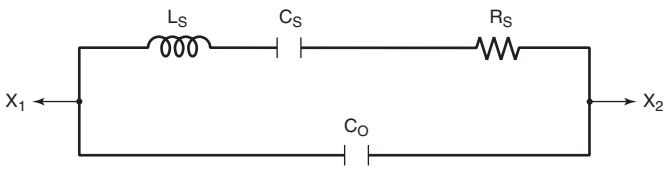


Figure 5.17 Typical Connection to 32.768-kHz Crystal Resonator



C_O = 1.5 pF (typ.)
R_S = 14 kΩ (typ.)
f_W = 32.768 kHz

Note: Values here are reference values.

Figure 5.18 Equivalent Circuit of 32.768-kHz Crystal Resonator

Figure 5.19 Pin Connection when not Using Subclock

5.7 Prescaler

5.7.1 Prescaler S

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

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

5.7.2 Prescaler W

Prescaler W is a 5-bit counter using a 32.768-kHz signal divided by 4 as its input clock. divided output is used for clock time base operation of the RTC. Prescaler W is initializ by a reset, and starts counting on exit from the reset state. Even in standby mode, subact or subsleep mode, prescaler W continues functioning.

5.8.2 Notes on Board Design

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

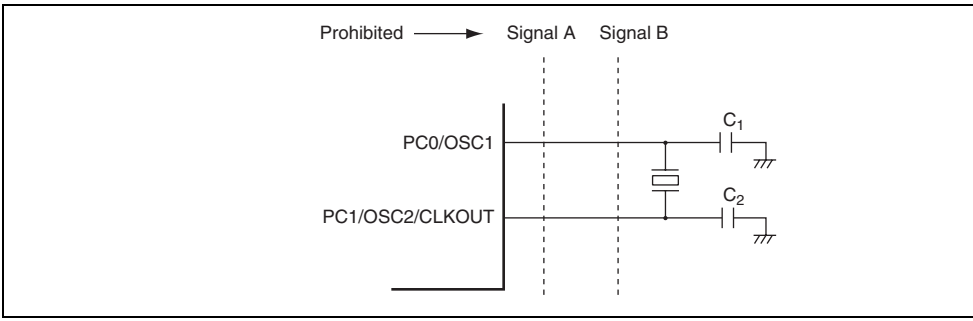


Figure 5.20 Example of Incorrect Board Design

The CPU and all on-chip peripheral modules are operable on the subclock. The subc
frequency can be selected from ϕw/2, ϕw/4, and ϕw/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 functio
  selected, the RTC is operable.
- Module standby mode

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

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

Bit	Bit Name	Initial Value	R/W	Description
7	SSBY	0	R/W	Software Standby This bit selects the mode to transit after the exec the SLEEP instruction. 0: Enters sleep mode or subsleep mode. 1: Enters standby mode. For details, see table 6.2.

				STS2 to STS0 values and the wait time. When using an external clock, set the wait time 100 μs or longer. These bits also set the wait states for external c stabilization when system clock is switched fron chip oscillator clock to the external clock by use software. The relationship between Nwait (number of wai oscillation stabilization) and Nstby (number of w for recovering to the standby mode) is as follow Nstby ≤ Nwait ≤ 2 × Nstby
3	NESEL	0	R/W	Noise Elimination Sampling Frequency Select The subclock pulse generator generates the wa signal (ϕ_w) and the external clock pulse generat generates the oscillator clock (ϕ_{osc}). This bit sel sampling frequency of the oscillator clock when clock signal (ϕ_w) is sampled. When ϕ_{osc}= 4 to 20 clear NESEL to 0. 0: Sampling rate is ϕ_{osc}/16 1: Sampling rate is ϕ_{osc}/4
2 to 0	—	All 0	—	Reserved These bits are always read as 0.

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

Note: Time unit is ms.

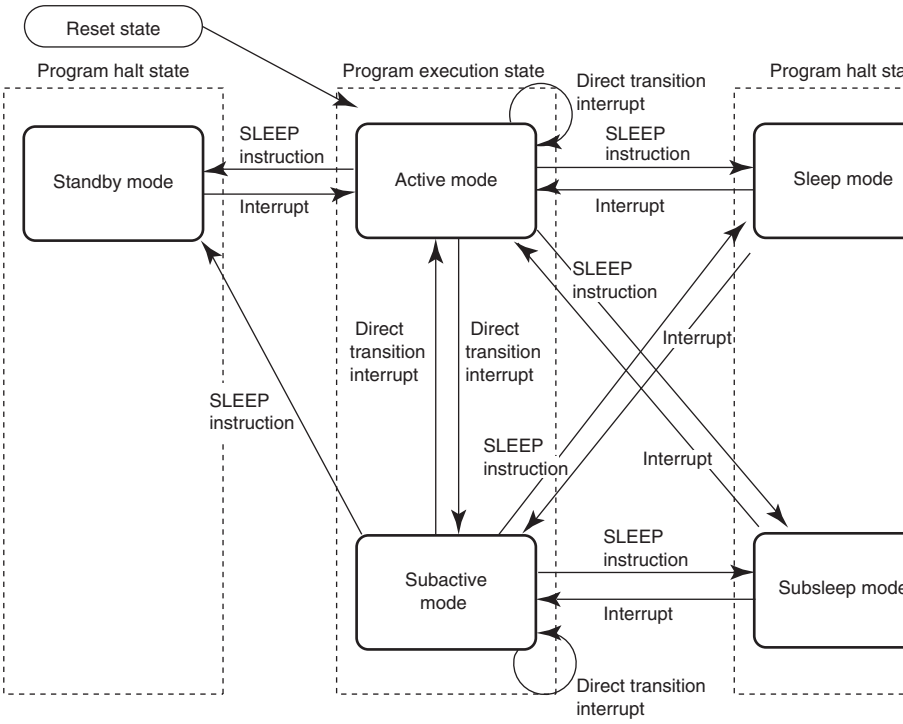
				a SLEEP instruction, as well as bit SSBY of SY For details, see table 6.2.
4	MA2	0	R/W	Active Mode Clock Select 2 to 0
3	MA1	0	R/W	These bits select the operating clock frequency
2	MA0	0	R/W	and sleep modes. The operating clock frequenc changes to the set frequency after the SLEEP i is executed. 0XX: ϕ_{osc} 100: $\phi_{osc}/8$ 101: $\phi_{osc}/16$ 110: $\phi_{osc}/32$ 111: $\phi_{osc}/64$
1	SA1	0	R/W	Subactive Mode Clock Select 1 and 0
0	SA0	0	R/W	These bits select the operating clock frequency subactive and subsleep modes. The operating frequency changes to the set frequency after th instruction is executed. 00: $\phi_w/8$ 01: $\phi_w/4$ 1X: $\phi_w/2$

[Legend] X: Don't care.

5	MSTS3	0	R/W	SCI3 Module Standby SCI3 enters standby mode when this bit is set to
4	MSTAD	0	R/W	A/D Converter Module Standby A/D converter enters standby mode when this bi to 1
3	MSTWD	0	R/W	Watchdog Timer Module Standby Watchdog timer enters standby mode when this to 1.When the internal oscillator is selected for th watchdog timer clock, the watchdog timer operat regardless of the setting of this bit
2	—	0	—	Reserved This bit is always read as 0.
1	MSTTV	0	R/W	Timer V Module Standby Timer V enters standby mode when this bit is se
0	MSTTA	0	R/W	RTC Module Standby RTC enters standby mode when this bit is set to

4	MSTTB1	0	R/W	Timer B1 Module Standby Timer B1 enters standby mode when this bit is
3, 2	—	All 0	—	Reserved 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 s
0	MSTPWM	0	R/W	PWM Module Standby PWM enters standby mode when this bit is set

by an interrupt. Table 6.3 shows the internal states of the LSI in each mode.



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

Figure 6.1 Mode Transition Diagram

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

X: Don't care.

Note: * When a state transition is performed while SMSEL is 1, timer V, SCI3, SCI3_ A/D converter are reset, and all registers are set to their initial values. To use functions after entering active mode, reset the registers.

						conte retain outpu high-i state.
External interrupts	IRQ3 to IRQ0	Functioning	Functioning	Functioning	Functioning	Funct
	WKP5 to WKP0	Functioning	Functioning	Functioning	Functioning	Funct
Peripheral functions	RTC	Functioning	Functioning	Functioning if the timekeeping time-ba function is selected, and retained if no		
	Timer V	Functioning	Functioning	Reset	Reset	Reset
	Watchdog timer	Functioning	Functioning	Retained (functioning if the internal os selected as a count clock*)		
	SCI3, SCI3_2	Functioning	Functioning	Reset	Reset	Reset
	IIC2	Functioning	Functioning	Retained*	Retained	Retai
	Timer B1	Functioning	Functioning	Retained*	Retained	Retai
	Timer Z	Functioning	Functioning	Retained (the counter increments acc subclocks if the internal clock (ϕ) is se a count clock*)		
	A/D converter	Functioning	Functioning	Reset	Reset	Reset
	LVD	Functioning	Functioning	Functioning	Functioning	Funct

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

6.2.2 Standby Mode

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

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

When the $\overline{\text{RES}}$ pin is driven low in standby mode, the on-chip oscillator starts functionin the oscillator starts, the system clock is supplied to the entire chip. The $\overline{\text{RES}}$ pin must be for the rated period set by the power-on reset circuit, until the oscillator stabilizes. If the is driven high after the oscillator has stabilized, the internal reset signal is cleared and th starts reset exception handling.

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

When the $\overline{\text{RES}}$ pin is driven low in standby mode, the on-chip oscillator starts functionin the oscillator starts, the system clock is supplied to the entire chip. The $\overline{\text{RES}}$ pin must be for the rated period set by the power-on reset circuit, until the oscillator stabilizes. If the is driven high after the oscillator has stabilized, the internal reset signal is cleared and the starts reset exception handling.

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 S SA0 bits in SYSCR2. After the SLEEP instruction is executed, the operating frequency c the frequency which is set before the execution.

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

When the $\overline{\text{RES}}$ pin is driven low in standby mode, the on-chip oscillator starts functionin the oscillator starts, the system clock is supplied to the entire chip. The $\overline{\text{RES}}$ pin must be for the rated period set by the power-on reset circuit, until the oscillator stabilizes. If the is driven high after the oscillator has stabilized, the internal reset signal is cleared and the starts reset exception handling.

transition also enables operating frequency modification in active or subactive mode. Af
mode transition, direct transition interrupt exception handling starts.

If the direct transition interrupt is disabled in interrupt enable register 1, a transition is m
instead to sleep or subsleep mode. Note that if a direct transition is attempted while the I
CCR is set to 1, sleep or subsleep mode will be entered, and the resulting mode cannot b
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
(the direct transition time) is calculated by equation (1).

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

Example
Direct transition time = (2 + 1) × tosc + 14 × 8tw = 3tosc + 112tw
(when the CPU operating clock of $\phi_{osc} \rightarrow \phi_w/8$ 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

(when the CPU operating clock of $\phi_w/8 \rightarrow \phi_{osc}$ and a waiting time of 8192 states are se

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 module standby mo
clock supply to modules stops to enter the power-down mode. Module standby mode ena
on-chip peripheral module to enter the standby state by setting a bit that corresponds to e
module to 1 and cancels the mode by clearing the bit to 0.

— On-board programming/erasing can be done in boot mode, in which the boot pro into the chip is started to erase or program of the entire flash memory. In normal program mode, individual blocks can be erased or programmed.

- 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 flash memory can be read with low power consumption.

7.1 Block Configuration

Figure 7.1 shows the block configuration of flash memory of each product model. The t indicate erasing units, the narrow lines indicate programming units, and the values are a Erasing is performed in erasing units shown in these figures. Programming is performed byte units starting from an address with lower eight bits H'00 or H'80.

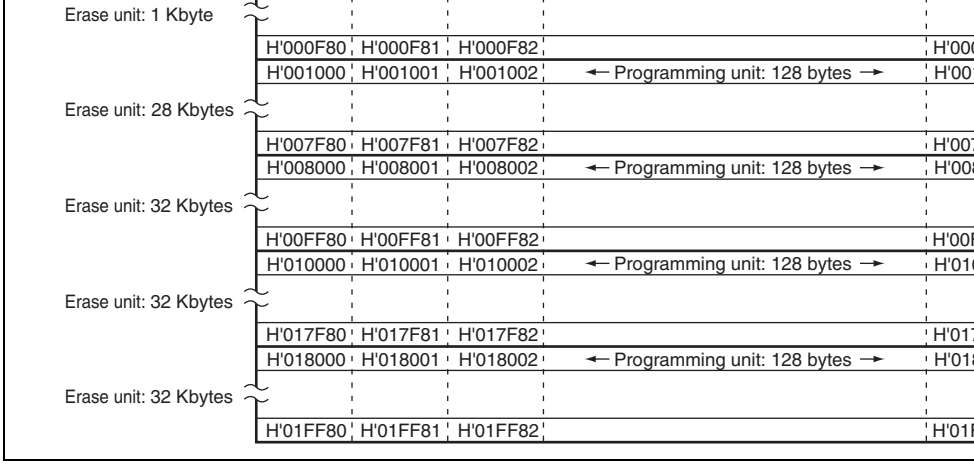


Figure 7.1 Flash Memory Block Configuration (1)
(H8/36079GF and H8/36079LF)

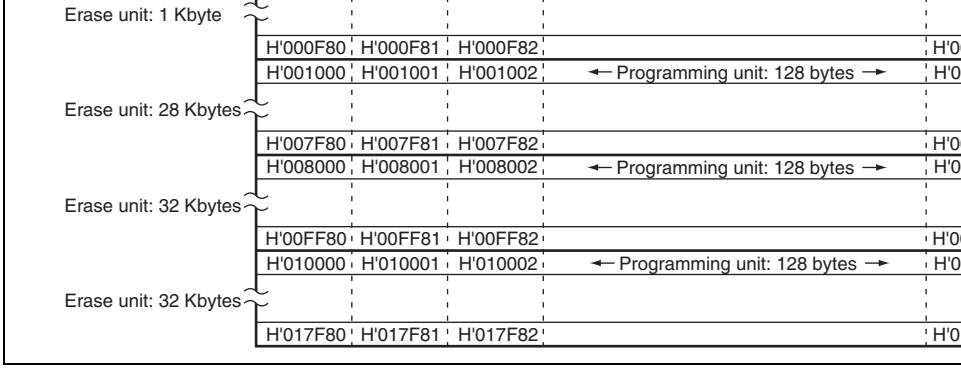


Figure 7.1 Flash Memory Block Configuration (2)
(H8/36078GF and H8/36078LF)

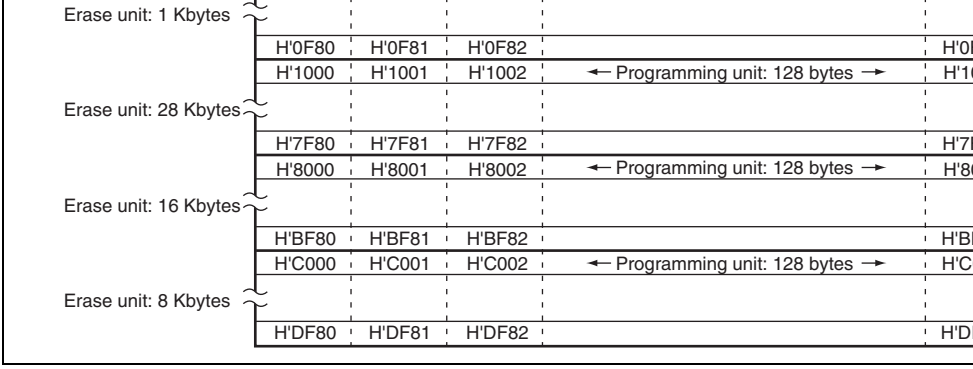


Figure 7.1 Flash Memory Block Configuration (3)
(H8/36077GF and H8/36077LF)

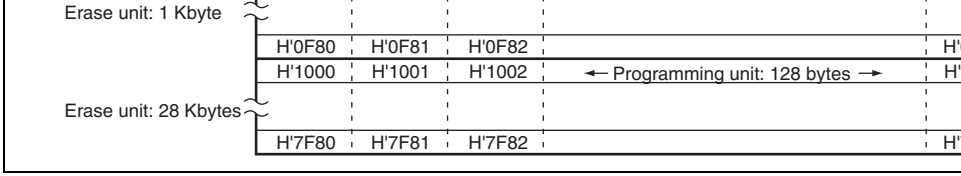


Figure 7.1 Flash Memory Block Configuration (4)
(H8/36074GF and H8/36074LF)

7.2.1 Flash Memory Control Register 1 (FLMCR1)

FLMCR1 is a register that makes the flash memory change to program mode, program-ve 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 to 0, other FLMCR1 register bits and all EBR1 bi be set.
5	ESU	0	R/W	Erase Setup When this bit is set to 1, the flash memory chang erase setup state. When it is cleared to 0, the era state is cancelled. Set this bit to 1 before setting to 1 in FLMCR1.
4	PSU	0	R/W	Program Setup When this bit is set to 1, the flash memory chang program setup state. When it is cleared to 0, the setup state is cancelled. Set this bit to 1 before s P bit in FLMCR1.

				When this bit is set to 1 while SWE=1 and ESU flash memory changes to erase mode. When it to 0, erase mode is cancelled.
0	P	0	R/W	Program When this bit is set to 1 while SWE=1 and PSU flash memory changes to program mode. Wher cleared to 0, program mode is cancelled.

7.2.2 Flash Memory Control Register 2 (FLMCR2)

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

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

6	EB6	0	R/W	When this bit is set to 1, 32 Kbytes of H'010000 to H'017FFF will b
5	EB5	0	R/W	When this bit is set to 1, H'008000 to H'00FFFF 32 Kbytes of H'01 H'017FFF will be erased.
4	EB4	0	R/W	When this bit is set to 1, 28 Kbytes of H'001000 to H'007FFF will b
3	EB3	0	R/W	When this bit is set to 1, 1 Kbyte of H'000C00 to H'000FFF will be
2	EB2	0	R/W	When this bit is set to 1, 1 Kbyte of H'000800 to H'000BFF will be
1	EB1	0	R/W	When this bit is set to 1, 1 Kbyte of H'000400 to H'0007FF will be
0	EB0	0	R/W	When this bit is set to 1, 1 Kbyte of H'000000 to H'0003FF will be

(2) H8/36078GF and H8/36078LF

Bit	Bit Name	Initial Value	R/W	Description
7	—	0	—	Reserved. Although this bit is readable/writable, do not set this bit
6	EB6	0	R/W	When this bit is set to 1, 32 Kbytes of H'010000 to H'017FFF will b
5	EB5	0	R/W	When this bit is set to 1, 32 Kbytes of H'008000 to H'00FFFF will b
4	EB4	0	R/W	When this bit is set to 1, 28 Kbytes of H'001000 to H'007FFF will b
3	EB3	0	R/W	When this bit is set to 1, 1 Kbyte of H'000C00 to H'000FFF will be
2	EB2	0	R/W	When this bit is set to 1, 1 Kbyte of H'000800 to H'000BFF will be e
1	EB1	0	R/W	When this bit is set to 1, 1 Kbyte of H'000400 to H'0007FF will be e
0	EB0	0	R/W	When this bit is set to 1, 1 Kbyte of H'000000 to H'0003FF will be e

1	EB1	0	R/W	When this bit is set to 1, 1 Kbyte of H'0400 to H'07FF will be eras
0	EB0	0	R/W	When this bit is set to 1, 1 Kbyte of H'0000 to H'03FF will be eras

(4) H8/36074GF and H8/36074LF

Bit	Bit Name	Initial Value	R/W	Description
7	—	0	—	Reserved. This bit is always read as 0.
6	—	0	—	Reserved. Although this bit is readable/writable, do not set this bi
5	—	0	—	Reserved. Although this bit is readable/writable, do not set this bi
4	EB4	0	R/W	When this bit is set to 1, 28 Kbytes of H'1000 to H'7FFF will be e
3	EB3	0	R/W	When this bit is set to 1, 1 Kbyte of H'0C00 to H'0FFF will be eras
2	EB2	0	R/W	When this bit is set to 1, 1 Kbyte of H'0800 to H'0BFF will be eras
1	EB1	0	R/W	When this bit is set to 1, 1 Kbyte of H'0400 to H'07FF will be eras
0	EB0	0	R/W	When this bit is set to 1, 1 Kbyte of H'0000 to H'03FF will be eras

				When this bit is 0 and a transition is made to sub mode, the flash memory enters the power-down When this bit is 1, the flash memory remains in t normal mode even after a transition is made to s 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 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 this bit is set to 1. Flash memory control register be accessed when this bit is set to 0.
6 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 ex
This can be used for programming initial values in the on-board state or for a forcible re
programming/erasing can no longer be done in user program mode. In user program mo
individual blocks can be erased and programmed by branching to the user program/eras
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.

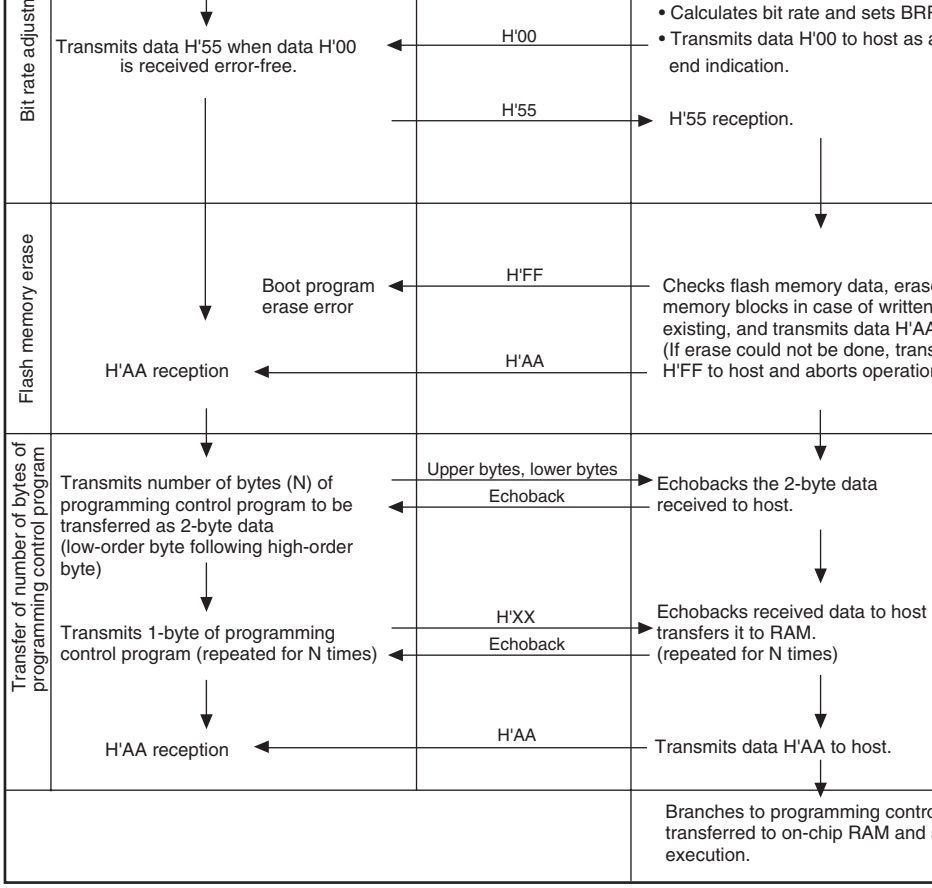
7.3.1 Boot Mode

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

1. When boot mode is used, the flash memory programming control program must be p
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 d
bit, and no parity.
3. When the boot program is initiated, the chip measures the low-level period of asynch
SCI communication data (H'00) transmitted continuously from the host. The chip the
calculates the bit rate of transmission from the host, and adjusts the SCI3 bit rate to m
of the host. The reset should end with the RxD pin high. The RxD and TxD pins sho

to H'FFEF for microcontrollers supporting advanced mode of H'FF700 to H'FFEF f microcontrollers supporting normal mode, is the area to which the programming contr program is transferred from the host. The boot program area cannot be used until the c state in boot mode switches to the programming control program.

6. Before branching to the programming control program, the chip terminates transfer op by SCI3 (by clearing the RE and TE bits in SCR to 0), however the adjusted bit rate v remains set in BRR. Therefore, the programming control program can still use it for tr program data or verify data with the host. The TxD pin is high (PCR22 = 1, P22 = 1). contents of the CPU general registers are undefined immediately after branching to th programming control program. These registers must be initialized at the beginning of programming control program, as the stack pointer (SP), in particular, is used implicit subroutine calls, etc.
7. Boot mode can be cleared by a reset. End the reset after driving the reset pin low, wai least 20 states, and then setting the $\overline{\text{NMI}}$ pin. Boot mode is also cleared when a WDT occurs.
8. Do not change the TEST pin and $\overline{\text{NMI}}$ pin input levels in boot mode.


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

program mode by branching to a user program/erase control program. The user must set b
conditions and provide on-board means of supplying programming data. The flash memo
contain the user program/erase control program or a program that provides the user progr
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,
mode. Figure 7.2 shows a sample procedure for programming/erasing in user program mo
Prepare a user program/erase control program in accordance with the description in sectio
Flash Memory Programming/Erasing.

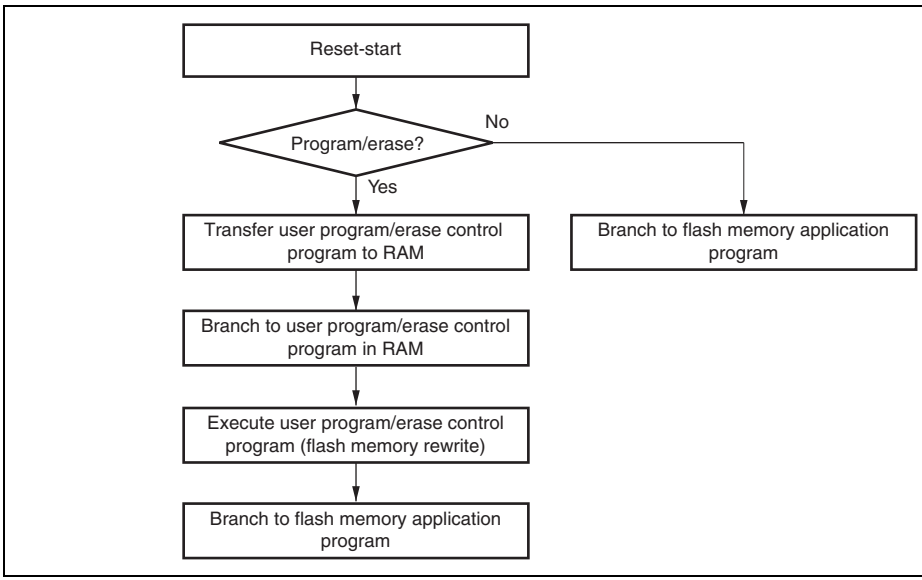
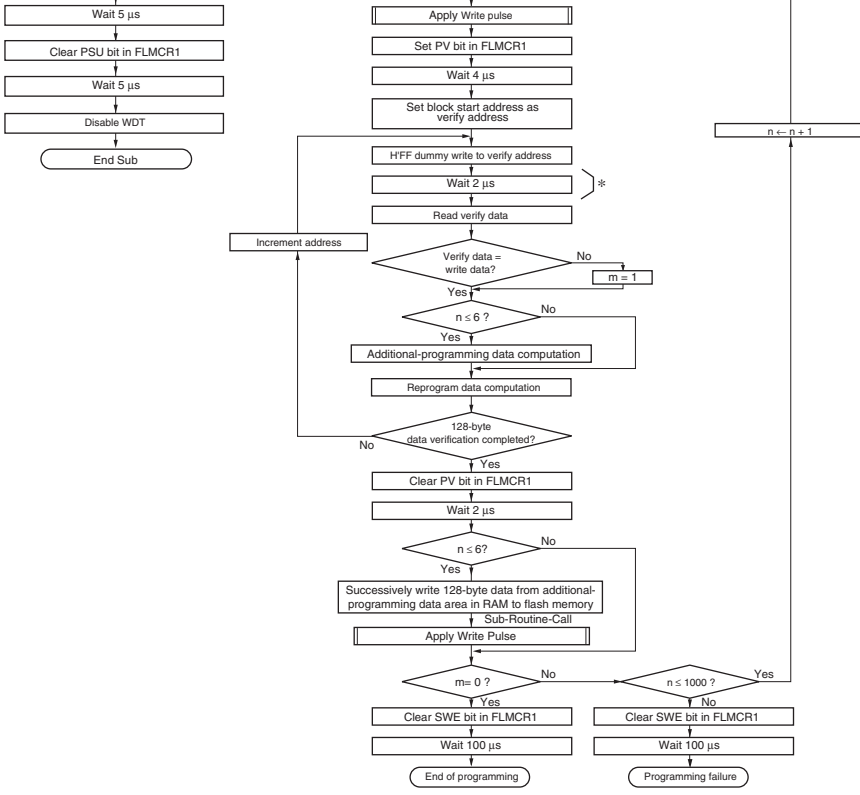


Figure 7.2 Programming/Erasing Flowchart Example in User Program Mo

7.4.1 Program/Program-Verify

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

1. Programming must be done to an empty address. Do not reprogram an address to wh programming has already been performed.
2. Programming should be carried out 128 bytes at a time. A 128-byte data transfer mu performed even if writing fewer than 128 bytes. In this case, H'FF data must be writt extra addresses.
3. Prepare the following data storage areas in RAM: A 128-byte programming data are byte reprogramming data area, and a 128-byte additional-programming data area. Pe reprogramming data computation according to table 7.4, and additional programmin computation according to table 7.5.
4. Consecutively transfer 128 bytes of data in byte units from the reprogramming data a additional-programming data area to the flash memory. The program address and 12 data are latched in the flash memory. The lower 8 bits of the start address in the flash 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 allowable programming times.
6. The watchdog timer (WDT) is set to prevent overprogramming due to program runa 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 l are B'00. Verify data can be read in words or in longwords from the address to which 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 H'FF 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
0	1	1	No additional progr
1	0	1	No additional progr
1	1	1	No additional progr

Table 7.6 Programming Time

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

Note: Time shown in μs.

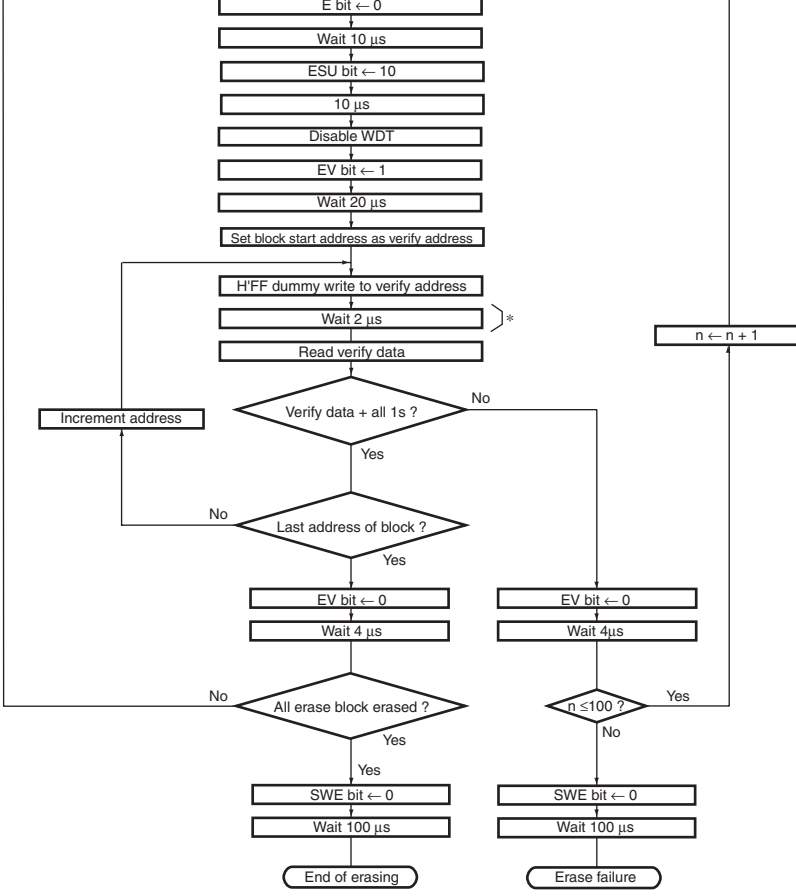
overflow cycle of approximately 19.8 ms is allowed.

5. For a dummy write to a verify address, write 1-byte data H'FF to an address whose lo bits are B'00. Verify data can be read in longwords from the address to which a dumm was performed.
6. If the read data is not erased successfully, set erase mode again, and repeat the erase/e verify sequence as before. The maximum number of repetitions of the erase/erase-ver 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 prog 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 e 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 can carried out.



Note: *The RTS instruction must not be used during a period between dummy writing of H'FF to a verify address and verify data rea

Figure 7.4 Erase/Erase-Verify Flowchart

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

7.5.2 Software Protection

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

7.5.3 Error Protection

In error protection, an error is detected when CPU runaway occurs during flash memory
programming/erasing, or operation is not performed in accordance with the program/eras
algorithm, and the program/erase operation is forcibly aborted. Aborting the program/era
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
bit in FLMCR2 is set to 1, and the error protection state is entered.

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

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 me
  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 fl
memory. In subactive mode, the flash memory can be set to operate in power-down mo
PDWND bit in FLPWCR. When the flash memory returns to its normal operating state
power-down mode or standby mode, a period to stabilize operation of the power supply
that were stopped is needed. When the flash memory returns to its normal operating stat
STS2 to STS0 in SYSCR1 must be set to provide a wait time of at least 20 µs, even whe
external clock is being used.

	3.3-V specification	H8/36078LF		
H8/36077 Group	5.0-V specification	H8/36077GF	4 Kbytes	H'E800 to H' H'F780 to H'
	3.3-V specification	H8/36077LF		
	5.0-V specification	H8/36074GF		
	3.3-V specification	H8/36074LF		

Notes: 1. When the E7 or E8 is used, area H'FFF780 to H'FFFB7F must not be access
2. When the E7 or E8 is used, area H'F780 to H'FB7F must not be accessed.

For functions in each port, see appendix B.1, I/O Port Block Diagrams. For the executio
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 p
bit PWM output pin, a timer B1 input pin, and a timer V input pin. Figure 9.1 shows its
configuration.

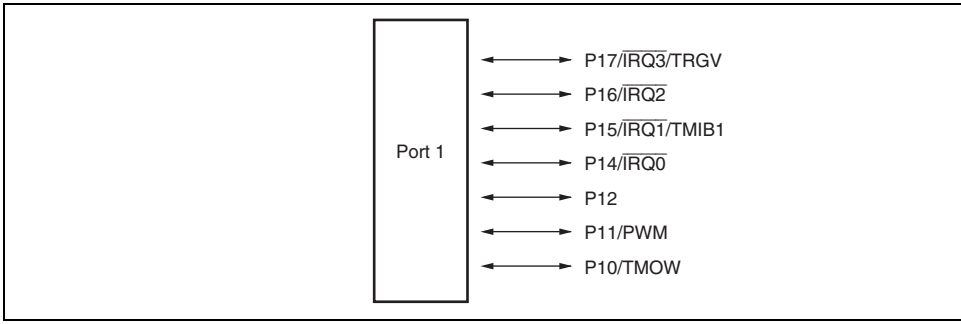


Figure 9.1 Port 1 Pin Configuration

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	This bit selects the function of pin P15/$\overline{\text{IRQ1}}$/TMI 0: General I/O port 1: $\overline{\text{IRQ1}}$/TMIB1 input pin
4	IRQ0	0	R/W	This bit 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	This bit selects the function of pin P72/TXD_2. 0: General I/O port 1: TXD_2 output pin
2	PWM	0	R/W	This bit selects the function of pin P11/PWM. 0: General I/O port 1: PWM output pin
1	TXD	0	R/W	This bit selects the function of pin P22/TXD. 0: General I/O port 1: TXD output pin
0	TMOW	0	R/W	This bit 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, th
5	P15	0	R/W	stored in PDR1 are read. If PDR1 is read while
4	P14	0	R/W	are cleared to 0, the pin states are read regardl value stored in PDR1.
3	—	1	—	Bit 3 is a reserved bit. This bit is always read as
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 be

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.

Register	PMR1	PCR1	
Bit Name	**IRQ1**	**PCR15**	**Pin Function**
Setting value	0	0	P15 input pin
		1	P15 output pin
	1	x	$\overline{\text{IRQ1}}$ input/TMIB1 input pin

[Legend] x: Don't care.

P14/$\overline{\text{IRQ0}}$ pin

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

Register	PMR1	PCR1	
Bit Name	**TMOW**	**PCR10**	**Pin Function**
Setting value	0	0	P10 input pin
		1	P10 output pin
	1	x	TMOW output pin

[Legend] x: Don't care.

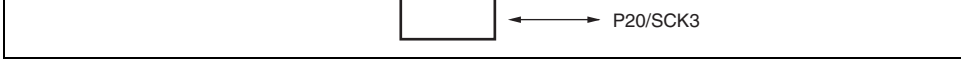


Figure 9.2 Port 2 Pin Configuration

Port 2 has the following registers.

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

9.2.1 Port Control Register 2 (PCR2)

PCR2 selects inputs/outputs in bit units for pins to be used as general I/O ports of port 2

Bit	Bit Name	Initial Value	R/W	Description
7 to 5	—	—	—	Reserved
4	PCR24	0	W	When each of the port 2 pins P24 to P20 functi
3	PCR23	0	W	general I/O port, setting a PCR2 bit to 1 makes corresponding pin an output port, while clearing
2	PCR22	0	W	0 makes the pin an input port.
1	PCR21	0	W	
0	PCR20	0	W	

2	P22	0	R/W	stored in PDR2 is read. If PDR2 is read while PC are cleared to 0, the pin states are read regardle value stored in PDR2.
1	P21	0	R/W	
0	P20	0	R/W	

9.2.3 Port Mode Register 3 (PMR3)

PMR3 selects the CMOS output or NMOS open-drain output for port 2.

Bit	Bit Name	Initial Value	R/W	Description
7 to 5	—	All 0	—	Reserved These bits are always read as 0.
4	POF24	0	R/W	When the bit is set to 1, the corresponding pin is by PMOS and it functions as the NMOS open-dr output. When cleared to 0, the pin functions as tl output.
3	POF23	0	R/W	
2 to 0	—	All 1	—	Reserved These bits are always read as 1.

P23 pin

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

P22/TXD pin

Register	PMR1	PCR2	
Bit Name	**TXD**	**PCR22**	**Pin Function**
Setting Value	0	0	P22 input pin
		1	P22 output pin
	1	x	TXD output pin

[Legend] x: Don't care.

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.

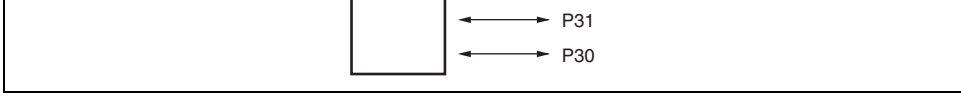


Figure 9.3 Port 3 Pin Configuration

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 correspondi
6	PCR36	0	W	output port, while clearing the bit to 0 makes th
5	PCR35	0	W	input port.
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 be

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

Bit Name	PCR34	Pin Function
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

Bit Name	PCR30	Pin Function
Setting Value	0	P30 input pin
	1	P30 output pin

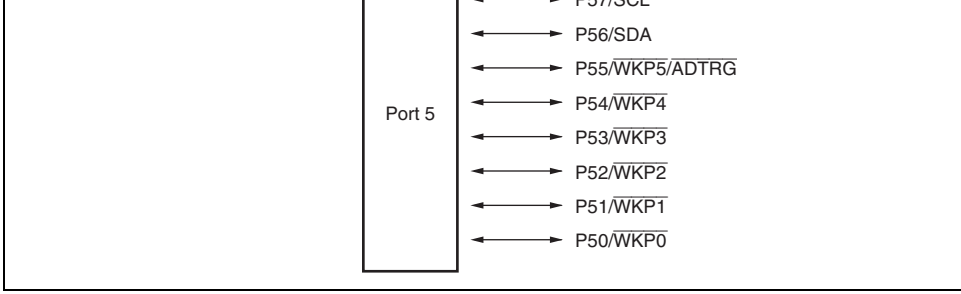


Figure 9.4 Port 5 Pin Configuration

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: $\overline{\text{WKP5}}$/$\overline{\text{ADTRG}}$ input pin
4	WKP4	0	R/W	This bit 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	This bit 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	This bit 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	This bit 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	This bit 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	Stores output data for port 5 pins.
6	P56	0	R/W	If PDR5 is read while PCR5 bits are set to 1, th
5	P55	0	R/W	stored in PDR5 are read. If PDR5 is read while
4	P54	0	R/W	are cleared to 0, the pin states are read regardl
3	P53	0	R/W	value stored in PDR5.
2	P52	0	R/W	
1	P51	0	R/W	
0	P50	0	R/W	

3	PUCR53	0	R/W	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 be

P57/SCL pin

Register	ICCR1	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.

Register	PMR5	PCR5	
Bit Name	**WKP2**	**PCR52**	**Pin Function**
Setting Value	0	0	P52 input pin
		1	P52 output pin
	1	x	$\overline{\text{WKP2}}$ input pin

[Legend] x: Don't care.

P51/$\overline{\text{WKP1}}$ pin

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.

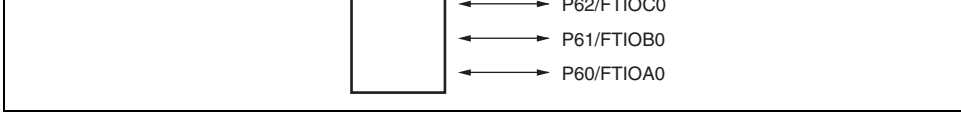


Figure 9.5 Port 6 Pin Configuration

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 functi
6	PCR66	0	W	general I/O port, setting a PCR6 bit to 1 makes
5	PCR65	0	W	corresponding pin an output port, while clearing
4	PCR64	0	W	0 makes the pin an input port.
3	PCR63	0	W	
2	PCR62	0	W	
1	PCR61	0	W	
0	PCR60	0	W	

3	P63	0	R/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 be

P67/FTIOD1 pin

Register	TOER	TFCR	TPMR	TIORC1	PCR6	
Bit Name	ED1	CMD1 and CMD0	PWMD1	IOD2 to IOD0	PCR67	Pin Function
Setting Value	1	00	0	000 or 1xx	0	P67 input/FTIOD1 inp
					1	P67 output pin
	0	00	0	001 or 01x	x	FTIOD1 output pin
			1	xxx		
		Other than 00	x	xxx		

[Legend] x: Don't care.

		Other than 00	x	xxx		

[Legend] x: Don't care.

P65/FTIOB1 pin

Register	TOER	TFCR	TPMR	TIORA1	PCR6	
Bit Name	EB1	CMD1 to CMD0	PWMB1	IOB2 to IOB0	PCR65	Pin Function
Setting Value	1	00	0	000 or 1xx	0	P65 input/FTIOB1 in
					1	P65 output pin
	0	00	0	001 or 01x	x	FTIOB1 output pin
			1	xxx		
		Other than 00	x	xxx		

[Legend] x: Don't care.

P63/FTIOD0 pin

Register	TOER	TFCR	TPMR	TIORC0	PCR6	
Bit Name	ED0	CMD1 to CMD0	PWMD0	IOD2 to IOD0	PCR63	Pin Function
Setting Value	1	00	0	000 or 1xx	0	P63 input/FTIOD0 inp
					1	P63 output pin
	0	00	0	001 or 01x	x	FTIOD0 output pin
			1	xxx		
		Other than 00	x	xxx		

[Legend] x: Don't care.

P62/FTIOC0 pin

Register	TOER	TFCR	TPMR	TIORC0	PCR6	
Bit Name	EC0	CMD1 to CMD0	PWMC0	IOC2 to IOC0	PCR62	Pin Function
Setting Value	1	00	0	000 or 1xx	0	P62 input/FTIOC0 inp
					1	P62 output pin
	0	00	0	001 or 01x	x	FTIOC0 output pin
			1	xxx		
		Other than 00	x	xxx		

[Legend] x: Don't care.

		Other than 00	x	xxx		

[Legend] x: Don't care.

P60/FTIOA0 pin

Register	TOER	TFCR	TFCR	TIORA0	PCR6	
Bit Name	EA0	CMD1 to CMD0	STCLK	IOA2 to IOA0	PCR60	Pin Function
Setting Value	1	xx	x	000 or 1xx	0	P60 input/FTIOA0 in
					1	P60 output pin
	0	00	0	001 or 01x	x	FTIOA0 output pin

[Legend] x: Don't care.

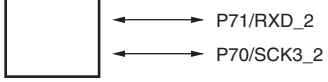


Figure 9.6 Port 7 Pin Configuration

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	—	—	—	When each of the port 7 pins P76 to P74 and P7 functions as a general I/O port, setting a PCR7 b makes the corresponding pin an output port, whi clearing the bit to 0 makes the pin an input port.
6	PCR76	0	W	
5	PCR75	0	W	
4	PCR74	0	W	Bits 7 and 3 are reserved bits.
3	—	—	—	
2	PCR72	0	W	
1	PCR71	0	W	
0	PCR70	0	W	

3	—	1	—	Bits 7 and 3 are reserved bits. These bits are al
2	P72	0	R/W	as 1.
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 b

P76/TMOV pin

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

[Legend] x: Don't care.

P75/TMCIV pin

Register	PCR7	
Bit Name	**PCR75**	**Pin Function**
Setting Value	0	P75 input/TMCIV input pin
	1	P75 output/TMCIV input pin

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.

P70/SCK3_2 pin

Register	SCR3_2		SMR2	PCR7	
Bit Name	CKE1	CKE0	COM	PCR70	Pin Function
Setting Value	0	0	0	0	P70 input pin
				1	P70 output pin
	0	0	1	x	SCK3_2 output pin
	0	1	x	x	SCK3_2 output pin
	1	x	x	x	SCK3_2 input pin

[Legend] x: Don't care.

Port 8 has the following registers.

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

9.7.1 Port Control Register 8 (PCR8)

PCR8 selects inputs/outputs in bit units for pins to be used as general I/O ports of port 8

Bit	Bit Name	Initial Value	R/W	Description
7	PCR87	0	W	When each of the port 8 pins P87 to P85 functi
6	PCR86	0	W	general I/O port, setting a PCR8 bit to 1 makes
5	PCR85	0	W	corresponding pin an output port, while clearing 0 makes the pin an input port.
4 to 0	—	—	—	Reserved

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, th
5	P85	0	R/W	stored in PDR8 is read. If PDR8 is read while P are cleared to 0, the pin states are read regardl value stored in PDR8.
4 to 0	—	All 1	—	Reserved These bits are always read as 1.

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

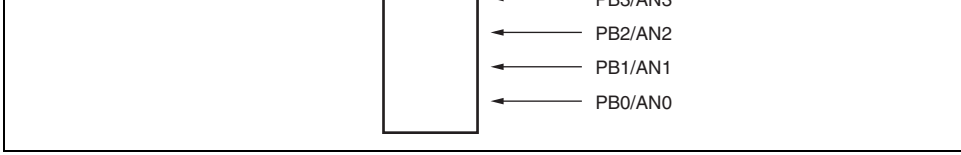


Figure 9.8 Port B Pin Configuration

Port B has the following register.

- Port data register B (PDRB)

9.8.1 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 t register. However, if a port B pin is designated as an ana channel by ADCSR in the A/D converter or as a comparison voltage input pin by LVDCR in the voltage detection circuit, the corresponding bit i as 0.
6	PB6	—	R	
5	PB5	—	R	
4	PB4	—	R	
3	PB3	—	R	
2	PB2	—	R	
1	PB1	—	R	
0	PB0	—	R	

	Other than above				PB0 o

[Legend] x: Don't care.

PB1/AN1 pin

Register			ADCSR		
Bit Name	**SCAN**	**CH2**	**CH1**	**CH0**	**Pin Fu**
Setting Value	0	0	0	1	AN1 in
	1		0	1	
			1	x	
	Other than above				PB1 o

[Legend] x: Don't care.

PB2/AN2 pin

Register			ADCSR		
Bit Name	**SCAN**	**CH2**	**CH1**	**CH0**	**Pin Fu**
Setting Value	0	0	1	0	AN2 in
	1		1	x	
	Other than above				PB2 o

[Legend] x: Don't care.

Register			ADCSR		
Bit Name	**SCAN**	**CH2**	**CH1**	**CH0**	**Pin F**
Setting Value	0	1	0	0	AN4
	1		x	x	
	Other than above				PB4

[Legend] x: Don't care.

PB5/AN5 pin

Register			ADCSR		
Bit Name	**SCAN**	**CH2**	**CH1**	**CH0**	**Pin F**
Setting Value	0	1	0	1	AN5
	1		0	1	
			1	x	
	Other than above				PB5

[Legend] x: Don't care.

					1	PB6 in

[Legend] x: Don't care.

