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

μPD780308, 780308Y Subseries

8-bit Single-Chip Microcontrollers

μPD780306
μPD780308
μPD78P0308
μPD780306Y
μPD780308Y
μPD78P0308Y

Document No. U11377EJ3V0UD00 (3rd edition)
Date Published October 2006 N CP(K)

[MEMO]

NOTES FOR CMOS DEVICES

① VOLTAGE APPLICATION WAVEFORM AT INPUT PIN

Waveform distortion due to input noise or a reflected wave may cause malfunction. If the input of the CMOS device stays in the area between V_{IL} (MAX) and V_{IH} (MIN) due to noise, etc., the device may malfunction. Take care to prevent chattering noise from entering the device when the input level is fixed, and also in the transition period when the input level passes through the area between V_{IL} (MAX) and V_{IH} (MIN).

② HANDLING OF UNUSED INPUT PINS

Unconnected CMOS device inputs can be cause of malfunction. If an input pin is unconnected, it is possible that an internal input level may be generated due to noise, etc., causing malfunction. CMOS devices behave differently than Bipolar or NMOS devices. Input levels of CMOS devices must be fixed high or low by using pull-up or pull-down circuitry. Each unused pin should be connected to V_{DD} or GND via a resistor if there is a possibility that it will be an output pin. All handling related to unused pins must be judged separately for each device and according to related specifications governing the device.

③ PRECAUTION AGAINST ESD

A strong electric field, when exposed to a MOS device, can cause destruction of the gate oxide and ultimately degrade the device operation. Steps must be taken to stop generation of static electricity as much as possible, and quickly dissipate it when it has occurred. Environmental control must be adequate. When it is dry, a humidifier should be used. It is recommended to avoid using insulators that easily build up static electricity. Semiconductor devices must be stored and transported in an anti-static container, static shielding bag or conductive material. All test and measurement tools including work benches and floors should be grounded. The operator should be grounded using a wrist strap. Semiconductor devices must not be touched with bare hands. Similar precautions need to be taken for PW boards with mounted semiconductor devices.

④ STATUS BEFORE INITIALIZATION

Power-on does not necessarily define the initial status of a MOS device. Immediately after the power source is turned ON, devices with reset functions have not yet been initialized. Hence, power-on does not guarantee output pin levels, I/O settings or contents of registers. A device is not initialized until the reset signal is received. A reset operation must be executed immediately after power-on for devices with reset functions.

⑤ POWER ON/OFF SEQUENCE

In the case of a device that uses different power supplies for the internal operation and external interface, as a rule, switch on the external power supply after switching on the internal power supply. When switching the power supply off, as a rule, switch off the external power supply and then the internal power supply. Use of the reverse power on/off sequences may result in the application of an overvoltage to the internal elements of the device, causing malfunction and degradation of internal elements due to the passage of an abnormal current.

The correct power on/off sequence must be judged separately for each device and according to related specifications governing the device.

⑥ INPUT OF SIGNAL DURING POWER OFF STATE

Do not input signals or an I/O pull-up power supply while the device is not powered. The current injection that results from input of such a signal or I/O pull-up power supply may cause malfunction and the abnormal current that passes in the device at this time may cause degradation of internal elements.

Input of signals during the power off state must be judged separately for each device and according to related specifications governing the device.

These commodities, technology or software, must be exported in accordance
with the export administration regulations of the exporting country.
Diversion contrary to the law of that country is prohibited.

- **The information in this document is current as of January, 2006. The information is subject to change without notice. For actual design-in, refer to the latest publications of NEC Electronics data sheets or data books, etc., for the most up-to-date specifications of NEC Electronics products. Not all products and/or types are available in every country. Please check with an NEC Electronics sales representative for availability and additional information.**
- No part of this document may be copied or reproduced in any form or by any means without the prior written consent of NEC Electronics. NEC Electronics assumes no responsibility for any errors that may appear in this document.
- NEC 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 NEC Electronics products listed in this document or any other liability arising from the use of such products. No license, express, implied or otherwise, is granted under any patents, copyrights or other intellectual property rights of NEC Electronics or others.
- Descriptions of circuits, software and other related information in this document are provided for illustrative purposes in semiconductor product operation and application examples. The incorporation of these circuits, software and information in the design of a customer's equipment shall be done under the full responsibility of the customer. NEC Electronics assumes no responsibility for any losses incurred by customers or third parties arising from the use of these circuits, software and information.
- While NEC Electronics endeavors to enhance the quality, reliability and safety of NEC Electronics products, customers agree and acknowledge that the possibility of defects thereof cannot be eliminated entirely. To minimize risks of damage to property or injury (including death) to persons arising from defects in NEC Electronics products, customers must incorporate sufficient safety measures in their design, such as redundancy, fire-containment and anti-failure features.
- NEC Electronics products are classified into the following three quality grades: "Standard", "Special" and "Specific".