PB7/AN7/ExtU pin

Register			ADCSR		LVDCR	
Bit Name	**SCAN**	**CH2**	**CH1**	**CH0**	**VDDII**	**Pin Fu**
Setting Value	0		1		0	AN7 in input p
	1					
	0				1	AN7 in
	1					
	Other than above				0	PB7 in input p
					1	PB7 in

[Legend] x: Don't care.

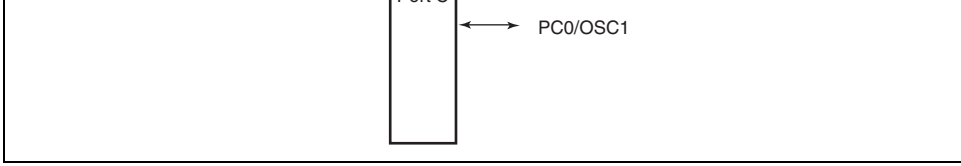


Figure 9.9 Port C Pin Configuration

Port C has the following registers.

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

9.9.1 Port Control Register C (PCRC)

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

Bit	Bit Name	Initial Value	R/W	Description
7 to 2	—	—	—	Reserved
1	PCRC1	0	W	When each of the port C pins, PC1 and PC0, fu
0	PCRC0	0	W	an general I/O port, setting a PCRC bit to 1 ma corresponding pin an output port, while clearing 0 makes the pin an input port.

value stored in PDRC.

9.9.3 Pin Functions

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

PC1/OSC2/CLKOUT pin

Register	CKCSR		PCRC	
Bit Name	**PMRC1**	**PMRC0**	**PCRC1**	**Pin Function**
Setting value	0	0	0	PC1 input pin
			1	PC1 output pin
		1	x	Open
	1	0	x	CLKOUT output pin
		1	x	OSC2 oscillation pin

[Legend] x: Don't care.

PC0/OSC1 pin

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

[Legend] x: Don't care.

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

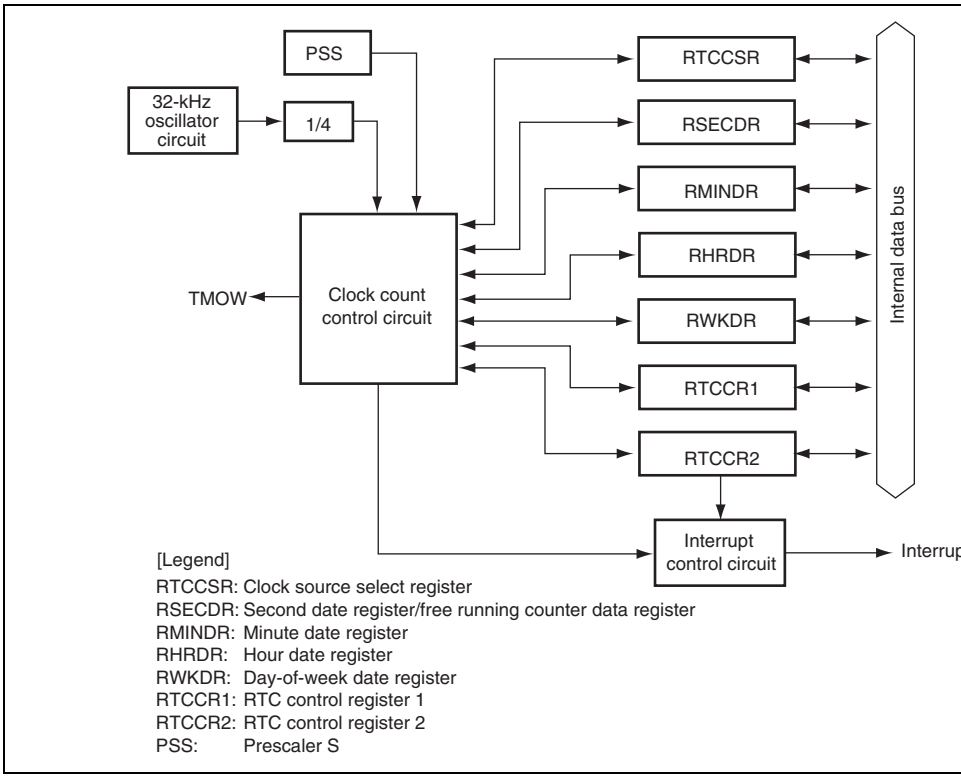


Figure 10.1 Block Diagram of RTC

- Clock source select register (RTCCSR)

10.3.1 Second Data Register/Free Running Counter Data Register (RSECDR)

RSECDR counts the BCD-coded second value. The setting range is decimal 00 to 59. It read register used as a counter, when it operates as a free running counter. For more info on reading seconds, minutes, hours, and day-of-week, see section 10.4.3, Data Reading

Bit	Bit Name	Initial Value	R/W	Description
7	BSY	—	R	RTC Busy This bit is set to 1 when the RTC is updating (the values of second, minute, hour, and day-c data registers. When this bit is 0, the values c 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 ca 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 car
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 o 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
1	HR01	—	R/W	generated, 1 is added to the ten's position.
0	HR00	—	R/W	

				minute, hour, and day-of-week data registers n adopted.
6 to 3	—	All 0	—	Reserved These bits are always read as 0.
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 co to 11.
				1: RTC operates in 24-hour mode. RHRDR co to 23.
5	PM	—	R/W	a.m./p.m.
				0: Indicates a.m. when RTC is in the 12-hour
				1: Indicates p.m. when RTC is in the 12-hour
4	RST	0	R/W	Reset
				0: Normal operation
				1: Resets registers and control circuits except and this bit. Clear this bit to 0 after having b 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 complet busy period.
2 to 0	—	All 0	—	Reserved
				These bits are always read as 0.

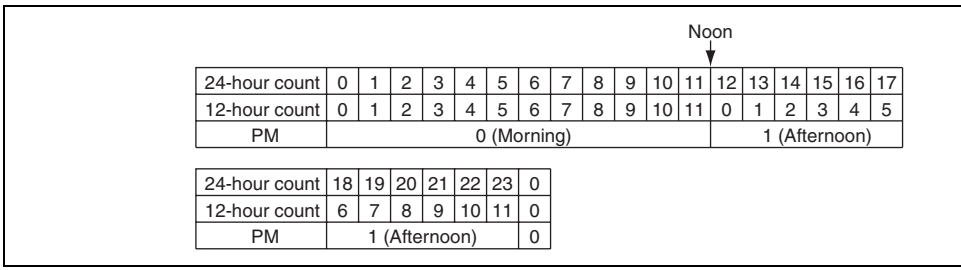


Figure 10.2 Definition of Time Expression

5	FOIE	—	R/W	Free Running Counter Overflow Interrupt Enab 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 wh TMOW in PMR1 to 1. 00: ϕ/4 01: ϕ/8 10: ϕ/16 11: ϕ/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: ϕ/8····· Free running counter ope
1	RCS1	0	R/W	0001: ϕ/32····· Free running counter ope
0	RCS0	0	R/W	0010: ϕ/128····· Free running counter ope 0011: ϕ/256····· Free running counter ope 0100: ϕ/512····· Free running counter ope 0101: ϕ/2048····· Free running counter ope 0110: ϕ/4096····· Free running counter ope 0111: ϕ/8192····· Free running counter ope 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.

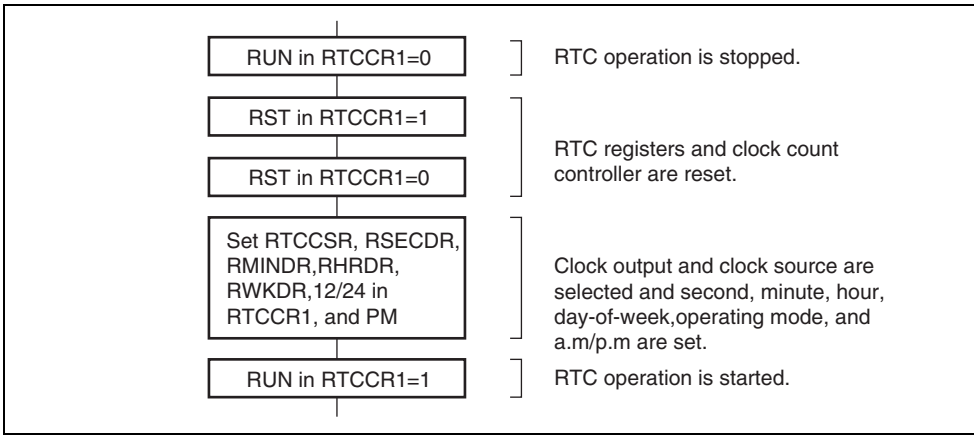


Figure 10.3 Initial Setting Procedure

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 regis
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
no change in the read data, the read data is used.

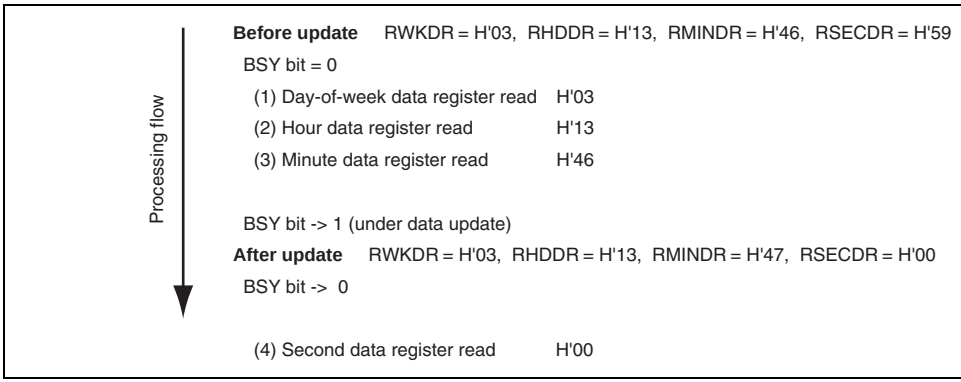


Figure 10.4 Example: Reading of Inaccurate Time Data

Table 10.2 Interrupt Sources

Interrupt Name	Interrupt Source	Interrupt Enal
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

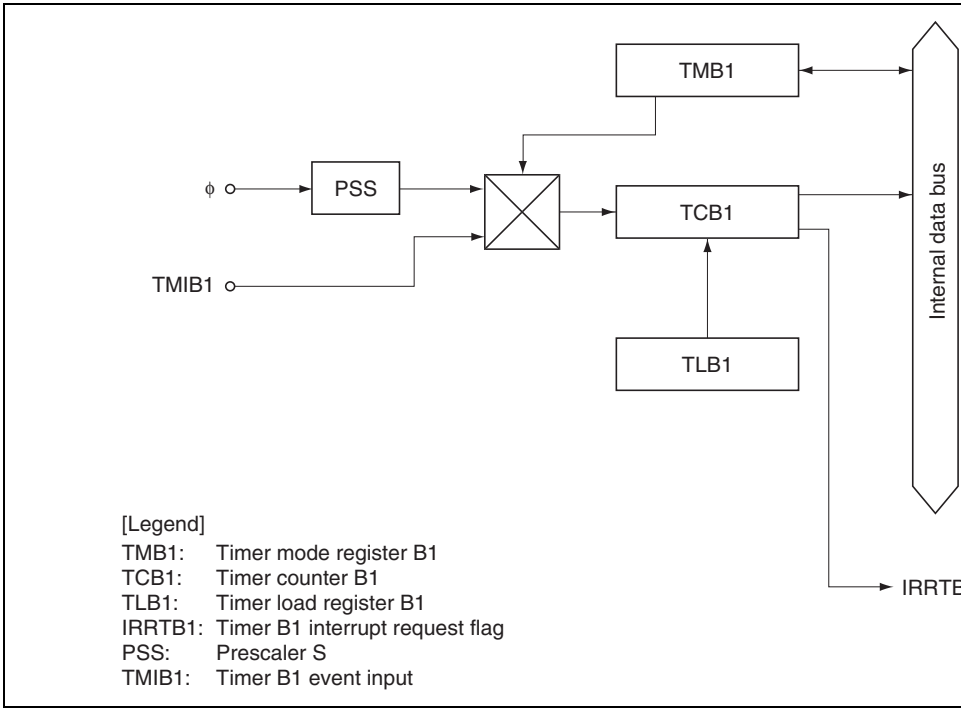


Figure 11.1 Block Diagram of Timer B1

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 to 3	—	All 1	—	Reserved These bits are always read as 1.
2	TMB12	0	R/W	Clock select
1	TMB11	0	R/W	000: Internal clock: ϕ/8192
0	TMB10	0	R/W	001: Internal clock: ϕ/2048 010: Internal clock: ϕ/512 011: Internal clock: ϕ/256 100: Internal clock: ϕ/64 101: Internal clock: ϕ/16 110: Internal clock: ϕ/4 111: External event (TMIB1): rising or falling ed Note: * The edge of the external event signal by bit IEG1 in the interrupt edge selec (IEGR1). See section 3.2.1, Interrupt E Select Register 1 (IEGR1), for details. setting TMB12 to TMB10 to 1, IRQ1 in mode register 1 (PMR1) should be set

TLB1 is an 8-bit write-only register for setting the reload value of TCB1. When a reload
set in TLB1, the same value is loaded into TCB1 as well, and TCB1 starts counting up fr
value. When TCB1 overflows during operation in auto-reload mode, the TLB1 value is l
into TCB1. Accordingly, overflow periods can be set within the range of 1 to 256 input c
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. U
reset, TCB1 is cleared to H'00 and bit TMB17 is cleared to 0, so up-counting and interval
resume immediately. The operating clock of timer B1 is selected from seven internal clo
output by prescaler S, or an external clock input at pin TMB1. The selection is made by b
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 req
the CPU.

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

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. E
event counting is selected by setting bits TMB12 to TMB10 in TMB1 to 1. TCB1 count
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
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	S
TCB1	Interval	Reset	Functions	Functions	Halted	Halted	H
	Auto-reload	Reset	Functions	Functions	Halted	Halted	H
TMB1		Reset	Functions	Retained	Retained	Retained	R

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

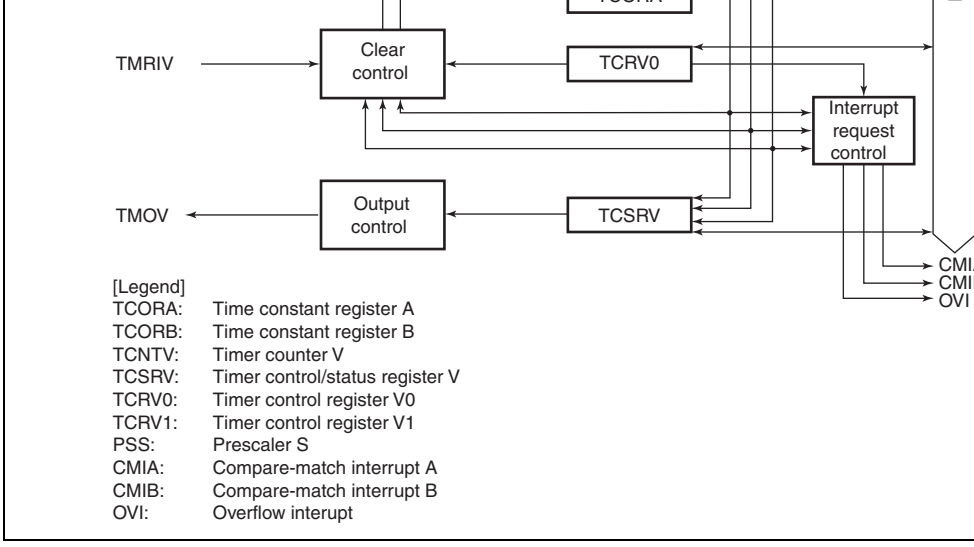


Figure 12.1 Block Diagram of Timer V

Trigger input	TRGV	Input	Trigger input to initiate co

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

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

TCNTV is initialized to H'00.

and the settings of bits OS3 to OS0 in TCSRV.

TCORA and TCORB are initialized to H'FF.

12.3.3 Timer Control Register V0 (TCRV0)

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

Bit	Bit Name	Initial Value	R/W	Description
7	CMIEB	0	R/W	Compare Match Interrupt Enable B When this bit is set to 1, interrupt request from th bit in TCSRV is enabled.
6	CMIEA	0	R/W	Compare Match Interrupt Enable A When this bit is set to 1, interrupt request from th bit in TCSRV is enabled.
5	OVIE	0	R/W	Timer Overflow Interrupt Enable When this bit is set to 1, interrupt request from th bit in TCSRV is enabled.
4	CCLR1	0	R/W	Counter Clear 1 and 0
3	CCLR0	0	R/W	These bits specify the clearing conditions of TCN 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. operation of TCNTV after clearing depends o in TCRV1.

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

12.3.4 Timer Control/Status Register V (TCSRV)

TCSRV 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 v Clearing condition: After reading CMFB = 1, cleared by writing 0 to

				Clearing condition: After reading OVF = 1, cleared by writing 0 to OV
4	—	1	—	Reserved This bit is always read as 1.
3	OS3	0	R/W	Output Select 3 and 2
2	OS2	0	R/W	These bits select an output method for the TMO the compare match of TCORB and TCNTV. 00: No change 01: 0 output 10: 1 output 11: Output toggles
1	OS1	0	R/W	Output Select 1 and 0
0	OS0	0	R/W	These bits select an output method for the TMO 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 outpu
for compare match A. The two output levels can be controlled independently. After a rese
timer output is 0 until the first compare match.

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

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

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

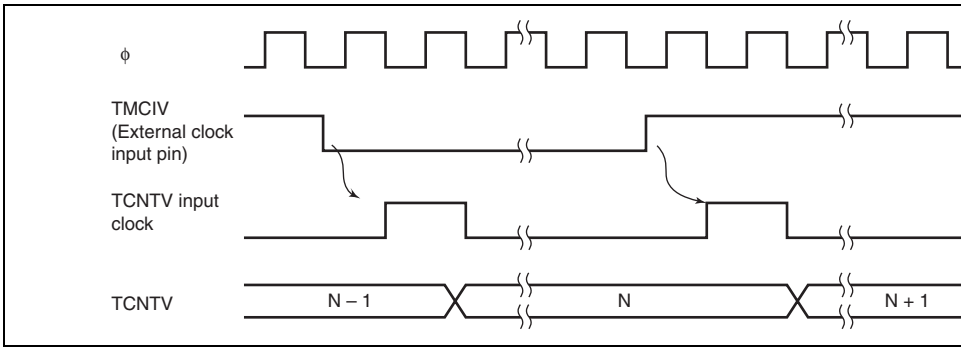


Figure 12.3 Increment Timing with External Clock

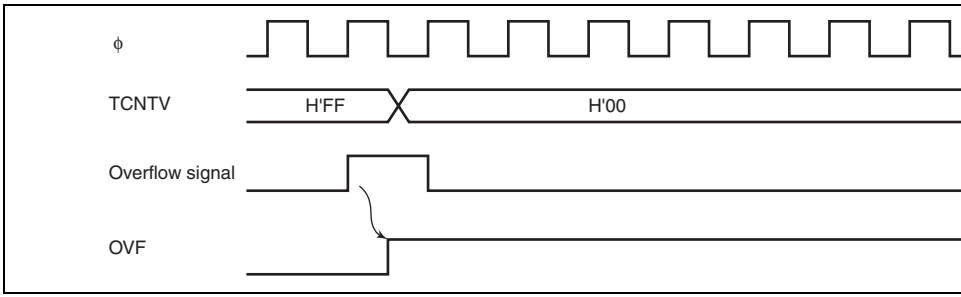


Figure 12.4 OVF Set Timing

Figure 12.5 CMFA and CMFB Set Timing

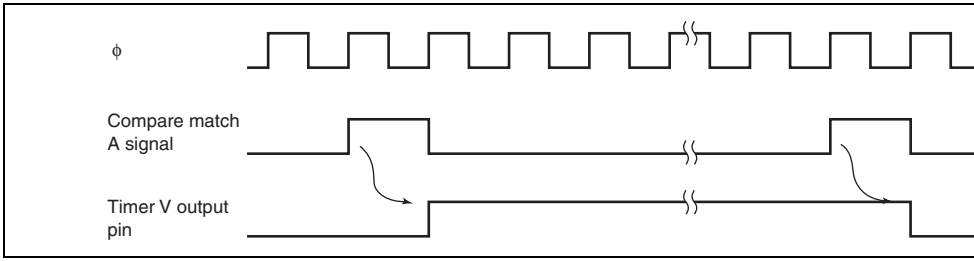


Figure 12.6 TMOV Output Timing

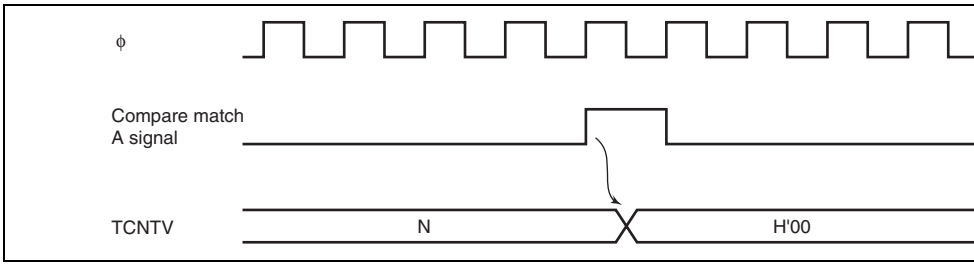


Figure 12.7 Clear Timing by Compare Match

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

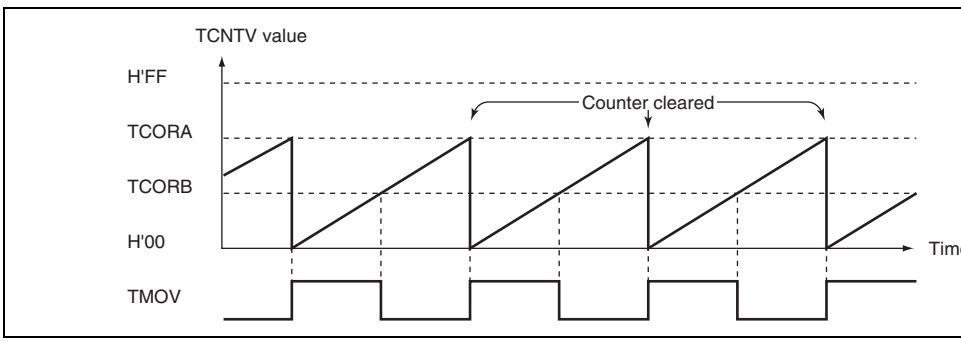


Figure 12.9 Pulse Output Example

input.

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

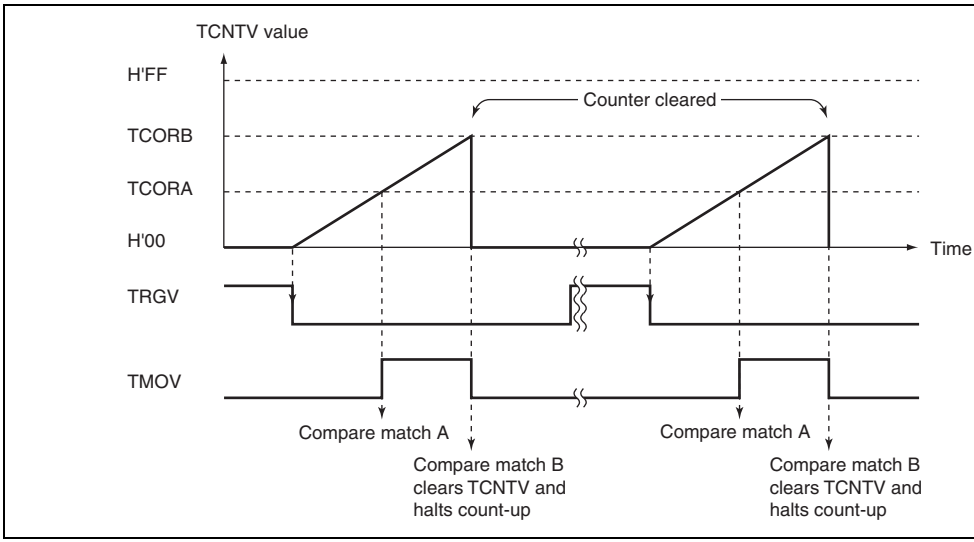


Figure 12.10 Example of Pulse Output Synchronized to TRGV Input

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

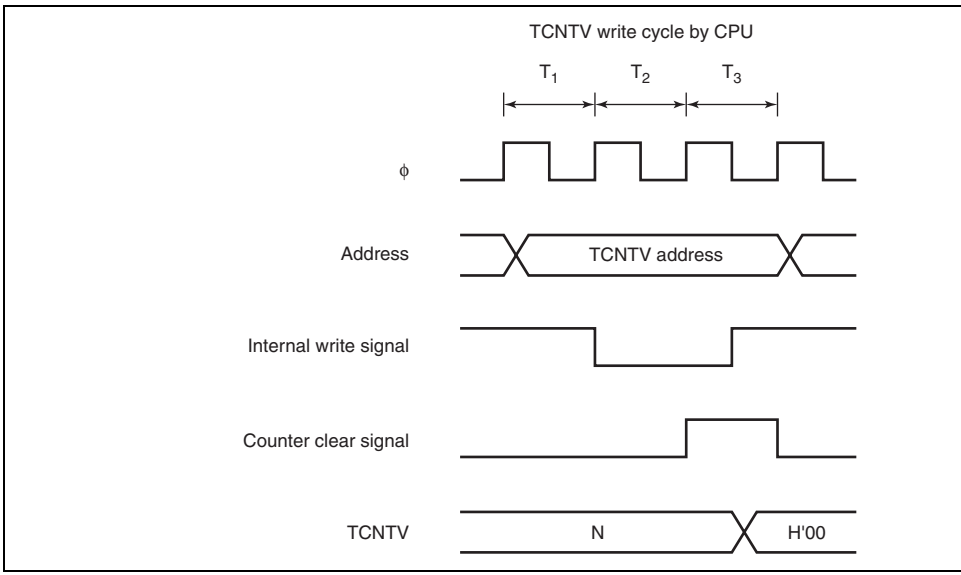


Figure 12.11 Contention between TCNTV Write and Clear

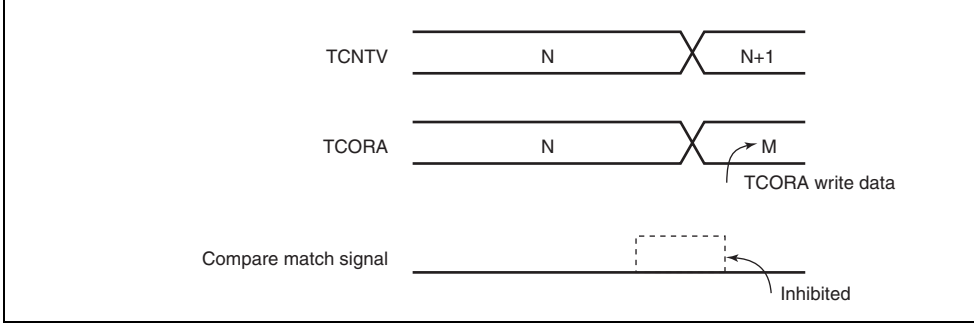


Figure 12.12 Contention between TCORA Write and Compare Match

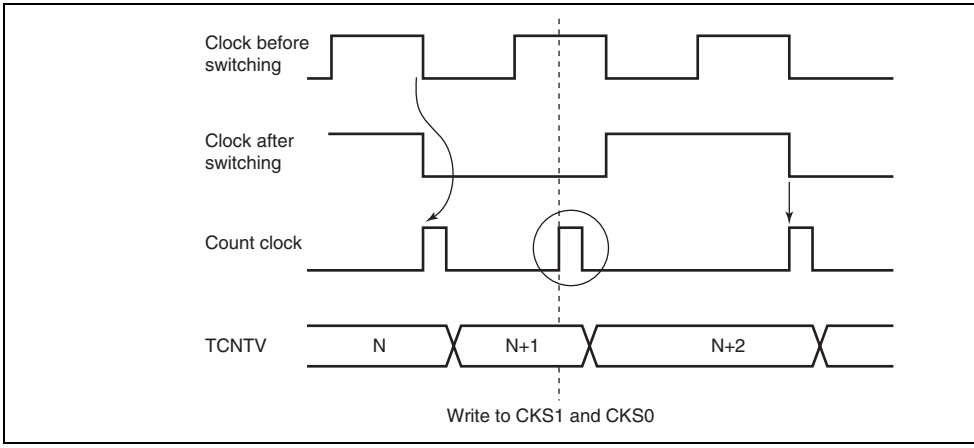


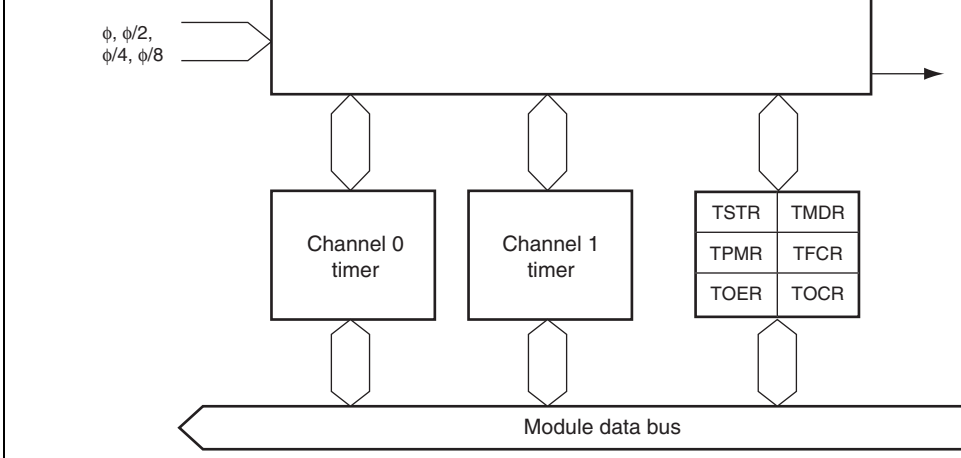
Figure 12.13 Internal Clock Switching and TCNTV Operation

— Independently assignable output compare or input capture functions

- Selection of five counter clock sources: four internal clocks (ϕ, ϕ/2, ϕ/4, and ϕ/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 inte
- Any initial timer output value can be set
- Output of the timer is disabled by external trigger

capture registers)

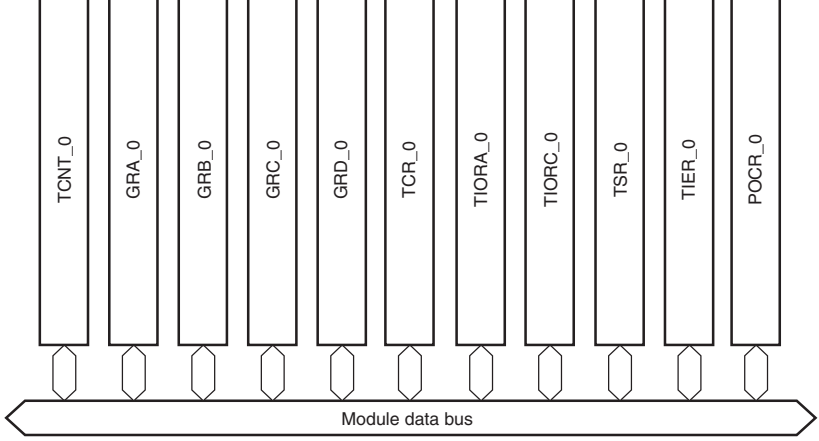
Buffer register		GRC_0, GRD_0	GRC_1, GRD_1
I/O pins		FTIOA0, FTIOB0, FTIOC0, FTIOD0	FTIOA1, FTIOB1, FTIOC1 FTIOD1
Counter clearing function		Compare match/input capture of GRA_0, GRB_0, GRC_0, or GRD_0	Compare match/input cap GRA_1, GRB_1, GRC_1, GRD_1
Compare match output	0 output	Yes	Yes
	1 output	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 cap 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)
$\overline{\text{ADTRG}}$: A/D conversion start trigger output signal
ITMZ0: Channel 0 interrupt
ITMZ1: Channel 1 interrupt

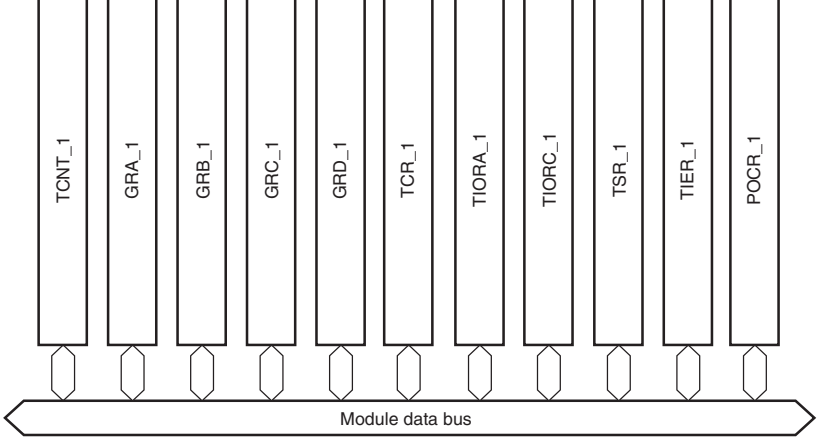
Figure 13.1 Timer Z Block Diagram



[Legend]

TCNT_0:	Timer counter_0 (16 bits)
GRA_0, GRB_0, GRC_0, GRD_0:	General registers A_0, B_0, C_0, and D_0 (input capture/output compare registers: 16 bits × 4)
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)
POCR_0:	PWM mode output level control register_0 (8 bits)
ITMZ0:	Channel 0 interrupt

Figure 13.2 Timer Z (Channel 0) Block Diagram



[Legend]

TCNT_1:	Timer counter_1 (16 bits)
GRA_1, GRB_1, GRC_1, GRD_1:	General registers A_1, B_1, C_1, and D_1 (input capture/output compare registers: 16 bits × 4)
TCR_1:	Timer control register_1 (8 bits)
TIORA_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)
POCR_1:	PWM mode output level control register_1 (8 bits)
ITMZ1:	Channel 1 interrupt

Figure 13.3 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)

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

13.3.2 Timer Mode Register (TMDR)

TMDR selects buffer operation settings and synchronized operation.

Bit	Bit Name	Initial Value	R/W	Description
7	BFD1	0	R/W	Buffer Operation D1 0: GRD_1 operates normally 1: GRB_1 and GRD_1 are used together for buff operation
6	BFC1	0	R/W	Buffer Operation C1 0: GRC_1 operates normally 1: GRA_1 and GRD_1 are used together for buff operation
5	BFD0	0	R/W	Buffer Operation D0 0: GRD_0 operates normally 1: GRB_0 and GRD_0 are used together for buff operation

				counter 1: TCNT_1 and TCNT_0 are synchronized TCNT_1 and TCNT_0 can be pre-set or clea synchronously

13.3.3 Timer PWM Mode Register (TPMR)

TPMR sets the pin to enter PWM mode.

Bit	Bit Name	Initial Value	R/W	Description
7	—	1	—	Reserved This bit is always read as 1, and cannot be mod
6	PWMD1	0	R/W	PWM Mode D1 0: FTIOD1 operates normally 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 mod

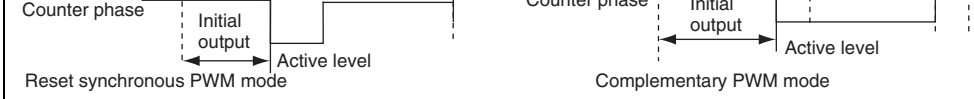
1: FTIOB0 operates in PWM mode

13.3.4 Timer Function Control Register (TFCR)

TFCR selects the settings and output levels for each operating mode.

Bit	Bit Name	Initial Value	R/W	Description
7	—	1	—	Reserved This bit is always read as 1.
6	STCLK	0	R/W	External Clock Input Select 0: External clock input is disabled 1: External clock input is enabled
5	ADEG	0	R/W	A/D Trigger Edge Select A/D module should be set to start an A/D conver the external trigger 0: A/D trigger at the crest in complementary PWI 1: A/D trigger at the trough in complementary PW
4	ADTRG	0	R/W	External Trigger Disable 0: A/D trigger for PWM cycles is disabled in complementary PWM mode 1: A/D trigger for PWM cycles is enabled in complementary PWM mode

				0: Initial output is high and the active level is lo
				1: Initial output is low and the active level is hig
				Figure 13.4 shows an example of outputs in re synchronous PWM mode and complementary P mode when OLS1 = 0 and OLS0 = 0.
1	CMD1	0	R/W	Combination Mode 1 and 0
0	CMD0	0	R/W	00: Channel 0 and channel 1 operate normally
				01: Channel 0 and channel 1 are used together operate in reset synchronous PWM mode
				10: Channel 0 and channel 1 are used together operate in complementary PWM mode (tra the trough)
				11: Channel 0 and channel 1 are used together operate in complementary PWM mode (tra the crest)
				Note: When reset synchronous PWM mode or complementary PWM mode is selected b bits, this setting has the priority to the set PWM mode by each bit in TPMR. Stop T and TCNT_1 before making settings for r synchronous PWM mode or complement mode.



Note: Write H'00 to TOCR to start initial outputs after stopping the counter.

Figure 13.4 Example of Outputs in Reset Synchronous PWM Mode and Complementary PWM Mode

13.3.5 Timer Output Master Enable Register (TOER)

TOER enables/disables the outputs for channel 0 and channel 1. When $\overline{\text{WKP4}}$ is selected inputs, if a low level signal is input to $\overline{\text{WKP4}}$, the bits in TOER are set to 1 to disable the for timer Z.

Bit	Bit Name	Initial Value	R/W	Description
7	ED1	1	R/W	Master Enable D1 0: FTIOD1 pin output is enabled according to the TFCR, and TIORC_1 settings 1: FTIOD1 pin output is disabled regardless of th TFCR, and TIORC_1 settings (FTIOD1 pin is as an I/O port).
6	EC1	1	R/W	Master Enable C1 0: FTIOC1 pin output is enabled according to the TFCR, and TIORC_1 settings 1: FTIOC1 pin output is disabled regardless of th TFCR, and TIORC_1 settings (FTIOC1 pin is as an I/O port).

				1: FTIOA1 pin output is disabled regardless of t TFCR, and TIORA_1 settings (FTIOA1 pin is as an I/O port).
3	ED0	1	R/W	Master Enable D0 0: FTIOD0 pin output is enabled according to th TFCR, and TIORC_0 settings 1: FTIOD0 pin output is disabled regardless of t TFCR, and TIORC_0 settings (FTIOD0 pin is as an I/O port).
2	EC0	1	R/W	Master Enable C0 0: FTIOC0 pin output is enabled according to th TFCR, and TIORC_0 settings 1: FTIOC0 pin output is disabled regardless of t TFCR, and TIORC_0 settings (FTIOC0 pin is as an I/O port).
1	EB0	1	R/W	Master Enable B0 0: FTIOB0 pin output is enabled according to th TFCR, and TIORA_0 settings 1: FTIOB0 pin output is disabled regardless of t TFCR, and TIORA_0 settings (FTIOB0 pin is as an I/O port).
0	EA0	1	R/W	Master Enable A0 0: FTIOA0 pin output is enabled according to th TFCR, and TIORA_0 settings 1: FTIOA0 pin output is disabled regardless of t TFCR, and TIORA_0 settings (FTIOA0 pin is 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.

13.3.8 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. registers are dual function 16-bit readable/writable registers, functioning as either outpu 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 regi 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 afte external signals. At this point, IMFA to IMFD flags in the corresponding TSR are set to 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 se values in TIORA and TIORC are ignored. Upon reset, the GR registers are set as output registers (no output) and initialized to H'FFFF. The GR registers cannot be accessed in 8 they must always be accessed as a 16-bit unit.

				010: Clears TCNT by GRB compare match/input capture*1
				011: Synchronization clear; Clears TCNT in sync with counter clearing of the other channel's
				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 sync with counter clearing of the other channel's
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) pi

Notes: 1. When GR functions as an output compare register, TCNT is cleared by compa
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 fa
3	—	1	—	Reserved This bit is always read as 1.

				101: Input capture to GRA at the falling edge
				11X: Input capture to GRA at both rising and fall

[Legend] X: Don't care.

TIORC: TIORC selects whether GRC or GRD is used as an output compare register or a capture register. When an output compare register is selected, the output setting is selecte an input capture register is selected, an input edge of an input capture signal is selected. T also selects the function of FTIOC or FTIOD pin.

Bit	Bit Name	Initial value	R/W	Description
7	—	1	—	Reserved This bit is always read as 1.
6	IOD2	0	R/W	I/O Control D2 to D0
5	IOD1	0	R/W	GRD is an output compare register:
4	IOD0	0	R/W	000: Disables pin output by compare match 001: 0 output by GRD compare match 010: 1 output by GRD compare match 011: Toggle output by GRD compare match GRD is an input capture register: 100: Input capture to GRD at the rising edge 101: Input capture to GRD at the falling edge 11X: Input capture to GRD at both rising and fall edges
3	—	1	—	Reserved This bit is always read as 1.

101: Input capture to GRC at the falling edge

11X: Input capture to GRC at both rising and fa
edges

[Legend] X: Don't care.

				These bits are always read as 1.
5	UDF*	0	R/W	Underflow Flag [Setting condition] • When TCNT_1 underflows [Clearing condition] • When 0 is written to UDF after reading UDF =
4	OVF	0	R/W	Overflow Flag [Setting condition] • When the TCNT value underflows [Clearing condition] • When 0 is written to OVF after reading OVF =
3	IMFD	0	R/W	Input Capture/Compare Match Flag D [Setting conditions] • When TCNT = GRD and GRD is functioning compare register • When TCNT value is transferred to GRD by i capture signal and GRD is functioning as inp capture register [Clearing condition] • When 0 is written to IMFD after reading IMFD

				• When 0 is written to IMFC after reading IMF
1	IMFB	0	R/W	Input Capture/Compare Match Flag B [Setting conditions] • When TCNT = GRB and GRB is functioning compare register • When TCNT value is transferred to GRB by capture signal and GRB is functioning as in capture register [Clearing condition] • When 0 is written to IMFB after reading IMF
0	IMFA	0	R/W	Input Capture/Compare Match Flag A [Setting conditions] • When TCNT = GRA and GRA is functioning compare register • When TCNT value is transferred to GRA by capture signal and GRA is functioning as in capture register [Clearing condition] • When 0 is written to IMFA after reading IMF

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

				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

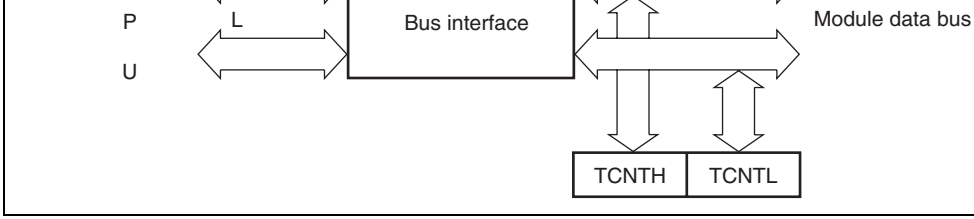


Figure 13.5 Accessing Operation of 16-Bit Register (between CPU and TCNT (16

2. 8-bit register

Registers other than TCNT and GR are 8-bit registers that are connected internally wi
CPU in an 8-bit width. Figure 13.6 shows an example of accessing the 8-bit registers.

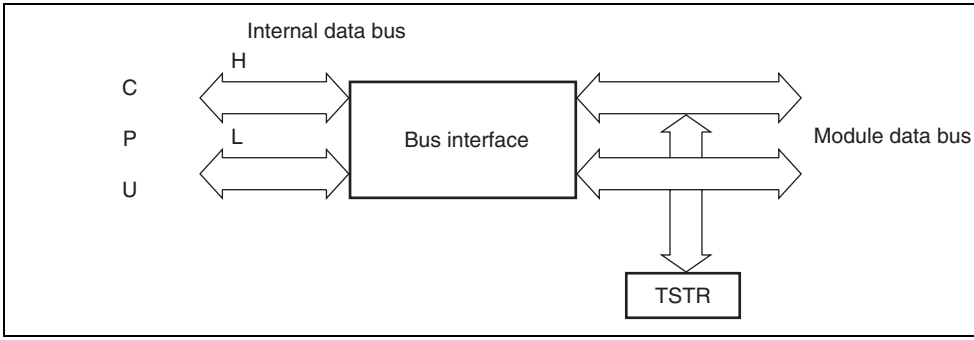
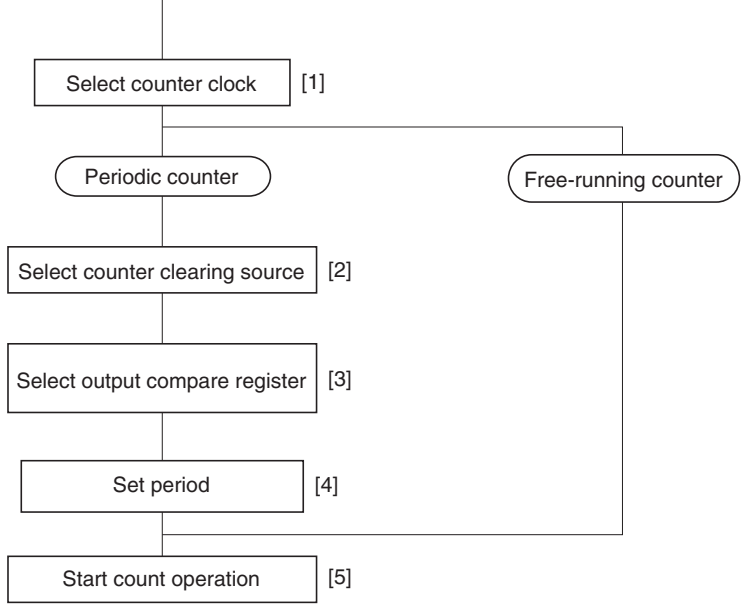


Figure 13.6 Accessing Operation of 8-Bit Register (between CPU and TSTR (8



TPSC2 to TPS
TCR. When an
clock is select
the external cl
with bits CKEG
and CKEG0 in
[2] For periodic co
operation, sele
TCNT clearing
source with bit
CCLR2 to CCL
TCR.
[3] Designate the
register select
as an output c
register by me
TIOR.
[4] Set the period
cycle in the ge
register select
in [2].
[5] Set the STR b
to 1 to start the
operation.

Figure 13.7 Example of Counter Operation Setting Procedure

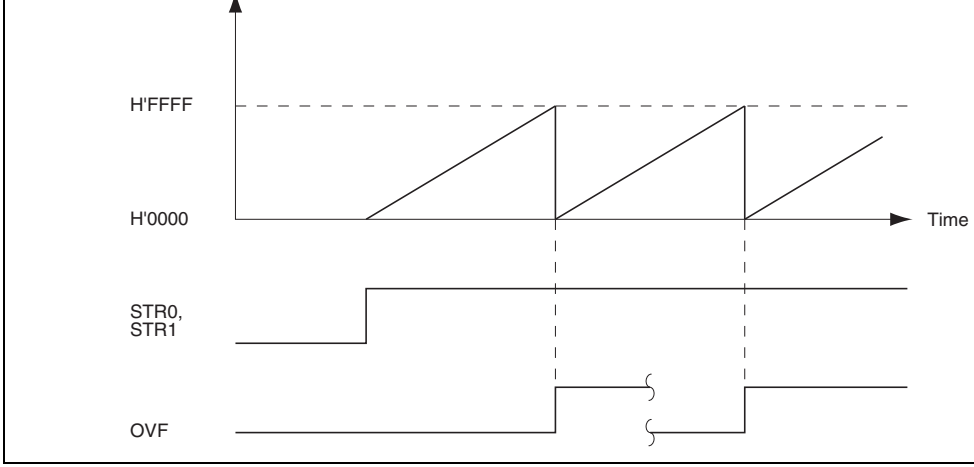


Figure 13.8 Free-Running Counter Operation

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 de
as output compare registers, and counter clearing by compare match is selected by means
CCLR1 and CCLR0 in TCR. After the settings have been made, TCNT starts an increme
operation as a periodic counter when the corresponding bit in TSTR is set to 1. When the
value matches the value in GR, the IMFA, IMFB, IMFC, or IMFD flag in TSR is set to 1
TCNT is cleared to H'0000.

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

Figure 13.9 illustrates periodic counter operation.

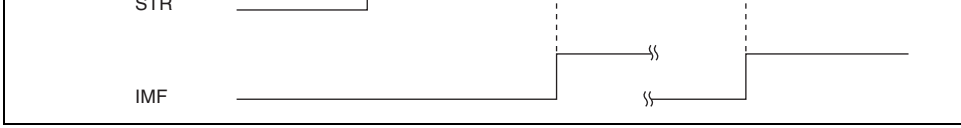


Figure 13.9 Periodic Counter Operation

2. TCNT count timing

   A. Internal clock operation

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

   Figure 13.10 illustrates this timing.

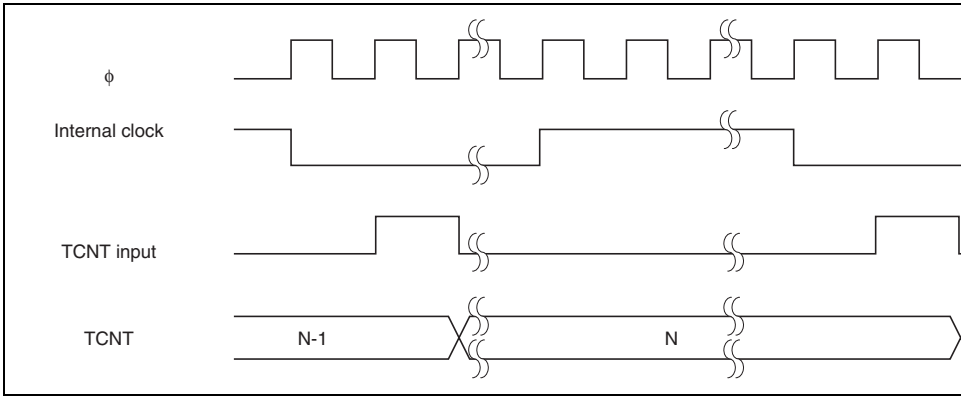


Figure 13.10 Count Timing at Internal Clock Operation

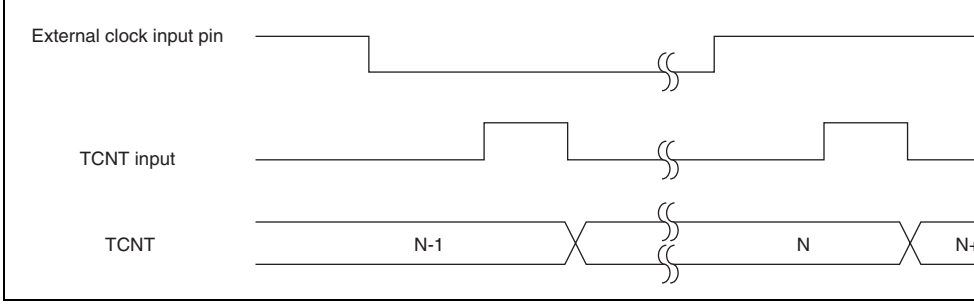


Figure 13.11 Count Timing at External Clock Operation (Both Edges Detect

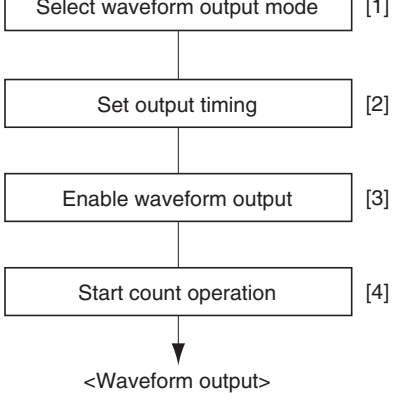


Figure 13.12 Example of Setting Procedure for Waveform Output by Compare

1. Examples of waveform output operation

   Figure 13.13 shows an example of 0 output/1 output.

   In this example, TCNT has been designated as a free-running counter, and settings h made such that 0 is output by compare match A, and 1 is output by compare match B the set level and the pin level coincide, the pin level does not change.

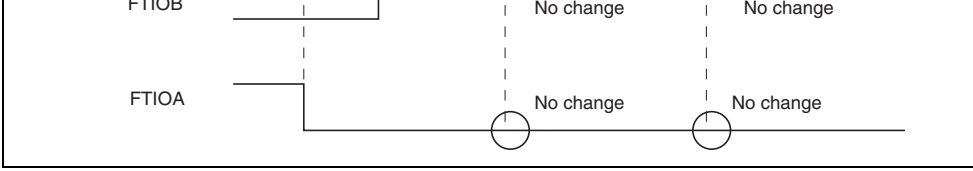


Figure 13.13 Example of 0 Output/1 Output Operation

Figure 13.14 shows an example of toggle output.

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

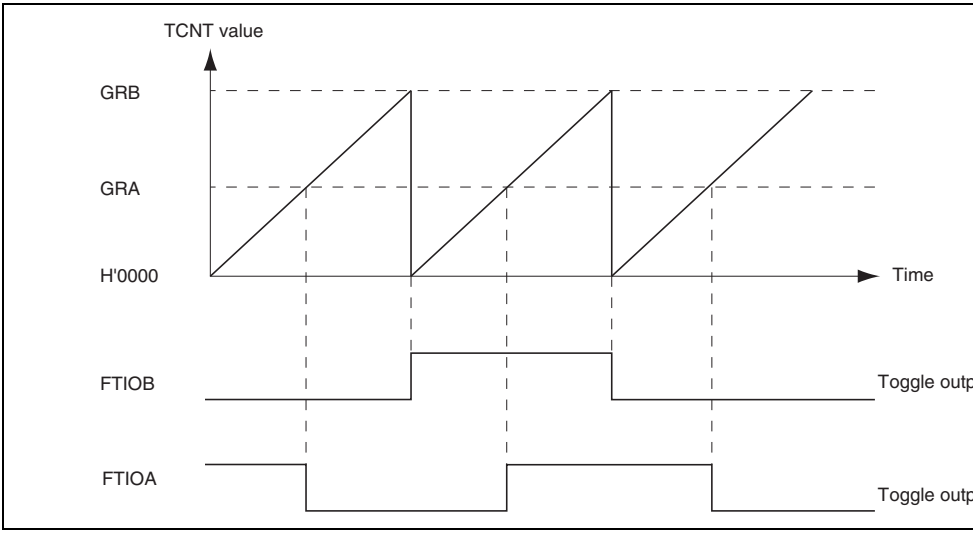


Figure 14.14 Example of Toggle Output Operation

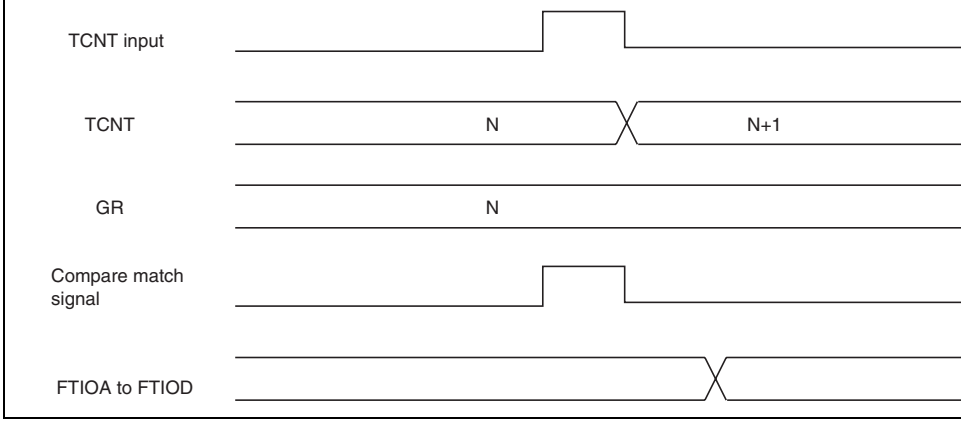


Figure 13.15 Output Compare Timing

13.4.3 Input Capture Function

The TCNT value can be transferred to GR on detection of the input edge of the input capture/output compare pin (FTIOA, FTIOB, FTIOC, or FTIOD). Rising edge, falling e both edges can be selected as the detected edge. When the input capture function is used width or period can be measured.

Figure 13.16 shows an example of the input capture operation setting procedure.

<Input capture operation>

Figure 13.16 Example of Input Capture Operation Setting Procedure

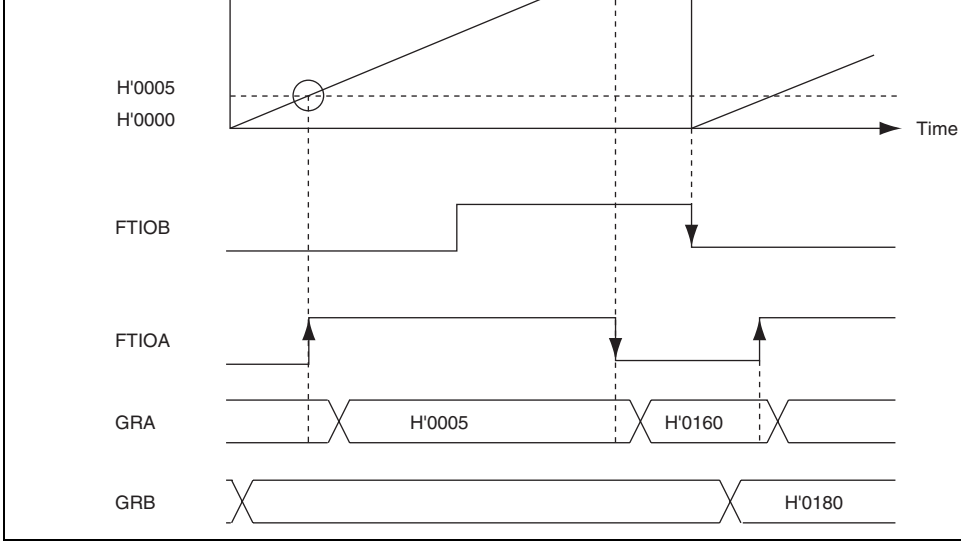


Figure 13.17 Example of Input Capture Operation

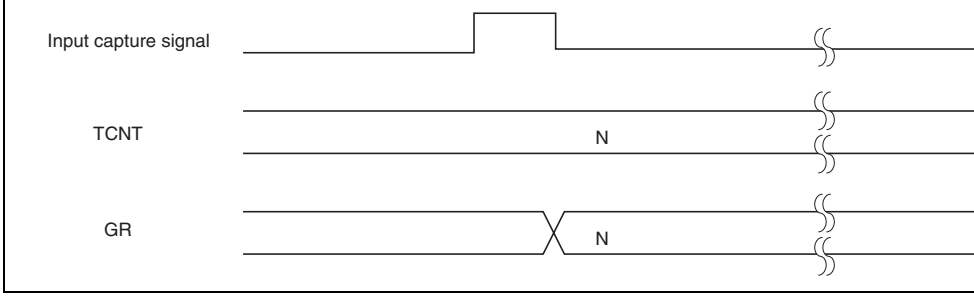
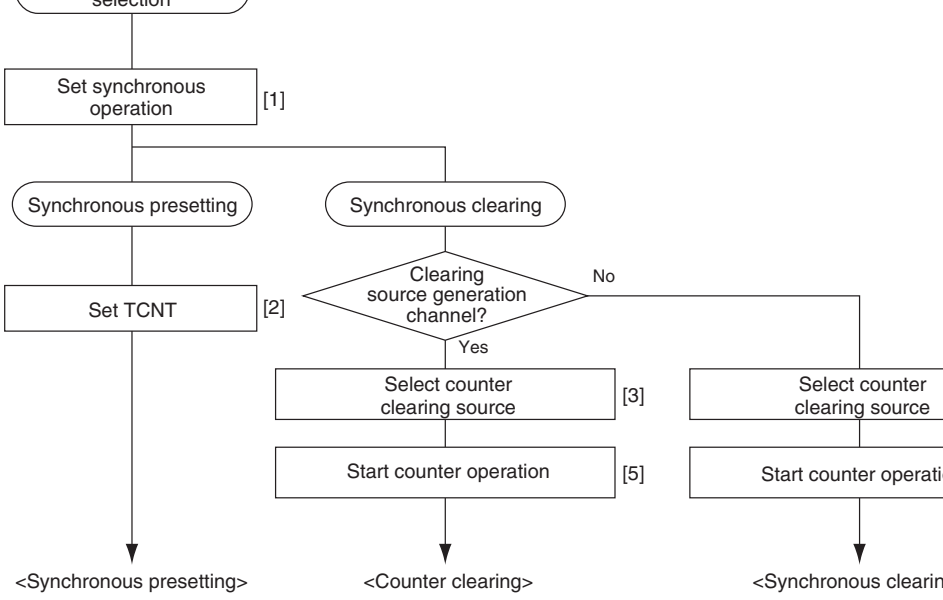


Figure 13.18 Input Capture Signal Timing



[1] Set the SYNC bits in TMDR to 1.
[2] When a value is written to either of the TCNT counters, the same value is simultaneously written to t other TCNT counter.
[3] Set bits CCLR1 and CCLR0 in TCR to specify counter clearing by compare match/input capture.
[4] Set bits CCLR1 and CCLR0 in TCR to designate synchronous clearing for the counter clearing sourc
[5] Set the STR bit in TSTR to 1 to start the count operation.

Figure 13.19 Example of Synchronous Operation Setting Procedure

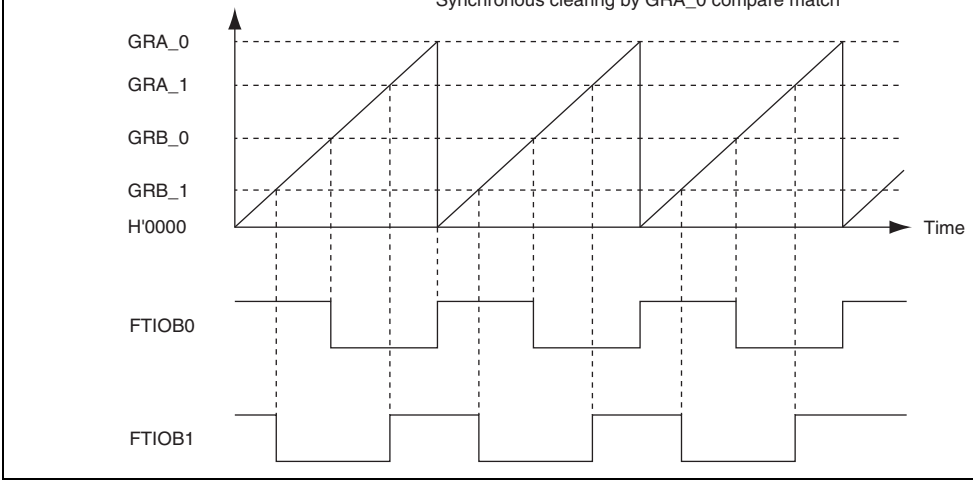


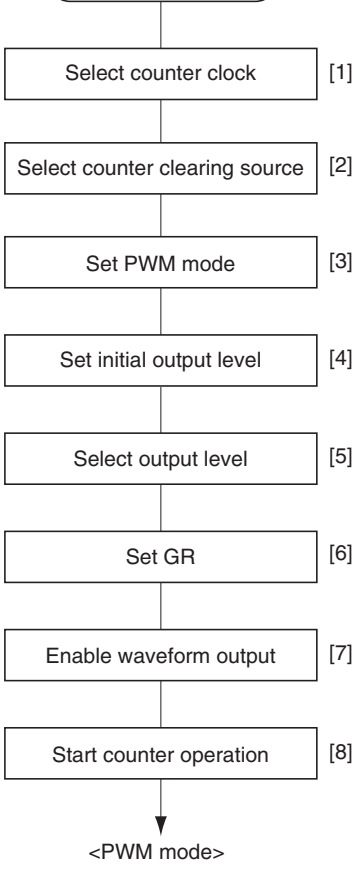
Figure 13.20 Example of Synchronous Operation

13.4.5 PWM Mode

In PWM mode, PWM waveforms are output from the FTIOB, FTIOC, and FTIOD outpu with GRA as a cycle register and GRB, GRC, and GRD as duty registers. The initial outp of the corresponding pin depends on the setting values of TOCR and POCR. Table 13.3 s example of the initial output level of the FTIOB0 pin.

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

Figure 13.21 shows an example of the PWM mode setting procedure.



[1] Select the counter clock with bits TPSC2 to TOSC0 in TCR. When an external clock is selected, select the external clock edge with bits CKEG1 and CKEG0 in TCR.
[2] Use bits CCLR1 and CCLR0 in TCR to select the counter clearing source.
[3] Select the PWM mode with bits PWMB0 to PWMD0 and PWMB1 to PWMD1 in TPMR.
[4] Set the initial output value with bits TOB0 to TOD0 and TOB1 to TOD1 in TOCR.
[5] Set the output level with bits POLB to POLD in POCR.
[6] Set the cycle in GRA, and set the duty in the other GR.
[7] Enable or disable the timer output by TOER.
[8] Set the STR bit in TSTR to 1 and start the counter operation.

Figure 13.21 Example of PWM Mode Setting Procedure

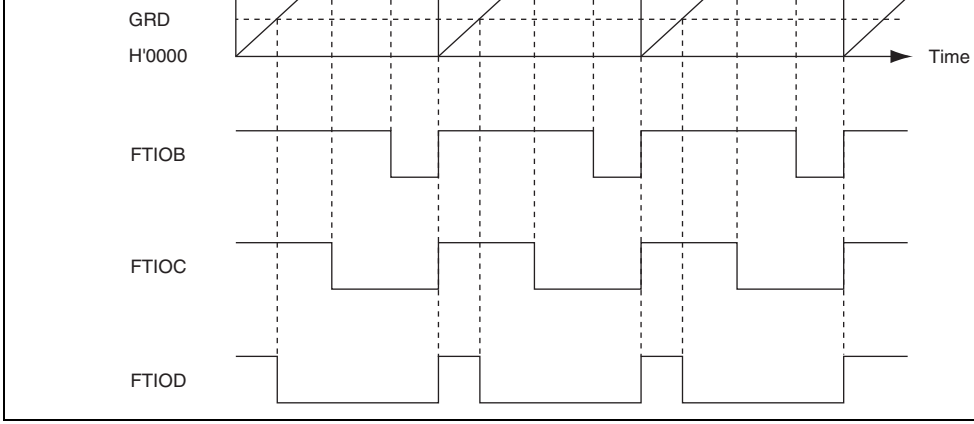


Figure 13.22 Example of PWM Mode Operation (1)

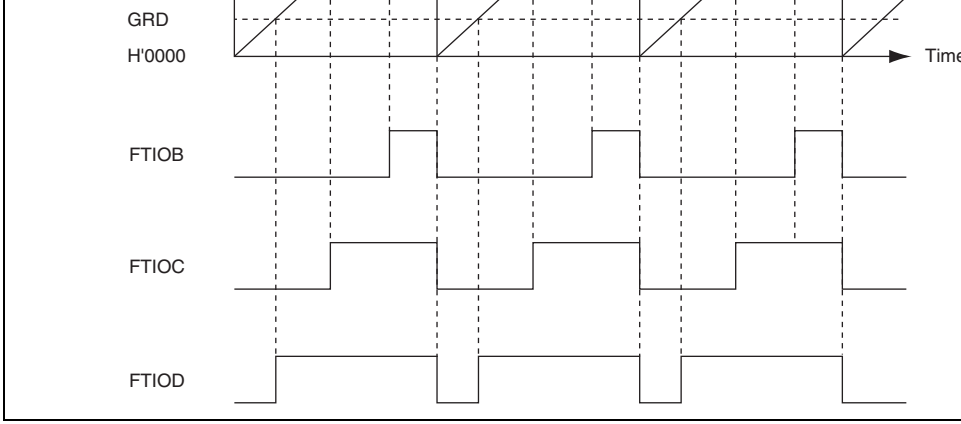


Figure 13.23 Example of PWM Mode Operation (2)

Figures 13.24 (when TOB, TOC, and TOD = 0, POLB, POLC, and POLD = 0) and 13.2
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.

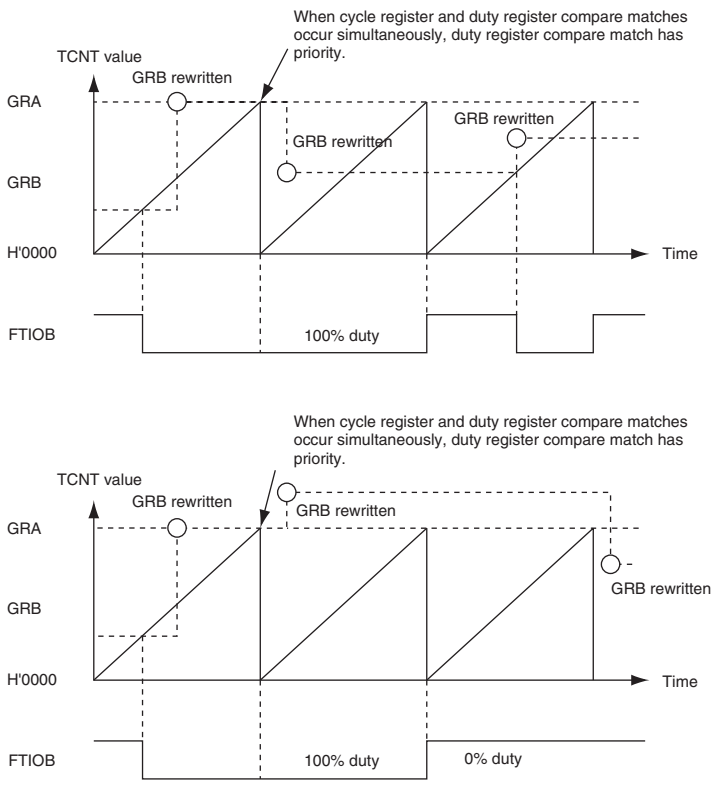


Figure 13.24 Example of PWM Mode Operation (3)

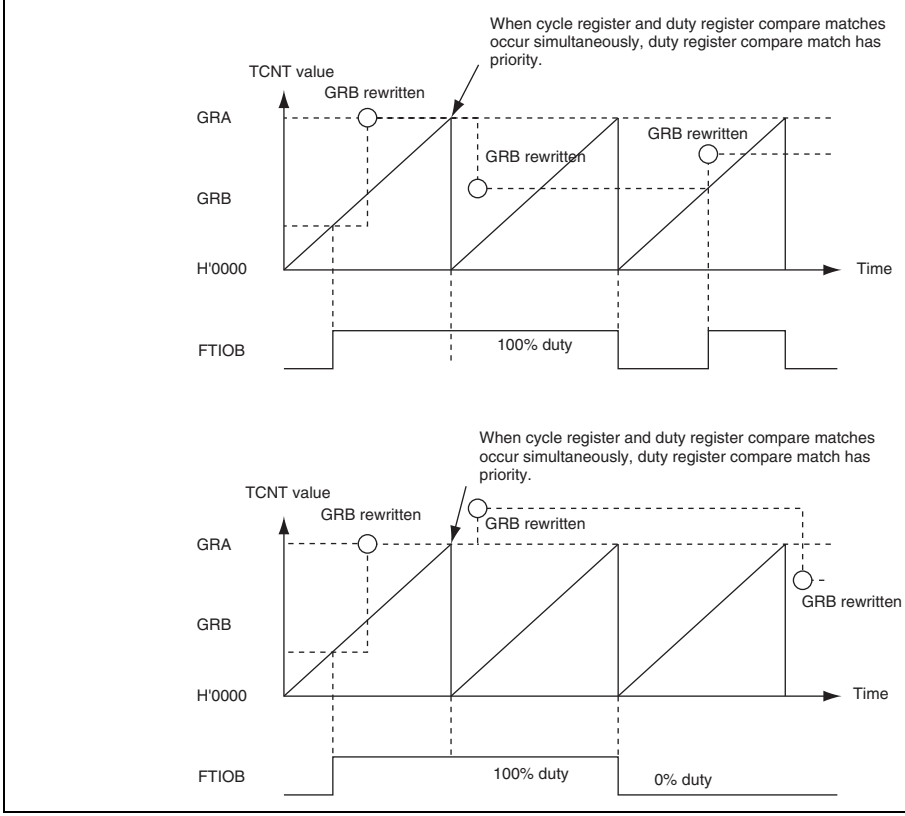
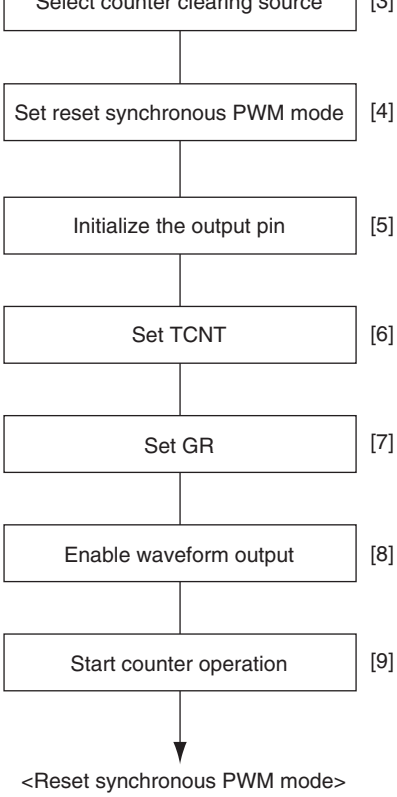


Figure 13.25 Example of PWM Mode Operation (4)

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 PW output 1)
1	FTIOA1	Output	PWM output 2
1	FTIOC1	Output	PWM output 2 (counter-phase waveform of PW output 2)
1	FTIOB1	Output	PWM output 3
1	FTIOD1	Output	PWM output 3 (counter-phase waveform of PW output 3)

Table 13.5 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 an FTIOD0.
GRA_1	Set a changing point of the PWM waveform output from pins FTIOA1 an FTIOC1.
GRB_1	Set a changing point of the PWM waveform output from pins FTIOB1 an FTIOD1.



[4] Select the reset synchronous PWM m with 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 need to be set.
[7] GRA_0 is a cycle register. Set a cycl GRA_0. Set the changing point timin 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 star counter operation.

Figure 13.26 Example of Reset Synchronous PWM Mode Setting Procedu

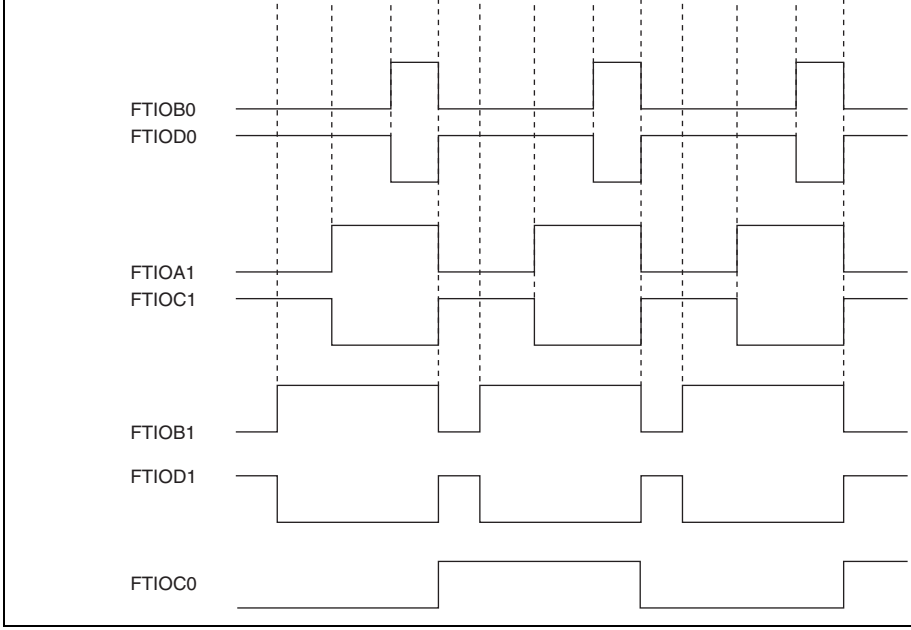


Figure 13.27 Example of Reset Synchronous PWM Mode Operation (OLS0 = OL

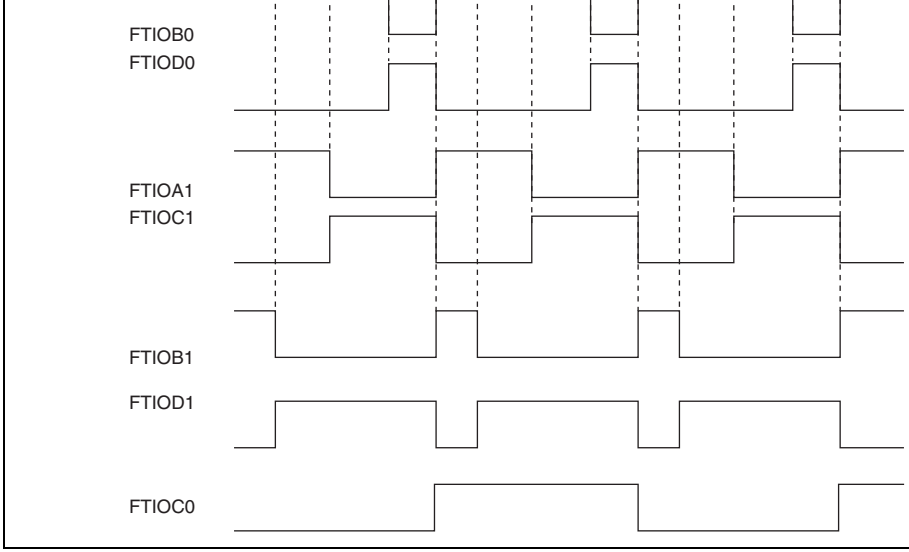


Figure 13.28 Example of Reset Synchronous PWM Mode Operation (OLS0 = O

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

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

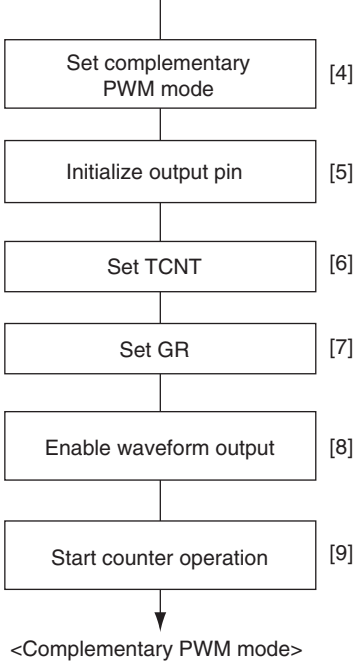
For details on operations when reset synchronous PWM mode and buffer operation are simultaneously set, refer to section 13.4.8, Buffer Operation.

Table 13.6 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 13.7 Register Settings in Complementary PWM Mode

Register	Description
TCNT_0	Initial setting of non-overlapped periods (non-overlapped periods are diff with 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 an FTIOD0.
GRA_1	Set a changing point of the PWM waveform output from pins FTIOA1 an FTIOC1.
GRB_1	Set a changing point of the PWM waveform output from pins FTIOB1 an FTIOD1.



[4] Use bits CMD1 and CMD0 in TFCR to set complementary PWM mode. FTIOB0 to FTIOD0 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-overlapped 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 13.4.7, 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 complemer 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 s in 2. Examples of Complementary PWM Mode Operation and 3. Setting GR Value in Complementary PWM Mode in section 13.4.7, Complementary PWM Mode.

Figure 13.29 Example of Complementary PWM Mode Setting Procedur

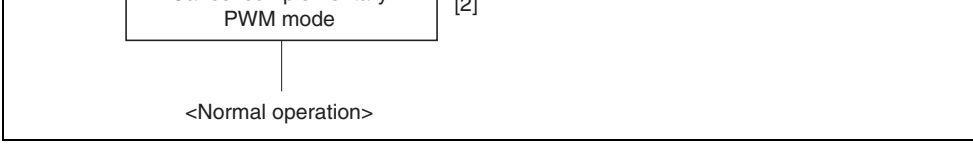


Figure 13.30 Canceling Procedure of Complementary PWM Mode

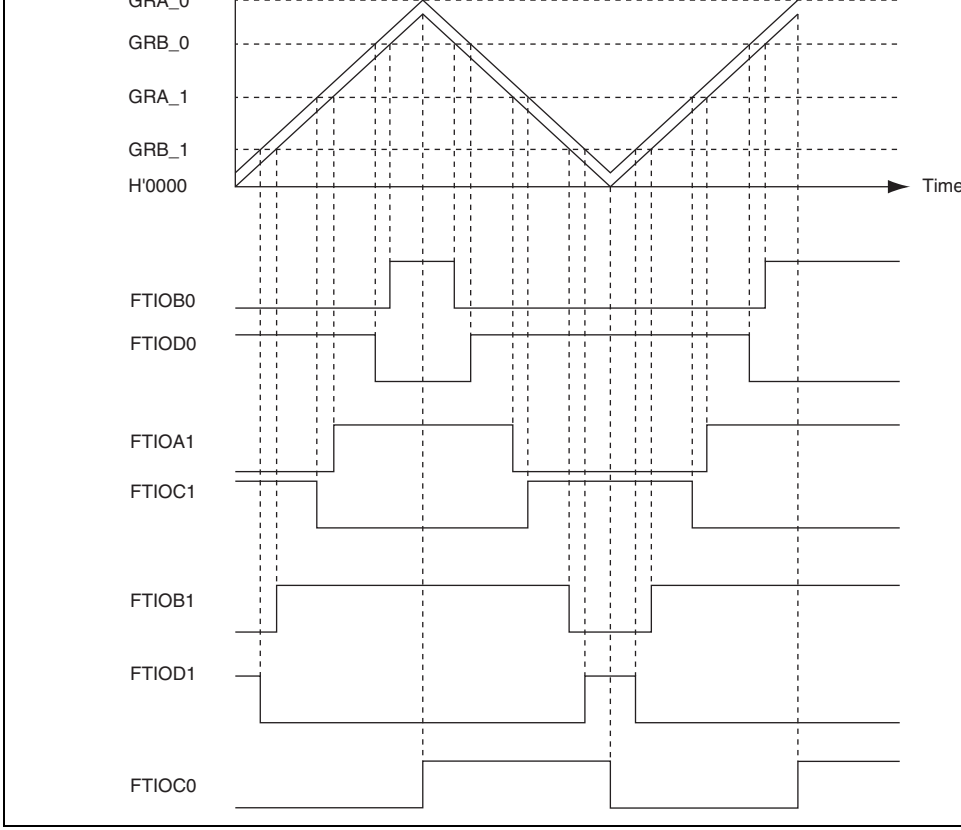


Figure 13.31 Example of Complementary PWM Mode Operation (1)

cycle waveform output, see 3.C., Outputting a waveform with a duty cycle of 0% and section 13.4.7, Complementary PWM Mode.

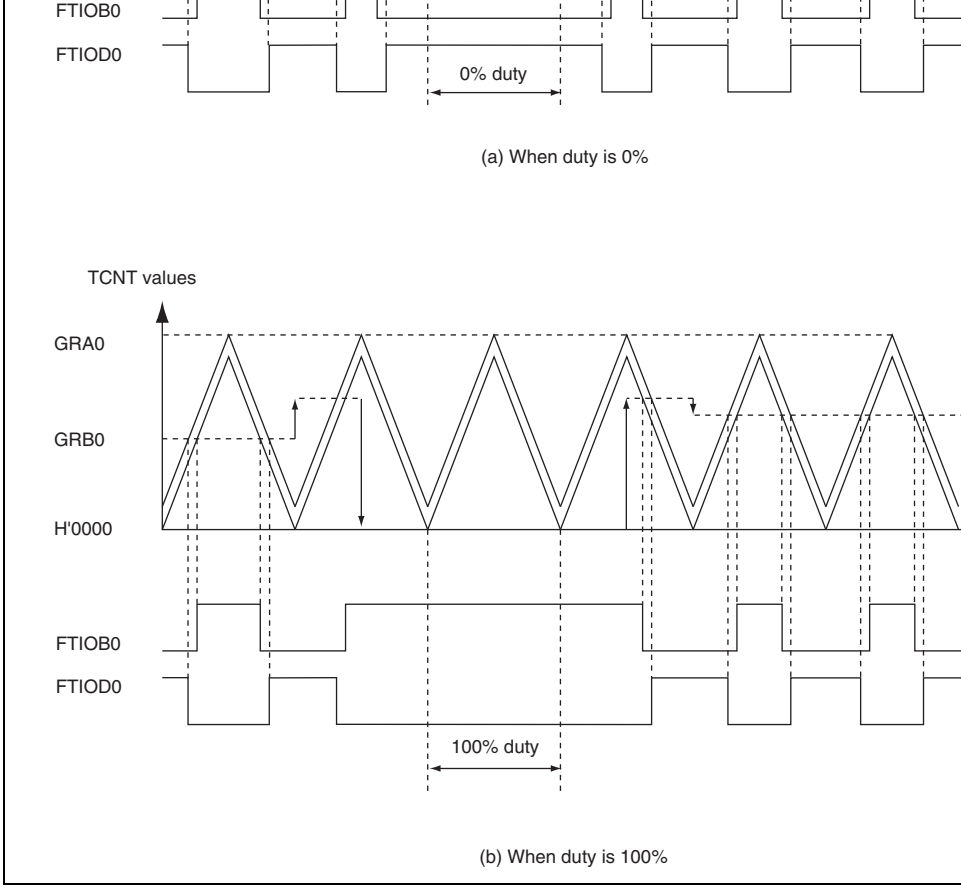


Figure 13.32 (1) Example of Complementary PWM Mode Operation (TPSC2 = TPSC1 = TPSC0 = 0) (2)

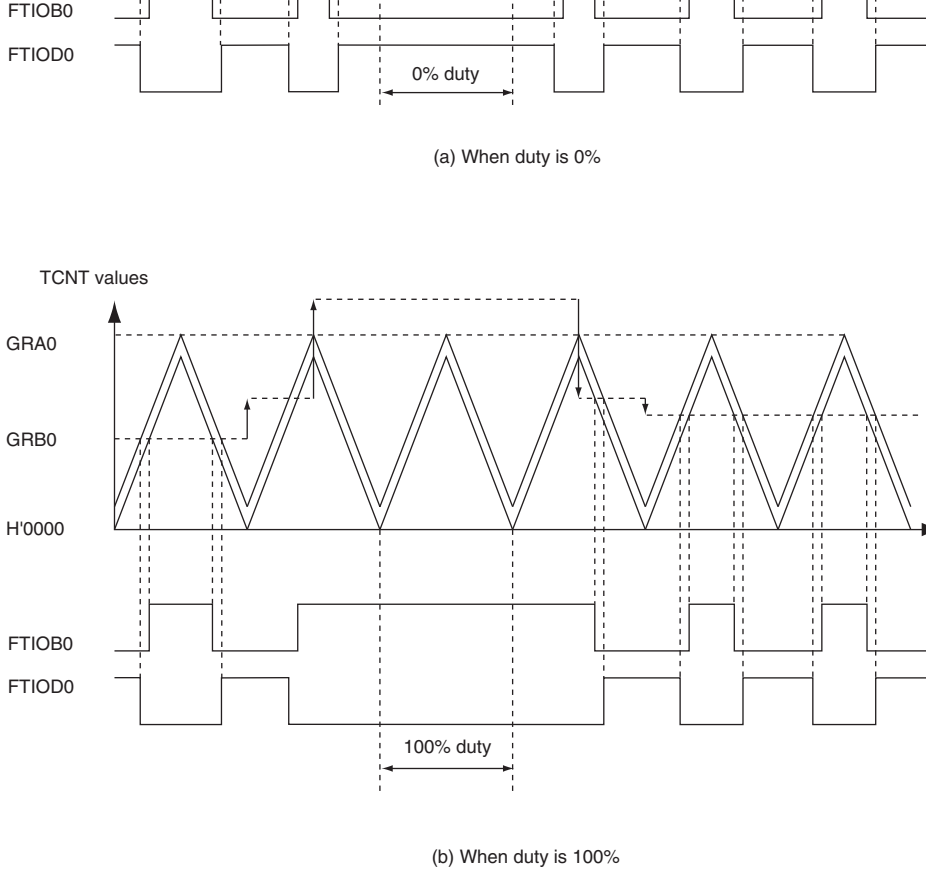


Figure 13.32 (2) Example of Complementary PWM Mode Operation (TPSC2 = TPSC1 = TPSC0 ≠ 0) (3)

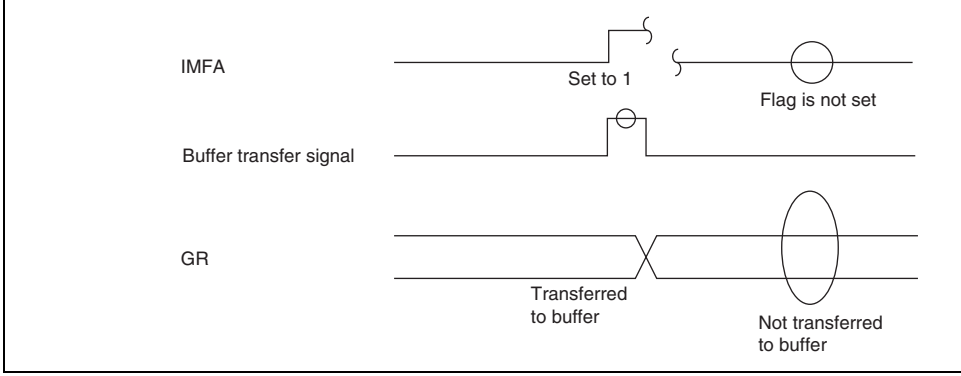


Figure 13.33 Timing of Overshooting

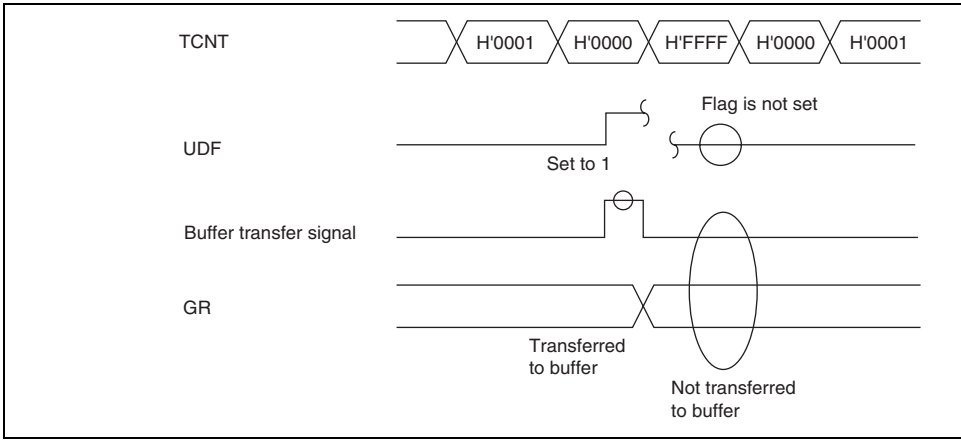


Figure 13.34 Timing of Undershooting

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

b. H'0000 to T – 1 (T: Initial value of TCNT0) must not be set for the initial valu

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 register
corresponding general registers.

B. Modifying the setting value

a. Writing to GR directly must be performed while the TCNT_1 and TCNT_0 va
should satisfy the following expression: H'0000 ≤ TCNT_1 < previous GR val
previous GR value < TCNT_0 ≤ GRA_0. Otherwise, a waveform is not output
correctly. For details on outputting a waveform with a duty cycle of 0% and 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 w
is not output correctly.

H'0000 ≤ GR ≤ T − 1 and GRA_0 − (T − 1) ≤ GR < GRA_0 when TPSC2 = T
TPSC0 = 0

H'0000 < GR ≤ T − 1 and GRA_0 − (T − 1) ≤ GR < GRA_0 + 1 when TPSC2
= 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
timing shown below.

- To output a 0%-duty cycle waveform, write a value equal to or more than the G
value while H'0000 ≤ TCNT_1 < previous GR value
- To output a 100%-duty cycle waveform, write H'0000 while previous GR valu
TCNT_0 ≤ GRA_0

- To output a 0%-duty cycle waveform, write a value equal to or more than the 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 13.4.8, 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 th shown below.

- To output a 0%-duty cycle waveform, write the value while H'0000 ≤ TCNT_ previous GR value
- To output a 100%-duty cycle waveform, write the value while previous GR v TCNT_0 ≤ GRA_0

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

- To change duty cycles while a 0%-duty cycle waveform is being output, writ while H'0000 ≤ TCNT_1 < previous GR value
- To change duty cycles while a 100%-duty cycle waveform is being output, w while previous GR value< TCNT_0 ≤ GRA_0

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

d. Buffer operation is used and other than TPSC2 = TPSC1 = TPSC0 = 0

Write a value which satisfies GRA_0 + 1 < GR < H'FFFF to the buffer regist waveform with a duty cycle of 0% can be output. However, a waveform with cycle of 100% cannot be output using the buffer operation. Also, the buffer o cannot be used to change duty cycles while a waveform with a duty cycle of being output. For details on buffer operation, see section 13.4.8, Buffer Opera

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 ch transferred to the general register.

   This operation is illustrated in figure 13.35.

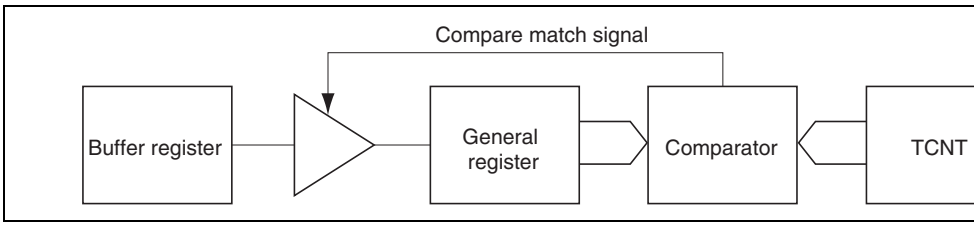


Figure 13.35 Compare Match Buffer Operation

2. When GR is an input capture register

   When an input capture occurs, the value in TCNT is transferred to the general register value previously stored in the general register is transferred to the buffer register.

   This operation is illustrated in figure 13.36.

buffer register is transferred to the general register. Here, the value of the buffer regi
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

5. Example of Buffer Operation Setting Procedure

   Figure 13.37 shows an example of the buffer operation setting procedure.