  The "Specific" quality grade applies only to NEC Electronics products developed based on a customer-designated "quality assurance program" for a specific application. The recommended applications of an NEC Electronics product depend on its quality grade, as indicated below. Customers must check the quality grade of each NEC Electronics product before using it in a particular application.

  "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.

  "Special": 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, life support systems and medical equipment for life support, etc.

The quality grade of NEC Electronics products is "Standard" unless otherwise expressly specified in NEC Electronics data sheets or data books, etc. If customers wish to use NEC Electronics products in applications not intended by NEC Electronics, they must contact an NEC Electronics sales representative in advance to determine NEC Electronics' willingness to support a given application.

(Note)

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

(2) "NEC Electronics products" means any product developed or manufactured by or for NEC Electronics (as defined above).

M8E 02.11-1

PREFACE

Readers

This manual has been prepared for user engineers who understand the functions of the μPD780308 and 780308Y Subseries and design and develop its application systems and programs.

- μPD780308 Subseries: μPD780306, 780308, 78P0308, 780306(A), 780308(A)
- μPD780308Y Subseries: μPD780306Y, 780308Y, 78P0308Y

Purpose

This manual is intended for users to understand the functions described in the Organization below.

Organization

The μPD780308, 780308Y Subseries manual is separated into two parts: this manual and the instruction edition (common to the 78K/0 Series).

μPD780308, 780308Y Subseries User's Manual (This Manual)	78K/0 Series Instructions User's Manual
• Pin functions	• CPU functions
• Internal block functions	• Instruction set
• Interrupt	• Explanation of each instruction
• Other on-chip peripheral functions	
• Electrical specifications	

How to Read This Manual

Before reading this manual, you should have general knowledge of electric and logic circuits and microcontrollers.

- To those who use this manual as the manual of the μPD780306(A) and 780308(A):
  → The μPD780306 and 780308, and μPD780306(A) and 780308(A) differ only in their quality grade. Regarding (A) models read the product name as follows:
  μPD780306 → μPD780306(A)
  μPD780308 → μPD780308(A)
- When you want to understand the functions in general:
  → Read this manual in the order of the contents. The mark <R> shows major revised points. The revised points can be easily searched by copying an "<R>" in the PDF file and specifying it in the "Find what:" field.

- How to interpret the register format:
  → For the circled bit number, the bit name is defined as a reserved word in the RA78K0, and is defined as an sfr variable using the #pragma sfr directive in the CC78K0.
- When you know a register name and want to confirm its details:
  → Read **APPENDIX B REGISTER INDEX**.
- To know the μPD780308 and 780308Y Subseries instruction function in detail:
  → Refer to the **78K/0 Series Instructions User's Manual (U12326E)**.
- To know the electrical specifications of the μPD780308 and 780308Y Subseries:
  → Refer to **CHAPTER 25 ELECTRICAL SPECIFICATIONS**.

Caution The application examples in this manual are for the "standard" quality grade for general-purpose electronic systems. If the examples in this manual are to be used for applications where a quality higher than that of the "standard" quality grade is required, determine the required quality grade of the respective components and circuits to be used.

Chapter Organization: This manual divides the descriptions for the μPD780308 and 780308Y Subseries into different chapters as shown below. Read only the chapters related to the device you use.