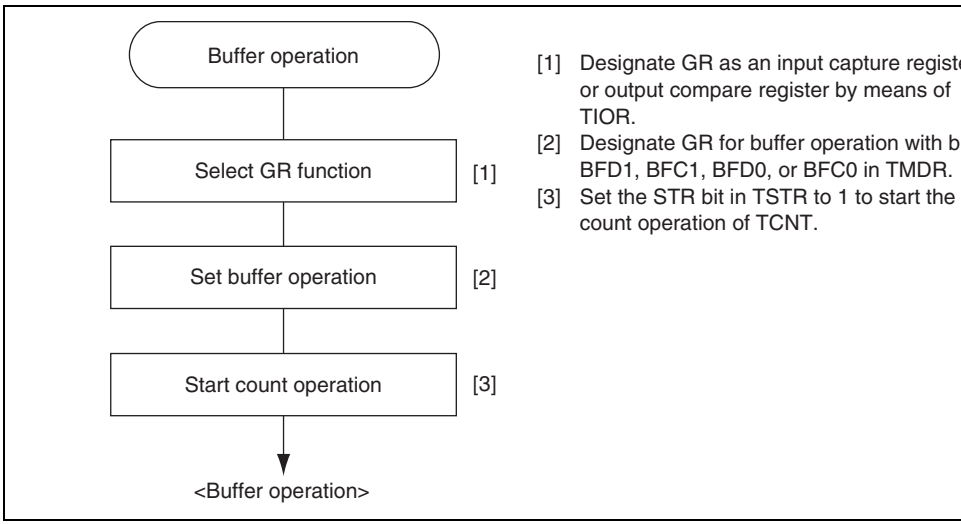


Figure 13.37 Example of Buffer Operation Setting Procedure

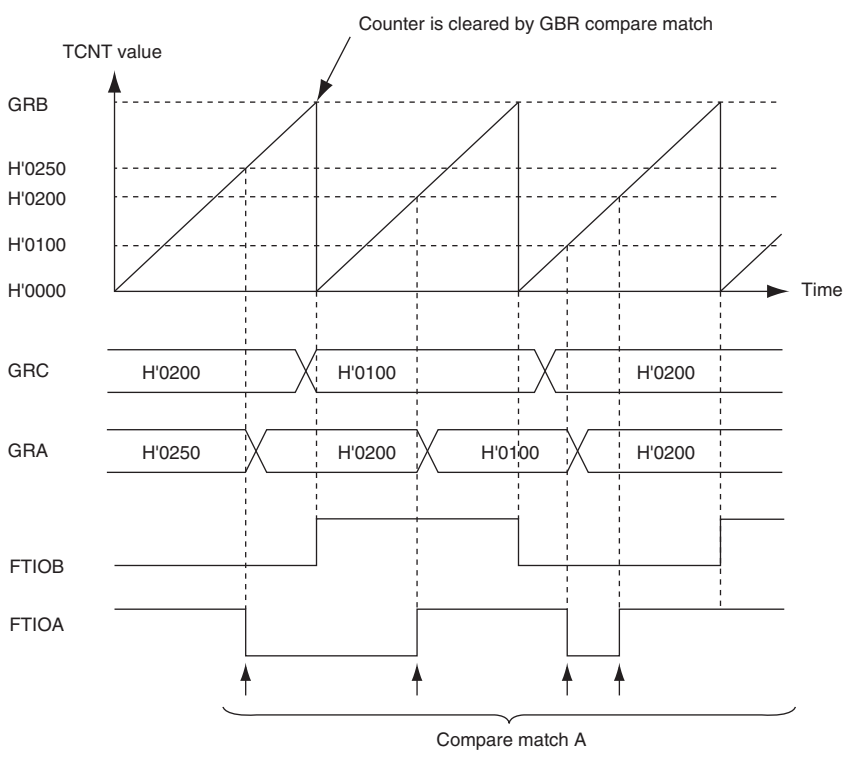


Figure 13.38 Example of Buffer Operation (1)
(Buffer Operation for Output Compare Register)

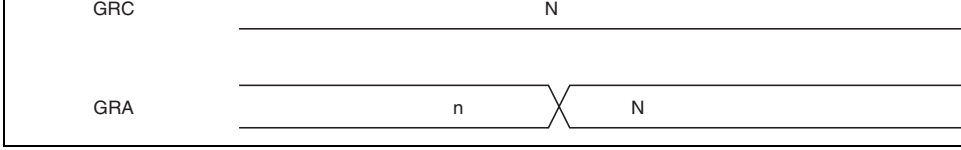


Figure 13.39 Example of Compare Match Timing for Buffer Operation

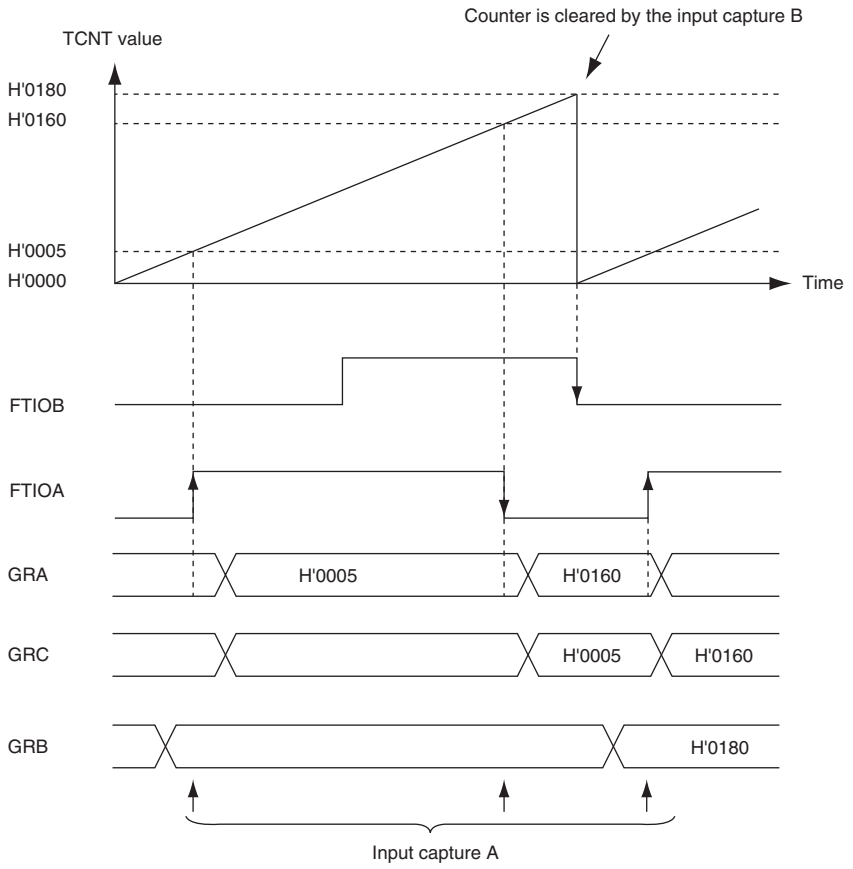


Figure 13.40 Example of Buffer Operation (2)
(Buffer Operation for Input Capture Register)

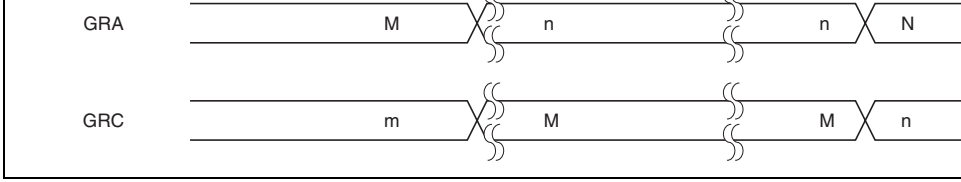


Figure 13.41 Input Capture Timing of Buffer Operation

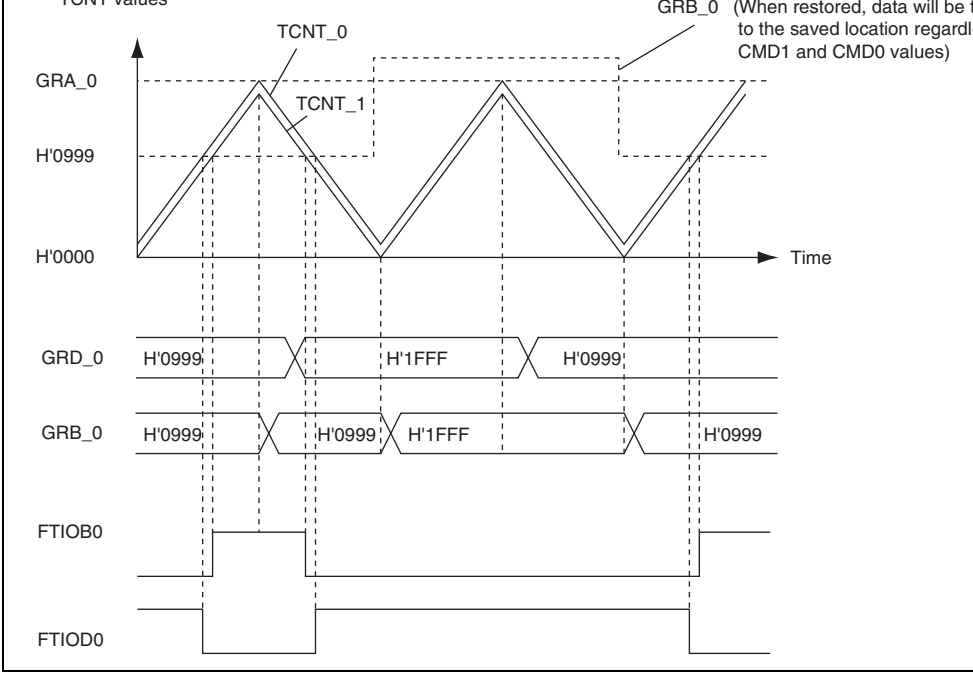


Figure 13.42 Buffer Operation (3)
(Buffer Operation in Complementary PWM Mode CMD1 = CMD0 = 1)

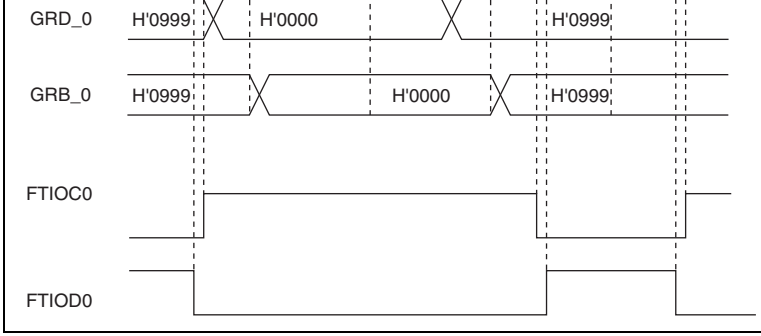


Figure 13.43 Buffer Operation (4)
(Buffer Operation in Complementary PWM Mode CMD1 = CMD0 = 1)

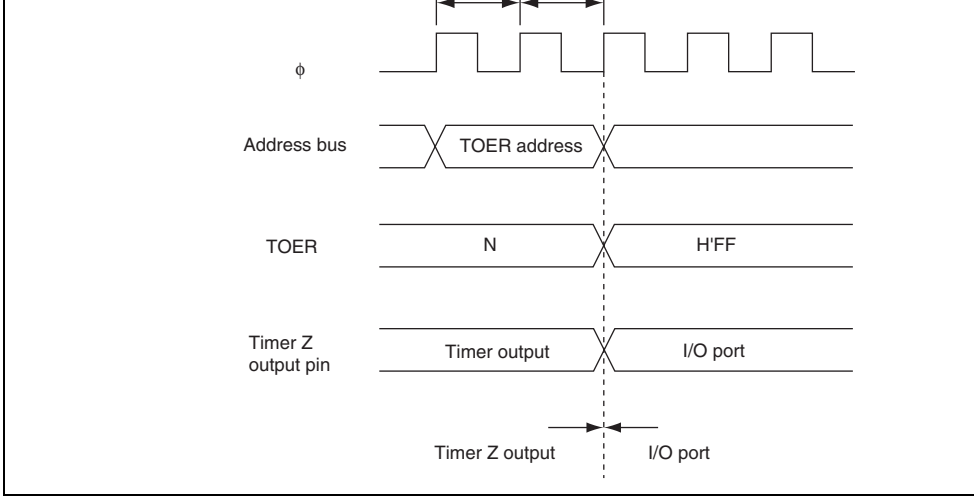


Figure 13.44 Example of Output Disable Timing of Timer Z by Writing to TO

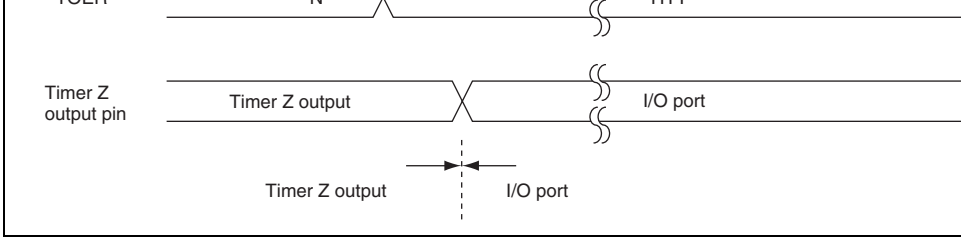


Figure 13.45 Example of Output Disable Timing of Timer Z by External Tri

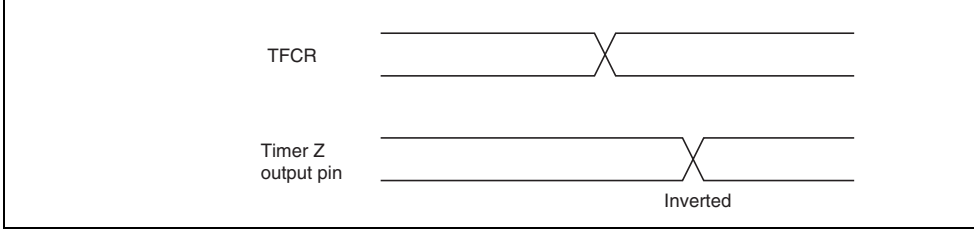


Figure 13.46 Example of Output Inverse Timing of Timer Z by Writing to TF

4. Output Inverse Timing by POCR: The output level can be inverted by inverting the PO
POLC, and POLB bits in POCR in PWM mode. Figure 13.47 shows the timing.

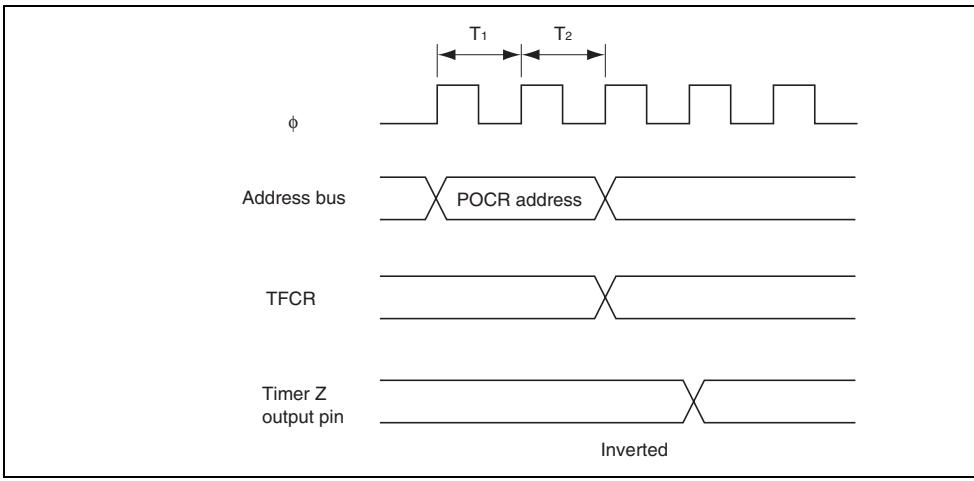


Figure 13.47 Example of Output Inverse Timing of Timer Z by Writing to PO

when the TCNT and GR matches, the compare match signal will not be generated un
TCNT input clock is generated. Figure 13.48 shows the timing to set the IMF flag.

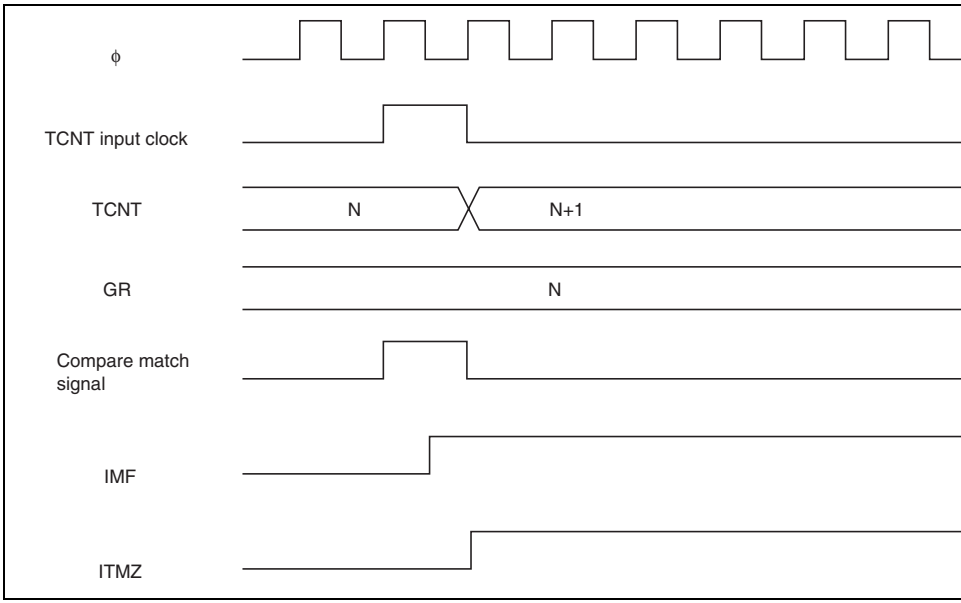


Figure 13.48 IMF Flag Set Timing when Compare Match Occurs

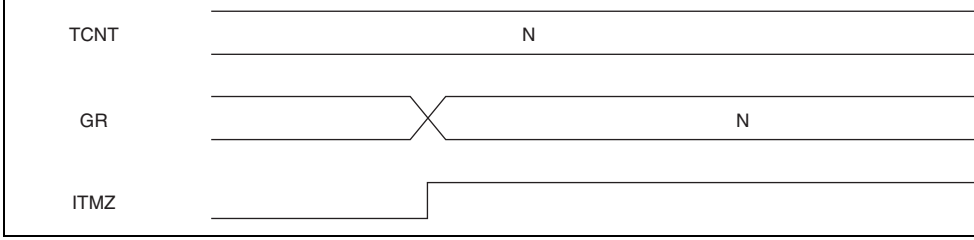


Figure 13.49 IMF Flag Set Timing at Input Capture

3. Overflow Flag (OVF) Set Timing: The overflow flag is set to 1 when the TCNT overf
Figure 13.50 shows the timing.

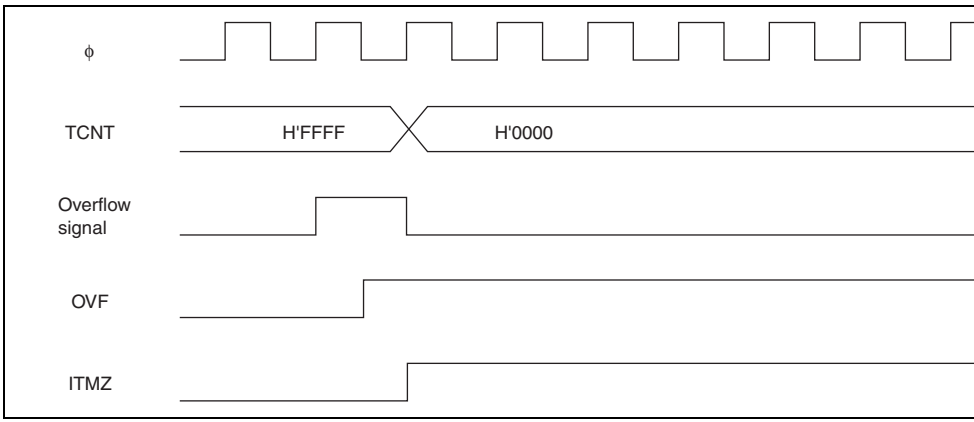


Figure 13.50 OVF Flag Set Timing

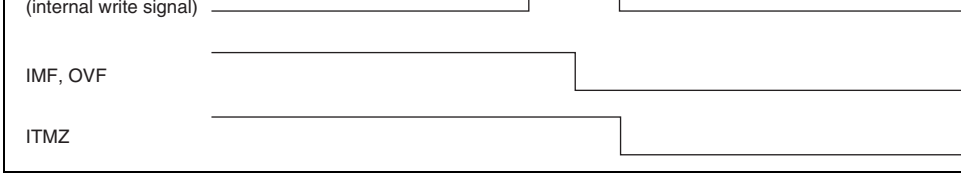


Figure 13.51 Status Flag Clearing Timing

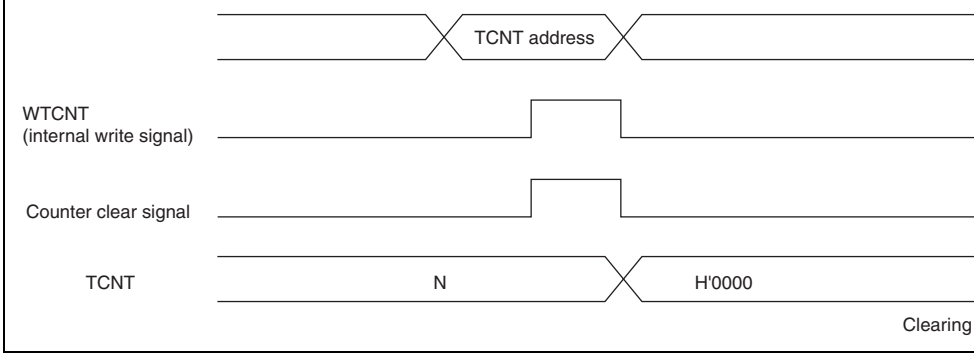


Figure 13.52 Contention between TCNT Write and Clear Operations

2. Contention between TCNT Write and Increment Operations: If increment is done in T a TCNT write cycle, TCNT writing has priority. Figure 13.53 shows the timing in this

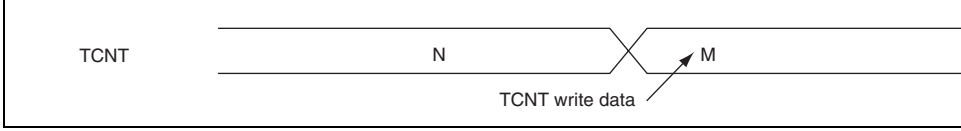


Figure 13.53 Contention between TCNT Write and Increment Operation

3. Contention between GR Write and Compare Match: If a compare match occurs in th of a GR write cycle, GR write has priority and the compare match signal is disabled. 13.54 shows the timing in this case.

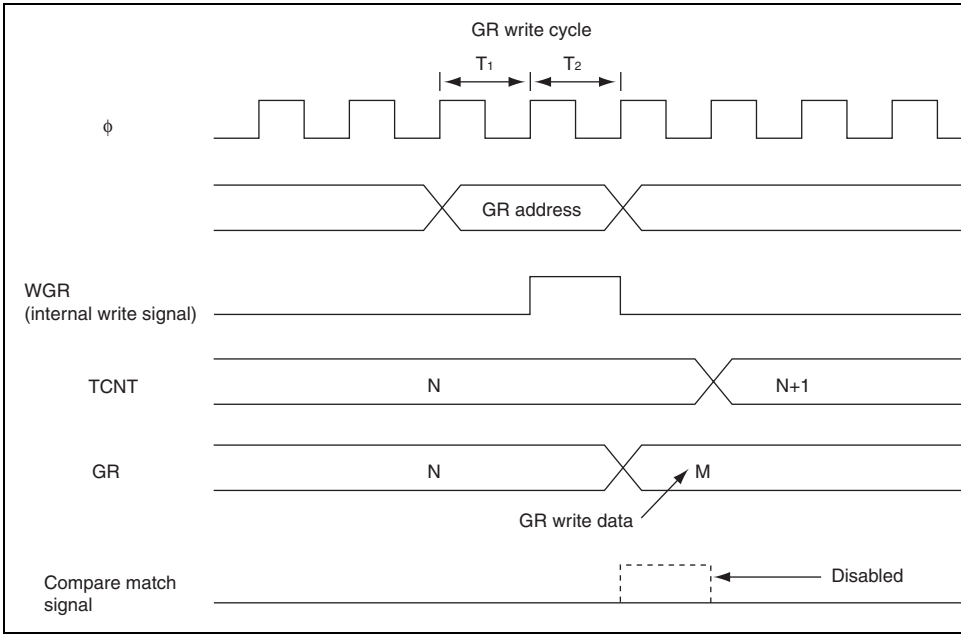


Figure 13.54 Contention between GR Write and Compare Match

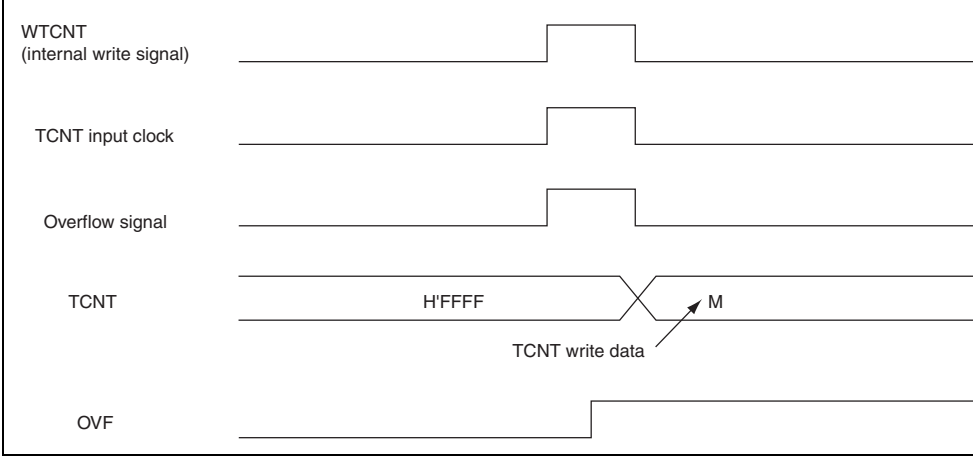


Figure 13.55 Contention between TCNT Write and Overflow

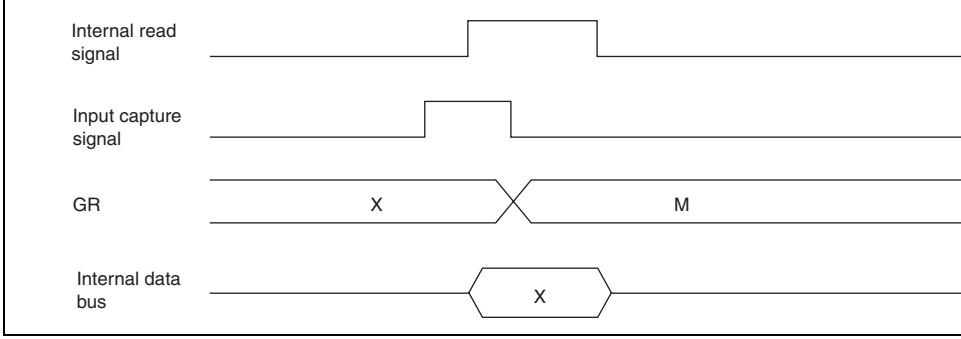


Figure 13.56 Contention between GR Read and Input Capture

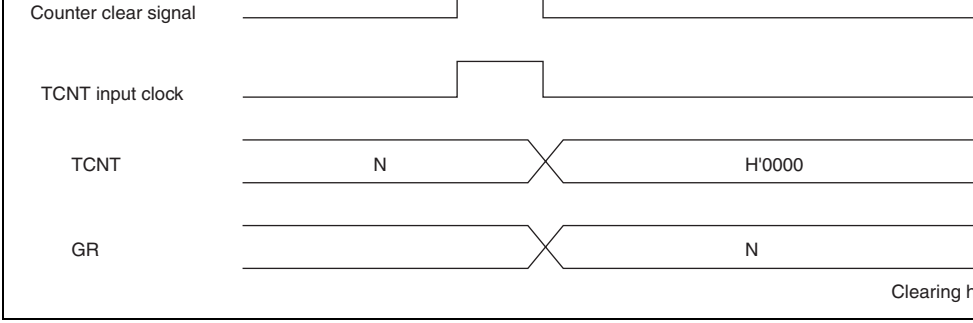


Figure 13.57 Contention between Count Clearing and Increment Operation by Input Capture

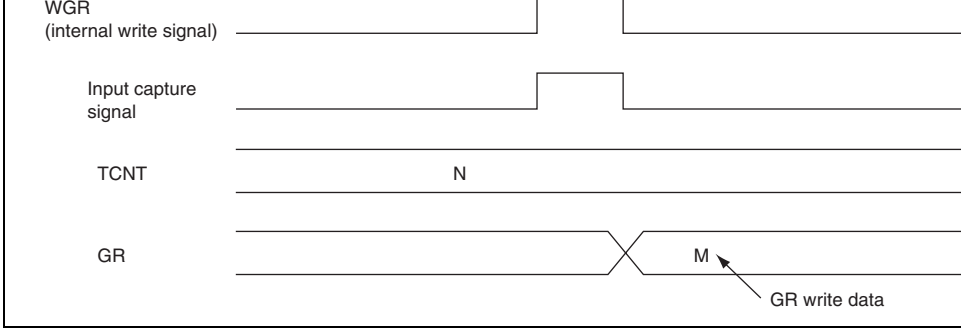


Figure 13.58 Contention between GR Write and Input Capture

8. Notes on Setting Reset Synchronous PWM Mode/Complementary PWM Mode: Wh
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
vice versa is disabled. Set reset synchronous PWM mode or complementary PW
after the normal operation (bits CMD1 and CMD0 are cleared to 0) has been set.

is to be written to while compare match is operating, stop the counter once before acc
TOCR, read the port 6 state to reflect the values of FTIOA0 to FTIOD0 and FTIOA1
FTIOD1 output, to TOA0 to TOD0 and TOA1 to TOD1, and then restart the counter.
13.59 shows an example when the compare match and the bit manipulation instructio
TOCR occur at the same timing.

TOCR has been set to H'06. Compare match B0 and compare match C0 are used. The FTIOB0 pin is in the 1 output state,
and is set to the toggle output or the 0 output by compare match B0.
When BCLR#2, @TOCR is executed to clear the TOC0 bit (the FTIOC0 signal is low) and compare match B0 occurs
at the same timing as shown below, the H'02 writing to TOCR has priority and compare match B0 does not drive the FTIOB
the FTIOB0 signal remains high.

Bit	7	6	5	4	3	2	1	0
TOCR	TOD1	TOC1	TOB1	TOA1	TOD0	TOC0	TOB0	TOA0
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

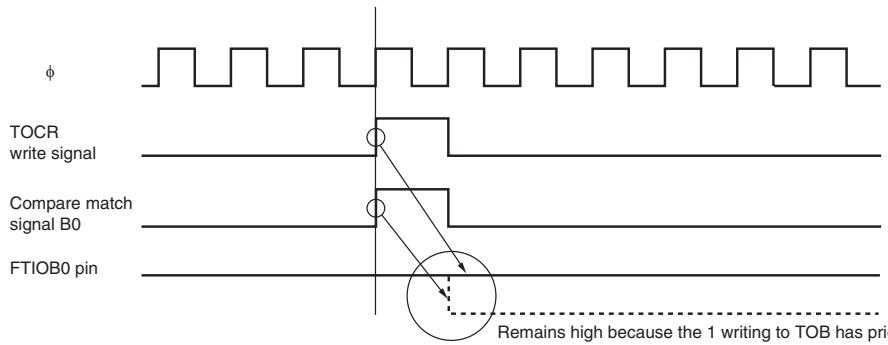


**Figure 13.59 When Compare Match and Bit Manipulation Instruction to TO
Occur at the Same Timing**

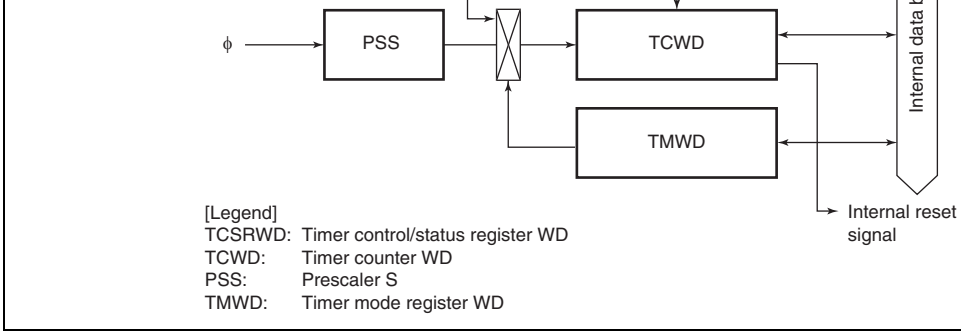


Figure 14.1 Block Diagram of Watchdog Timer

14.1 Features

- Selectable from nine counter input clocks.

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

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

  It starts operating after the reset state is canceled.

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

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

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

Bit	Bit Name	Initial Value	R/W	Description
7 to 4	—	All 1	—	Reserved These bits are always read as 1.
3	CKS3	1	R/W	Clock Select 3 to 0
2	CKS2	1	R/W	Select the clock to be input to TCWD.
1	CKS1	1	R/W	1000: Internal clock: counts on ϕ/64
0	CKS0	1	R/W	1001: Internal clock: counts on ϕ/128 1010: Internal clock: counts on ϕ/256 1011: Internal clock: counts on ϕ/512 1100: Internal clock: counts on ϕ/1024 1101: Internal clock: counts on ϕ/2048 1110: Internal clock: counts on ϕ/4096 1111: Internal clock: counts on ϕ8192 0XXX: WDT dedicated internal oscillator For the overflow periods of the WDT dedicated i oscillator, see section 22, Electrical Characteristi

[Legend] X: Don't care.

Figure 14.2 shows an example of watchdog timer operation.

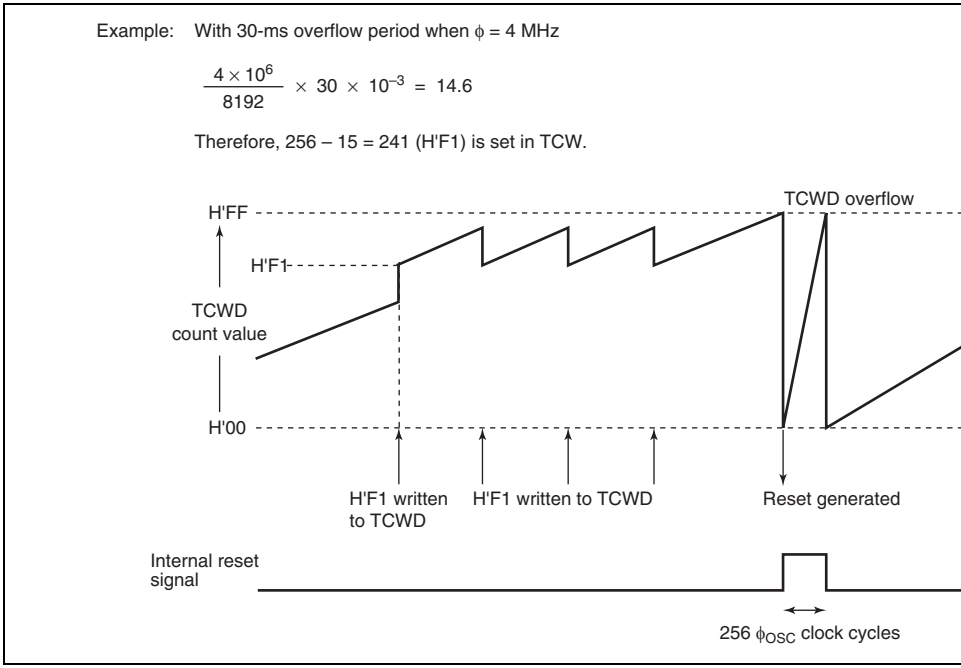


Figure 14.2 Watchdog Timer Operation Example

- Pulse division method for less ripple

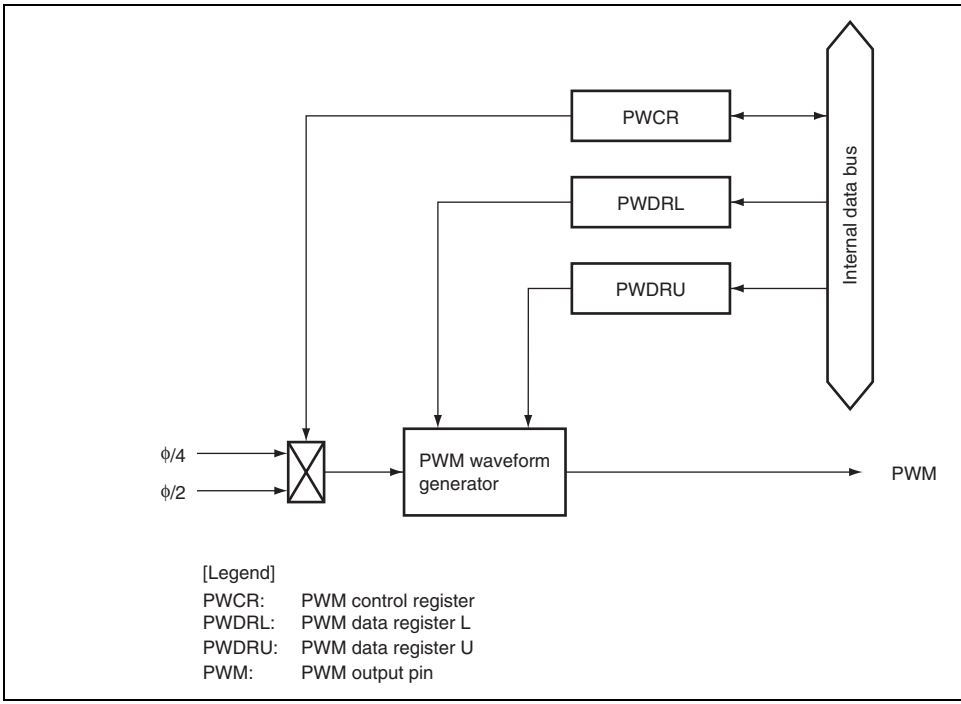


Figure 15.1 Block Diagram of 14-Bit PWM

PWCR selects the conversion period.

Bit	Bit Name	Initial Value	R/W	Description
7 to 1	—	All 1	—	Reserved These bits are always read as 1, and cannot be
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 a modulation width of $1/\phi$ 1: The input clock is $\phi/4$ ($t\phi = 4/\phi$) — The conversion period is $32768/\phi$, with a modulation width of $2/\phi$

[Legend]

$t\phi$: Period of PWM clock input

15.3.2 PWM Data Registers U and L (PWDRU, PWDRL)

PWDRU and PWDRL indicate high level width in one PWM waveform cycle. PWDRU PWDRL are 14-bit write-only registers, with the upper 6 bits assigned to PWDRU and t bits to PWDRL. When read, all bits are always read as 1.

Both PWDRU and PWDRL are accessible only in bytes. Note that the operation is not g if word access is performed. When 14-bit data is written in PWDRU and PWDRL, the d are latched in the PWM waveform generator and the PWM waveform generation data is When writing the 14-bit data, the order is as follows: PWDRL to PWDRU.

PWDRU and PWDRL are initialized to H'C000.

data is updated in synchronization with internal signals.

One conversion period consists of 64 pulses, as shown in figure 15.2. The total high-leve during this period (T_H) corresponds to the data in PWDRU and PWDRL. This relation ca 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 st When the data value is H'C000, T_H is calculated as follows:

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

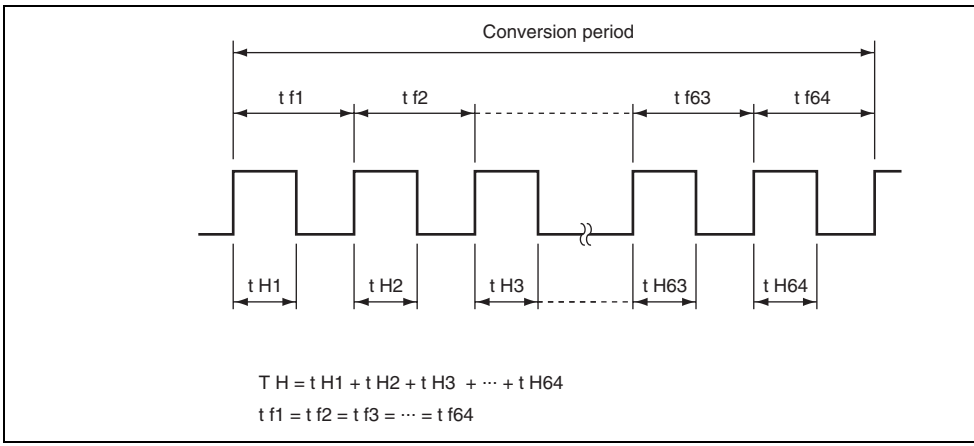


Figure 15.2 Waveform Output by 14-Bit PWM

SCI5. Since pin functions are identical for each of the two channels (SCI5 and SCI5_2), explanations are not given in this section.

16.1 Features

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

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

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

  Transmit-end, transmit-data-empty, receive-data-full, overrun error, framing error, a error.

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 c framing error

			TDR	H'FFAB
			SSR	H'FFAC
			RDR	H'FFAD
			RSR	—
			TSR	—
Channel 2	SCI3_2	SCK3_2 RXD_2 TXD_2	SMR_2	H'F740
			BRR_2	H'F741
			SCR3_2	H'F742
			TDR_2	H'F743
			SSR_2	H'F744
			RDR_2	H'F745
			RSR_2	—
			TSR_2	—

Note: * The channel 1 of the SCI3 is used in on-board programming mode by boot mo

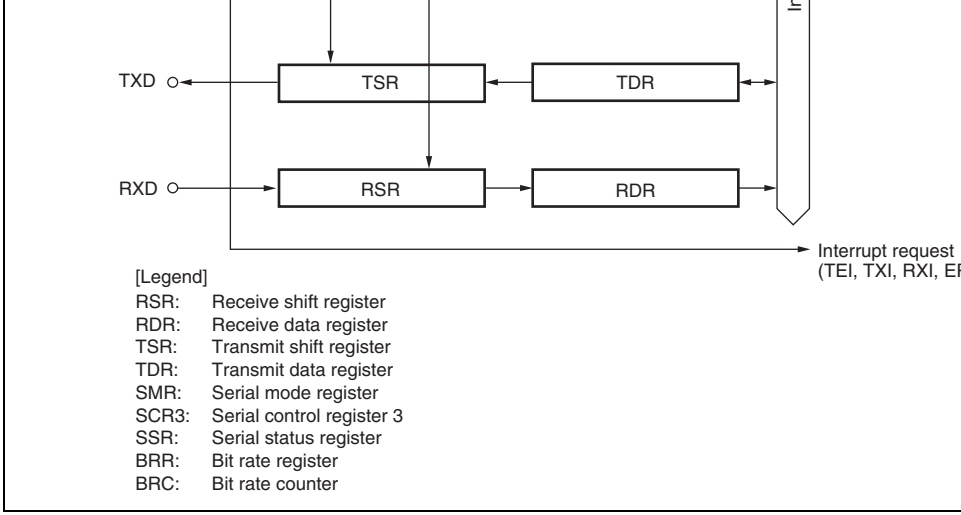


Figure 16.1 Block Diagram of SCI3

- Serial status register (SSR)
- Bit rate register (BRR)

16.3.1 Receive Shift Register (RSR)

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

16.3.2 Receive Data Register (RDR)

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

16.3.3 Transmit Shift Register (TSR)

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

16.3.5 Serial Mode Register (SMR)

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

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

				When this bit is set to 1, the multiprocessor communication function is enabled. The PE bit bit settings are invalid in multiprocessor mode. synchronous mode, clear this bit to 0.
1	CKS1	0	R/W	Clock Select 0 and 1
0	CKS0	0	R/W	These bits select the clock source for the baud generator. 00: ϕ clock (n = 0) 01: ϕ/4 clock (n = 1) 10: ϕ/16 clock (n = 2) 11: ϕ/64 clock (n = 3) For the relationship between the bit rate registe and the baud rate, see section 16.3.8, Bit Rate (BRR). n is the decimal representation of the va BRR (see section 16.3.8, Bit Rate Register (BR

6	RIE	0	R/W	Receive Interrupt Enable When this bit is set to 1, RXI and ERI interrupt re are enabled.
5	TE	0	R/W	Transmit Enable When this bit s set to 1, transmission is enabled.
4	RE	0	R/W	Receive Enable When this bit is set to 1, reception is enabled.
3	MPIE	0	R/W	Multiprocessor Interrupt Enable (enabled only wh MP bit in SMR is 1 in asynchronous mode) When this bit is set to 1, receive data in which th multiprocessor bit is 0 is skipped, and setting of t RDRF, FER, and OER status flags in SSR is dis On receiving data in which the multiprocessor bi bit is automatically cleared and normal reception resumed. For details, refer to section 16.6, Multip Communication Function.
2	TEIE	0	R/W	Transmit End Interrupt Enable When this bit is set to 1, TEI interrupt request is

Inputs a clock with a frequency 16 times the
from the SCK3 pin.

11:Reserved

- Clock synchronous mode

00: On-chip clock (SCK3 pin functions as clock

01: Reserved

10: External clock (SCK3 pin functions as clock

11: Reserved

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

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

[Asynchronous Mode]

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

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

[Clock 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	—	—

	Operating Frequency ϕ (MHz)										
	3.6864			4			4.9152				
Bit Rate (bits/s)	n	N	Error (%)	n	N	Error (%)	n	N	Error (%)	n	N
110	2	64	0.70	2	70	0.03	2	86	0.31	2	88
150	1	191	0.00	1	207	0.16	1	255	0.00	2	64
300	1	95	0.00	1	103	0.16	1	127	0.00	1	12
600	0	191	0.00	0	207	0.16	0	255	0.00	1	64
1200	0	95	0.00	0	103	0.16	0	127	0.00	0	12
2400	0	47	0.00	0	51	0.16	0	63	0.00	0	64
4800	0	23	0.00	0	25	0.16	0	31	0.00	0	32
9600	0	11	0.00	0	12	0.16	0	15	0.00	0	15
19200	0	5	0.00	0	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

	Operating Frequency ϕ (MHz)									
	8			9.8304			10			1
Bit Rate (bit/s)	n	N	Error (%)	n	N	Error (%)	n	N	Error (%)	n	N
110	2	141	0.03	2	174	–0.26	2	177	–0.25	2	212
150	2	103	0.16	2	127	0.00	2	129	0.16	2	155
300	1	207	0.16	1	255	0.00	2	64	0.16	2	77
600	1	103	0.16	1	127	0.00	1	129	0.16	1	155
1200	0	207	0.16	0	255	0.00	1	64	0.16	1	77
2400	0	103	0.16	0	127	0.00	0	129	0.16	0	155
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	10
2400	0	159	0.00	0	181	0.16	0	191	0.00	0	20
4800	0	79	0.00	0	90	0.16	0	95	0.00	0	10
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

	Operating Frequency ϕ (MHz)					
	18			20		
Bit Rate (bit/s)	n	N	Error (%)	n	N	Error (%)
110	3	79	–0.12	3	88	–0.25
150	2	233	0.16	3	64	0.16
300	2	116	0.16	2	129	0.16
600	1	233	0.16	2	64	0.16
1200	1	116	0.16	1	129	0.16
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.

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

5k	0	99	0	199	1	99	1	124	1
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.

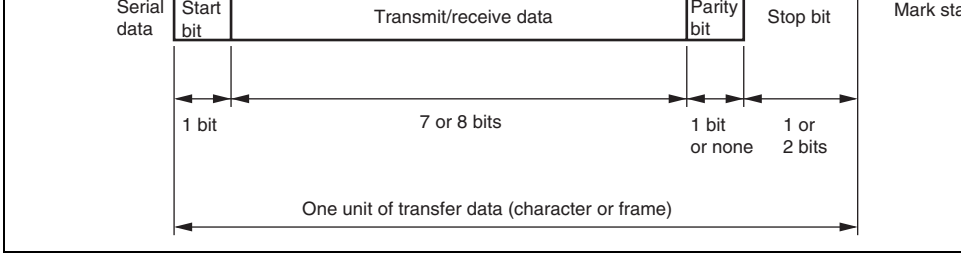


Figure 16.2 Data Format in Asynchronous Communication

16.4.1 Clock

Either an internal clock generated by the on-chip baud rate generator or an external cloc the SCK3 pin can be selected as the SCI3's serial clock, according to the setting of the C SMR and the CKE0 and CKE1 bits in SCR3. When an external clock is input at the SCI 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 frequency of the clock output in this case is equal to the bit rate, and the phase is such th rising edge of the clock is in the middle of the transmit data, as shown in figure 16.3.

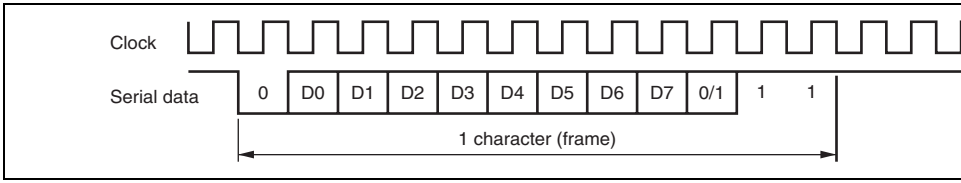
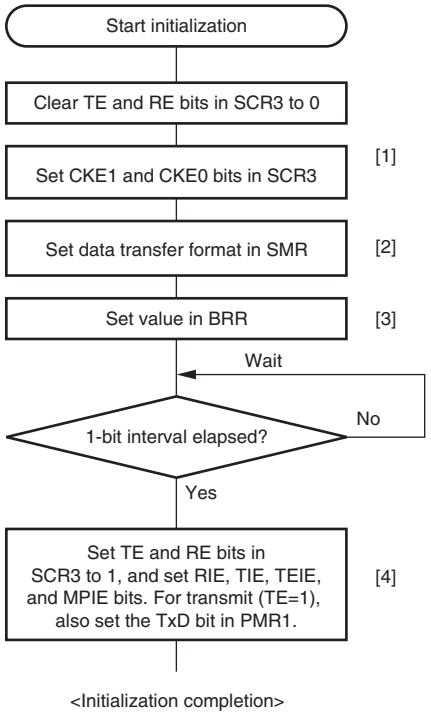


Figure 16.3 Relationship between Output Clock and Transfer Data Phas (Asynchronous Mode) (Example with 8-Bit Data, Parity, Two Stop Bits)



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

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

[2] Set the data transfer format in SMR.

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

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

Figure 16.4 Sample SCI3 Initialization Flowchart

to TDR before transmission of the current transmit data has been completed.

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, a 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 state" is entered, in which 1 is output. If the TEIE bit in SCR3 is set to 1 at this time, interrupt request is generated.
6. Figure 16.6 shows a sample flowchart for transmission in asynchronous mode.

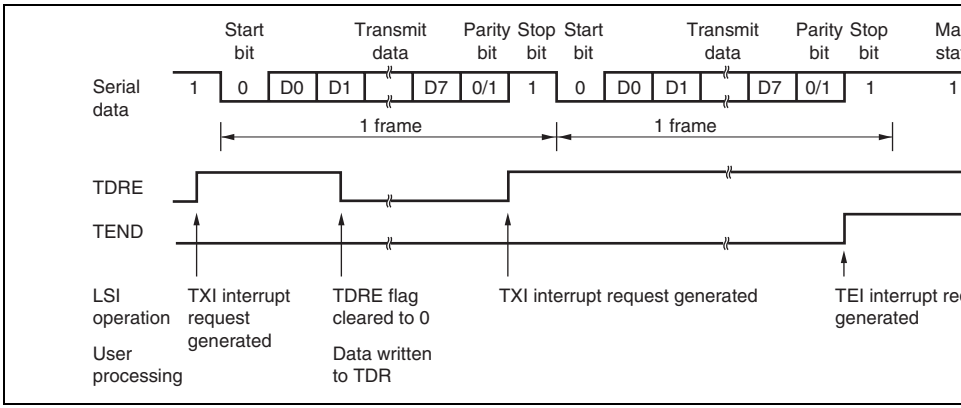
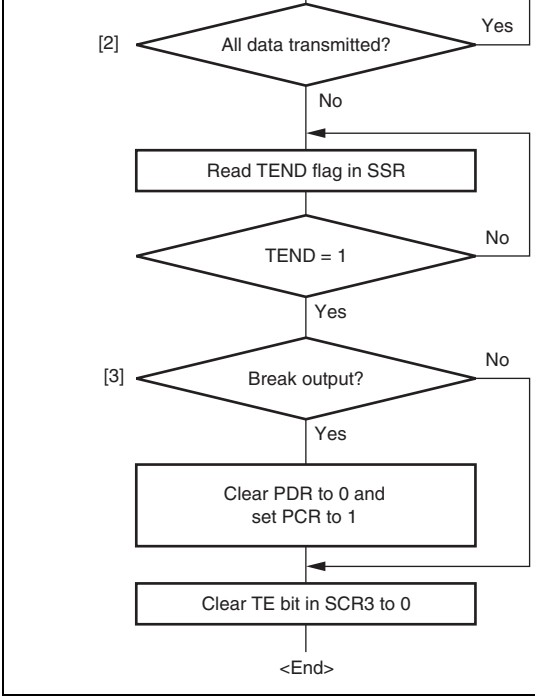


Figure 16.5 Example of SCI3 Transmission in Asynchronous Mode (8-Bit Data, Parity, One Stop Bit)



and PDR to 0, clear TxD in PMR1
to 0, then clear the TE bit in SCR3
to 0.

Figure 16.6 Sample Serial Transmission Data Flowchart (Asynchronous Mo

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

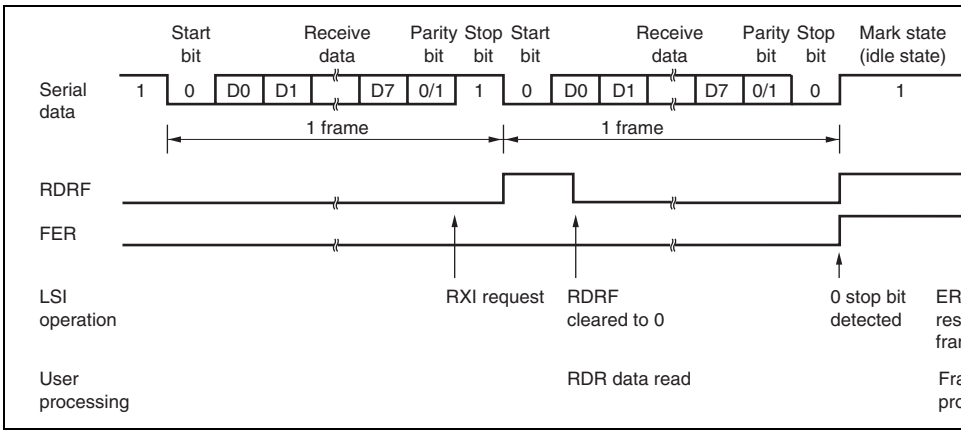
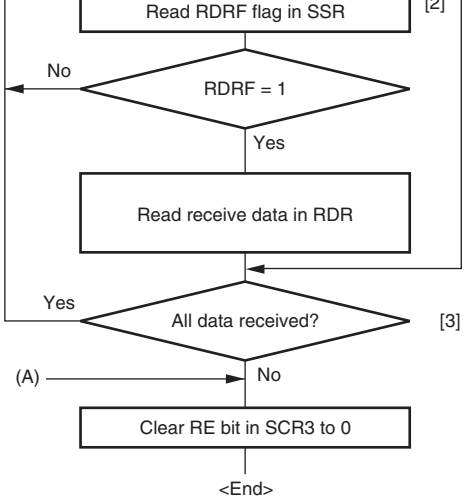


Figure 16.7 Example of SCI3 Reception in Asynchronous Mode (8-Bit Data, Parity, One Stop Bit)

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 + framin
1	1	0	1	Lost	Overrun error + parity
0	0	1	1	Transferred to RDR	Framing error + parity
1	1	1	1	Lost	Overrun error + framin parity error

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



the error. After performing the
appropriate error processing, e
that the OER, PER, and FER fl
all cleared to 0. Reception can
resumed if any of these flags a
1. In the case of a framing erro
break can be detected by readi
value of the input port correspo
the RxD pin.

Figure 16.8 Sample Serial Reception Data Flowchart (Asynchronous Mode)

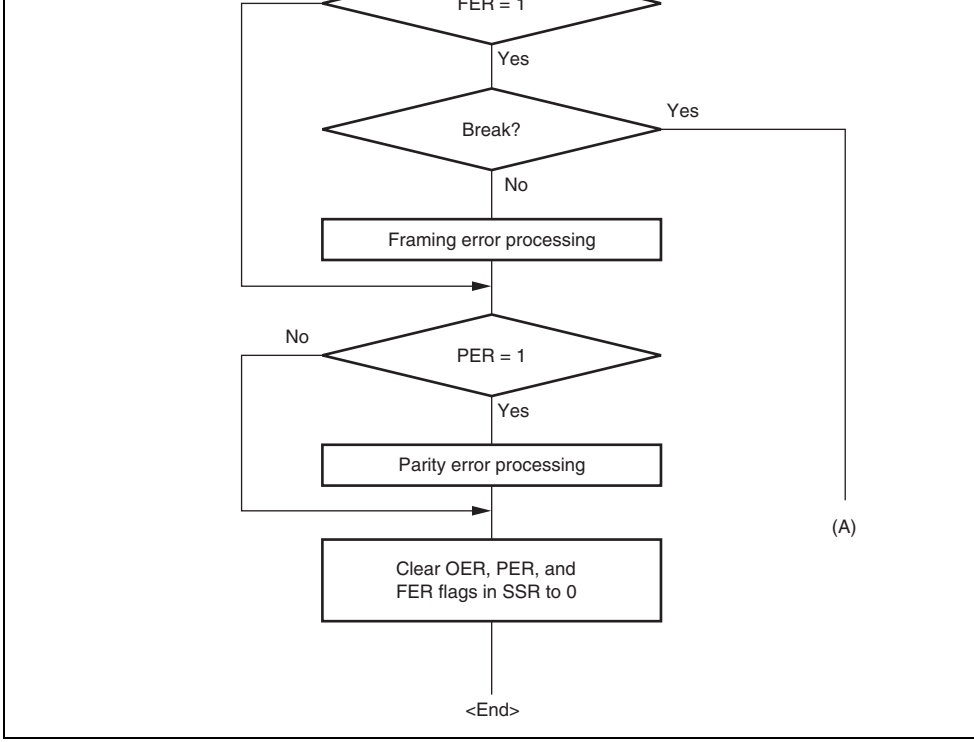


Figure 16.8 Sample Serial Reception Data Flowchart (Asynchronous Mode)

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

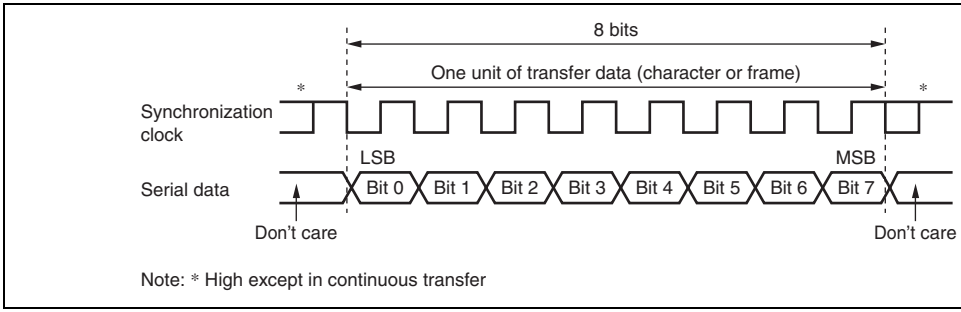


Figure 16.9 Data Format in Clock Synchronous Communication

16.5.1 Clock

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

16.5.2 SCI3 Initialization

Before transmitting and receiving data, the SCI3 should be initialized as described in a s flowchart in figure 16.4.

mode has been specified, and synchronized with the input clock when use of an exter
has been specified. Serial data is transmitted sequentially from the LSB (bit 0), from t
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 trans
of the next frame is started.
6. If the TDRE flag is set to 1, the TEND flag in SSR is set to 1, and the TDRE flag ma
the output state of the last bit. If the TEIE bit in SCR3 is set to 1 at this time, a TEI i
request is generated.
7. The SCK3 pin is fixed high at the end of transmission.

Figure 16.11 shows a sample flow chart for serial data transmission. Even if the TDRE fl
cleared to 0, transmission will not start while a receive error flag (OER, FER, or PER) is
Make sure that the receive error flags are cleared to 0 before starting transmission.

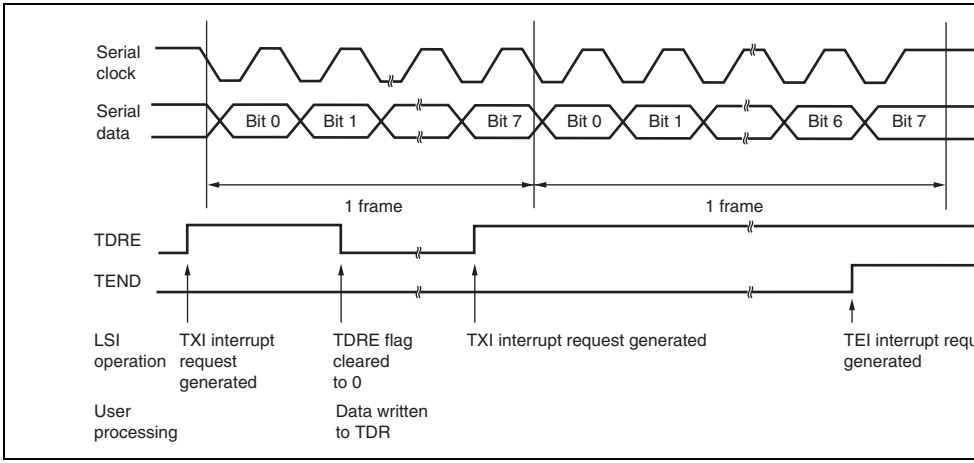


Figure 16.10 Example of SCI3 Transmission in Clock Synchronous Mode

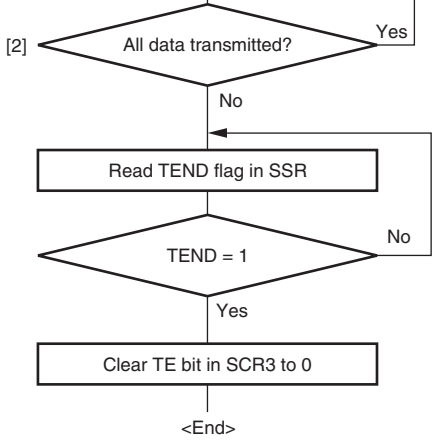


Figure 16.11 Sample Serial Transmission Flowchart (Clock Synchronous M

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 da transferred to RDR. If the RIE bit in SCR3 is set to 1 at this time, an RXI interrupt rec generated.

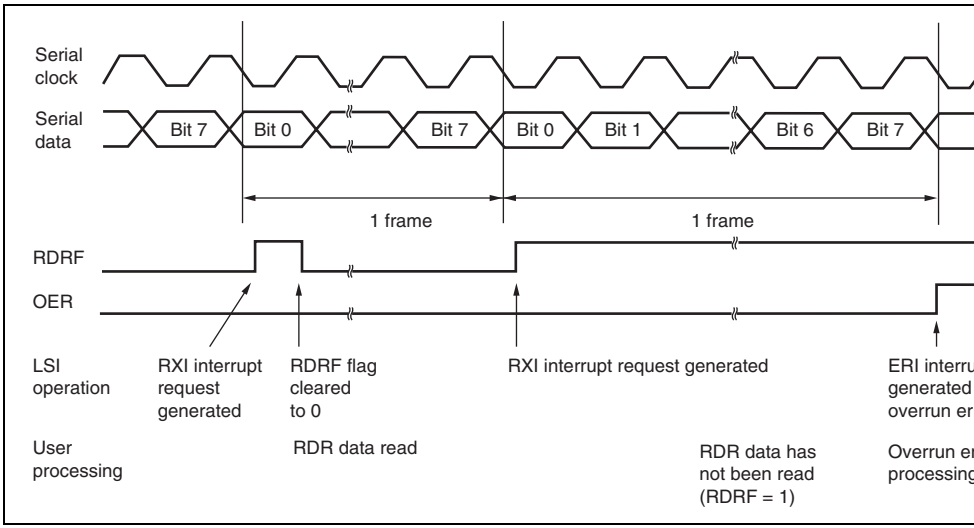
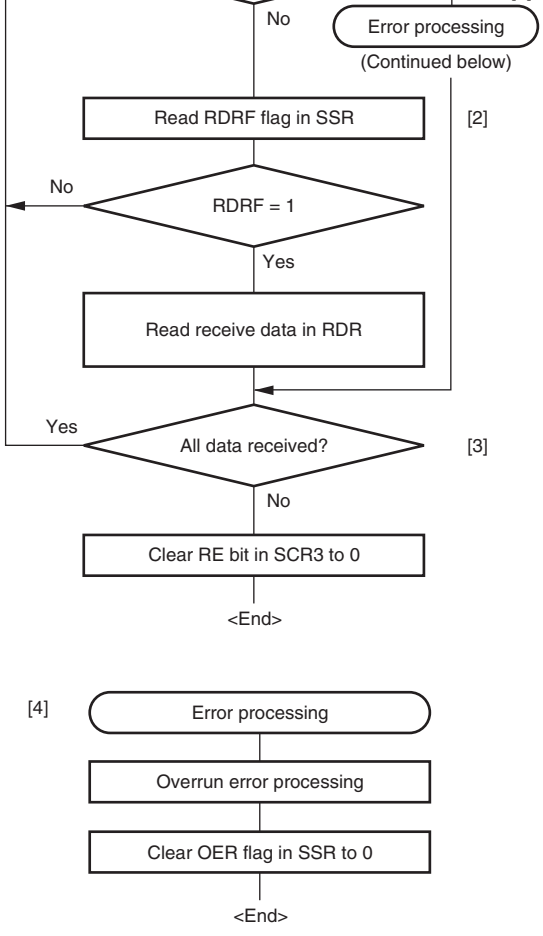


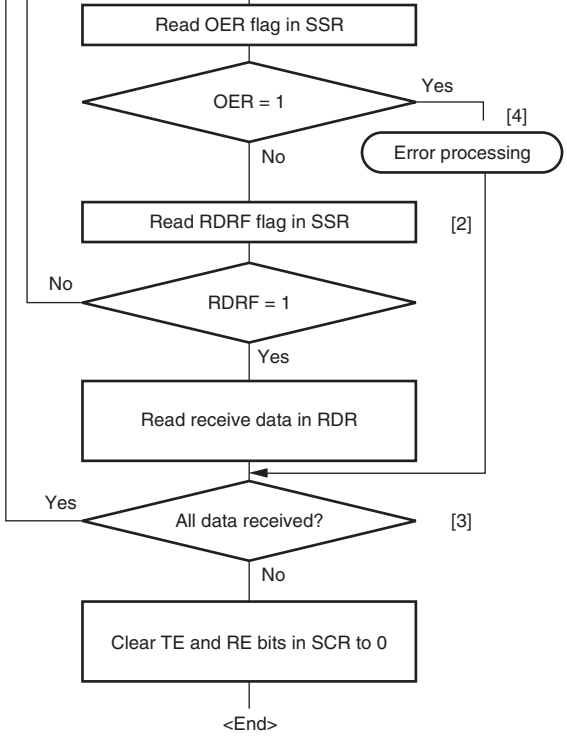
Figure 16.12 Example of SCI3 Reception in Clock Synchronous Mode



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

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

Figure 16.13 Sample Serial Reception Flowchart (Clock Synchronous Mod



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

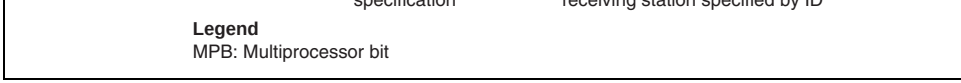
[4] If an overrun error occurs, read
OER flag in SSR, and after
performing the appropriate erro
processing, clear the OER flag
Transmission/reception cannot
resumed if the OER flag is set
For overrun error processing, s
figure 16.13.

**Figure 16.14 Sample Flowchart of Simultaneous Serial Transmit and Receive Op
(Clock Synchronous Mode)**

cycle is a data transmission cycle. Figure 16.15 shows an example of inter-processor
communication using the multiprocessor format. The transmitting station first sends the I
of the receiving station with which it wants to perform serial communication as data with
multiprocessor bit added. It then sends transmit data as data with a 0 multiprocessor bit a
When data with a 1 multiprocessor bit is received, the receiving station compares that dat
own ID. The station whose ID matches then receives the data sent next. Stations whose ID
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 s
transfer of receive data from RSR to RDR, error flag detection, and setting the SSR status
RDRF, FER, and OER, to 1, are inhibited until data with a 1 multiprocessor bit is receive
reception of a receive character with a 1 multiprocessor bit, the MPBR bit in SSR is set to
the MPIE bit is automatically cleared, thus normal reception is resumed. If the RIE bit in
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
settings are the same as those in normal asynchronous mode. The clock used for multipro
communication is the same as that in normal asynchronous mode.

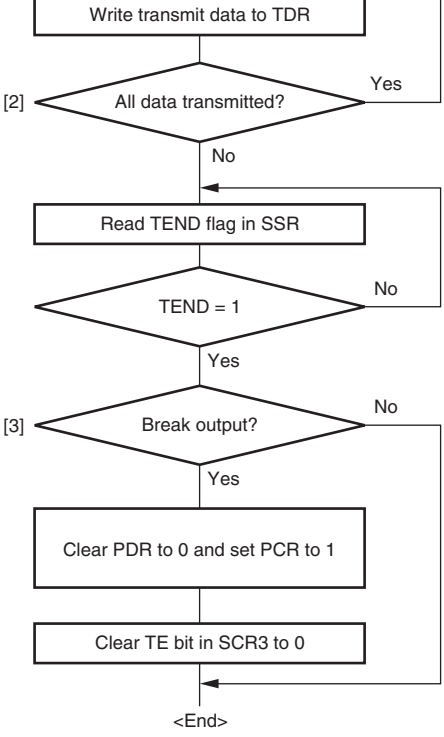


Legend
MPB: Multiprocessor bit

Figure 16.15 Example of Inter-Processor Communication Using Multiprocessor (Transmission of Data H'AA to Receiving Station A)

16.6.1 Multiprocessor Serial Data Transmission

Figure 16.16 shows a sample flowchart for multiprocessor serial data transmission. For a transmission cycle, set the MPBT bit in SSR to 1 before transmission. For a data transm cycle, clear the MPBT bit in SSR to 0 before transmission. All other SCI3 operations ar as those in asynchronous mode.



[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 16.16 Sample Multiprocessor Serial Transmission Flowchart

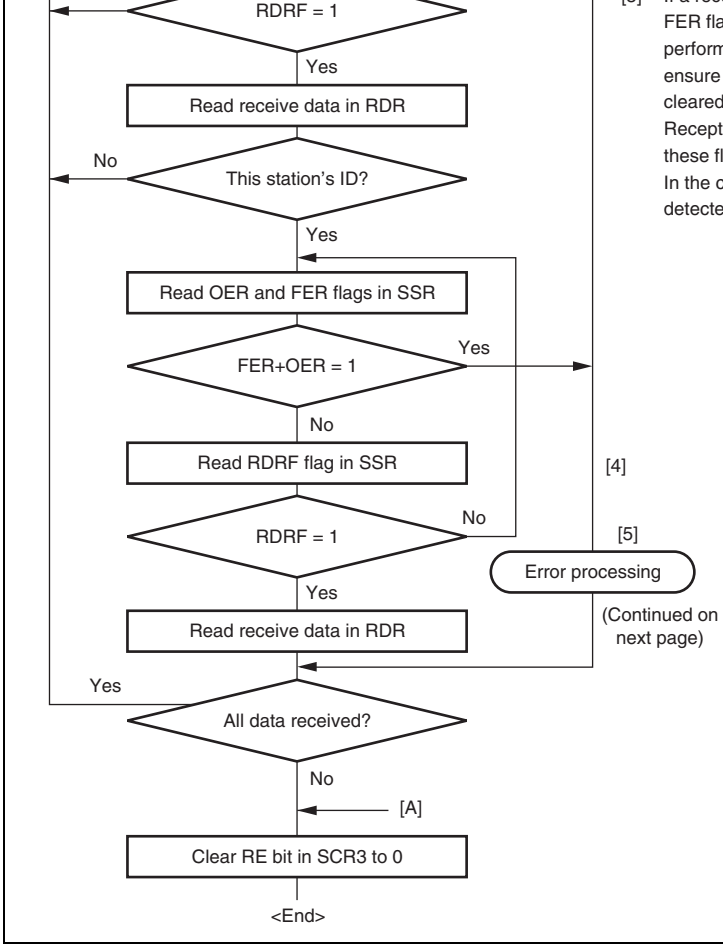


Figure 16.17 Sample Multiprocessor Serial Reception Flowchart (1)

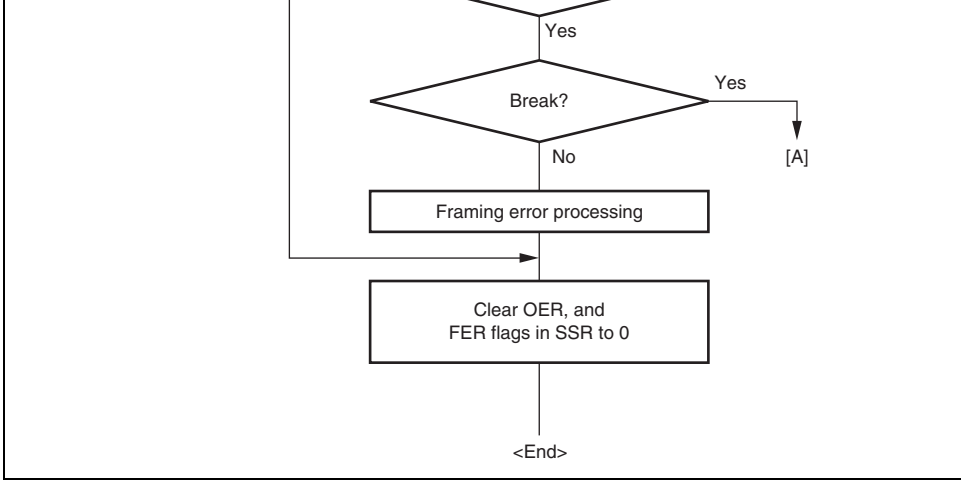


Figure 16.17 Sample Multiprocessor Serial Reception Flowchart (2)

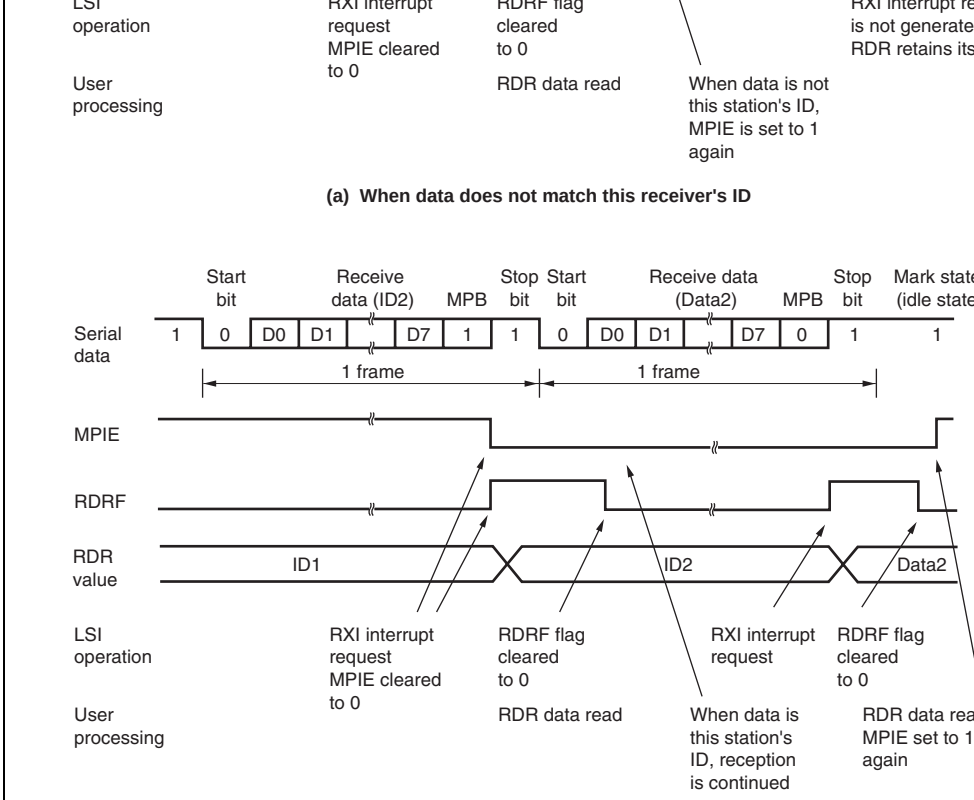


Figure 16.18 Example of SCI3 Reception Using Multiprocessor Format (Example with 8-Bit Data, Multiprocessor Bit, One Stop Bit)

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

When the TXD or TXD2 bit in PMR1 is 1, the TxD pin is used as an I/O port whose direc (input or output) and level are determined by PCR and PDR. This can be used to set the T to mark state (high level) or send a break during serial data transmission. To maintain the communication line at mark state until TE is set to 1, set both PCR and PDR to 1, and the TXD bit to 1. At this point, the TxD pin becomes an I/O port, and 1 is output from the Tx To send a break during serial transmission, first set PCR to 1 and clear PDR to 0, and then TXD bit to 1. At this point, the TxD pin becomes an I/O port regardless of the current transmission state, and 0 is output from the TxD pin.

16.8.3 Receive Error Flags and Transmit Operations (Clock Synchronous Mode (

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

[Legend]

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

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

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

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

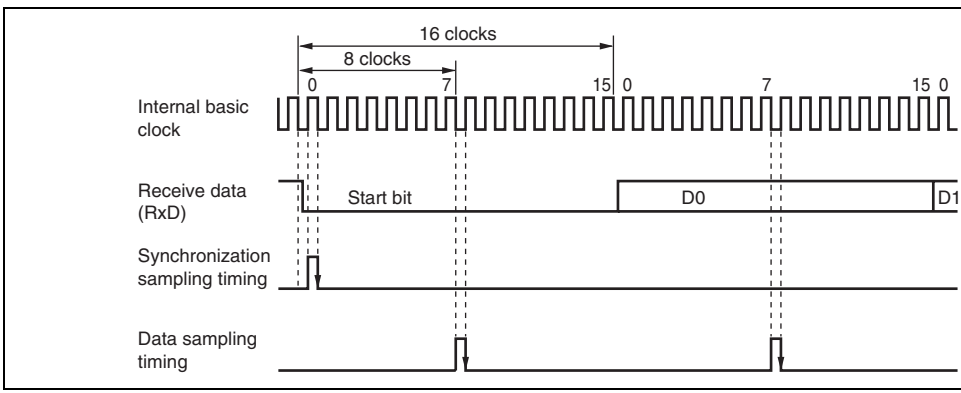


Figure 16.19 Receive Data Sampling Timing in Asynchronous Mode

17.1 Features

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

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

I^2C bus format

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

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

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

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

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

Clock synchronous format

- Four interrupt sources

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

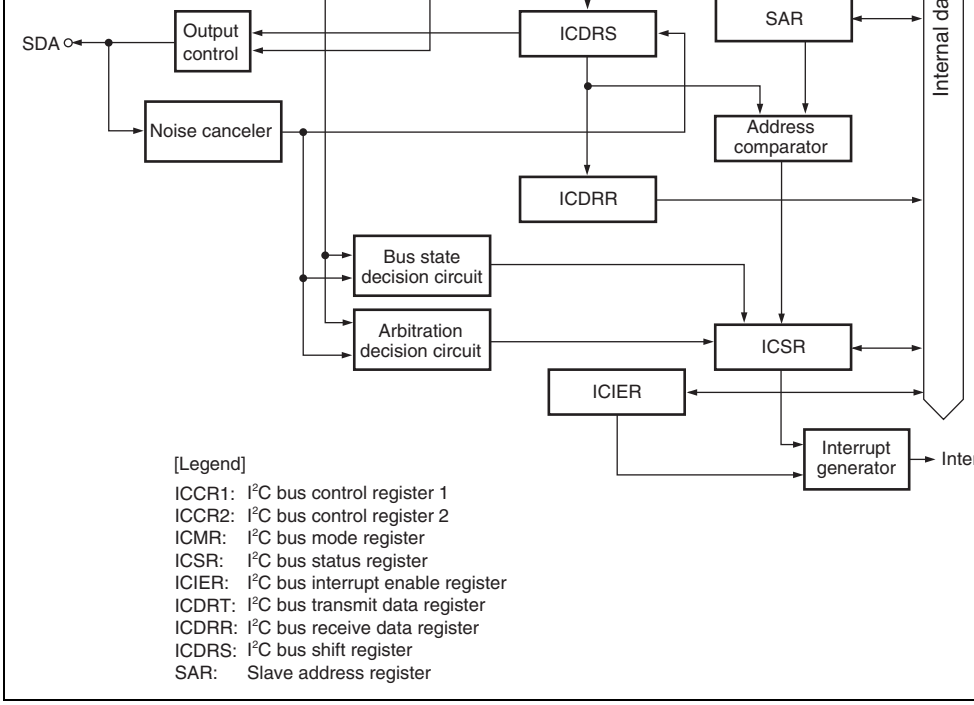


Figure 17.1 Block Diagram of I²C Bus Interface 2

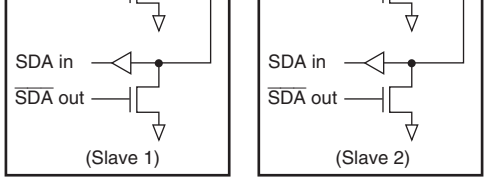


Figure 17.2 External Circuit Connections of I/O Pins

17.2 Input/Output Pins

Table 17.1 summarizes the input/output pins used by the I²C bus interface 2.

Table 17.1 I²C Bus Interface Pins

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)

17.3.1 I²C Bus Control Register 1 (ICCR1)

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

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

				bit is 1, TRS is automatically set to 1. If an ove occurs in master mode with the clock synchron format, MST is cleared to 0 and slave receive m entered. Operating modes are described below accordin and TRS combination. When clock synchronou format is selected and MST is 1, clock is output 00: Slave receive mode 01: Slave transmit mode 10: Master receive mode 11: Master transmit mode
3	CKS3	0	R/W	Transfer Clock Select 3 to 0
2	CKS2	0	R/W	These bits should be set according to the neces transfer rate (see table 17.2) in master mode. I mode, these bits are used for reservation of the time in transmit mode. The time is 10 t_{cyc} when and 20 t_{cyc} when CKS3 = 1.
1	CKS1	0	R/W	
0	CKS0	0	R/W	

		1	0	φ/112	44.6 kHz	71.4 kHz	89.3 kHz	143 kHz	17
			1	φ/128	39.1 kHz	62.5 kHz	78.1 kHz	125 kHz	15
1	0	0	0	φ/56	89.3 kHz	143 kHz	179 kHz	286 kHz	35
			1	φ/80	62.5 kHz	100 kHz	125 kHz	200 kHz	25
		1	0	φ/96	52.1 kHz	83.3 kHz	104 kHz	167 kHz	20
			1	φ/128	39.1 kHz	62.5 kHz	78.1 kHz	125 kHz	15
	1	0	0	φ/160	31.3 kHz	50.0 kHz	62.5 kHz	100 kHz	12
			1	φ/200	25.0 kHz	40.0 kHz	50.0 kHz	80.0 kHz	10
		1	0	φ/224	22.3 kHz	35.7 kHz	44.6 kHz	71.4 kHz	89
			1	φ/256	19.5 kHz	31.3 kHz	39.1 kHz	62.5 kHz	78

				this bit has no meaning. With the I^2C bus forma set to 1 when the SDA level changes from high under the condition of SCL = high, assuming th condition has been issued. This bit is cleared to the SDA level changes from low to high under t condition of SCL = high, assuming that the stop has been issued. Write 1 to BBSY and 0 to SCI a start condition. Follow this procedure when al transmitting a start condition. Write 0 in BBSY a SCP to issue a stop condition. To issue start/st conditions, use the MOV instruction.
6	SCP	1	W	Start/Stop Issue Condition Disable The SCP bit controls the issue of start/stop con master mode. To issue a start condition, write 1 in BBSY and A retransmit start condition is issued in the sam issue a stop condition, write 0 in BBSY and 0 in This bit is always read as 1. If 1 is written, the c stored.
5	SDAO	1	R/W	SDA Output Value Control This bit is used with SDAOP when modifying ou of SDA. This bit should not be manipulated duri transfer. 0: When reading, SDA pin outputs low. When writing, SDA pin is changed to output 1: When reading, SDA pin outputs high. When writing, SDA pin is changed to output (outputs high by external pull-up resistance).

				This bit is always read as 1, and cannot be modi
1	IICRST	0	R/W	IIC Control Part Reset This bit resets the control part except for I²C regi this bit is set to 1 when hang-up occurs because communication failure during I²C operation, I²C c part can be reset without setting ports and initiali registers.
0	—	1	—	Reserved This bit is always read as 1, and cannot be modi

17.3.3 I²C Bus Mode Register (ICMR)

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

Bit	Bit Name	Initial Value	R/W	Description
7	MLS	0	R/W	MSB-First/LSB-First Select 0: MSB-first 1: LSB-first Set this bit to 0 when the I²C bus format is used.

5, 4	—	All 1	—	Reserved These bits are always read as 1, and cannot be
3	BCWP	1	R/W	BC Write Protect This bit controls the BC2 to BC0 modifications. modifying BC2 to BC0, this bit should be cleare use the MOV instruction. In clock synchronous mode, BC should not be modified. 0: When writing, values of BC2 to BC0 are set. 1: When reading, 1 is always read. When writing, settings of BC2 to BC0 are inv

synchronous serial format, these bits should not
modified.

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

17.3.4 I^2C Bus Interrupt Enable Register (ICIER)

ICIER enables or disables interrupt sources and acknowledge bits, sets acknowledge bits
transferred, and confirms acknowledge bits to be received.

Bit	Bit Name	Initial Value	R/W	Description
7	TIE	0	R/W	Transmit Interrupt Enable When the TDRE bit in ICSR is set to 1, this bit e disables the transmit data empty interrupt (TXI). 0: Transmit data empty interrupt request (TXI) is 1: Transmit data empty interrupt request (TXI) is

				This bit enables or disables the receive data ful request (RXI) and the overrun error interrupt re (ERI) with the clock synchronous format, when data is transferred from ICDRS to ICDRR and t bit in ICSR is set to 1. RXI can be canceled by the RDRF or RIE bit to 0. 0: Receive data full interrupt request (RXI) and error interrupt request (ERI) with the clock synchronous format are disabled. 1: Receive data full interrupt request (RXI) and error interrupt request (ERI) with the clock synchronous format are enabled.
4	NAKIE	0	R/W	NACK Receive Interrupt Enable This bit enables or disables the NACK receive i request (NAKI) and the overrun error (setting of bit in ICSR) interrupt request (ERI) with the cloc synchronous format, when the NACKF and AL ICSR are set to 1. NAKI can be canceled by cle NACKF, OVE, or NAKIE bit to 0. 0: NACK receive interrupt request (NAKI) is dis 1: NACK receive interrupt request (NAKI) is ena
3	STIE	0	R/W	Stop Condition Detection Interrupt Enable 0: Stop condition detection interrupt request (ST disabled. 1: Stop condition detection interrupt request (ST enabled.

				0: Receive acknowledge = 0 1: Receive acknowledge = 1
0	ACKBT	0	R/W	Transmit Acknowledge In receive mode, this bit specifies the bit to be se acknowledge timing. 0: 0 is sent at the acknowledge timing. 1: 1 is sent at the acknowledge timing.

17.3.5 I²C Bus Status Register (ICSR)

ICSR performs confirmation of interrupt request flags and status.

Bit	Bit Name	Initial Value	R/W	Description
7	TDRE	0	R/W	Transmit Data Register Empty [Setting conditions] • When data is transferred from ICDRT to ICD ICDRT becomes empty • When TRS is set • When a start condition (including re-transfer) been issued • When transmit mode is entered from receive slave mode [Clearing conditions] • When 0 is written in TDRE after reading TDF • When data is written to ICDRT with an instru

				• When data is written to ICDRT with an instr
5	RDRF	0	R/W	Receive Data Register Full [Setting condition] • When a receive data is transferred from ICD ICDRR [Clearing conditions] • When 0 is written in RDRF after reading RD • When ICDRR is read with an instruction
4	NACKF	0	R/W	No Acknowledge Detection Flag [Setting condition] • When no acknowledge is detected from the device in transmission while the ACKE bit in 1 [Clearing condition] • When 0 is written in NACKF after reading N

				— The first byte in the slave address matche address set in the SAR [Clearing condition] • When 0 is written in STOP after reading STO
2	AL/OVE	0	R/W	Arbitration Lost Flag/Overrun Error Flag This flag indicates that arbitration was lost in ma mode with the I^2C bus format and that the final b been received while RDRF = 1 with the clock synchronous format. When two or more master devices attempt to sei bus at nearly the same time, if the I^2C bus interfa detects data differing from the data it sent, it sets to indicate that the bus has been taken by anoth master. [Setting conditions] • If the internal SDA and SDA pin disagree at t SCL in master transmit mode • When the SDA pin outputs high in master mo 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

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

17.3.6 Slave Address Register (SAR)

SAR selects the communication format and sets the slave address. When the chip is in s with the I²C bus format, if the upper 7 bits of SAR match the upper 7 bits of the first fra 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 differing form the addresses of other slave devi connected to the I²C bus.
0	FS	0	R/W	Format Select 0: I²C bus format is selected. 1: Clock 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. ICDI receive-only register, therefore the CPU cannot write to this register. The initial value of is H'FF.

17.3.9 I²C Bus Shift Register (ICDRS)

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

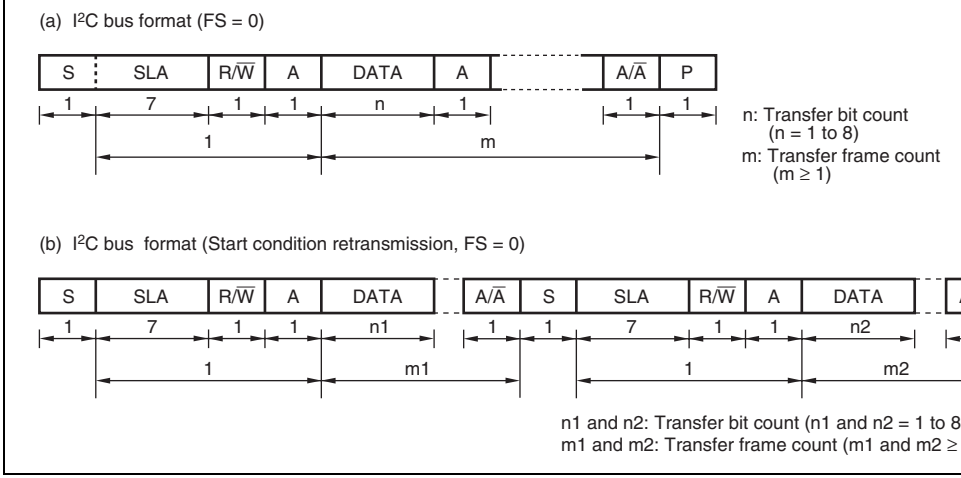


Figure 17.3 I²C Bus Formats

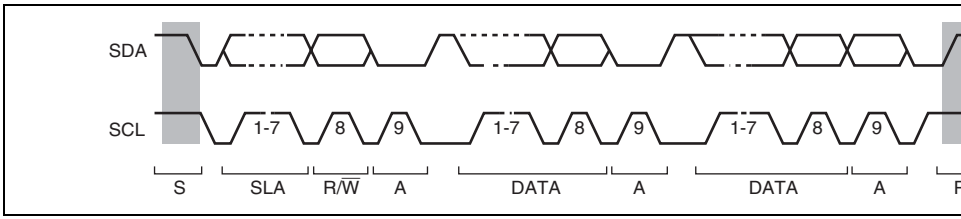


Figure 17.4 I²C Bus Timing

In master transmit mode, the master device outputs the transmit clock and transmit data, a slave device returns an acknowledge signal. For master transmit mode operation timing, r figures 17.5 and 17.6. The transmission procedure and operations in master transmit mod described below.

1. Set the ICE bit in ICCR1 to 1. Set the MLS and WAIT bits in ICMR and the CKS3 to 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 b ICCR1 to select master transmit mode. Then, write 1 to BBSY and 0 to SCP using M instruction. (Start condition issued) This generates the start condition.
3. After confirming that TDRE in ICSR has been set, write the transmit data (the first by show the slave address and R/$\overline{W}$) to ICDRT. At this time, TDRE is automatically clea 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 at the rise of the 9th transmit clock pulse. Read the ACKBR bit in ICIER, and confirm slave device has been selected. Then, write second byte data to ICDRT. When ACKB the slave device has not been acknowledged, so issue the stop condition. To issue the condition, write 0 to BBSY and SCP using MOV instruction. SCL is fixed low until t 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 en byte data transmission) while TDRE is 1, or wait for NACK (NACKF in ICSR = 1) fr receive device while ACKE in ICIER is 1. Then, issue the stop condition to clear TEN NACKF.
7. When the STOP bit in ICSR is set to 1, the operation returns to the slave receive mod

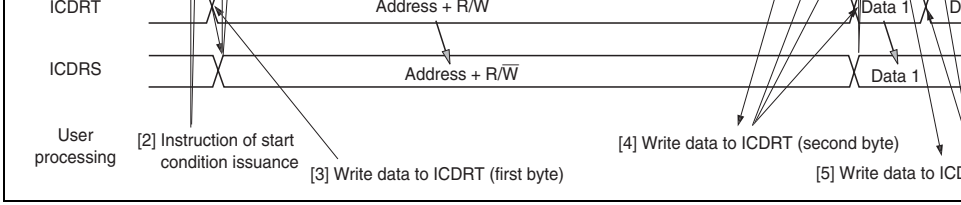


Figure 17.5 Master Transmit Mode Operation Timing (1)

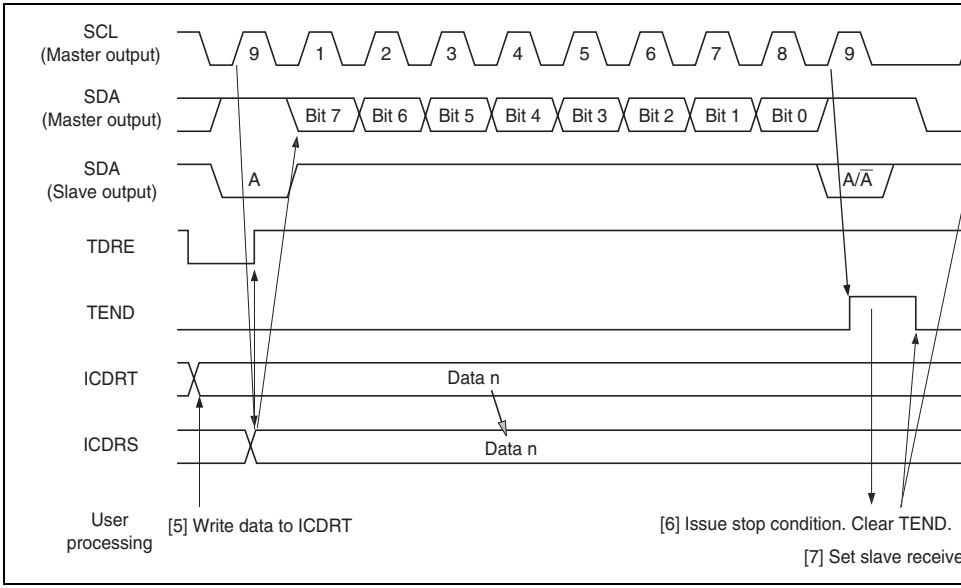


Figure 17.6 Master Transmit Mode Operation Timing (2)

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

3. After the reception of first frame data is completed, the RDRF bit in ICST is set to 1 a of 9th receive clock pulse. At this time, the receive data is read by reading ICDRR, an is cleared to 0.
4. The continuous reception is performed by reading ICDRR every time RDRF is set. If 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 I 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 co
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.

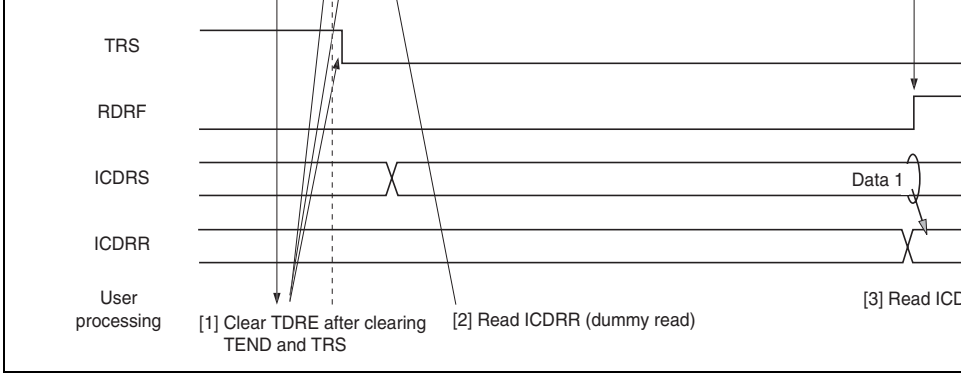


Figure 17.7 Master Receive Mode Operation Timing (1)

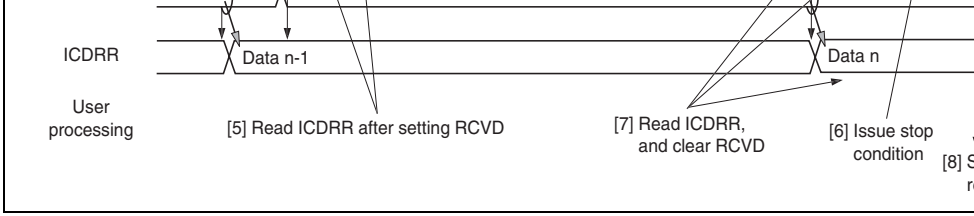


Figure 17.8 Master Receive Mode Operation Timing (2)

17.4.4 Slave Transmit Operation

In slave transmit mode, the slave device outputs the transmit data, while the master devic the receive clock and returns an acknowledge signal. For slave transmit mode operation refer to figures 17.9 and 17.10.

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 bits in ICCR1 to 1. (Initial setting) Set the MST and TRS bits in ICCR1 to select slav mode, and wait until the slave address matches.
2. When the slave address matches in the first frame following detection of the start con the slave device outputs the level specified by ACKBT in ICIER to SDA, at the rise o clock pulse. At this time, if the 8th bit data (R/$\overline{W}$) is 1, the TRS and ICSR bits in ICC 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 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.

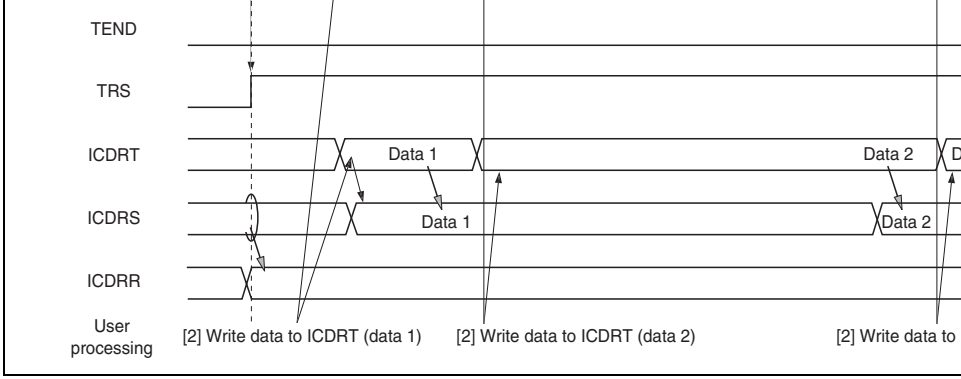


Figure 17.9 Slave Transmit Mode Operation Timing (1)

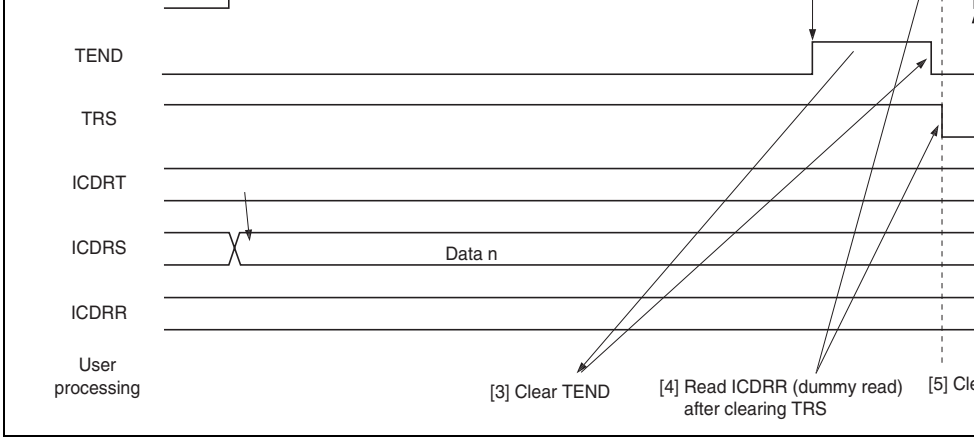


Figure 17.10 Slave Transmit Mode Operation Timing (2)

17.4.5 Slave Receive Operation

In slave receive mode, the master device outputs the transmit clock and transmit data, and slave device returns an acknowledge signal. For slave receive mode operation timing, ref figures 17.11 and 17.12. The reception procedure and operations in slave receive mode ar described below.