Chapter		μPD780308 Subseries	μPD780308Y Subseries
Chapter 1	Outline (μPD780308 Subseries)	√	—
Chapter 2	Outline (μPD780308Y Subseries)	—	√
Chapter 3	Pin Function (μPD780308 Subseries)	√	—
Chapter 4	Pin Function (μPD780308Y Subseries)	—	√
Chapter 5	CPU Architecture	√	√
Chapter 6	Port Functions	√	√
Chapter 7	Clock Generator	√	√
Chapter 8	16-bit Timer/Event Counter	√	√
Chapter 9	8-bit Timer/Event Counter	√	√
Chapter 10	Watch Timer	√	√
Chapter 11	Watchdog Timer	√	√
Chapter 12	Clock Output Controller	√	√
Chapter 13	Buzzer Output Controller	√	√
Chapter 14	A/D Converter	√	√
Chapter 15	Serial Interface Channel 0 (μPD780308 Subseries)	√	—
Chapter 16	Serial Interface Channel 0 (μPD780308Y Subseries)	—	√
Chapter 17	Serial Interface Channel 2	√	√
Chapter 18	Serial Interface Channel 3	√	√
Chapter 19	LCD Controller/Driver	√	√
Chapter 20	Interrupt and Test Functions	√	√
Chapter 21	Standby Function	√	√
Chapter 22	Reset Function	√	√
Chapter 23	μPD78P0308, μPD78P0308Y	√	√
Chapter 24	Instruction Set	√	√
Chapter 25	Electrical Specifications	√	√
Chapter 26	Package Drawings	√	√
Chapter 27	Recommended Soldering Conditions	√	√

Differences between μPD780308 and μPD780308Y Subseries:

The μPD780308 and μPD780308Y Subseries are different in the following functions of serial interface channel 0.

Modes of Serial Interface Channel 0	μPD780308 Subseries	μPD780308Y Subseries
3-wire serial I/O mode	√	√
2-wire serial I/O mode	√	√
SBI (serial bus interface) mode	√	—
I^2C (Inter IC) bus mode	—	√

√: Supported

—: Not supported

Conventions

Data significance:	Higher digits on the left and lower digits on the right
Active low representation:	$\overline{\times\times\times}$ (overscore over pin or signal name)
Note:	Footnote for item marked with **Note** in the text
Caution:	Information requiring particular attention
Remark:	Supplementary information
Numerical representation:	Binary ··· ×××× or ××××B
	Decimal ··· ××××
	Hexadecimal ··· ××××H

<R> **Related Documents** The related documents indicated in this publication may include preliminary versions. However, preliminary versions are not marked as such.

Documents Related to Devices

Document Name	Document No.
μPD780308, 780308Y Subseries User's Manual	This manual
78K/0 Series Instructions User's Manual	U12326E
78K/0 Series Basic (III) Application Note	U10182E

Caution The related documents listed above are subject to change without notice. Be sure to use the latest version of each document for designing.

Documents Related to Development Tools (Software) (User's Manuals)

Document Name		Document No.
RA78K0 Ver. 3.80 Assembler Package	Operation	U17199E
	Language	U17198E
	Structured Assembly Language	U17197E
CC78K0 Ver. 3.70 C Compiler	Operation	U17201E
	Language	U17200E
SM78K Series Ver. 2.52 System Simulator	Operation	U16768E
ID78K Series Integrated Debugger Ver. 2.30 or Later	Operation (Windows^{TM} Based)	U15185E
PM plus Ver. 5.20		U16934E

Documents Related to Development Tools (Hardware) (User's Manuals)

Document Name	Document No.
IE-78K0-NS In-Circuit Emulator	U13731E
IE-78K0-NS-A In-Circuit Emulator	U14889E
IE-780308-NS-EM1 Emulation Board	U13304E
IE-78001-R-A In-Circuit Emulator	U14142E
IE-780308-R-EM Emulation Board	U11362E

Documents Related to PROM Writing (User's Manuals)

Document Name		Document No.
PG-1500 PROM Programmer		U11940E
PG-1500 Controller	PC-9800 Series (MS-DOS^{TM}) Based	EEU-1291
	IBM PC Series (PC-DOS^{TM}) Based	U10540E

Other Documents

Document Name	Document No.
SEMICONDUCTOR SELECTION GUIDE – Products and Packages –	X13769X
Semiconductor Device Mount Manual	**Note**
Quality Grades on NEC Semiconductor Devices	C11531E
NEC Semiconductor Device Reliability/Quality Control System	C10983E
Guide to Prevent Damage for Semiconductor Devices by Electrostatic Discharge (ESD)	C11892E

Note See the "Semiconductor Device Mount Manual" website (http://www.necel.com/pkg/en/mount/index.html).

Caution The related documents listed above are subject to change without notice. Be sure to use the latest version of each document for designing.