1. Set the ICE bit in ICCR1 to 1. Set the MLS and WAIT bits in ICMR and the CKS3 to bits in ICCR1 to 1. (Initial setting) Set the MST and TRS bits in ICCR1 to select slave mode, and wait until the slave address matches.
2. When the slave address matches in the first frame following detection of the start con the slave device outputs the level specified by ACKBT in ICIER to SDA, at the rise o clock pulse. At the same time, RDRF in ICSR is set to read ICDRR (dummy read). (S read data show the slave address and R/$\overline{W}$, it is not used.)

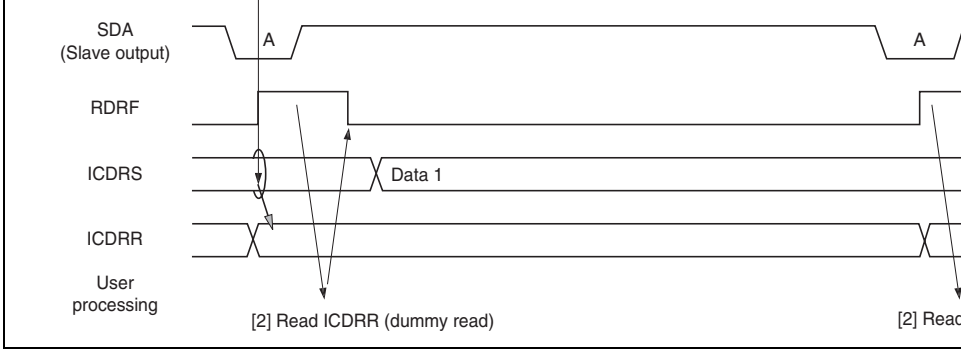


Figure 17.11 Slave Receive Mode Operation Timing (1)

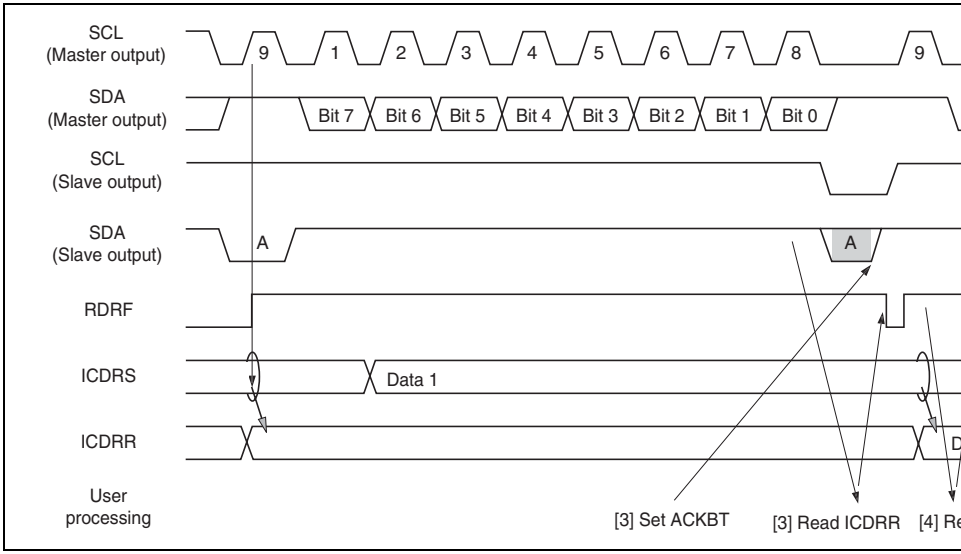


Figure 17.12 Slave Receive Mode Operation Timing (2)

MSB first or LSB first. The output level of SDA can be changed during the transfer wait, SDAO bit in ICCR2.

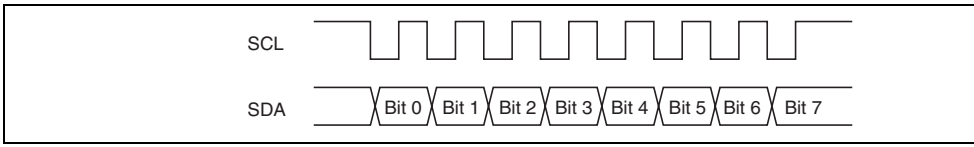


Figure 17.13 Clock Synchronous Serial Transfer Format

transferred from ICDRT to ICDRS, and TDRE is set automatically. The continuous transmission is performed by writing data to ICDRT every time TDRE is set. When from transmit mode to receive mode, clear TRS while TDRE is 1.

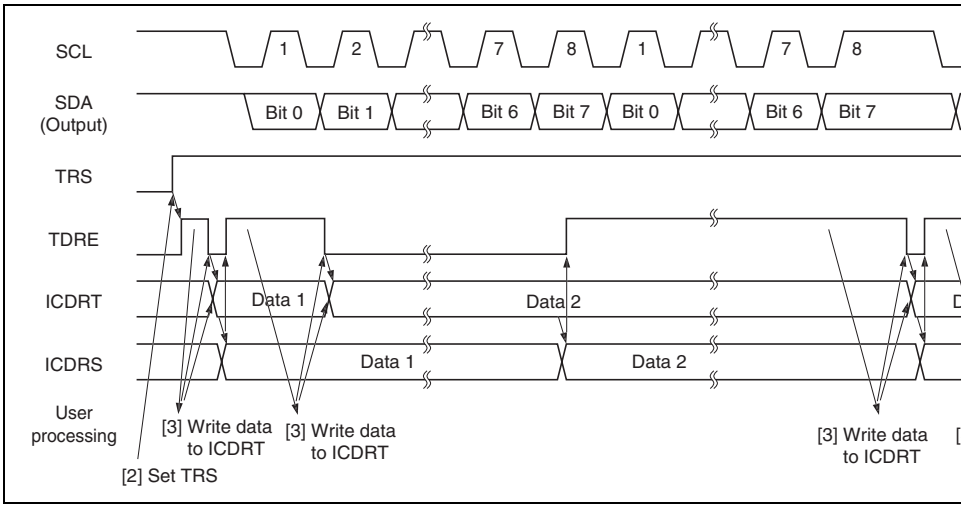


Figure 17.14 Transmit Mode Operation Timing

continually output. The continuous reception is performed by reading ICDRR every ti
RDRF is set. When the 8th clock is risen while RDRF is 1, the overrun is detected and
AL/OVE in ICSR is set. At this time, the previous reception data is retained in ICDRR

4. To stop receiving when MST = 1, set RCVD in ICCR1 to 1, then read ICDRR. Then,
fixed high after receiving the next byte data.

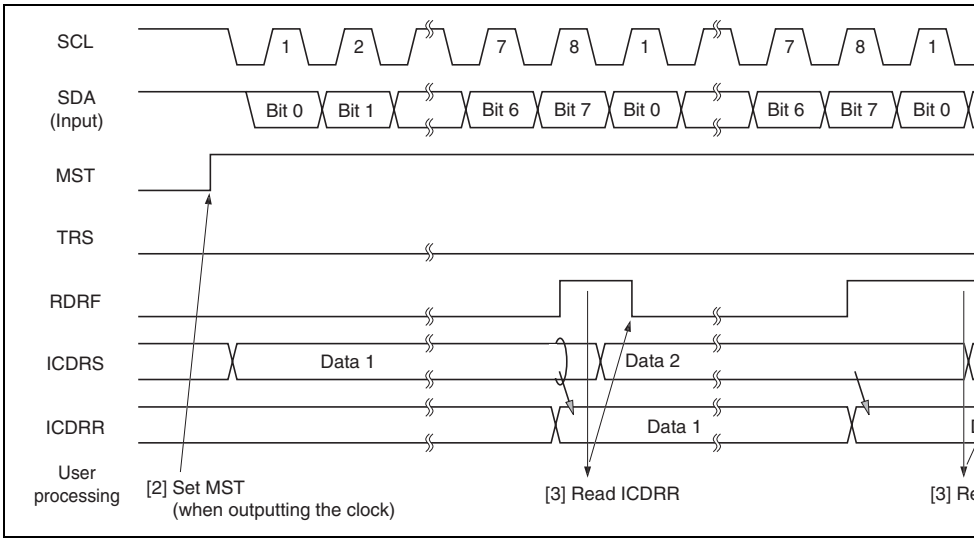


Figure 17.15 Receive Mode Operation Timing

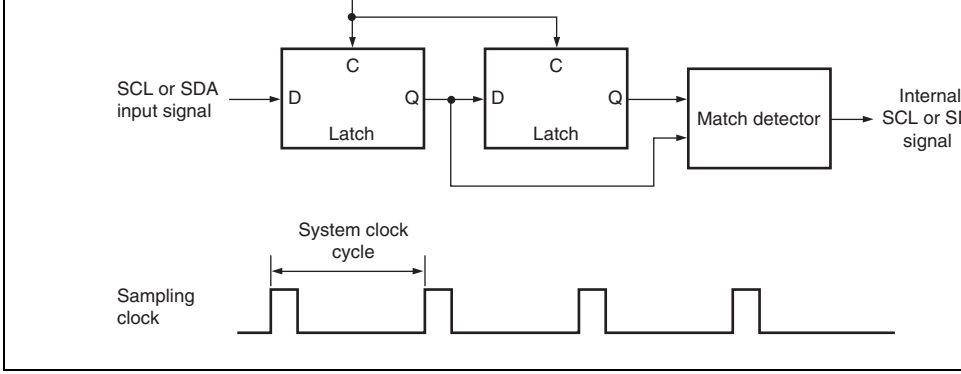


Figure 17.16 Block Diagram of Noise Filter

17.4.8 Example of Use

Flowcharts in respective modes that use the I^2C bus interface are shown in figures 17.17

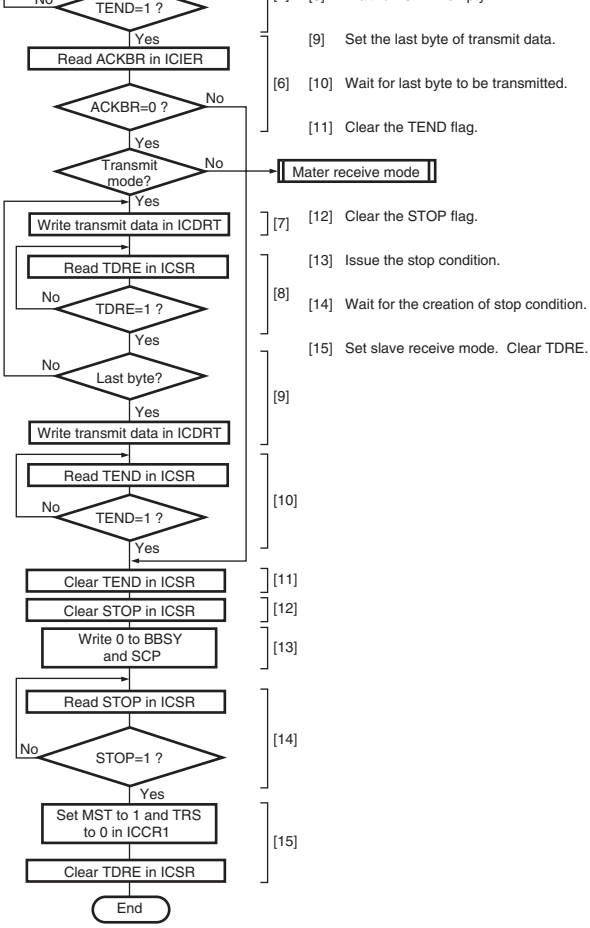
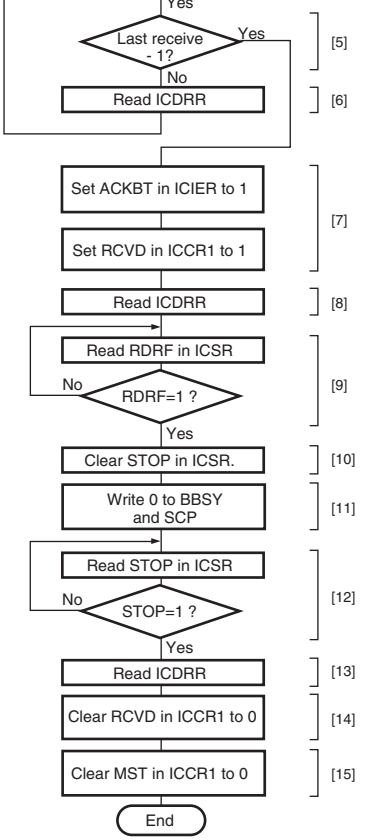


Figure 17.17 Sample Flowchart for Master Transmit Mode



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

[10] Clear the STOP flag.

[11] Issue the stop condition.

[12] Wait for the creation of stop condition.

[13] Read the last byte of receive data.

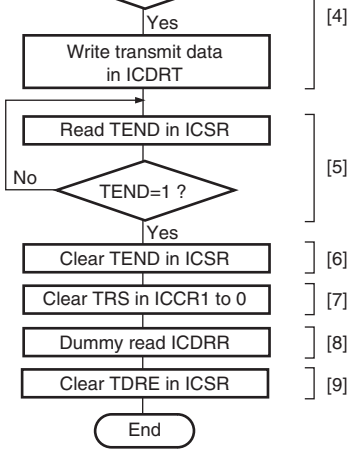
[14] Clear RCVD.

[15] Set slave receive mode.

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

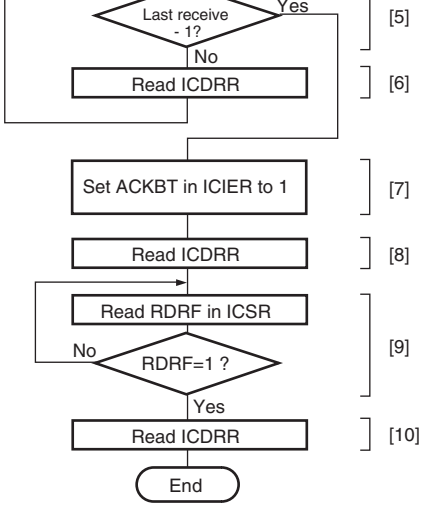
Supplementary explanation: When one byte is received, steps [2] to [6] skipped after step [1], before jumping to ste
The step [8] is dummy-read in ICDRR.

Figure 17.18 Sample Flowchart for Master Receive Mode



[9] Clear the TDRE flag.

Figure 17.19 Sample Flowchart for Slave Transmit Mode



[8] Read the (last byte - 1) of receive data.

[9] Wait the last byte to be received.

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

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

Figure 17.20 Sample Flowchart for Slave Receive Mode

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

When interrupt conditions described in table 17.3 are 1 and the I bit in CCR is 0, the CPU executes an interrupt exception processing. Interrupt sources should be cleared in the exc processing. TDRE and TEND are automatically cleared to 0 by writing the transmit data ICDRT. RDRF are automatically cleared to 0 by reading ICDRR. TDRE is set to 1 again same time when transmit data is written to ICDRT. When TDRE is cleared to 0, then an e data of one byte may be transmitted.

Figure 17.21 shows the timing of the bit synchronous circuit and table 17.4 shows the ti
SCL output changes from low to Hi-Z then SCL is monitored.

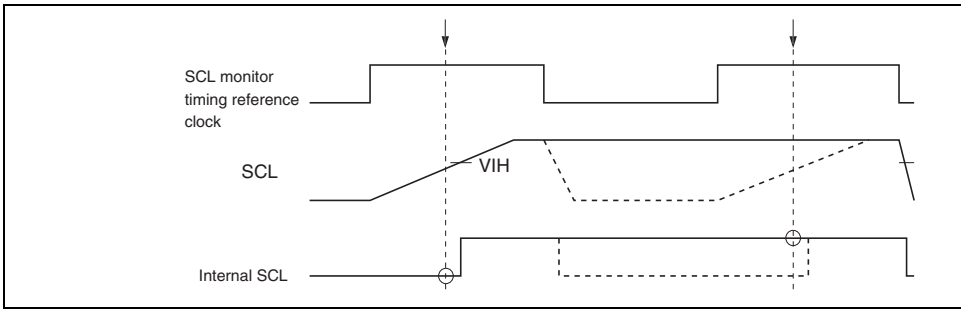


Figure 17.21 The Timing of the Bit Synchronous Circuit

Table 17.4 Time for Monitoring SCL

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

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

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

17.7.2 WAIT Setting in I²C Bus Mode Register (ICMR)

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

(2) Restriction on Use of Bit Manipulation Instructions to Set MST and TRS

When master transmission is selected by consecutively manipulating the MST and TRS multi-master usage, an arbitration loss during execution of the bit-manipulation instructi TRS leads to the contradictory situation where AL in ICSR is 1 in master transmit mode 1, TRS = 1).

Ways to avoid this effect are listed below.

- Use the MOV instruction to set MST and TRS in multi-master usage.
- When arbitration is lost, confirm that MST = 0 and TRS = 0. If the setting of MST = TRS = 0 is not confirmed, set MST = 0 and TRS = 0 again.

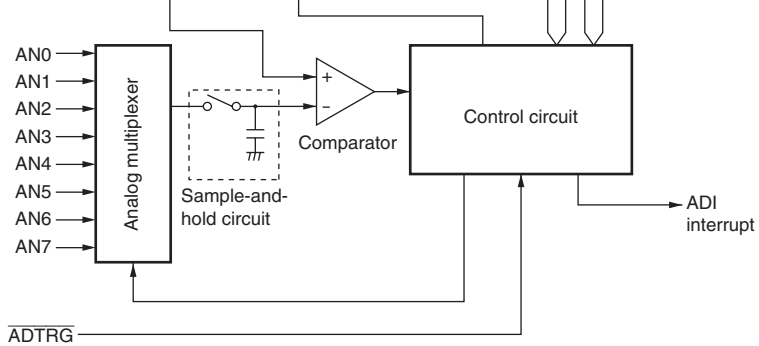
17.7.4 Continuous Data Reception in Master Receive Mode

In master receive mode, when SCL is fixed low on the falling edge of the 8th clock whil RDRF bit is set to 1 and ICDRR is read around the falling edge of the 8th clock, the cloc fixed low in the 8th clock of the next round of data reception. The SCL is then released fixed state without reading ICDRR and the 9th clock is output. As a result, some receive lost.

Ways to avoid this phenomenon are listed below.

- Read ICDRR in master receive mode before the rising edge of the 8th clock.
- Set RCVD to 1 in master receive mode and perform communication in units of one b

- Conversion time: at least 3.5 μs 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 request
  - An A/D conversion end interrupt request (ADI) can be generated



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

Figure 18.1 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 A/D conversion

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

There are four 16-bit read-only ADDR registers; ADDRA to ADDRD, used to store the r
A/D conversion. The ADDR registers, which store a conversion result for each analog inp
channel, are shown in table 18.2.

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
directly from the CPU, however the lower byte should be read via a temporary register. T
temporary register contents are transferred from the ADDR when the upper byte data is r
When reading ADDR, read the upper byte first then the lower one, or read in word units.
ADDR is initialized to H'0000.

Table 18.2 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 Conversi**
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 en 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 sin this bit is cleared to 0 automatically when conve the specified channel is complete. In scan mode conversion continues sequentially on the specifi channels until this bit is cleared to 0 by software a transition to standby mode.
4	SCAN	0	R/W	Scan Mode Selects single mode or scan mode as the A/D c 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 cor time.

	101: AN5	101: AN4 and AN5
	110: AN6	110: AN4 to AN6
	111: AN7	111: AN4 to AN7

18.3.3 A/D Control Register (ADCR)

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

Bit	Bit Name	Initial Value	R/W	Description
7	TRGE	0	R/W	Trigger Enable A/D conversion is started at the falling edge and t edge of the external trigger signal ($\overline{\text{ADTRG}}$) when set to 1. The selection between the falling edge and rising the external trigger pin ($\overline{\text{ADTRG}}$) conforms to the bit in the interrupt edge select register 2 (IEGR2)
6 to 1	—	All 1	—	Reserved These bits are always read as 1.
0	—	0	R/W	Reserved Do not set this bit to 1, though the bit is readable/

channel as follows:

1. A/D conversion is started when the ADST bit in ADCSR is set to 1, according to sof external trigger input.
2. When A/D conversion is completed, the result is transferred to the corresponding A/ register of the channel.
3. On completion of conversion, the ADF bit in ADCSR is set to 1. If the ADIE bit is s 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, bit is automatically cleared to 0 and the A/D converter enters the wait state.

18.4.2 Scan Mode

In scan mode, A/D conversion is performed sequentially for the analog input of the spec 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 C
2. When A/D conversion for each channel is completed, the result is sequentially transf the A/D data register corresponding to each channel.
3. When conversion of all the selected channels is completed, the ADF flag in ADCSR If the ADIE bit is set to 1 at this time, an ADI interrupt requested is generated. A/D 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 lon 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 18.3 apply to the first conversion time. In the secc
subsequent conversions, the conversion time is 128 states (fixed) when CKS = 0 and 66 s
(fixed) when CKS = 1.

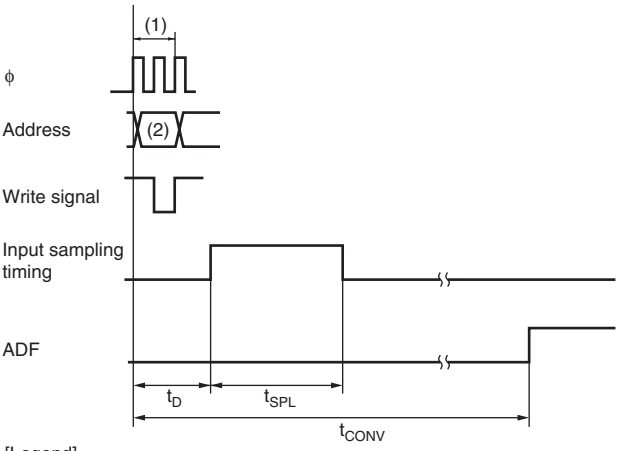


Figure 18.2 A/D Conversion Timing

18.4.4 External Trigger Input Timing

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

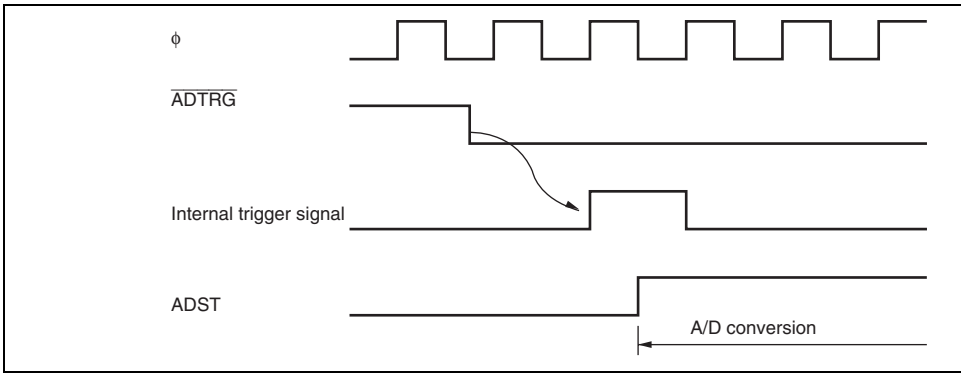


Figure 18.3 External Trigger Input Timing

when the digital output changes from the minimum voltage value 0000000000 to 000
(see figure 18.5).

- Full-scale error

  The deviation of the analog input voltage value from the ideal A/D conversion charac
  when the digital output changes from 1111111110 to 1111111111 (see figure 18.5).

- Nonlinearity error

  The deviation from the ideal A/D conversion characteristic as the voltage changes fro
  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 err
  scale error, quantization error, and nonlinearity error.

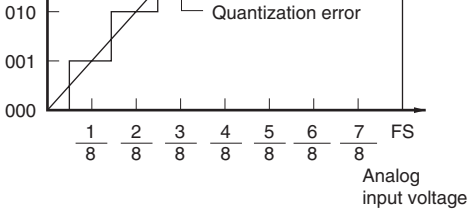


Figure 18.4 A/D Conversion Accuracy Definitions (1)

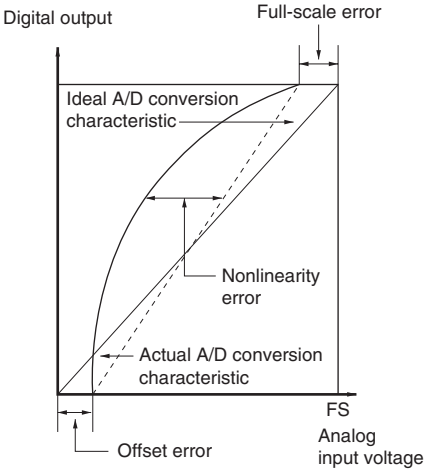


Figure 18.5 A/D Conversion Accuracy Definitions (2)

filter effect is obtained in this case, it may not be possible to follow an analog signal with
differential coefficient (e.g., 5 mV/µs or greater) (see figure 18.6). When converting a hig
analog signal or converting in scan mode, a low-impedance buffer should be inserted.

18.6.2 Influences on Absolute Accuracy

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

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

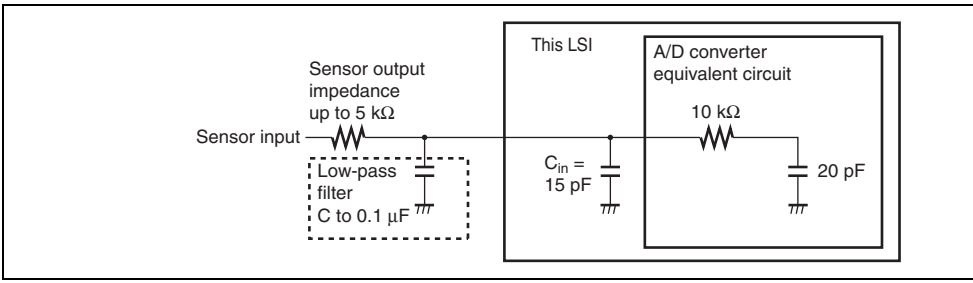


Figure 18.6 Analog Input Circuit Example

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

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

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

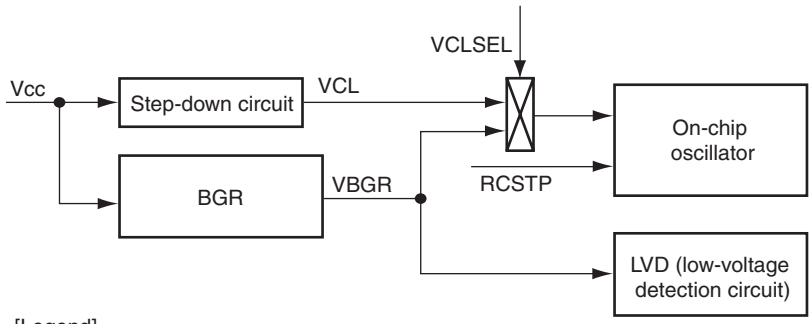
Figure 19.2 is a block diagram of the power-on reset circuit and the low-voltage detectio

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

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

- Reset source decision

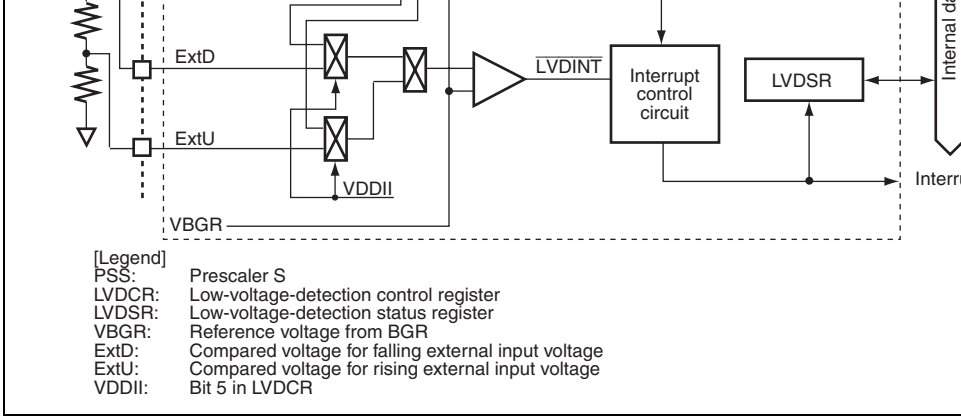
  The source of a reset can be decided by reading the reset source decision register in th
  exception handler.



[Legend]

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

Figure 19.1 Block Diagram around BGR



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

Figure 19.2 Block Diagram of Power-On Reset Circuit and Low-Voltage Detectio

LVDCR selects the compared voltage of the LVDI circuit, sets the detection levels for the
circuit, enables or disables the LVDR circuit, and enables or disables generation of an int
when the power-supply voltage rises above or falls below the respective levels.

Table 19.1 shows the relationship between the LVDCR settings and functions to be selec
LVDCR should be set according to table 19.1.

Bit	Bit Name	Initial Value	R/W	Description
7, 6	—	All 1	—	Reserved These bits are always read as 1 and cannot be r
5	VDDII	1*[1]	R/W	LVDR External Compared Voltage Input Inhibit 0: Use external voltage as LVDI compared voltag 1: Use internal voltage as LVDI compared voltag
4	—	1	—	Reserved This bit is always read as 1 and cannot be modif
3	LVDSEL*[2]	1	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 interr used, the reset detection voltage of 2.3 V (Typ.) be used. When only a reset detection interrupt is reset detection voltage of 3.6 V (Typ.) should be
2	—	1	—	Reserved This bit is always read as 1 and cannot be modif

0) and the reset detection voltage is 2.3 V (typ.).

Table 19.1 LVDCR Settings and Select Functions

LVDCR Settings				Select Functions			
VDDII	LVDSEL	LVDDE	LVDUE	Power-On Reset	LVDR	Low-Voltage-Detection Fall Interrupt	Low Dete Inte
*	1	0	0	√	√	—	—
*	0	1	0	√	√	√	—
*	0	1	1	√	√	√	√

Note: * Set these bits if necessary.

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

Note: * Initialized by an LVDR.

				[Setting conditions] • When a power-on reset has occurred • When an LVDR has occurred [Clearing condition] When writing 0
0	WRST	*2	R/W	WDT Reset Detection [Setting condition] • When a reset by the WDT has occurred [Clearing conditions] • When a power-on reset has occurred • When an LVDR has occurred • When an reset signal input on the external p asserted • When writing 0

Notes: 1. The initial value depends on the condition when the PRST bit is set or cleared
2. The initial value depends on the condition when the WRST bit is set or cleare

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

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

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

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

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

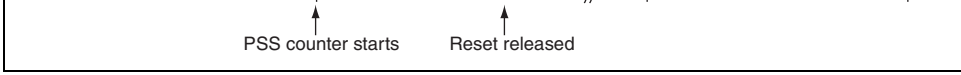


Figure 19.3 Operational Timing of Power-On Reset Circuit

19.3.2 Low-Voltage Detection Circuit

LVDR (Reset by Low Voltage Detection) Circuit:

Figure 19.4 shows the timing of the operation of the LVDR circuit. The LVDR circuit is enabled during the LSI's operation.

When the power-supply voltage falls below the Vreset voltage (the value selected by the bit: Typ. = 2.3 V or 3.6 V), the LVDR circuit clears the $\overline{\text{LVDRES}}$ signal to 0, and resets S. The low-voltage detection reset state remains in place until a power-on reset is genera the power-supply voltage rises above the Vreset voltage (Typ. = 3.6 V) regardless of LV setting) again, the LVDR circuit sets the $\overline{\text{LVDRES}}$ signal to 1 and prescaler S starts cou When 131,072 clock (ϕ) cycles have been counted, the internal reset signal is released. I case, the LVDSEL bit in LVDCR is initialized (the Vreset voltage: Typ. = 3.6 V) thoug VDDII bit is not initialized.

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

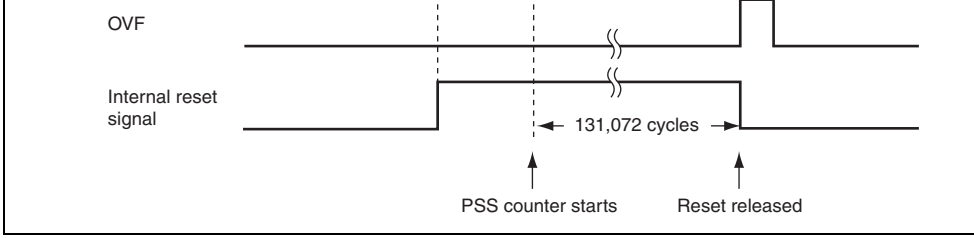


Figure 19.4 Operating Timing of LVDR Circuit

IRQ0 interrupt request is generated. In this case, the necessary data must be saved in the EEPROM and a transition to standby mode or subsleep mode must be made. Until this p is completed, the power supply voltage must be higher than the lower limit of the guaran operating voltage.

When the power-supply voltage does not fall below the Vreset1 (Typ. = 2.3 V) voltage a above the Vint (U) (Typ. = 4.0 V) voltage, the LVDI circuit sets the $\overline{\text{LVDINT}}$ signal to LVDUE bit is 1 at this time, the LVDUF bit in LVDSR is set to 1 and an IRQ0 interrupt simultaneously generated.

If the power supply voltage (Vcc) falls below the Vreset1 (Typ. = 2.3 V) voltage, this LS low voltage detection reset operation (when LVDRE = 1).

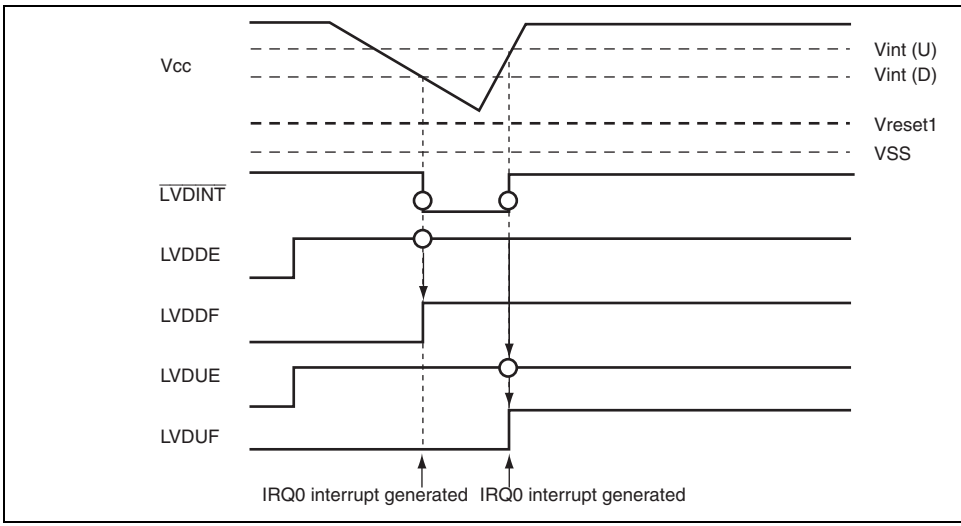


Figure 19.5 Operational Timing of LVDI Circuit

When the external comparison voltage of ExtD pin falls below the Vexd (D) (Typ. = 1.15 voltage, the LVDI clears the $\overline{\text{LVDINT}}$ signal to 0 and sets the LVDDF bit in LVDSR to 1 LVDDE bit is 1 at this time, an IRQ0 interrupt request is generated. In this case, the nece data must be saved in the external EEPROM, and a transition to standby mode or subslee must be made. Until this processing is completed, the power supply voltage must be high the lower limit of the guaranteed operating voltage.

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

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

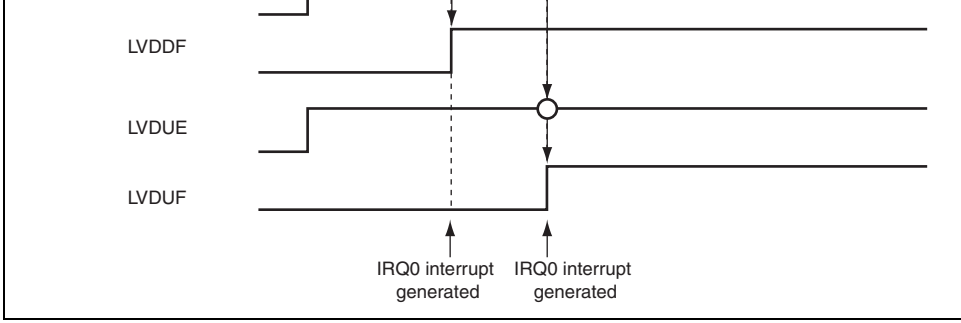


Figure 19.6 Operational Timing of LVDI Circuit (When Compared Voltage is through ExtU and ExtD Pins)

PRST	WRST	Reset Source
1	0	Power-on reset or LVDR occurred
0	0	Reset signal input on external reset pin
0	1	WDT reset occurred

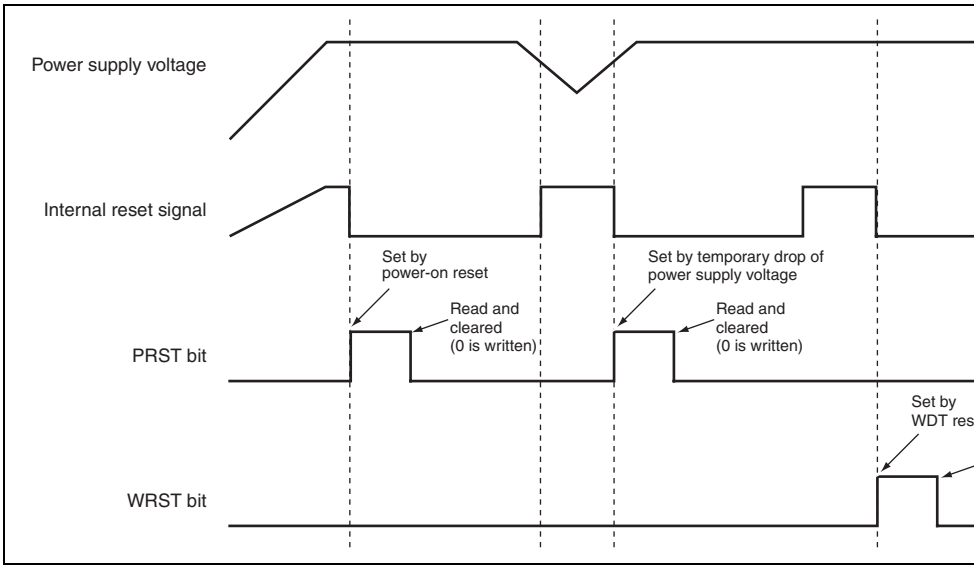


Figure 19.7 Timing of Setting Bits in Reset Source Decision Register

Connect the external power supply to the V_{cc} pin, and connect a capacitor of approximat between V_{CL} and V_{ss}, as shown in figure 20.1. The internal step-down circuit is made effe simply by adding this external circuit. In the external circuit interface, the external powe voltage connected to V_{cc} and the GND potential connected to V_{ss} are the reference levels example, for port input/output levels, the V_{cc} level is the reference for the high level, and level is that for the low level. The A/D converter analog power supply is not affected by internal step-down circuit.

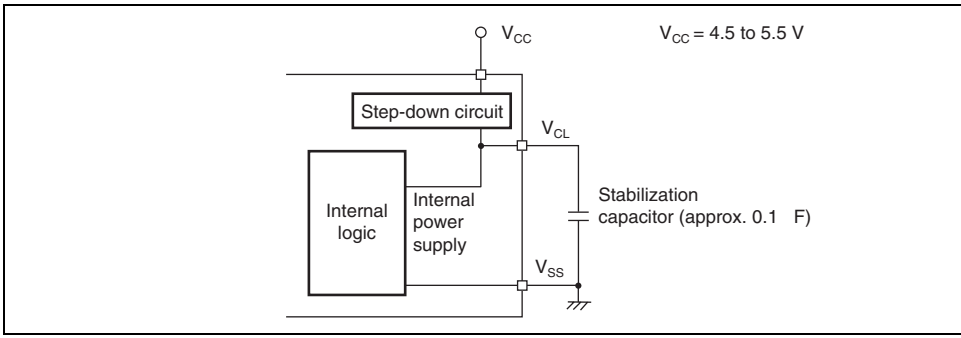


Figure 20.1 Power Supply Connection of 5.0-V-Specification Microcontroll

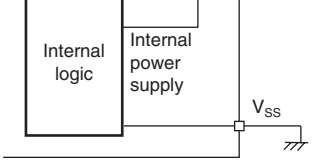


Figure 20.2 Power Supply Connection of 3.3-V-Specification Microcontrolle

- Registers are listed from the lower allocation addresses.
- The symbol — in the register-name column represents a reserved address or range o addresses.
  Do not attempt to access reserved addresses.
- When the register address is 16-bit wide, the address of the upper byte is given in the
- Registers are classified by functional modules.
- The data bus width is indicated.
- The number of access states is indicated.

2. Register bits

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

3. Register states in each operating mode

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

—	—	—	H'F000 to H'F6FF	—	—
Timer control register_0	TCR_0	8	H'F700	Timer Z	8
Timer I/O control register A_0	TIORA_0	8	H'F701	Timer Z	8
Timer I/O control register C_0	TIORC_0	8	H'F702	Timer Z	8
Timer status register_0	TSR_0	8	H'F703	Timer Z	8
Timer interrupt enable register_0	TIER_0	8	H'F704	Timer Z	8
PWM mode output level control register_0	POCR_0	8	H'F705	Timer Z	8
Timer counter_0	TCNT_0	16	H'F706	Timer Z	16
General register A_0	GRA_0	16	H'F708	Timer Z	16
General register B_0	GRB_0	16	H'F70A	Timer Z	16
General register C_0	GRC_0	16	H'F70C	Timer Z	16
General register D_0	GRD_0	16	H'F70E	Timer Z	16
Timer control register_1	TCR_1	8	H'F710	Timer Z	8
Timer I/O control register A_1	TIORA_1	8	H'F711	Timer Z	8
Timer I/O control register C_1	TIORC_1	8	H'F712	Timer Z	8
Timer status register_1	TSR_1	8	H'F713	Timer Z	8
Timer interrupt enable register_1	TIER_1	8	H'F714	Timer Z	8
PWM mode output level control register_1	POCR_1	8	H'F715	Timer Z	8
Timer counter_1	TCNT_1	16	H'F716	Timer Z	16
General register A_1	GRA_1	16	H'F718	Timer Z	16
General register B_1	GRB_1	16	H'F71A	Timer Z	16

register					
Timer output control register	TOCR	8	H'F725	Timer Z	8
—	—	—	H'F726, H'F727	Timer Z	—
Second data register/free running counter data register	RSECDR	8	H'F728	RTC	8
Minute data register	RMINDR	8	H'F729	RTC	8
Hour data register	RHRDR	8	H'F72A	RTC	8
Day-of-week data register	RWKDR	8	H'F72B	RTC	8
RTC control register 1	RTCCR1	8	H'F72C	RTC	8
RTC control register 2	RTCCR2	8	H'F72D	RTC	8
—	—	—	H'F72E	RTC	—
Clock source select register	RTCCSR	8	H'F72F	RTC	8
Low-voltage-detection control register	LVDCR	8	H'F730	LVDC*[1]	8
Low-voltage-detection status register	LVDSR	8	H'F731	LVDC*[1]	8
Reset source decision register	LVDRF	8	H'F732	LVDC*[1]	8
—	—	—	H'F733	—	—
Clock control/status register	CKCSR	8	H'F734	CPG	8
RC control register	RCCR	8	H'F735	On-chip oscillator	8
RC trimming data protect register	RCTRMDPR	8	H'F736	On-chip oscillator	8

Transmit data register_2	TDR_2	8	H'F743	SCI3_2	8
Serial status register_2	SSR_2	8	H'F744	SCI3_2	8
Receive data register_2	RDR_2	8	H'F745	SCI3_2	8
—	—	—	H'F746, H'F747	SCI3_2	—
I2C bus control register 1	ICCR1	8	H'F748	IIC2	8
I2C bus control register 2	ICCR2	8	H'F749	IIC2	8
I2C bus mode register	ICMR	8	H'F74A	IIC2	8
I2C bus interrupt enable register	ICIER	8	H'F74B	IIC2	8
I2C status register	ICSR	8	H'F74C	IIC2	8
Slave address register	SAR	8	H'F74D	IIC2	8
I2C bus transmit data register	ICDRT	8	H'F74E	IIC2	8
I2C bus receive data register	ICDRR	8	H'F74F	IIC2	8
—	—	—	H'F750 to H'F75F	—	—
Timer mode register B1	TMB1	8	H'F760	Timer B1	8
Timer counter B1	TCB1	8	H'F761	Timer B1	8
Timer load register B1	TLB1	8	H'F761	Timer B1	8
—	—	—	H'F762 to H'FF8F	—	—
Flash memory control register 1	FLMCR1	8	H'FF90	ROM	8
Flash memory control register 2	FLMCR2	8	H'FF91	ROM	8
Flash memory power control register	FLPWCR	8	H'FF92	ROM	8

Timer control/status register V	TCSRV	8	H'FFA1	Timer V	8
Time constant register A	TCORA	8	H'FFA2	Timer V	8
Time constant register B	TCORB	8	H'FFA3	Timer V	8
Timer counter V	TCNTV	8	H'FFA4	Timer V	8
Timer control register V1	TCRV1	8	H'FFA5	Timer V	8
—	—	—	H'FFA6, H'FFA7	—	—
Serial mode register	SMR	8	H'FFA8	SCI3	8
Bit rate register	BRR	8	H'FFA9	SCI3	8
Serial control register 3	SCR3	8	H'FFAA	SCI3	8
Transmit data register	TDR	8	H'FFAB	SCI3	8
Serial status register	SSR	8	H'FFAC	SCI3	8
Receive data register	RDR	8	H'FFAD	SCI3	8
—	—	—	H'FFAE, H'FFAF	SCI3	—
A/D data register	ADDRA	16	H'FFB0	A/D converter	8
A/D data register	ADDRB	16	H'FFB2	A/D converter	8
A/D data register	ADDRC	16	H'FFB4	A/D converter	8
A/D data register	ADDRD	16	H'FFB6	A/D converter	8
A/D control/status register	ADCSR	8	H'FFB8	A/D converter	8

—	—	—	H'FFBF	14-bit PWM	—
Timer control/status register WD	TCSRWD	8	H'FFC0	WDT*[1]	8
Timer counter WD	TCWD	8	H'FFC1	WDT*[1]	8
Timer mode register WD	TMWD	8	H'FFC2	WDT*[1]	8
—	—	—	H'FFC3	WDT*[1]	—
—	—	—	H'FFC4 to H'FFC7	—	—
Address break control register	ABRKCR	8	H'FFC8	Address break	8
Address break status register	ABRKSR	8	H'FFC9	Address break	8
Break address register H	BARH	8	H'FFCA	Address break	8
Break address register L	BARL	8	H'FFCB	Address break	8
Break data register H	BDRH	8	H'FFCC	Address break	8
Break data register L	BDRL	8	H'FFCD	Address break	8
—	—	—	H'FFCE	—	—
Break address register E*[2]	BARE	8	H'FFCF	Address break	8
Port pull-up control register 1	PUCR1	8	H'FFD0	I/O port	8
Port pull-up control register 5	PUCR5	8	H'FFD1	I/O port	8
—	—	—	H'FFD2, H'FFD3	I/O port	—
Port data register 1	PDR1	8	H'FFD4	I/O port	8
Port data register 2	PDR2	8	H'FFD5	I/O port	8
Port data register 3	PDR3	8	H'FFD6	I/O port	8
—	—	—	H'FFD7	I/O port	—
Port data register 5	PDR5	8	H'FFD8	I/O port	8

Port mode register 1	PMR1	8	H'FFE0	I/O port	8
Port mode register 5	PMR5	8	H'FFE1	I/O port	8
Port mode register 3	PMR3	8	H'FFE2	I/O port	8
—	—	—	H'FFD3	I/O port	—
Port control register 1	PCR1	8	H'FFE4	I/O port	8
Port control register 2	PCR2	8	H'FFE5	I/O port	8
Port control register 3	PCR3	8	H'FFE6	I/O port	8
—	—	—	H'FFE7	I/O port	—
Port control register 5	PCR5	8	H'FFE8	I/O port	8
Port control register 6	PCR6	8	H'FFE9	I/O port	8
Port control register 7	PCR7	8	H'FFEA	I/O port	8
Port control register 8	PCR8	8	H'FFEB	I/O port	8
—	—	—	H'FFEC, H'FFED	I/O port	—
Port control register C	PCRC	8	H'FFEE	I/O port	8
—	—	—	H'FFEF	I/O port	—
System control register 1	SYSCR1	8	H'FFF0	Low power	8
System control register 2	SYSCR2	8	H'FFF1	Low power	8
Interrupt edge select register 1	IEGR1	8	H'FFF2	Interrupt	8
Interrupt edge select register 2	IEGR2	8	H'FFF3	Interrupt	8
Interrupt enable register 1	IENR1	8	H'FFF4	Interrupt	8
Interrupt enable register 2	IENR2	8	H'FFF5	Interrupt	8
Interrupt flag register 1	IRR1	8	H'FFF6	Interrupt	8

H'FFFF

Notes: 1. WDT: Watchdog timer
2. Only provided for microcontrollers that supports advanced mode.