CONTENTS

CHAPTER 1 OUTLINE (μPD780308 SUBSERIES)

1.1 Features

○ On-chip high-capacity ROM and RAM

Type / Part Number	Program Memory (ROM)	Data Memory		
		Internal High-Speed RAM	Internal Expansion RAM	LCD RAM
μPD780306	48 KB	1024 bytes	1024 bytes	40 × 4 bits
μPD780308	60 KB			
μPD78P0308	60 KB[Note]			

Note The capacity of internal PROM can be changed by means of the internal memory size switching register (IMS).

○ Minimum instruction execution time changeable from high speed (0.4 μs: @ 5.0 MHz operation with main system clock) to ultra-low speed (122 μs: @ 32.768 kHz operation with subsystem clock)

○ Instruction set suited to system control
- Bit manipulation possible in all address spaces
- Multiply and divide instructions

○ Fifty-seven I/O ports (including alternate-function pins for segment signal output)

○ LCD controller/driver
- Segment signal output: Max. 40
- Common signal output: Max. 4
- Bias: 1/2, 1/3 bias switching possible
- Power supply voltage: V_{DD} = 2.0 to 5.5 V (can operate in all modes)

○ 8-bit resolution A/D converter: 8 channels

○ Serial interface: 3 channels
- 3-wire serial I/O/SBI/2-wire serial I/O mode: 1 channel
- 3-wire serial I/O/UART mode: 1 channel
- 3-wire serial I/O mode: 1 channel

○ Timer: 5 channels
- 16-bit timer/event counter: 1 channel
- 8-bit timer/event counter: 2 channels
- Watch timer: 1 channel
- Watchdog timer: 1 channel

○ Twenty-one vectored interrupt sources

○ Two test inputs

○ Two types of on-chip clock oscillators (main system clock and subsystem clock)

○ Power supply voltage: V_{DD} = 2.0 to 5.5 V

1.2 Applications

Cellular phones, CD players, cameras, meters, etc.

1.3 Ordering Information

Part Number	Package	Internal ROM
μPD780306GC-×××-8EU	100-pin plastic LQFP (Fine pitch) (14 × 14)	Mask ROM
μPD780306GC-×××-8EU-A	100-pin plastic LQFP (Fine pitch) (14 × 14)	Mask ROM
μPD780306GF-×××-3BA	100-pin plastic QFP (14 × 20)	Mask ROM
μPD780306GF-×××-3BA-A	100-pin plastic QFP (14 × 20)	Mask ROM
μPD780308GC-×××-8EU	100-pin plastic LQFP (Fine pitch) (14 × 14)	Mask ROM
μPD780308GC-×××-8EU-A	100-pin plastic LQFP (Fine pitch) (14 × 14)	Mask ROM
μPD780308GF-×××-3BA	100-pin plastic QFP (14 × 20)	Mask ROM
μPD780308GF-×××-3BA-A	100-pin plastic QFP (14 × 20)	Mask ROM
μPD780306GF(A)-×××-3BA	100-pin plastic QFP (14 × 20)	Mask ROM
μPD780308GF(A)-×××-3BA	100-pin plastic QFP (14 × 20)	Mask ROM
μPD78P0308GC-8EU	100-pin plastic LQFP (Fine pitch) (14 × 14)	One-time PROM
μPD78P0308GC-8EU-A	100-pin plastic LQFP (Fine pitch) (14 × 14)	One-time PROM
μPD78P0308GF-3BA	100-pin plastic QFP (14 × 20)	One-time PROM
μPD78P0308GF-3BA-A	100-pin plastic QFP (14 × 20)	One-time PROM

Remarks **1.** ××× indicates ROM code suffix.

2. Products with -A at the end of the part number are lead-free products.

1.4 Quality Grade

Part Number	Package	Quality Grade
μPD780306GC-×××-8EU	100-pin plastic LQFP (Fine pitch) (14 × 14)	Standard
μPD780306GC-×××-8EU-A	100-pin plastic LQFP (Fine pitch) (14 × 14)	Standard
μPD780306GF-×××-3BA	100-pin plastic QFP (14 × 20)	Standard
μPD780306GF-×××-3BA-A	100-pin plastic QFP (14 × 20)	Standard
μPD780308GC-×××-8EU	100-pin plastic LQFP (Fine pitch) (14 × 14)	Standard
μPD780308GC-×××-8EU-A	100-pin plastic LQFP (Fine pitch) (14 × 14)	Standard
μPD780308GF-×××-3BA	100-pin plastic QFP (14 × 20)	Standard
μPD780308GF-×××-3BA-A	100-pin plastic QFP (14 × 20)	Standard
μPD780306GF(A)-×××-3BA	100-pin plastic QFP (14 × 20)	Special
μPD780308GF(A)-×××-3BA	100-pin plastic QFP (14 × 20)	Special
μPD78P0308GC-8EU	100-pin plastic LQFP (Fine pitch) (14 × 14)	Standard
μPD78P0308GC-8EU-A	100-pin plastic LQFP (Fine pitch) (14 × 14)	Standard
μPD78P0308GF-3BA	100-pin plastic QFP (14 × 20)	Standard
μPD78P0308GF-3BA-A	100-pin plastic QFP (14 × 20)	Standard

Remarks 1. ××× indicates ROM code suffix.

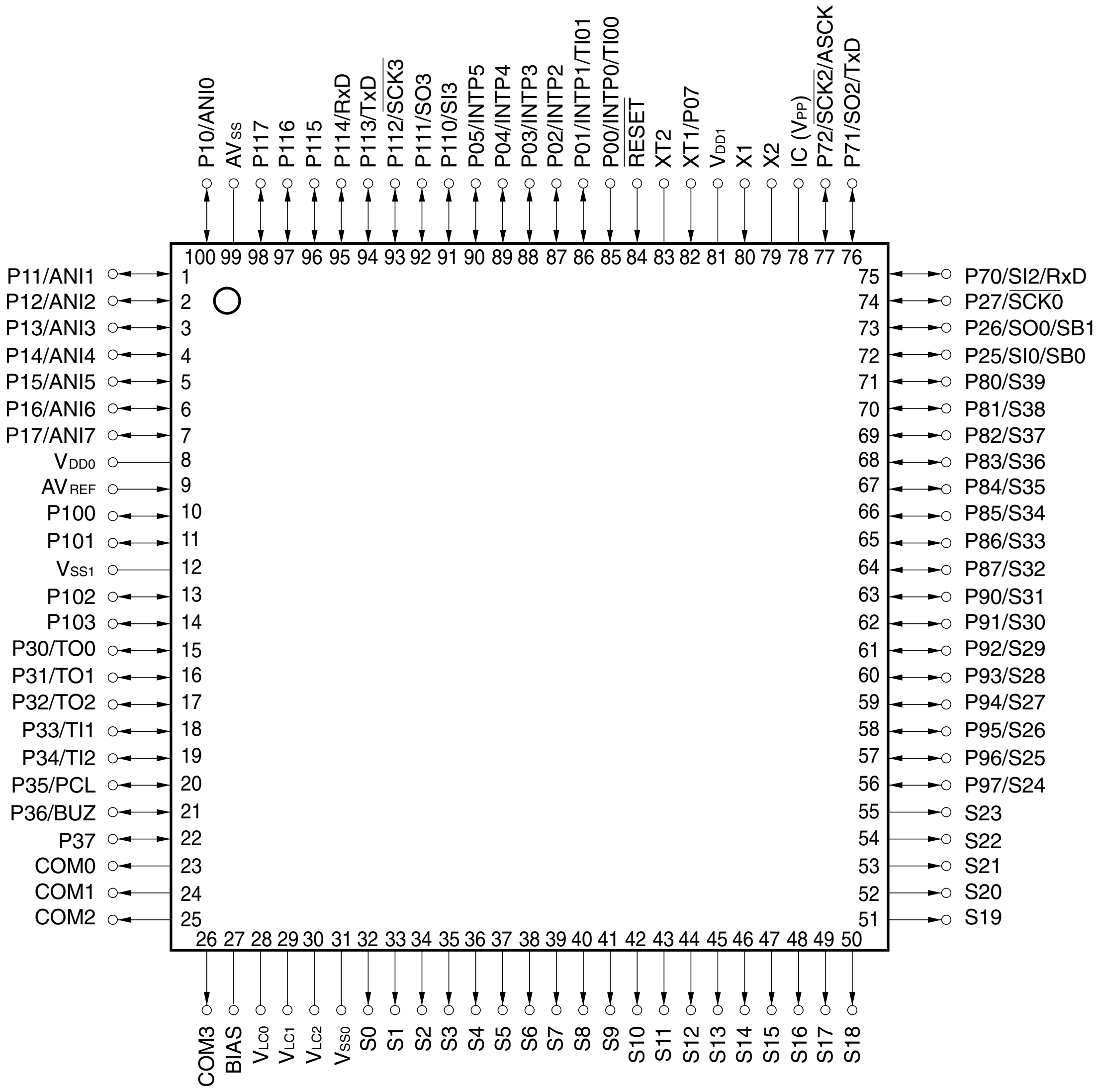
2. Products with -A at the end of the part number are lead-free products.