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
TSR_1	—	—	UDF	OVF	IMFD	IMFC	IMFB	IMFA
TIER_1	—	—	—	OVIE	IMIED	IMIEC	IMIEB	IMIEA
POCR_1	—	—	—	—	—	POLD	POLC	POLB
TCNT_1	TCNT1H7	TCNT1H6	TCNT1H5	TCNT1H4	TCNT1H3	TCNT1H2	TCNT1H1	TCNT1H0
	TCNT1L7	TCNT1L6	TCNT1L5	TCNT1L4	TCNT1L3	TCNT1L2	TCNT1L1	TCNT1L0
GRA_1	GRA1H7	GRA1H6	GRA1H5	GRA1H4	GRA1H3	GRA1H2	GRA1H1	GRA1H0
	GRA1L7	GRA1L6	GRA1L5	GRA1L4	GRA1L3	GRA1L2	GRA1L1	GRA1L0

TMDR	BFDT	BFOT	BFD0	BFC0	—	—	—	STNC
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	—	—	VDDII	—	LVDSEL*[1]	—	LVDDE	LVDUE
LVDSR	—	—	—	—	—	—	LVDDF	LVDUF
LVDRF	—	—	—	—	—	—	PRST	WRST
—	—	—	—	—	—	—	—	—
CKCSR	PMRC1	PMRC0	OSCBAKE	OSCSEL	CKSWIE	CKSWIF	OSCHLT	CKSTA
RCCR	RCSTP	FSEL	VCLSEL	—	—	—	RCPSC1	RCPSC0
RCTRMDPR	WRI	PRWE	LOCKDW	TRMDRWE	—	—	—	—
RCTRMDR	TRMD7	TRMD6	TRMD5	TRMD4	TRMD3	TRMD2	TRMD1	TRMD0
—	—	—	—	—	—	—	—	—
SMR_2	COM	CHR	PE	PM	STOP	MP	CKS1	CKS0
BRR_2	BRR7	BRR6	BRR5	BRR4	BRR3	BRR2	BRR1	BRR0
SCR3_2	TIE	RIE	TE	RE	MPIE	TEIE	CKE1	CKE0

IICSR	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
—	—	—	—	—	—	—	—	—
FLMCR1	—	SWE	ESU	PSU	EV	PV	E	P
FLMCR2	FLER	—	—	—	—	—	—	—
FLPWCR	PDWND	—	—	—	—	—	—	—
EBR1*[1]	EB7	EB6	EB5	EB4	EB3	EB2	EB1	EB0
FENR	FLSHE	—	—	—	—	—	—	—
TCRV0	CMIEB	CMIEA	OVIE	CCLR1	CCLR0	CKS2	CKS1	CKS0
TCSRV	CMFB	CMFA	OVF	—	OS3	OS2	OS1	OS0
TCORA	TCORA7	TCORA6	TCORA5	TCORA4	TCORA3	TCORA2	TCORA1	TCORA0
TCORB	TCORB7	TCORB6	TCORB5	TCORB4	TCORB3	TCORB2	TCORB1	TCORB0
TCNTV	TCNTV7	TCNTV6	TCNTV5	TCNTV4	TCNTV3	TCNTV2	TCNTV1	TCNTV0
TCRV1	—	—	—	TVEG1	TVEG0	TRGE	—	ICKS0
—	—	—	—	—	—	—	—	—
SMR	COM	CHR	PE	PM	STOP	MP	CKS1	CKS0
BRR	BRR7	BRR6	BRR5	BRR4	BRR3	BRR2	BRR1	BRR0
SCR3	TIE	RIE	TE	RE	MPIE	TEIE	CKE1	CKE0

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	—	—	—	—	—	—	—
—	—	—	—	—	—	—	—	—
PWDRL	PWDRL7	PWDRL6	PWDRL5	PWDRL4	PWDRL3	PWDRL2	PWDRL1	PWDRL0
PWDRU	—	—	PWDRU5	PWDRU4	PWDRU3	PWDRU2	PWDRU1	PWDRU0
PWCR	—	—	—	—	—	—	—	PWCR0
TCSRWD	B6WI	TCWE	B4WI	TCSRWE	B2WI	WDON	B0WI	WRST
TCWD	TCWD7	TCWD6	TCWD5	TCWD4	TCWD3	TCWD2	TCWD1	TCWD0
TMWD	—	—	—	—	CKS3	CKS2	CKS1	CKS0
—	—	—	—	—	—	—	—	—
ABRKCR	RTINTE	CSEL1	CSEL0	ACMP2	ACMP1	ACMP0	DCMP1	DCMP0
ABRKSR	ABIF	ABIE	—	—	—	—	—	—
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*[3]	BDRE7	BDRE6	BDRE5	BDRE4	BDRE3	BDRE2	BDRE1	BDRE0
PUCR1	PUCR17	PUCR16	PUCR15	PUCR14	—	PUCR12	PUCR11	PUCR10
PUCR5	—	—	PUCR55	PUCR54	PUCR53	PUCR52	PUCR51	PUCR50
PDR1	P17	P16	P15	P14	—	P12	P11	P10

PDRC	—	—	—	—	—	—	PC1	PC0
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
PCR6	PCR67	PCR66	PCR65	PCR64	PCR63	PCR62	PCR61	PCR60
PCR7	—	PCR76	PCR75	PCR74	—	PCR72	PCR71	PCR70
PCR8	PCR87	PCR86	PCR85	—	—	—	—	—
PCRC	—	—	—	—	—	—	PCRC1	PCRC0
SYSCR1	SSBY	STS2	STS1	STS0	NESEL	—	—	—
SYSCR2	SMSEL	LSON	DTON	MA2	MA1	MA0	SA1	SA0
IEGR1	NMIEG	—	—	—	IEG3	IEG2	IEG1	IEG0
IEGR2	—	—	WPEG5	WPEG4	WPEG3	WPEG2	WPEG1	WPEG0
IENR1	IENDT	IENTA	IENWP	—	IEN3	IEN2	IEN1	IEN0
IENR2	—	—	IENTB1	—	—	—	—	—
IRR1	IRRDT	IRRTA	—	—	IRRI3	IRRI2	IRRI1	IRRI0
IRR2	—	—	IRRTB1	—	—	—	—	—
IWPR	—	—	IWPF5	IWPF4	IWPF3	IWPF2	IWPF1	IWPF0

- H8/36079G, H8/36079L

Register Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
EBR1	EB7	EB6	EB5	EB4	EB3	EB2	EB1	EB0

- H8/36078G, H8/36078L, H8/36077G, H8/36077L

Register Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
EBR1	—	EB6	EB5	EB4	EB3	EB2	EB1	EB0

- H8/36074G, H8/36074L

Register Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
EBR1	—	—	—	EB4	EB3	EB2	EB1	EB0

POCR_0	Initialized	—	—	—	—	—
TCNT_0	Initialized	—	—	—	—	—
GRA_0	Initialized	—	—	—	—	—
GRB_0	Initialized	—	—	—	—	—
GRC_0	Initialized	—	—	—	—	—
GRD_0	Initialized	—	—	—	—	—
TCR_1	Initialized	—	—	—	—	—
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	—	—	—	—	—
TSTR	Initialized	—	—	—	—	—
TMDR	Initialized	—	—	—	—	—
TPMR	Initialized	—	—	—	—	—
TFCR	Initialized	—	—	—	—	—
TOER	Initialized	—	—	—	—	—
TOCR	Initialized	—	—	—	—	—

LVDCR	Initialized	—	—	—	—	—	LVDC
LVDSR	Initialized	—	—	—	—	—	
LVDRF	—	—	—	—	—	—	
CKCSR	Initialized	—	—	—	—	—	CPG
RCCR	Initialized	—	—	—	—	—	On-ch
RCTRMDPR	Initialized	—	—	—	—	—	oscilla
RCTRMDR	Initialized	—	—	—	—	—	
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	—	—	—	—	—	Timer
TCB1	Initialized	—	—	—	—	—	
TLB1	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	
ADDRD	Initialized	—	—	Initialized	Initialized	Initialized	
ADCSR	Initialized	—	—	Initialized	Initialized	Initialized	
ADCR	Initialized	—	—	Initialized	Initialized	Initialized	
PWDRL	Initialized	—	—	—	—	—	14bi
PWDRU	Initialized	—	—	—	—	—	
PWCR	Initialized	—	—	—	—	—	
TCSRWD	Initialized	—	—	—	—	—	WD
TCWD	Initialized	—	—	—	—	—	
TMWD	Initialized	—	—	—	—	—	
ABRKCR	Initialized	—	—	—	—	—	Add
ABRKSR	Initialized	—	—	—	—	—	

PDR1	Initialized	—	—	—	—	—	
PDR2	Initialized	—	—	—	—	—	
PDR3	Initialized	—	—	—	—	—	
PDR5	Initialized	—	—	—	—	—	
PDR6	Initialized	—	—	—	—	—	
PDR7	Initialized	—	—	—	—	—	
PDR8	Initialized	—	—	—	—	—	
PDRB	Initialized	—	—	—	—	—	
PDRC	Initialized	—	—	—	—	—	
PMR1	Initialized	—	—	—	—	—	
PMR5	Initialized	—	—	—	—	—	
PMR3	Initialized	—	—	—	—	—	
PCR1	Initialized	—	—	—	—	—	
PCR2	Initialized	—	—	—	—	—	
PCR3	Initialized	—	—	—	—	—	
PCR5	Initialized	—	—	—	—	—	
PCR6	Initialized	—	—	—	—	—	
PCR7	Initialized	—	—	—	—	—	
PCR8	Initialized	—	—	—	—	—	
PCRC	Initialized	—	—	—	—	—	
SYSCR1	Initialized	—	—	—	—	—	Low p
SYSCR2	Initialized	—	—	—	—	—	

MSTCRH	Initialized	—	—	—	—	—	Low
MSTCR2	Initialized	—	—	—	—	—	

Notes: — is not initialized

1. WDT: Watchdog timer
2. The BARE register is only provided for microcontrollers that support advance

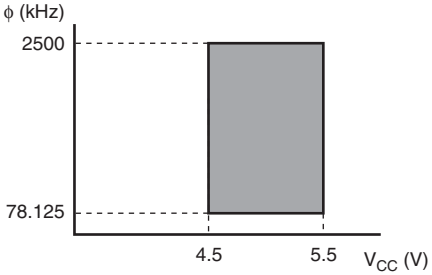
	B and X1	V_{IN}		
	Port B		–0.3 to AV_{CC} +0.3	V
	X1		–0.3 to 4.3	V
Operating temperature		T_{opr}	Regular specifications: –20 to +75	°C
			Wide-range specifications: –40 to +85	°C
Storage temperature		T_{stg}	–55 to +125	°C

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

- AV_{CC} = 4.5 to 5.5 V
- Active mode
- Sleep mode

- AV_{CC} = 4.5 to 5.5 V
- All operating modes

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

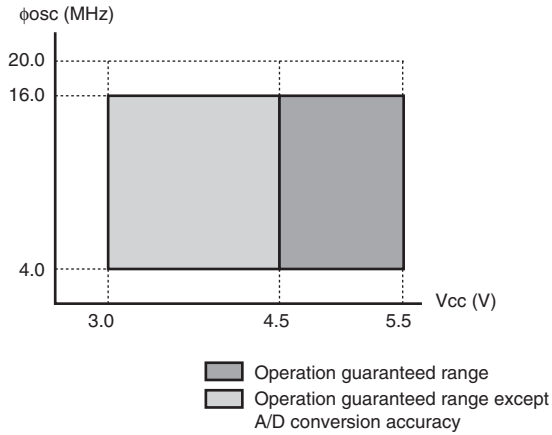


- AV_{CC} = 4.5 to 5.5 V
- Active mode
- Sleep mode
(When MA2 in SYSCR2 = 1)

- Subactive mode
- Subsleep mode

- V_{CC} = 4.5 to 5.5 V
- Active mode
- Sleep mode

(4) Range of Power Supply Voltage and Oscillation Frequency when Low-Voltage Detection Circuit is Used



		TMRIV, TMCIV, FTIOA0 to FTIOD0, FTIOA1 to FTIOD1, SCK3, SCK3_2, TRGV					
		RXD, RXD_2, 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 P76, P85 to P87, PC0, PC1		$V_{CC} \times 0.7$	—	$V_{CC} + 0.3$	V
		PB0 to PB7	AV_{CC} = 4.5 to 5.5 V	$AV_{CC} \times 0.7$	—	$AV_{CC} + 0.3$	V
		OSC1		$V_{CC} - 0.5$	—	$V_{CC} + 0.3$	V
Input low voltage	V_{IL}	$\overline{RES}$, $\overline{NMI}$, $\overline{WKP0}$ to $\overline{WKP5}$, $\overline{IRQ0}$ to $\overline{IRQ3}$, $\overline{ADTRG}$, TMIB1, TMRIV, TMCIV, FTIOA0 to FTIOD0, FTIOA1 to FTIOD1, SCK3, SCK3_2, TRGV		–0.3	—	$V_{CC} \times 0.2$	V

Note: Connect the TEST pin to Vss.

		PC0, PC1					
		PB0 to PB7	AV_{CC} = 4.5 to 5.5 V	–0.3	—	AV_{CC} × 0.3	V
		OSC1		–0.3	—	0.5	V
Output high voltage	V_{OH}	P10 to P12, P14 to P17, P20 to P24, P30 to P37, P50 to P55, P60 to P67, P70 to P72, P74 to P76, P85 to P87, PC0, PC1	$-I_{OH}$ = 1.5 mA	V_{CC} – 1.0	—	—	V
			$-I_{OH}$ = 0.1 mA	V_{CC} – 0.5	—	—	
		P56, P57	$-I_{OH}$ = 0.1 mA	V_{CC} – 2.5	—	—	V
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 P76, P85 to P87 PC0, PC1	I_{OL} = 1.6 mA	—	—	0.6	V
			I_{OL} = 0.4 mA	—	—	0.4	
		P60 to P67	I_{OL} = 20.0 mA	—	—	1.5	V
			I_{OL} = 10.0 mA	—	—	1.0	
			I_{OL} = 1.6 mA	—	—	0.4	
			I_{OL} = 0.4 mA	—	—	0.4	
		SCL, SDA	I_{OL} = 6.0 mA	—	—	0.6	
			I_{OL} = 3.0 mA	—	—	0.4	

		SDA					
		P10 to P12, P14 to P17, P20 to P24, P30 to P37, P50 to P57, P60 to P67, P70 to P72, P74 to P76, P85 to P87, PC0, PC1	V_{IN} = 0.5 V to (V_{CC} – 0.5 V)	—	—	1.0	µA
		PB0 to PB7	V_{IN} = 0.5 V to (AV_{CC} – 0.5 V)	—	—	1.0	µA
Pull-up MOS current	$-I_p$	P10 to P12, P14 to P17, P50 to P55	V_{CC} = 5.0 V, V_{IN} = 0.0 V	50.0	—	300.0	µA
Input capacitance	C_{in}	All input pins except power supply pins	f = 4 MHz, V_{IN} = 0.0 V, T_a = 25°C	—	—	15.0	pF
Active mode supply current	I_{OPE1}	V_{CC}	Active mode 1 V_{CC} = 5.0 V, f_{OSC} = 20 MHz	—	19.0	28.0	mA
			Active mode 1 V_{CC} = 5.0 V, f_{OSC} = 10 MHz	—	11.0	—	
	I_{OPE2}	V_{CC}	Active mode 2 V_{CC} = 5.0 V, f_{OSC} = 20 MHz	—	3.0	5.5	mA
			Active mode 2 V_{CC} = 5.0 V, f_{OSC} = 10 MHz	—	2.5	—	

			V_{CC} = 5.0 V, f_{OSC} = 10 MHz					
Subactive mode supply current	I_{SUB}	V_{CC}	V_{CC} = 5.0 V 32-kHz crystal resonator ($\phi_{SUB} = \phi_W/2$)	—	95.0	145.0	µA	*
			V_{CC} = 5.0 V 32-kHz crystal resonator ($\phi_{SUB} = \phi_W/8$)	—	85.0	—		*
Subsleep mode supply current	I_{SUBSP}	V_{CC}	V_{CC} = 5.0 V 32-kHz crystal resonator ($\phi_{SUB} = \phi_W/2$)	—	85.0	140.0	µA	*
Standby mode supply current	I_{STBY}	V_{CC}	32-kHz crystal resonator not used	—	—	135.0	µA	*
RAM data retention voltage	V_{RAM}	V_{CC}		2.0	—	—	V	

Subactive mode	V_{CC}	Operates	V_{CC}	Main clock: ceramic or cryst resonator, and c oscillator
Subsleep mode	V_{CC}	Only timers operate	V_{CC}	Subclock: crystal resonato
Standby mode	V_{CC}	CPU and timers both stop	V_{CC}	Main clock: ceramic or cryst resonator, and c oscillator Subclock: Pin X1 = V_{SS}

		SCL, and SDA			
		Port 6, SCL, and SDA	—	—	80.0
Permissible output high current (per pin)	$\lvert -I_{OH} \rvert$	All output pins	—	—	5.0
Permissible output high current (total)	$\lvert -\Sigma I_{OH} \rvert$	All output pins	—	—	50.0

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 (on-chip oscillator)	t_{rc}		—	—	500	µs
Oscillation stabilization time	t_{rcx}	X1, X2	—	—	2.0	s
External clock high width	t_{CPH}	OSC1	20.0	—	—	ns
External clock low width	t_{CPL}	OSC1	20.0	—	—	ns
External clock rise time	t_{CPr}	OSC1	—	—	10.0	ns
			—	—	15.0	
External clock fall time	t_{CPf}	OSC1	—	—	10.0	ns
			—	—	15.0	

		TMRIV, TRGV, $\overline{\text{ADTRG}}$, FTIOA0 to FTIOD0, FTIOA1 to FTIOD1				
Input pin low width	t_{IL}	$\overline{\text{NMI}}$, TMIB1, $\overline{\text{IRQ0}}$ to $\overline{\text{IRQ3}}$, $\overline{\text{WKP0}}$ to $\overline{\text{WKP5}}$, TMCIV, TMRIV, TRGV, $\overline{\text{ADTRG}}$, FTIOA0 to FTIOD0, FTIOA1 to FTIOD1	2	—	—	t_{cyc} t_{subcyc}

Vcc = 5.0 V, Ta = 25°C, FSEL = 0, VCLSEL = 0	15.76	16.0	16.24
FSEL = 0, Ta = -20 to +75°C, VCLSEL = 0	15.52	16.0	16.48
FSEL = 0, Ta = -40 to +85°C, VCLSEL = 0	15.36	16.0	16.64

Notes: 1. When an external clock is input, the minimum frequency of the external clock is 4.0 MHz.

2. Determined by MA2 to MA0, SA1, and SA0 in system control register 2 (SYS

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
Repeated 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}	—	—	250	ns

Input clock pulse width	t_{SCKW}	SCK3	0.4	—	0.6	t_{Scyc}	
Transmit data delay time (clock synchronous)	t_{TXD}	TXD	—	—	1	t_{cyc}	F
Receive data setup time (clock synchronous)	t_{RXS}	RXD	50.0	—	—	ns	
Receive data hold time (clock synchronous)	t_{RXH}	RXD	50.0	—	—	ns	

		AN7						
Analog power supply current	AI_{OPE}	AV_{CC}	AV_{CC} = 5.0 V f_{OSC} = 20 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	
Permissible signal source impedance	R_{AIN}	AN0 to AN7		—	—	5.0	kΩ	
Resolution (data length)				10	10	10	bit	
Conversion time (single mode)			AV_{CC} = 4.5 to 5.5 V	70	—	—	t_{cyc}	
Nonlinearity error				—	—	±7.5	LSB	
Offset error				—	—	±7.5	LSB	
Full-scale error				—	—	±7.5	LSB	
Quantization error				—	—	±0.5	LSB	
Absolute accuracy				—	—	±8.0	LSB	

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

22.2.5 Watchdog Timer Characteristics

Table 22.7 Watchdog Timer Characteristics

V_{CC} = 4.5 to 5.5 V, V_{SS} = 0.0 V, T_a = -20 to +75°C/-40 to +85°C, unless otherwise indica

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

Note: * Time until an internal reset is generated after the counter counts from 0 to 25 on-chip oscillator is selected

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

bit*[1]					
Wait time after setting EV bit*[1]	γ		20	—	—
Wait time after dummy write*[1]	ε		2	—	—
Wait time after clearing EV bit*[1]	η		4	—	—
Wait time after clearing SWE bit*[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 memory control register 1 (FLMCR1) is set. The program-verify time is not inc

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

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

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

   Programming count (n)

   $1 \leq n \leq 6$ z1 = 30 µs

   $7 \leq n \leq 1000$ z2 = 200 µs

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

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

Reset detection voltage 1*[1]	Vreset1	LVDSEL = 0	—	2.3	2.6
Reset detection voltage 2*[2]	Vreset2	LVDSEL = 1	3.3	3.6	3.9
Lower-limit voltage of LVDR operation	$V_{LVDRmin}$		1.0	—	—

Notes: 1. This voltage should be used when the falling and rising voltage detection funct used.
2. Select the low-voltage reset 2 when only the low-voltage detection reset is use

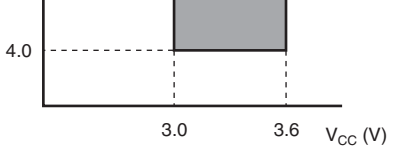
22.2.8 LVDI External Input Voltage Detection Circuit Characteristics

Table 22.10 LVDI External Input Voltage Detection Circuit Characteristics

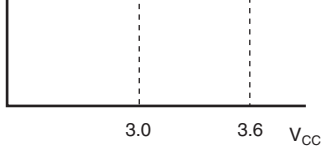
Vcc = 4.5 to 5.5 V, AVcc = 4.5 to 5.5 V, V_{SS} = 0.0 V, T_a = -20 to +75°C/-40 to +85°C

Item	Symbol	Test Condition	Values		
			Min.	Typ.	Max.
ExtD/ExtU input detection voltage	Vexd		1.0	1.15	1.30
ExtD/ExtU input voltage range	VextD/VextU	VextD > VextU	–0.3	—	Lower voltage of AV_{cc} + 0.3 or V_{cc} + 0.3

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

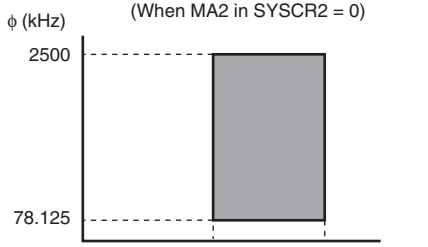


- AV_{CC} = 3.0 to 3.6 V
- Active mode
- Sleep mode



- AV_{CC} = 3.0 to 3.6 V
- All operating modes

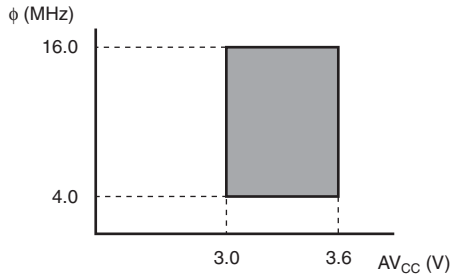
• AV_{CC} = 3.0 to 3.6 V
• Active mode
• Sleep mode
(When MA2 in SYSCR2 = 0)



• AV_{CC} = 3.0 to 3.6 V
• Active mode
• Sleep mode
(When MA2 in SYSCR2 = 1)

• AV_{CC} = 3.0 to 3.6 V
• Subactive mode
• Subsleep mode

(3) Analog Power Supply Voltage and A/D Converter Accuracy Guaranteed Ran



• V_{CC} = 3.0 to 3.6 V
• Active mode
• Sleep mode

		TMRIV, TMCIV, FTIOA0 to FTIOD0, FTIOA1 to FTIOD1, SCK3, SCK3_2, TRGV					
		RXD, RXD_2, 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 P76, P85 to P87, PC0, PC1		$V_{CC} \times 0.8$	—	$V_{CC} + 0.3$	V
		PB0 to PB7	AV_{CC} = 3.0 to 3.6 V	$AV_{CC} \times 0.8$	—	$AV_{CC} + 0.3$	V
		OSC1		$V_{CC} - 0.3$	—	$V_{CC} + 0.3$	V
Input low voltage	V_{IL}	$\overline{RES}$, $\overline{NMI}$, $\overline{WKP0}$ to $\overline{WKP5}$, $\overline{IRQ0}$ to $\overline{IRQ3}$, $\overline{ADTRG}$, TMIB1, TMRIV, TMCIV, FTIOA0 to FTIOD0, FTIOA1 to FTIOD1, SCK3, SCK3_2, TRGV		–0.3	—	$V_{CC} \times 0.1$	V

Note: Connect the TEST pin to Vss.

		PC0, PC1					
		PB0 to PB7	AV_{CC} = 3.0 to 3.6 V	–0.3	—	AV_{CC} × 0.2	V
		OSC1		–0.3	—	0.3	V
Output high voltage	V_{OH}	P10 to P12, P14 to P17, P20 to P24, P30 to P37, P50 to P55, P60 to P67, P70 to P72, P74 to P76, P85 to P87, PC0, PC1	$-I_{OH}$ = 1.5 mA	V_{CC} – 1.0	—	—	V
			$-I_{OH}$ = 0.1 mA	V_{CC} – 0.5	—	—	
		P56, P57	$-I_{OH}$ = 0.1 mA	V_{CC} – 2.0	—	—	V
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 P76, P85 to P87 PC0, PC1	I_{OL} = 1.6 mA	—	—	0.6	V
			I_{OL} = 0.4 mA	—	—	0.4	
		P60 to P67	I_{OL} = 20.0 mA	—	—	1.5	V
			I_{OL} = 10.0 mA	—	—	1.0	
			I_{OL} = 1.6 mA	—	—	0.4	
			I_{OL} = 0.4 mA	—	—	0.4	
		SCL, SDA	I_{OL} = 6.0 mA	—	—	0.6	
			I_{OL} = 3.0 mA	—	—	0.4	

		SDA						
		P10 to P12, P14 to P17, P20 to P24, P30 to P37, P50 to P57, P60 to P67, P70 to P72, P74 to P76, P85 to P87, PC0, PC1	V_{IN} = 0.5 V to (V_{CC} – 0.5 V)	—	—	1.0	µA	
		PB0 to PB7	V_{IN} = 0.5 V to (AV_{CC} – 0.5 V)	—	—	1.0	µA	
Pull-up MOS current	$-I_p$	P10 to P12, P14 to P17, P50 to P55	V_{CC} = 3.3 V, V_{IN} = 0.0 V	—	60.0	—	µA	
Input capacitance	C_{in}	All input pins except power supply pins	f = 4 MHz, V_{IN} = 0.0 V, T_a = 25°C	—	—	15.0	pF	
Active mode current consumption	I_{OPE1}	V_{CC}	Active mode 1 V_{CC} = 3.3 V, f_{OSC} = 16 MHz	—	15.0	22.0	mA	*
			Active mode 1 V_{CC} = 3.3 V, f_{OSC} = 10 MHz	—	11.0	—		* P v
	I_{OPE2}	V_{CC}	Active mode 2 V_{CC} = 3.3 V, f_{OSC} = 16 MHz	—	2.8	4.0	mA	*
			Active mode 2 V_{CC} = 3.3 V, f_{OSC} = 10 MHz	—	2.5	—		* P v

			V_{CC} = 3.3 V, f_{OSC} = 10 MHz				
Subactive mode supply current	I_{SUB}	V_{CC}	V_{CC} = 3.3 V 32-kHz crystal resonator ($\phi_{SUB} = \phi_W/2$)	—	95.0	145.0	µA
			V_{CC} = 3.3 V 32-kHz crystal resonator ($\phi_{SUB} = \phi_W/8$)	—	85.0	—	
Subsleep mode supply current	I_{SUBSP}	V_{CC}	V_{CC} = 3.3 V 32-kHz crystal resonator ($\phi_{SUB} = \phi_W/2$)	—	85.0	140.0	µA
Standby mode supply current	I_{STBY}	V_{CC}	32-kHz crystal resonator not used	—	—	135.0	µA
RAM data retention voltage	V_{RAM}	V_{CC}		2.0	—	—	V

Subactive mode	V_{cc}	Operates	V_{cc}	Main clock: ceramic or crysta resonator, and or oscillator
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 crysta resonator, and or oscillator Subclock: Pin X1 = V_{ss}

		SCL, and SDA			
		Port 6, SCL, and SDA	—	—	80.0
Permissible output high current (per pin)	$\lvert -I_{OH} \rvert$	All output pins	—	—	5.0
Permissible output high current (total)	$\lvert -\Sigma I_{OH} \rvert$	All output pins	—	—	50.0

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 (on-chip oscillator)	t_{rc}		—	—	500	µs
Oscillation stabilization time	t_{rcx}	X1, X2	—	—	2.0	s
External clock high width	t_{CPH}	OSC1	23.8	—	—	ns
External clock low width	t_{CPL}	OSC1	23.8	—	—	ns
External clock rise time	t_{CPr}	OSC1	—	—	15.0	ns
External clock fall time	t_{CPf}	OSC1	—	—	15.0	ns

		TMCIV, TMRIV, TRGV, $\overline{\text{ADTRG}}$, FTIOA0 to FTIOD0, FTIOA1 to FTIOD1				
Input pin low width	t_{IL}	$\overline{\text{NMI}}$, TMIB1, $\overline{\text{IRQ0}}$ to $\overline{\text{IRQ3}}$, $\overline{\text{WKP0}}$ to $\overline{\text{WKP5}}$, TMCIV, TMRIV, TRGV, $\overline{\text{ADTRG}}$, FTIOA0 to FTIOD0, FTIOA1 to FTIOD1	2	—	—	t_{cyc} t_{subcyc}

Vcc = 3.3 V, Ta = 25°C, FSEL = 0, VCLSEL = 0	15.76	16.00	16.24
FSEL = 0, Ta = -20 to +75°C VCLSEL = 0	15.52	16.0	16.48
FSEL = 0, Ta = -40 to +85°C, VCLSEL = 0	15.36	16.0	16.64

Notes: 1. When an external clock is input, the minimum frequency of the external clock c
is 4.0 MHz.

2. Determined by MA2 to MA0, SA1, and SA0 in system control register 2 (SYSC

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
Repeated 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}	—	—	250	ns

Input clock pulse width	t_{SCKW}	SCK3	0.4	—	0.6	t_{Scyc}	
Transmit data delay time (clock synchronous)	t_{TXD}	TXD	—	—	1	t_{cyc}	Fi
Receive data setup time (clock synchronous)	t_{RXS}	RXD	50.0	—	—	ns	
Receive data hold time (clock synchronous)	t_{RXH}	RXD	50.0	—	—	ns	

Analog power supply current	AI_{OPE}	AV_{CC}	AV_{CC} = 3.6 V f_{OSC} = 16 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
Permissible signal source impedance	R_{AIN}	AN0 to AN7		—	—	5.0	kΩ
Resolution (data length)				10	10	10	bit
Conversion time (single mode)			AV_{CC} = 3.0 to 3.6 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

Notes: 1. Set $AV_{CC} = V_{CC}$ when the A/D converter is not used.
2. AI_{STOP1} is the current in active and sleep modes while the A/D converter is idle
3. AI_{STOP2} is the current at reset and in standby, subactive, and subsleep modes w A/D converter is idle.

time

Note: * Time until an internal reset is generated after the counter counts from 0 to 255 on-chip oscillator is selected

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

	bit*[1]				
	Wait time after setting EV bit*[1]	γ	20	—	—
	Wait time after dummy write*[1]	ε	2	—	—
	Wait time after clearing EV bit*[1]	η	4	—	—
	Wait time after clearing SWE bit*[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 b memory control register 1 (FLMCR1) is set. The program-verify time is not incl

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

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

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

   Programming count (n)

   $1 \leq n \leq 6$ z1 = 30 µs

   $7 \leq n \leq 1000$ z2 = 200 µs

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

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

Reset detection voltage 1*	Vreset1	—	2.3	2.6
Lower-limit voltage of LVDR operation	$V_{LVDRmin}$	1.0	—	—

Note: * This voltage should be used when the falling and rising voltage detection func used.

22.3.8 LVDI External Input Voltage Detection Circuit Characteristics

Table 22.20 LVDI External Input Voltage Detection Circuit Characteristics

Vcc = 3.0 to 3.6 V, AVcc = 3.0 to 3.6 V, V_{SS} = 0.0 V, T_a = -20 to +75°C/-40 to +85°C

Item	Symbol	Test Condition	Values		
			Min.	Typ.	Max.
ExtD/ExtU input detection voltage	Vexd		0.95	1.15	1.35
ExtD/ExtU input voltage range	VextD/VextU	VextD > VextU	–0.3	—	Lower voltage of AV_{cc} + 0.3 o V_{cc} + 0.3

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

Figure 22.1 System Clock Input Timing

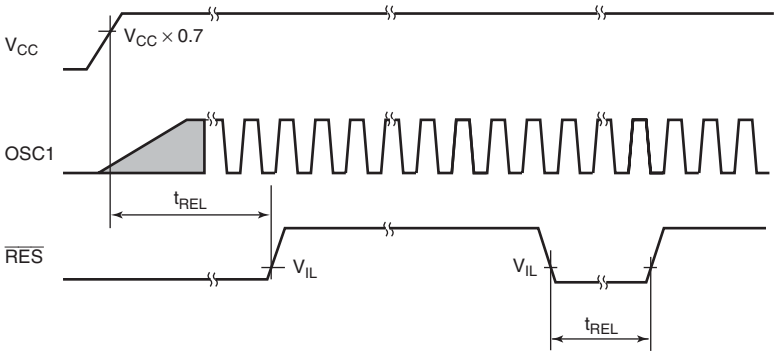


Figure 22.2 $\overline{\text{RES}}$ Low Width Timing

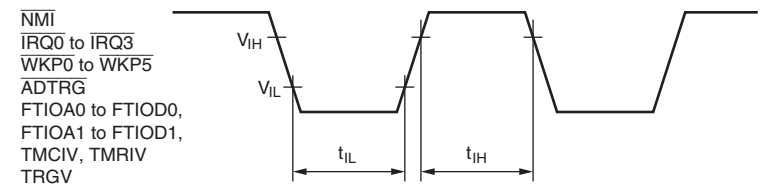


Figure 22.3 Input Timing

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

Figure 22.4 I²C Bus Interface Input/Output Timing

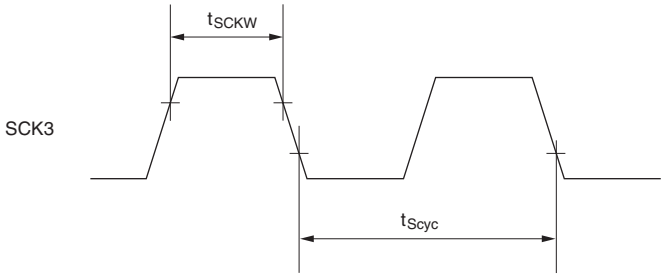
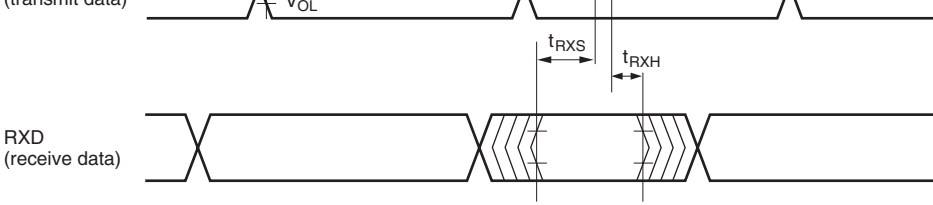


Figure 22.5 SCK3 Input Clock Timing



Note: * Output timing reference levels
Output high: V_{OH}= 2.0 V
Output low: V_{OL}= 0.8 V
Load conditions are shown in figure 22.7.

Figure 22.6 SCI Input/Output Timing in Clock Synchronous Mode

Figure 22.7 Output Load Circuit

ERd	General destination register (address register or 32-bit register)
ERs	General source register (address register or 32-bit register)
ERn	General register (32-bit register)
(EAd)	Destination operand
(EAs)	Source operand
PC	Program counter
SP	Stack pointer
CCR	Condition-code register
N	N (negative) flag in CCR
Z	Z (zero) flag in CCR
V	V (overflow) flag in CCR
C	C (carry) flag in CCR
disp	Displacement
→	Transfer from the operand on the left to the operand on the right, or trans the state on the left to the state on the right
+	Addition of the operands on both sides
–	Subtraction of the operand on the right from the operand on the left
×	Multiplication of the operands on both sides
÷	Division of the operand on the left by the operand on the right
∧	Logical AND of the operands on both sides
∨	Logical OR of the operands on both sides
⊕	Logical exclusive OR of the operands on both sides
¬	NOT (logical complement)
( ), < >	Contents of operand

Note: General registers include 8-bit registers (R0H to R7H and R0L to R7L) and 16-bit (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		2								MOV.L ERs, 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	-
MOVFPE	MOVFPE @aa:16, Rd	B						4				Cannot be used in this LSI	Cannot be used in this LSI					
MOVTPE	MOVTPE Rs, @aa:16	B						4				Cannot be used in this LSI	Cannot be used in this LSI					

	ADD.L #xx:32, ERd	L	6									ERd32+#xx:32 → ERd32	—	(2)	↕	↕	↕
	ADD.L ERs, ERd	L		2								ERd32+ERs32 → ERd32	—	(2)	↕	↕	↕
ADDX	ADDX.B #xx:8, Rd	B	2									Rd8+#xx:8 +C → Rd8	—	↕	↕	(3)	↕
	ADDX.B Rs, Rd	B		2								Rd8+Rs8 +C → Rd8	—	↕	↕	(3)	↕
ADDS	ADDS.L #1, ERd	L		2								ERd32+1 → ERd32	—	—	—	—	—
	ADDS.L #2, ERd	L		2								ERd32+2 → ERd32	—	—	—	—	—
	ADDS.L #4, ERd	L		2								ERd32+4 → ERd32	—	—	—	—	—
INC	INC.B Rd	B		2								Rd8+1 → Rd8	—	—	↕	↕	↕
	INC.W #1, Rd	W		2								Rd16+1 → Rd16	—	—	↕	↕	↕
	INC.W #2, Rd	W		2								Rd16+2 → Rd16	—	—	↕	↕	↕
	INC.L #1, ERd	L		2								ERd32+1 → ERd32	—	—	↕	↕	↕
	INC.L #2, ERd	L		2								ERd32+2 → ERd32	—	—	↕	↕	↕
DAA	DAA Rd	B		2								Rd8 decimal adjust → Rd8	—	*	↕	↕	*
SUB	SUB.B Rs, Rd	B		2								Rd8–Rs8 → Rd8	—	↕	↕	↕	↕
	SUB.W #xx:16, Rd	W	4									Rd16–#xx:16 → Rd16	—	(1)	↕	↕	↕
	SUB.W Rs, Rd	W		2								Rd16–Rs16 → Rd16	—	(1)	↕	↕	↕
	SUB.L #xx:32, ERd	L	6									ERd32–#xx:32 → ERd32	—	(2)	↕	↕	↕
	SUB.L ERs, ERd	L		2								ERd32–ERs32 → ERd32	—	(2)	↕	↕	↕
SUBX	SUBX.B #xx:8, Rd	B	2									Rd8–#xx:8–C → Rd8	—	↕	↕	(3)	↕
	SUBX.B Rs, Rd	B		2								Rd8–Rs8–C → Rd8	—	↕	↕	(3)	↕
SUBS	SUBS.L #1, ERd	L		2								ERd32–1 → ERd32	—	—	—	—	—
	SUBS.L #2, ERd	L		2								ERd32–2 → ERd32	—	—	—	—	—
	SUBS.L #4, ERd	L		2								ERd32–4 → ERd32	—	—	—	—	—
DEC	DEC.B Rd	B		2								Rd8–1 → Rd8	—	—	↕	↕	↕
	DEC.W #1, Rd	W		2								Rd16–1 → Rd16	—	—	↕	↕	↕
	DEC.W #2, Rd	W		2								Rd16–2 → Rd16	—	—	↕	↕	↕

	MULXU. W Rs, ERd	W		2								Rd16 × Rs16 → ERd32 (unsigned multiplication)	—	—	—	—	—	
MULXS	MULXS. B Rs, Rd	B		4								Rd8 × Rs8 → Rd16 (signed multiplication)	—	—	↕	↕	—	
	MULXS. W Rs, ERd	W		4								Rd16 × Rs16 → ERd32 (signed multiplication)	—	—	↕	↕	—	
DIVXU	DIVXU. B Rs, Rd	B		2								Rd16 ÷ Rs8 → Rd16 (RdH: remainder, RdL: quotient) (unsigned division)	—	—	(6)	(7)	—	
	DIVXU. W Rs, ERd	W		2								ERd32 ÷ Rs16 → ERd32 (Ed: remainder, Rd: quotient) (unsigned division)	—	—	(6)	(7)	—	
DIVXS	DIVXS. B Rs, Rd	B		4								Rd16 ÷ Rs8 → Rd16 (RdH: remainder, RdL: quotient) (signed division)	—	—	(8)	(7)	—	
	DIVXS. W Rs, ERd	W		4								ERd32 ÷ Rs16 → ERd32 (Ed: remainder, Rd: quotient) (signed division)	—	—	(8)	(7)	—	
CMP	CMP.B #xx:8, Rd	B	2									Rd8–#xx:8	—	↕	↕	↕	↕	
	CMP.B Rs, Rd	B		2								Rd8–Rs8	—	↕	↕	↕	↕	
	CMP.W #xx:16, Rd	W	4									Rd16–#xx:16	—	(1)	↕	↕	↕	
	CMP.W Rs, Rd	W		2								Rd16–Rs16	—	(1)	↕	↕	↕	
	CMP.L #xx:32, ERd	L	6									ERd32–#xx:32	—	(2)	↕	↕	↕	
	CMP.L ERs, ERd	L		2								ERd32–ERs32	—	(2)	↕	↕	↕	

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

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

	BSET Rn, @ERd	B			4							(Rn8 of @ERd) ← 1	—	—	—	—	—	
	BSET Rn, @aa:8	B						4				(Rn8 of @aa:8) ← 1	—	—	—	—	—	
BCLR	BCLR #xx:3, Rd	B		2								(#xx:3 of Rd8) ← 0	—	—	—	—	—	
	BCLR #xx:3, @ERd	B			4							(#xx:3 of @ERd) ← 0	—	—	—	—	—	
	BCLR #xx:3, @aa:8	B						4				(#xx:3 of @aa:8) ← 0	—	—	—	—	—	
	BCLR Rn, Rd	B		2								(Rn8 of Rd8) ← 0	—	—	—	—	—	
	BCLR Rn, @ERd	B			4							(Rn8 of @ERd) ← 0	—	—	—	—	—	
	BCLR Rn, @aa:8	B						4				(Rn8 of @aa:8) ← 0	—	—	—	—	—	
BNOT	BNOT #xx:3, Rd	B		2								(#xx:3 of Rd8) ← ¬ (#xx:3 of Rd8)	—	—	—	—	—	
	BNOT #xx:3, @ERd	B			4							(#xx:3 of @ERd) ← ¬ (#xx:3 of @ERd)	—	—	—	—	—	
	BNOT #xx:3, @aa:8	B						4				(#xx:3 of @aa:8) ← ¬ (#xx:3 of @aa:8)	—	—	—	—	—	
	BNOT Rn, Rd	B		2								(Rn8 of Rd8) ← ¬ (Rn8 of Rd8)	—	—	—	—	—	
	BNOT Rn, @ERd	B			4							(Rn8 of @ERd) ← ¬ (Rn8 of @ERd)	—	—	—	—	—	
	BNOT Rn, @aa:8	B						4				(Rn8 of @aa:8) ← ¬ (Rn8 of @aa:8)	—	—	—	—	—	
BTST	BTST #xx:3, Rd	B		2								¬ (#xx:3 of Rd8) → Z	—	—	—	↕	—	
	BTST #xx:3, @ERd	B			4							¬ (#xx:3 of @ERd) → Z	—	—	—	↕	—	
	BTST #xx:3, @aa:8	B						4				¬ (#xx:3 of @aa:8) → Z	—	—	—	↕	—	
	BTST Rn, Rd	B		2								¬ (Rn8 of @Rd8) → Z	—	—	—	↕	—	
	BTST Rn, @ERd	B			4							¬ (Rn8 of @ERd) → Z	—	—	—	↕	—	
	BTST Rn, @aa:8	B						4				¬ (Rn8 of @aa:8) → Z	—	—	—	↕	—	
BLD	BLD #xx:3, Rd	B		2								(#xx:3 of Rd8) → C	—	—	—	—	—	

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

	BHI d:8	—						2			C∨ Z = 0	—	—	—	—	—				
	BHI d:16	—						4				—	—	—	—	—				
	BLS d:8	—						2			C∨ Z = 1	—	—	—	—	—				
	BLS d:16	—						4				—	—	—	—	—				
	BCC d:8 (BHS d:8)	—						2			C = 0	—	—	—	—	—				
	BCC d:16 (BHS d:16)	—						4				—	—	—	—	—				
	BCS d:8 (BLO d:8)	—						2			C = 1	—	—	—	—	—				
	BCS d:16 (BLO d:16)	—						4				—	—	—	—	—				
	BNE d:8	—						2			Z = 0	—	—	—	—	—				
	BNE d:16	—						4				—	—	—	—	—				
	BEQ d:8	—						2			Z = 1	—	—	—	—	—				
	BEQ d:16	—						4				—	—	—	—	—				
	BVC d:8	—						2			V = 0	—	—	—	—	—				
	BVC d:16	—						4				—	—	—	—	—				
	BVS d:8	—						2			V = 1	—	—	—	—	—				
	BVS d:16	—						4				—	—	—	—	—				
	BPL d:8	—						2			N = 0	—	—	—	—	—				
	BPL d:16	—						4				—	—	—	—	—				
	BMI d:8	—						2			N = 1	—	—	—	—	—				
	BMI d:16	—						4				—	—	—	—	—				
	BGE d:8	—						2			N⊕V = 0	—	—	—	—	—				
	BGE d:16	—						4				—	—	—	—	—				
	BLT d:8	—						2			N⊕V = 1	—	—	—	—	—				
	BLT d:16	—						4				—	—	—	—	—				
	BGT d:8	—						2			Z∨ (N⊕V) = 0	—	—	—	—	—				
	BGT d:16	—						4				—	—	—	—	—				
	BLE d:8	—						2			Z∨ (N⊕V) = 1	—	—	—	—	—				
	BLE d:16	—						4				—	—	—	—	—				

													PC ← PC+d:16					
JSR	JSR @ERn	—			2								PC → @–SP PC ← ERn	—	—	—	—	—
	JSR @aa:24	—						4					PC → @–SP PC ← aa:24	—	—	—	—	—
	JSR @@aa:8	—								2			PC → @–SP PC ← @aa:8	—	—	—	—	—
RTS	RTS	—									2		PC ← @SP+	—	—	—	—	—

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

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

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

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

Instruction code:

1st byte		2nd byte	
AH	AL	BH	BL

Instruction when most significant bit of BH is 0

Instruction when most significant bit of BH is 1

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

Instruction code:

1st byte		2nd byte	
AH	AL	BH	BL

AH AL \ BH	0	1	2	3	4	5	6	7	8	9	A	B
01	MOV				LDC/STC				SLEEP			
0A	INC											
0B	ADDS					INC		INC	ADDS			
0F	DAA											
10	SHLL			SHLL					SHAL			SHA
11	SHLR			SHLR					SHAR			SHA
12	ROTXL			ROTXL					ROTL			ROT
13	ROTXR			ROTXR					ROTR			ROT
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	BM
79	MOV	ADD	CMP	SUB	OR	XOR	AND					
7A	MOV	ADD	CMP	SUB	OR	XOR	AND					

Instruction code:

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

Instruction when r
Instruction when r

AH / ALBH / BLCH \ CL	0	1	2	3	4	5	6	7	8	9	A	B
01406										LDC / STC		LDC / S
01C05	MULXS		MULXS									
01D05		DIVXS		DIVXS								
01F06					OR	XOR	AND					
7Cr06*1				BTST								
7Cr07*1				BTST	BOR / BIOR	BXOR / BIXOR	BAND / BIAND	BLD / BILD				
7Dr06*1	BSET	BNOT	BCLR					BST / BIST				
7Dr07*1	BSET	BNOT	BCLR									
7Eaa6*2				BTST								
7Eaa7*2				BTST	BOR / BIOR	BXOR / BIXOR	BAND / BIAND	BLD / BILD				
7Faa6*2	BSET	BNOT	BCLR					BST / BIST				
7Faa7*2	BSET	BNOT	BCLR									

Notes: 1. r is the register designation field.
2. aa is the absolute address field.