Please refer to "Quality Grades on NEC Semiconductor Devices" (Document No. C11531E) published by NEC Electronics Corporation to know the specification of the quality grade on the device and its recommended applications.

1.5 Pin Configuration (Top View)

(1) Normal operating mode

100-pin plastic LQFP (Fine pitch) (14 × 14)



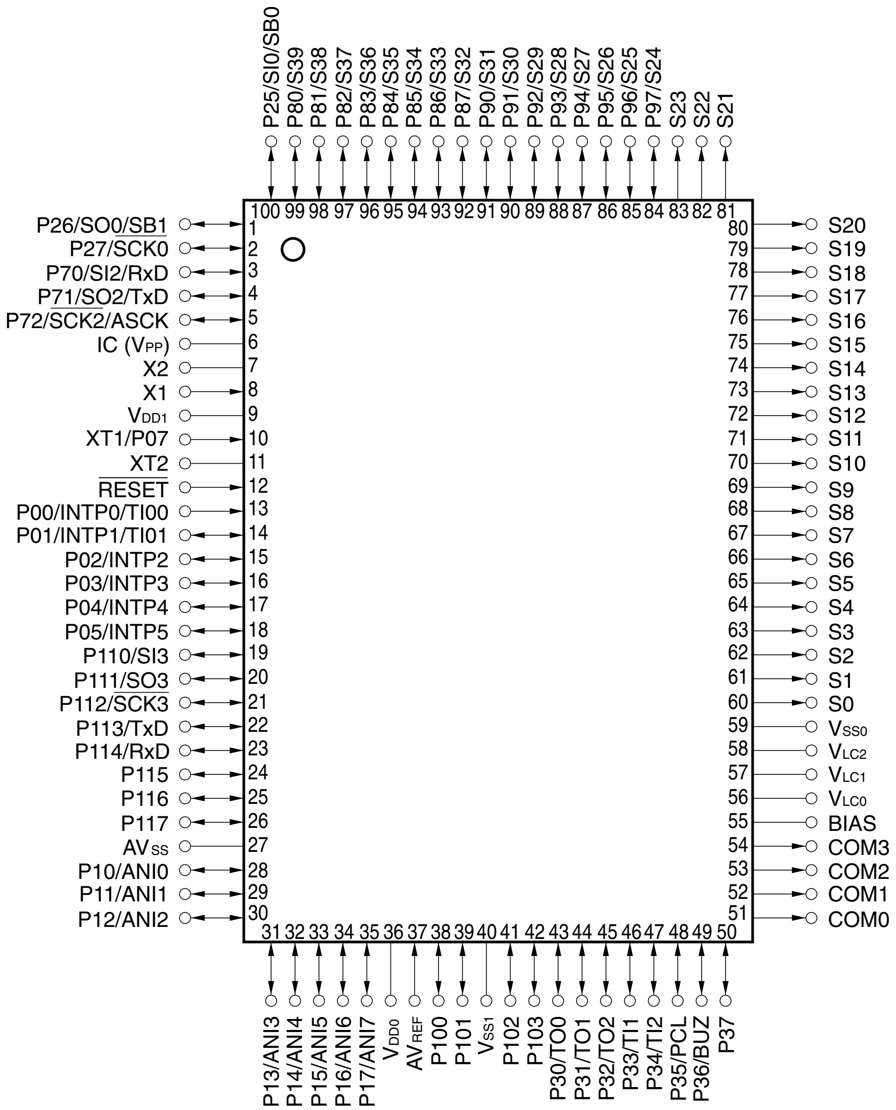
Cautions 1. Connect the IC (Internally Connected) pin directly to V_{SS0} or V_{SS1}.

2. Connect the AV$_{SS}$ pin to V_{SS0}.

Remarks **1.** Pin connection in parentheses is intended for the μPD78P0308.

2. When using the μPD780308 Subseries in an application field where the noise generated from the microcontroller must be reduced, it is recommended to take noise reduction measures by supplying separate power to V_{DD0} and V_{DD1}, and connecting V_{SS0} and V_{SS1} to separate ground lines.

100-pin plastic QFP (14 × 20)



Cautions 1. Connect the IC (Internally Connected) pin directly to V_{SS0} or V_{SS1}.

2. Connect the AV_{SS} pin to V_{SS0}.

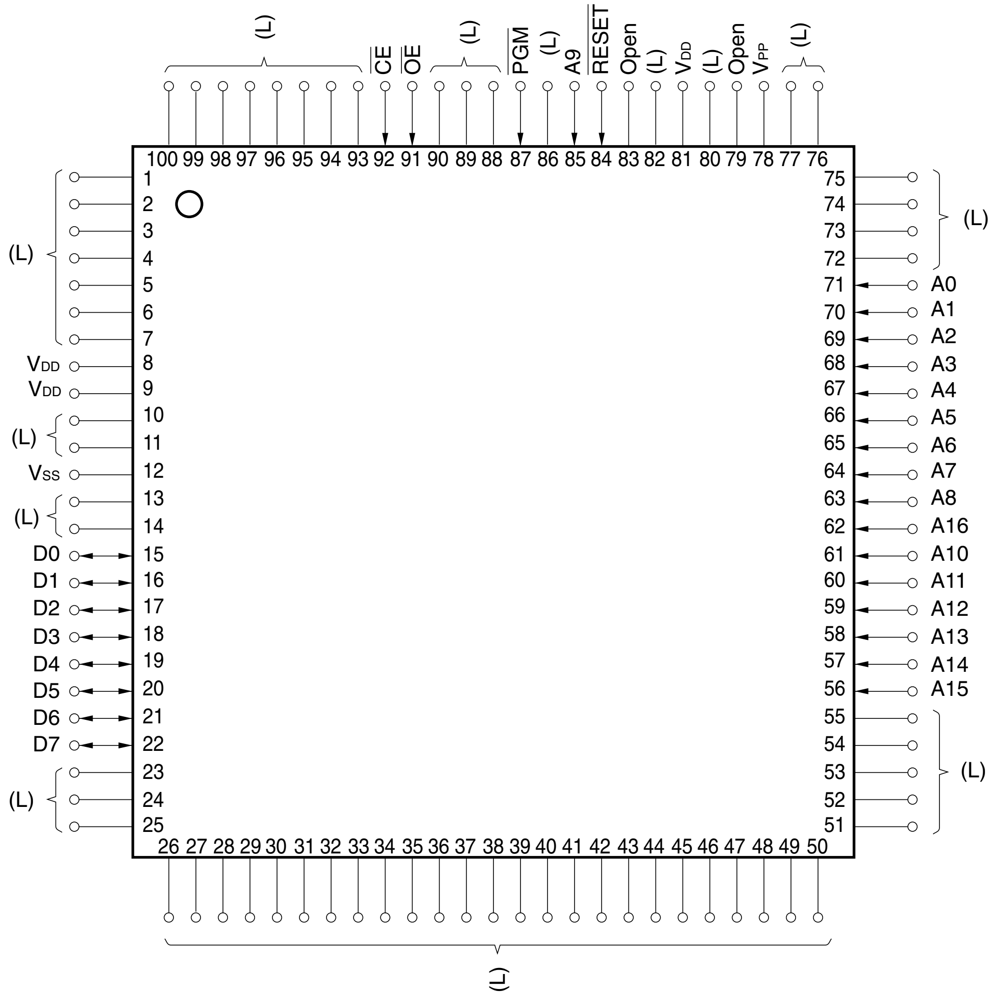
Remarks 1. Pin connection in parentheses is intended for the μPD78P0308.

2. When using the μPD780308 Subseries in an application field where the noise generated from the microcontroller must be reduced, it is recommended to take noise reduction measures by supplying separate power to V_{DD0} and V_{DD1}, and connecting V_{SS0} and V_{SS1} to separate ground lines.