BSET #0, @FF00

From table A.4:

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

From table A.3:

$S_I = 2$, $S_L = 2$

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

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

JSR @@ 30

From table A.4:

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

From table A.3:

$S_I = S_J = S_K = 2$

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

Note: * Depends on which on-chip peripheral module is accessed. See section 21.1, 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

	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	BLT d:8	2
	BGT d:8	2
	BLE d:8	2
	BRA d:16(BT d:16)	2
	BRN d:16(BF d:16)	2
	BHI d:16	2
	BLS d:16	2
	BCC d:16(BHS d:16)	2
	BCS d:16(BLO d:16)	2
	BNE d:16	2
	BEQ d:16	2
	BVC d:16	2
	BVS d:16	2
	BPL d:16	2
	BMI d:16	2
	BGE d:16	2
	BLT d:16	2
	BGT d:16	2
	BLE d:16	2

	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
BIOR	BIOR #xx:8, Rd	1	
	BIOR #xx:8, @ERd	2	1
	BIOR #xx:8, @aa:8	2	1
BIST	BIST #xx:3, Rd	1	
	BIST #xx:3, @ERd	2	2
	BIST #xx:3, @aa:8	2	2
BIXOR	BIXOR #xx:3, Rd	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

	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
BTST	BTST #xx:3, Rd	1		
	BTST #xx:3, @ERd	2		1
	BTST #xx:3, @aa:8	2		1
	BTST Rn, Rd	1		
	BTST Rn, @ERd	2		1
	BTST Rn, @aa:8	2		1
BXOR	BXOR #xx:3, Rd	1		
	BXOR #xx:3, @ERd	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		

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					
INC	INC.B Rd	1					
	INC.W #1/2, Rd	1					
	INC.L #1/2, ERd	1					
JMP	JMP @ERn	2					
	JMP @aa:24	2					
	JMP @@aa:8	2	1				
JSR	JSR @ERn	2		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.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	MOV.B Rs, @aa:16	2		1	
	MOV.B Rs, @aa:24	3		1	
	MOV.W #xx:16, Rd	2			
	MOV.W Rs, Rd	1			
	MOV.W @ERs, Rd	1			1
	MOV.W @(d:16,ERs), Rd	2			1
	MOV.W @(d:24,ERs), Rd	4			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

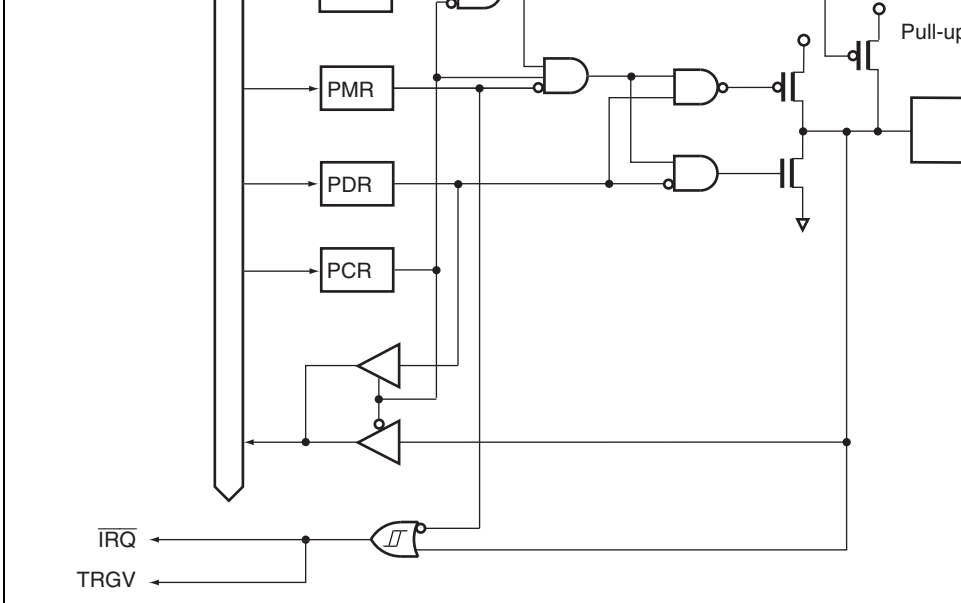
	MOV.L @(d:24,ERs), ERd	5		2
	MOV.L @ERs+, ERd	2		2
	MOV.L @aa:16, ERd	3		2
	MOV.L @aa:24, ERd	4		2
	MOV.L ERs,@ERd	2		2
	MOV.L ERs, @(d:16,ERd)	3		2
	MOV.L ERs, @(d:24,ERd)	5		2
	MOV.L ERs, @-ERd	2		2
	MOV.L ERs, @aa:16	3		2
	MOV.L ERs, @aa:24	4		2
MOVFPE	MOVFPE @aa:16, Rd*2	2	1	
MOVTPE	MOVTPE Rs,@aa:16*2	2	1	
MULXS	MULXS.B Rs, Rd	2		
	MULXS.W Rs, ERd	2		
MULXU	MULXU.B Rs, Rd	1		
	MULXU.W Rs, ERd	1		
NEG	NEG.B Rd	1		
	NEG.W Rd	1		
	NEG.L ERd	1		
NOP	NOP	1		
NOT	NOT.B Rd	1		
	NOT.W Rd	1		
	NOT.L ERd	1		

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		
ROTXR	ROTXR.B Rd	1		
	ROTXR.W Rd	1		
	ROTXR.L ERd	1		
RTE	RTE	2	2	
RTS	RTS	2	1	
SHAL	SHAL.B Rd	1		
	SHAL.W Rd	1		
	SHAL.L ERd	1		
SHAR	SHAR.B Rd	1		
	SHAR.W Rd	1		
	SHAR.L ERd	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				
SUBX	SUBX #xx:8, Rd	1				
	SUBX. Rs, Rd	1				
TRAPA	TRAPA #xx:2	2	1	2		
XOR	XOR.B #xx:8, Rd	1				
	XOR.B Rs, Rd	1				
	XOR.W #xx:16, Rd	2				
	XOR.W Rs, Rd	1				
	XOR.L #xx:32, ERd	3				
	XOR.L ERs, ERd	2				
XORC	XORC #xx:8, CCR	1				

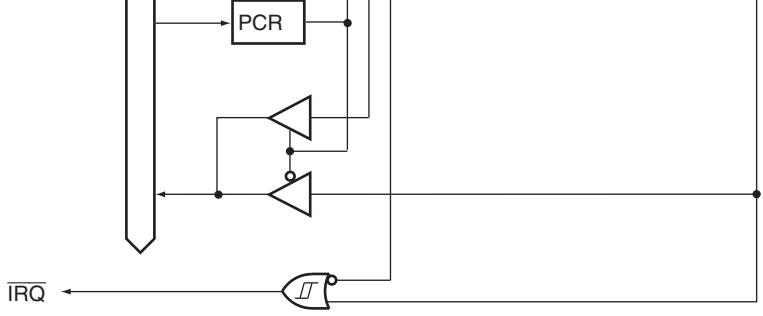
Notes: 1. n: Specified value in R4L and R4. The source and destination operands are a
n+1 times respectively.
2. Cannot be used in this LSI.

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



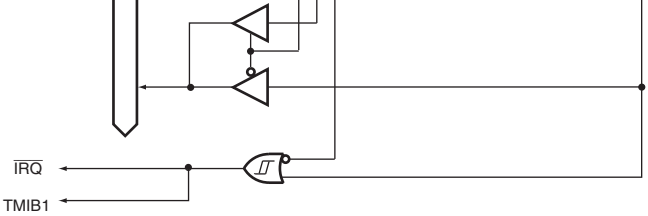
[Legend]
PUCR: Port pull-up control register
PMR: Port mode register
PDR: Port data register
PCR: Port control register

Figure B.1 Port 1 Block Diagram (P17)



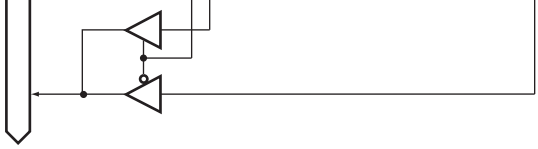
[Legend]
PUCR: Port pull-up control register
PMR: Port mode register
PDR: Port data register
PCR: Port control register

Figure B.2 Port 1 Block Diagram (P14, P16)



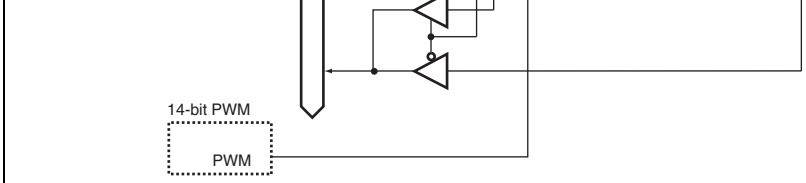
[Legend]
PUCR : Port pull-up control register
PMR : Port mode register
PDR : Port data register
PCR : Port control register

Figure B.3 Port 1 Block Diagram (P15)



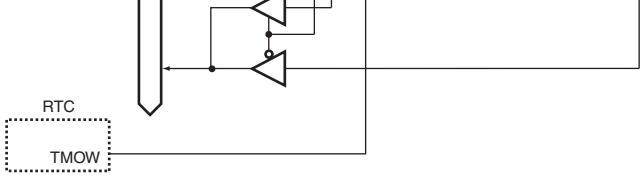
[Legend]
PUCR : Port pull-up control register
PDR : Port data register
PCR : Port control register

Figure B.4 Port 1 Block Diagram (P12)



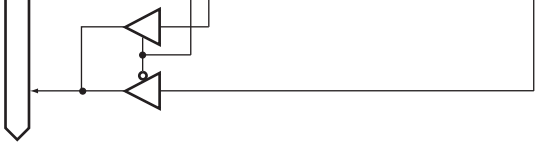
[Legend]
PUCR : Port pull-up control register
PMR : Port mode register
PDR : Port data register
PCR : Port control register

Figure B.5 Port 2 Block Diagram (P11)



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

Figure B.6 Port 1 Block Diagram (P10)



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

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

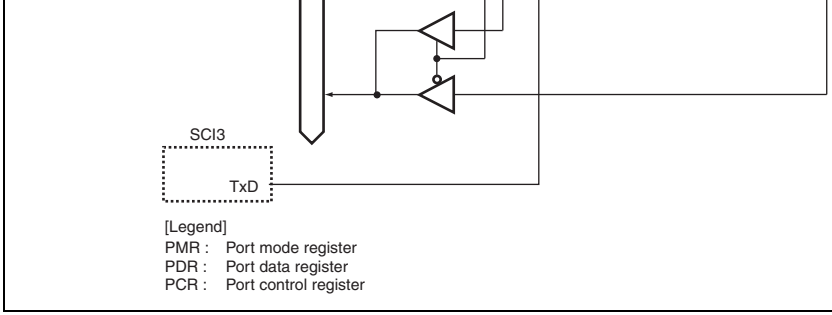


Figure B.8 Port 2 Block Diagram (P22)

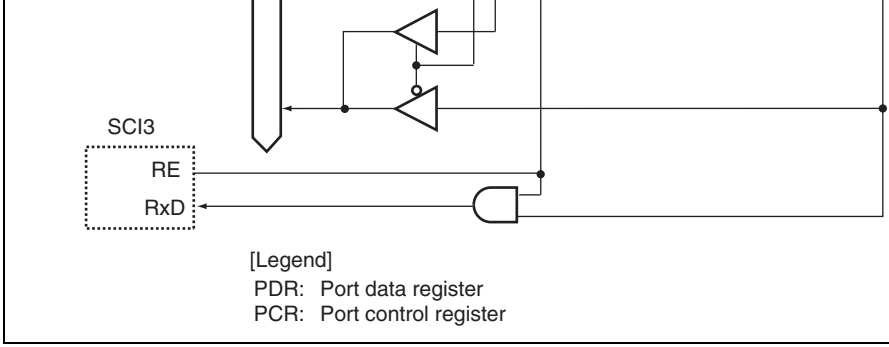


Figure B.9 Port 2 Block Diagram (P21)

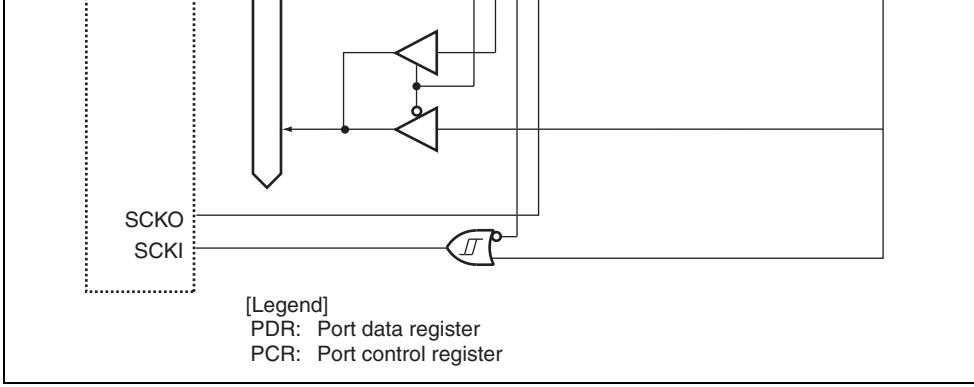
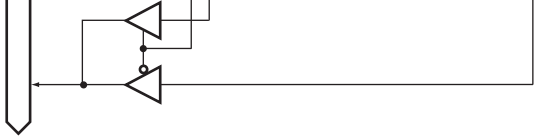
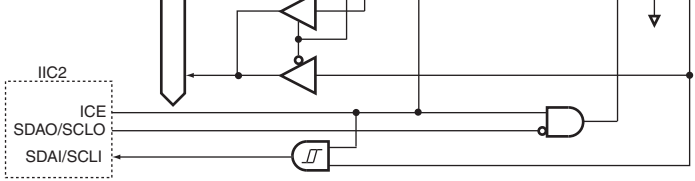


Figure B.10 Port 2 Block Diagram (P20)



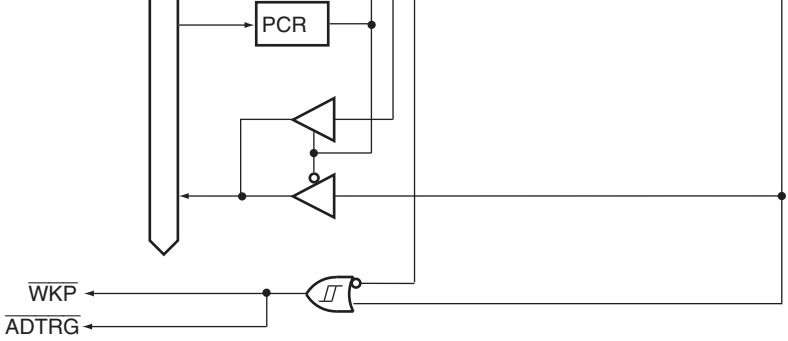
[Legend]
PDR : Port data register
PCR : Port control register

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



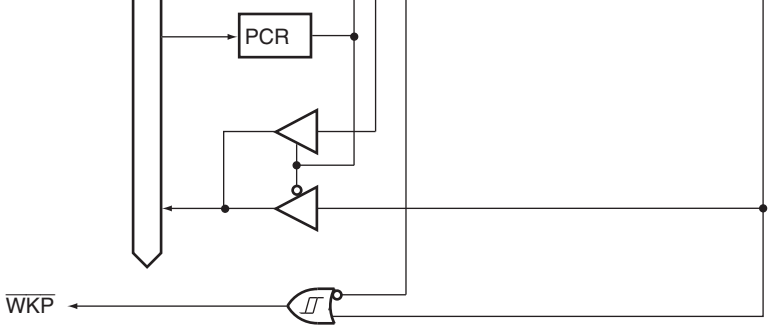
[Legend]
PMR : Port mode register
PDR : Port data register
PCR : Port control register

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



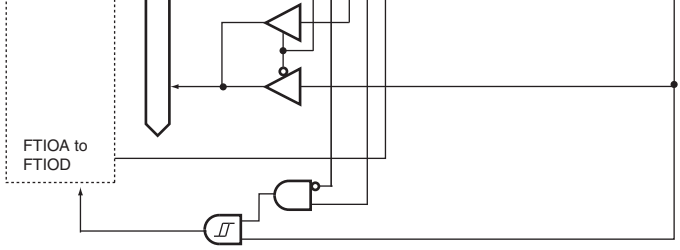
[Legend]
PUCR: Port pull-up control register
PMR: Port mode register
PDR: Port data register
PCR: Port control register

Figure B.13 Port 5 Block Diagram (P55)



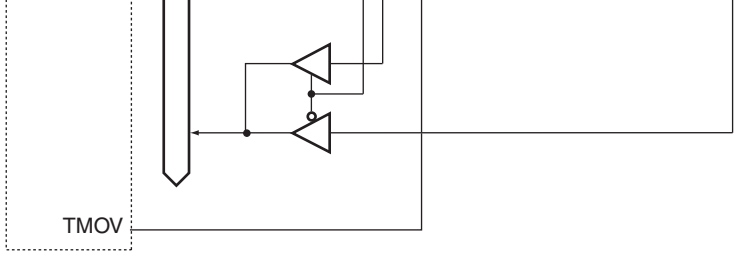
[Legend]
PUCR: Port pull-up control register
PMR: Port mode register
PDR: Port data register
PCR: Port control register

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



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

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



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

Figure B.16 Port 7 Block Diagram (P76)

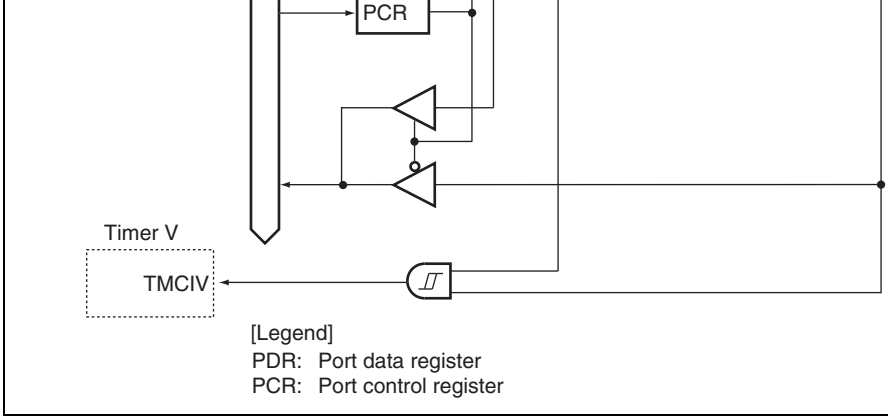


Figure B.17 Port 7 Block Diagram (P75)

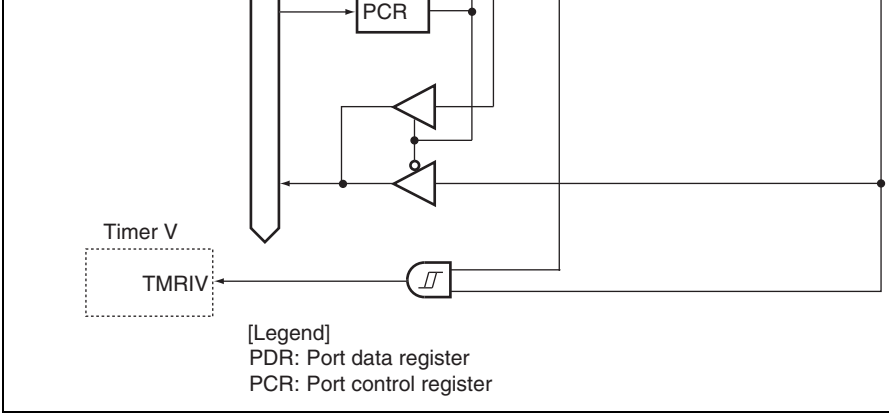


Figure B.18 Port 7 Block Diagram (P74)

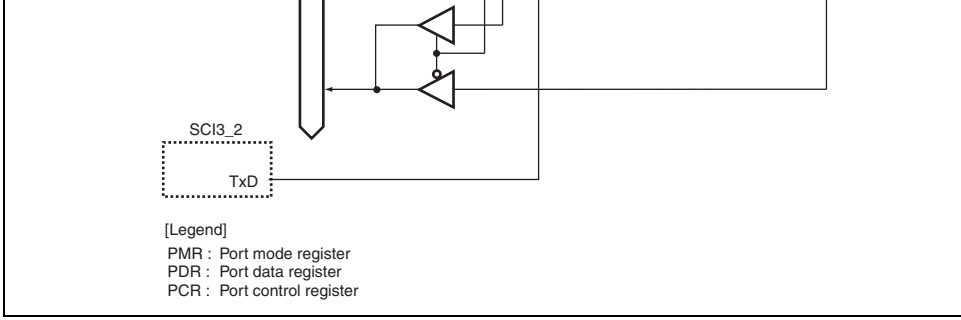
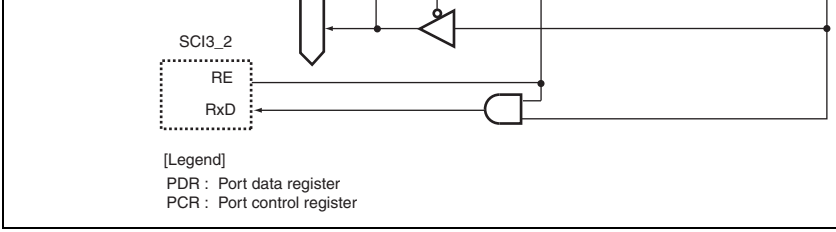
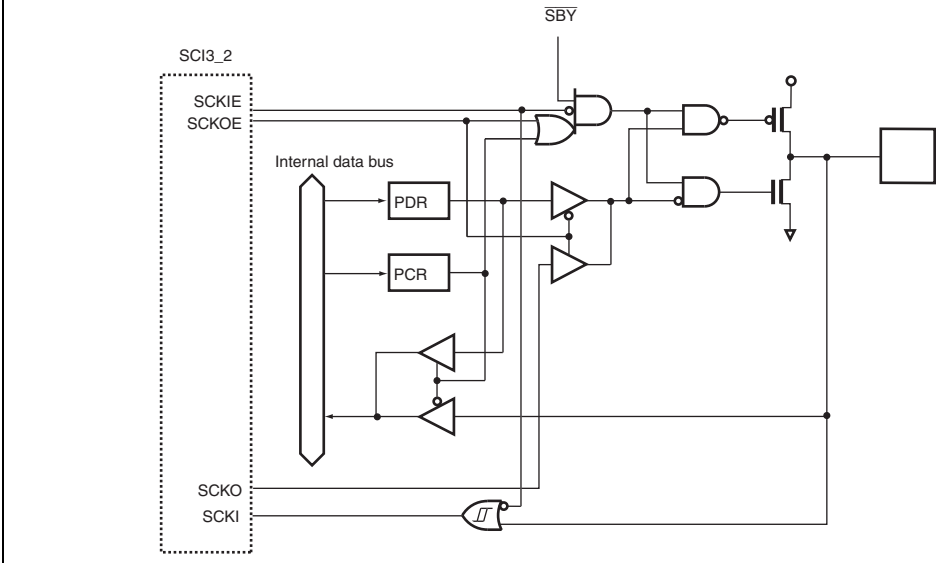


Figure B.19 Port 7 Block Diagram (P72)



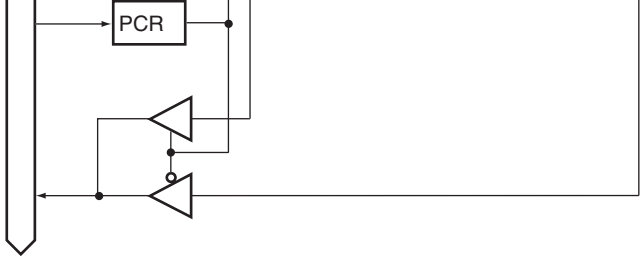
[Legend]
PDR : Port data register
PCR : Port control register

Figure B.20 Port 7 Block Diagram (P71)



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

Figure B.21 Port 7 Block Diagram (P70)



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

Figure B.22 Port 8 Block Diagram (P87 to P85)

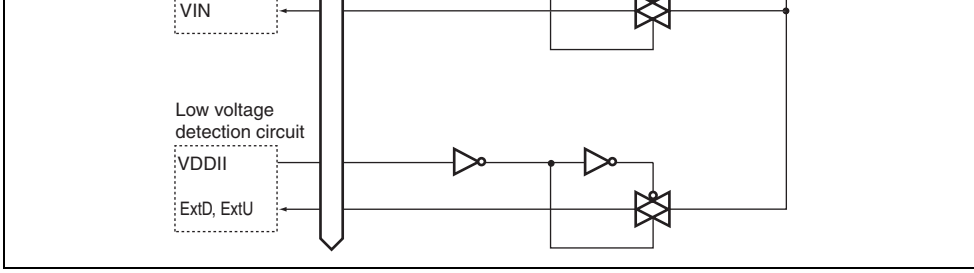


Figure B.23 Port B Block Diagram (PB7 and PB6)

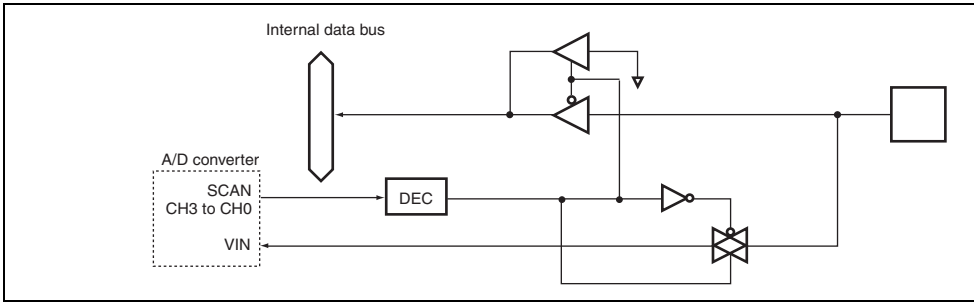
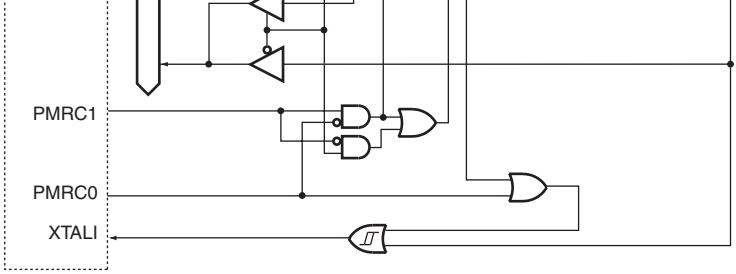
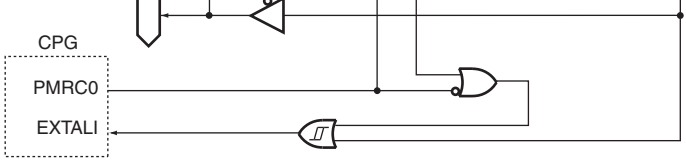


Figure B.24 Port B Block Diagram (PB5 to PB0)



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

Figure B.25 Port B Block Diagram (PC1)



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

Figure B.26 Port B Block Diagram (PC0)

P67 to P60	High impedance	Retained	Retained	High impedance	Functioning	Fu
P76 to P74, P72 to P70	High impedance	Retained	Retained	High impedance	Functioning	Fu
P87 to P85	High impedance	Retained	Retained	High impedance	Functioning	Fu
PB7 to PB0	High impedance	High impedance	High impedance	High impedance	High impedance	Hig im
PC1 and PC0	High impedance	Retained	Retained	High impedance	Functioning	Fu

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

H8/36079LF	3.3 V	Regular	HD64F36079LH	HD64F36079LH
		Wide range	HD64F36079LHW	HD64F36079LHW
		Regular	HD64F36079LFZ	HD64F36079LFZ
		Wide range	HD64F36079LFZW	HD64F36079LFZW
H8/36078GF	5.0 V	Regular	HD64F36078GH	HD64F36078GH
		Wide range	HD64F36078GHW	HD64F36078GHW
		Regular	HD64F36078GFZ	HD64F36078GFZ
		Wide range	HD64F36078GFZW	HD64F36078GFZW
H8/36078LF	3.3 V	Regular	HD64F36078LH	HD64F36078LH
		Wide range	HD64F36078LHW	HD64F36078LHW
		Regular	HD64F36078LFZ	HD64F36078LFZ
		Wide range	HD64F36078LFZW	HD64F36078LFZW

		Regular	HD64F36077LFZ	HD64F36077LFZ
		Wide range	HD64F36077LFZW	HD64F36077LFZ
H8/36074GF	5.0 V	Regular	HD64F36074GH	HD64F36074GH
		Wide range	HD64F36074GHW	HD64F36074GH
		Regular	HD64F36074GFZ	HD64F36074GFZ
		Wide range	HD64F36074GFZW	HD64F36074GFZ
H8/36074LF	3.3 V	Regular	HD64F36074LH	HD64F36074LH
		Wide range	HD64F36074LHW	HD64F36074LH
		Regular	HD64F36074LFZ	HD64F36074LFZ
		Wide range	HD64F36074LFZW	HD64F36074LFZ

Notes: 1. Operating voltage range: 4.5 to 5.5 V (5.0-V models), 3.0 to 3.6 V (3.3-V mod
2. Operating temperature range: Regular specifications: -20 to +75°C, wide-rang specifications: -40 to +85°C
3. The table includes products that are yet to be released. For details, please co of our sales representatives.

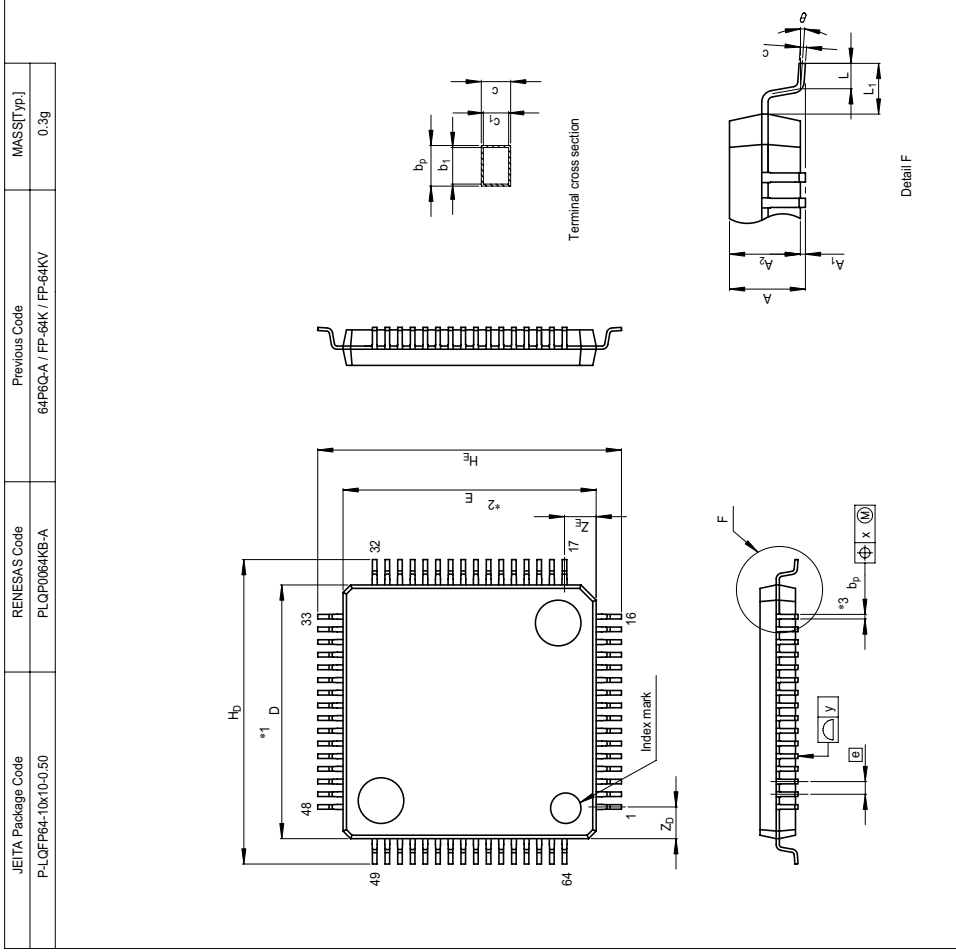


Figure D.1 FP-64K Package Dimensions

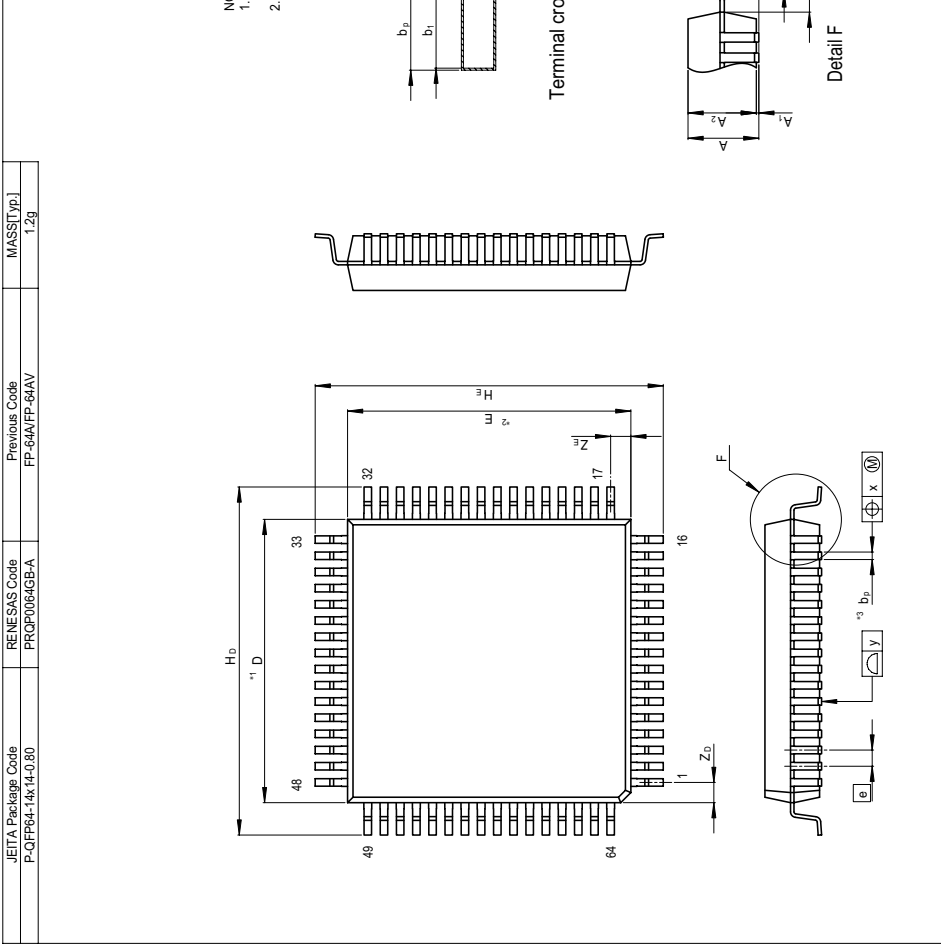


Figure D.2 FP-64A Package Dimensions

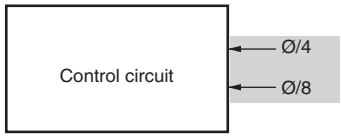
Communication Interface 3 (SCI3)

Table 16.3 Examples of BRR Settings for Various Bit Rates (Asynchronous Mode) (3)

		Operating Frequency ø (MHz)				
		12.288			14	
Bit Rate (bit/s)		n	N	Error (%)	n	N

Section 18 A/D Converter — 368 — Deleted

Figure 18.1 Block Diagram of A/D Converter



Section 21 List of Registers — 405 — Amended

Register Name	Bit 7
:	:
FLPWCR	PDWND
EBR1*4	EB7

Notes:

4. The bit configuration of EBR1 differs according to products configuration of each product, see the list below.

Register Name	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	
EBR1	–	–	–	EB4	EB3	EB2	EB1	

Section 22 Electrical Characteristics	415 to 458	The wide temperature range of Ta = –40 to +8 added to the conditions.
Table 22.3 AC Characteristics	427	Amended

Item	Test Condition	Value Min.	Value Typ
On-chip oscillator oscillation frequency	Vcc = 5.0 V Ta = 25°C FSEL= 1 VCLSEL= 0	19.70	20.0
	FSEL = 1, Ta = -20 to +75°C, VCLSEL = 0	19.40	20.0
	FSEL = 1, Ta = -40 to +85°C, VCLSEL = 0	19.20	20.0
	Vcc = 5.0 V, Ta = 25°C, FSEL = 0, VCLSEL = 0	15.76	16.0
	FSEL = 0, Ta = -20 to +75°C, VCLSEL = 0	15.52	16.0
	FSEL = 0, Ta = -40 to +85°C, VCLSEL = 0	15.36	16.0

Absolute accuracy	–	—	

Table 22.9 Power-Supply-Voltage Detection Circuit Characteristics	434	Amended

Item	Min	Typ	Valu
Reset detection voltage 2*2	3.3	3.6	

Table 22.12 DC Characteristics (1)	440	Amended

Item	Symbol	Min	Typ	Valu
:	:	:	:	:
Pull-up MOS current	$-I_P$	—	60.0	

Amended

Item	Symbol	Min	Typ	Valu
:	:	:	:	:
Active mode current consumption	:	:	:	:
	I_{OPE2}	—	2.8	
		—	2.5	

		Ta = -40 to +85°C, VCLSEL = 0		
		Vcc = 3.3 V, Ta = 25°C, FSEL = 0, VCLSEL = 0	15.76	16.00
		FSEL = 0, Ta = -20 to +75°C, VCLSEL = 0	15.36	16.0
		FSEL = 0, Ta = -40 to +85°C, VCLSEL = 0	15.52	16.0

Table 22.19 Power-Supply-Voltage Detection Circuit Characteristics	453	Amended		
			Values	
		Item	Min.	Min.
		Power-supply falling detection voltage	2.8	2.9
		Power-supply rising detection voltage	2.9	3.0

		Wide range	HD64F36079LHW	HD64F3607
		Regular	HD64F36079LFZ	HD64F3607
		Wide range	HD64F36079LFZW	HD64F3607
H8/36078GF	5.0 V	Regular	HD64F36078GH	HD64F3607
		Wide range	HD64F36078GHW	HD64F3607
		Regular	HD64F36078GFZ	HD64F3607
		Wide range	HD64F36078GFZW	HD64F3607
H8/36078LF	3.3 V	Regular	HD64F36078LH	HD64F3607
		Wide range	HD64F36078LHW	HD64F3607
		Regular	HD64F36078LFZ	HD64F3607
		Wide range	HD64F36078LFZW	HD64F3607

		Wide range	HD64F36077LHW	HD64F36077LH
		Regular	HD64F36077LFZ	HD64F36077LF
		Wide range	HD64F36077LFZW	HD64F36077LF
H8/36074GF	5.0 V	Regular	HD64F36074GH	HD64F36074GH
		Wide range	HD64F36074GHW	HD64F36074GH
		Regular	HD64F36074GFZ	HD64F36074GF
		Wide range	HD64F36074GFZW	HD64F36074GF
H8/36074LF	3.3 V	Regular	HD64F36074LH	HD64F36074LH
		Wide range	HD64F36074LHW	HD64F36074LH
		Regular	HD64F36074LFZ	HD64F36074LF

Address break 63
Addressing modes..... 30
Arithmetic operations instructions..... 21
Asynchronous mode 303
Auto-reload timer operation..... 183

B

Bit manipulation instructions..... 24
Bit rate 296
Bit synchronous circuit..... 363
Block data transfer instructions 28
Boot mode..... 117
Boot program..... 117
Branch instructions 26
Break..... 326
Buffer operation..... 254

C

Clock synchronous mode..... 311
Clock synchronous serial format 354
Complementary PWM mode 244
Condition field..... 29
Condition-code register (CCR)..... 14
CPU 9

E

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^2C bus format.....
I^2C bus interface 2 (IIC2).....
Immediate .....
Initial setting procedure .....
Input capture function .....
Instruction set.....

Logic operations instructions 23
Low-voltage detection circuit 379
LVDI (interrupt by low voltage detect) circuit 389, 390
LVDR (reset by low voltage detect) circuit 387

M

Memory indirect 32
Memory map 11
Module standby function 106
Multiprocessor communication function 318

N

NMI interrupt 56
Noise canceler 357

O

On-board programming modes 117
Operation field 29
Overrun error 307
Program/program-verify .....
Program-counter relative .....
Programmer mode .....
Programming units .....
Programming/erasing in user program mode .....
PWM mode .....

R

Realtime clock (RTC) .....
Register direct .....
Register field .....
Register indirect .....
Register indirect with displacement
Register indirect with post-increme
Register indirect with pre-decremer
Register settings .....
Registers
ABRKCR 64, 400,
ABRKSR 67, 400,
ADCR 372, 400,
ADCSR 371, 399,
ADDRA 370, 399,
ADDRB 370, 399,
ADDRC 370, 399,
ADDRD 370, 399,
BARE 67, 400,
BARH 67, 400,

GRB 213, 396, 403, 409
GRC 213, 396, 403, 409
GRD 213, 396, 403, 409
ICCR1 ..... 332, 398, 405, 410
ICCR2 ..... 335, 398, 405, 410
ICDRR 344, 398, 405, 410
ICDRS 344
ICDRT 344, 398, 405, 410
ICIER 338, 398, 405, 410
ICMR 336, 398, 405, 410
ICSR 340, 398, 405, 410
IEGR1 ..... 48, 401, 407, 413
IEGR2 ..... 49, 401, 407, 413
IENR1 ..... 50, 401, 407, 413
IENR2 ..... 51, 401, 407, 413
IRR1 ..... 51, 401, 407, 413
IRR2 ..... 53, 402, 407, 413
IWPR 53, 402, 407, 413
LVDCR 382, 397, 404, 410
LVDRF 385, 397, 404, 410
LVDSR 384, 397, 404, 410
MSTCR1 ..... 98, 402, 408, 413
MSTCR2 ..... 99, 402, 408, 413
PCR1 ..... 133, 401, 407, 412
PCR2 ..... 137, 401, 407, 412
PCR3 ..... 141, 401, 407, 412
PCR5 ..... 147, 401, 407, 412
PCR6 ..... 151, 401, 407, 412
PCR7 ..... 156, 401, 407, 412
PMR1 ..... 132, 40
PMR3 ..... 138, 40
PMR5 ..... 146, 40
POCR 221, 39
PUCR1 ..... 134, 40
PUCR5 ..... 148, 40
PWCR 283, 40
PWDRL 283, 40
PWDRU 283, 40
RCCR 73, 39
RCTRMDPR 74, 39
RCTRMDR 75, 39
RDR 289, 39
RHRDR 171, 39
RMINDR 170, 39
RSECDR 169, 39
RSR .....
RTCCR1 ..... 173, 39
RTCCR2 ..... 174, 39
RTCCSR 175, 39
RWKDR 172, 39
SAR 343, 39
SCR3 ..... 292, 39
SMR 290, 39
SSR 294, 39
SYSCR1 ..... 94, 40
SYSCR2 ..... 97, 40
TCB1 ..... 182, 39
TCNT 213, 39

TIER 220, 396, 403, 409
TIORA 215, 396, 403, 409
TIORC 216, 396, 403, 409
TLB1 ..... 182, 398, 405, 410
TMB1 ..... 181, 398, 405, 410
TMDR 206, 397, 404, 409
TMWD 278, 400, 406, 411
TOCR 212, 397, 404, 409
TOER 210, 397, 404, 409
TPMR 207, 397, 404, 409
TSR 218, 396, 403, 409
TSTR 206, 397, 404, 409
Reset exception handling 55
Reset synchronous PWM mode 240

S

Sample-and-hold circuit 374
Scan mode 373
Serial communication interface 3 (SCI3) 285
Shift instructions 23
Single mode 373
System clocks .....
System control instructions .....
System prescaler S .....

T

Timer B1 .....
Timer V .....
Timer Z .....
Transfer rate .....
Trap instruction .....

V

Vector address .....

W

Watchdog timer .....
Waveform output .....
Waveform output by compare matc
WKP5 to WKP0 interrupts .....

Renesas 16-Bit Single-Chip Microcomputer
Hardware Manual
H8/36079 Group, H8/36077 Group

Publication Date:	Rev.1.00, Sep. 16, 2005
	Rev.3.00, Sep. 10, 2007
Published by:	Sales Strategic Planning Div.
	Renesas Technology Corp.
Edited by:	Customer Support Department
	Global Strategic Communication Div.
	Renesas Solutions Corp.

RENESAS SALES OFFICES

http://www.ren

Refer to "**http://www.renesas.com/en/network**" for the latest and detailed information.

Renesas Technology America, Inc.
450 Holger Way, San Jose, CA 95134-1368, U.S.A
Tel: <1> (408) 382-7500, Fax: <1> (408) 382-7501

Renesas Technology Europe Limited
Dukes Meadow, Millboard Road, Bourne End, Buckinghamshire, SL8 5FH, U.K.
Tel: <44> (1628) 585-100, Fax: <44> (1628) 585-900

Renesas Technology (Shanghai) Co., Ltd.
Unit 204, 205, AZIACenter, No.1233 Lujiazui Ring Rd, Pudong District, Shanghai, China 200120
Tel: <86> (21) 5877-1818, Fax: <86> (21) 6887-7898

Renesas Technology Hong Kong Ltd.
7th Floor, North Tower, World Finance Centre, Harbour City, 1 Canton Road, Tsimshatsui, Kowloon, Hong Kong
Tel: <852> 2265-6688, Fax: <852> 2730-6071

Renesas Technology Taiwan Co., Ltd.
10th Floor, No.99, Fushing North Road, Taipei, Taiwan
Tel: <886> (2) 2715-2888, Fax: <886> (2) 2713-2999

Renesas Technology Singapore Pte. Ltd.
1 Harbour Front Avenue, #06-10, Keppel Bay Tower, Singapore 098632
Tel: <65> 6213-0200, Fax: <65> 6278-8001

Renesas Technology Korea Co., Ltd.
Kukje Center Bldg. 18th Fl., 191, 2-ka, Hangang-ro, Yongsan-ku, Seoul 140-702, Korea
Tel: <82> (2) 796-3115, Fax: <82> (2) 796-2145

Renesas Technology Malaysia Sdn. Bhd
Unit 906, Block B, Menara Amcorp, Amcorp Trade Centre, No.18, Jalan Persiaran Barat, 46050 Petaling Jaya, Selangor Darul Ehsan,
Tel: <603> 7955-9390, Fax: <603> 7955-9510

H8/36079 Group, H8/36077 Group
Hardware Manual

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

REJ09B0216-0300

X-ON Electronics

Largest Supplier of Electrical and Electronic Components

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

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

Other Similar products are found below :

MB90F036APMC-GSE1 MB90F342CASPMC-GSE1 MB90F345CESPMC-GE1 MB90F349CAPFR-GSE1 MB90F428GCPFR-GSE1 MB90F462APFM-GE1 MB90F462APMC-G-SNE1 MB90F497GPF-GE1 MB90F546GSPFR-GE1 MB90F947APFR-GS-SPE1 MB96F346RSBPMC-GS-N2E2 MB96F683RBPMC-GSAE1 R5F11BGEAFB#30 DF3026XBL25V S912ZVFP64F1VLL R4F24268NVRFQV R5F107DEGSP#X0 R5F11B7EANA#U0 R5F21172DSP#U0 M30622F8PGP#U3C MB90092PF-G-BNDE1 MB90F335APMC1-G-SPE1 MB90F342CASPFR-GS-N2E1 MB90F345CAPFR-GSE1 MB90F543GPF-GE1 MB90F546GSPF-GE1 MB90F568PMCR-GE1 MB90F594APFR-GE1 MB90F882ASPMC-GE1 MB96F346RSAPQCR-GS-N2E2 MB96F387RSBPMC-GSE2 MB96F387RSBPMC-GS-N2E2 MB96F395RSAPMC-GSE2 MB96F623RBPMC1-GSE1 MB96F646RBPMC-GSE1 XE167F96F66LACFXUMA1 MB96F696RBPMC-GSAE1 MB96F018RBPMC-GSE1 MB90F962SPMCR-GE1 MB90F867ASPFR-GE1 MB90F543GPF-G-FLE1 MB90F345CESPF-GE1 M30290FCHP#U3A DF2239FA20IV HD64F3672FPV R5F104AEASP#V0 R5F100BCANA#U0 R5F100BFANA#U0 S9S12H256J2VFVER R5F100ACASP#V0