ANI0 to ANI7:	Analog input	PCL:	Programmable clock
ASCK:	Asynchronous serial clock	$\overline{\text{RESET}}$:	Reset
AV_{REF}:	Analog reference voltage	RxD:	Receive data
AV_{SS}:	Analog ground	S0 to S39:	Segment output
BIAS:	LCD power supply bias control	SB0, SB1:	Serial bus
BUZ:	Buzzer clock	$\overline{\text{SCK0}}$, $\overline{\text{SCK2}}$, $\overline{\text{SCK3}}$:	Serial clock
COM0 to COM3:	Common output	SI0, SI2, SI3:	Serial input
INTP0 to INTP5:	Interrupt from peripherals	SO0, SO2, SO3:	Serial output
IC:	Internally connected	TI00, TI01:	Timer input
P00 to P05, P07:	Port 0	TI1, TI2:	Timer input
P10 to P17:	Port 1	TO0 to TO2:	Timer output
P25 to P27:	Port 2	TxD:	Transmit data
P30 to P37:	Port 3	V_{DD0}, V_{DD1}:	Power supply
P70 to P72:	Port 7	V_{LC0} to V_{LC2}:	LCD power supply
P80 to P87:	Port 8	V_{PP}:	Programming power supply
P90 to P97:	Port 9	V_{SS0}, V_{SS1}:	Ground
P100 to P103:	Port 10	X1, X2:	Crystal (main system clock)
P110 to P117:	Port 11	XT1, XT2:	Crystal (subsystem clock)

(2) PROM programming mode

100-pin plastic LQFP (Fine pitch) (14 × 14)



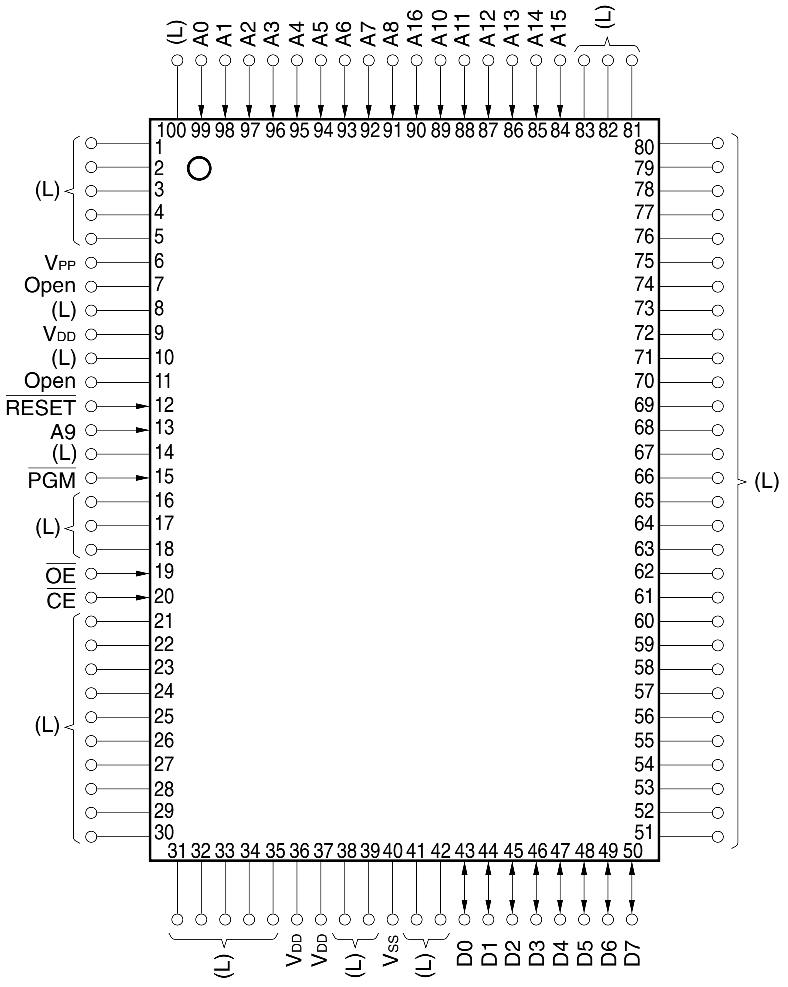
Cautions 1. (L): Independently connect to VSS via a pull-down resistor.

2. VSS: Connect to the ground.

3. $\overline{\text{RESET}}$: Set to the low level.

4. Open: Do not connect anything.

100-pin plastic QFP (14 × 20)



Cautions 1. (L): Independently connect to V_{SS} via a pull-down resistor.

2. V_{SS}: Connect to the ground.

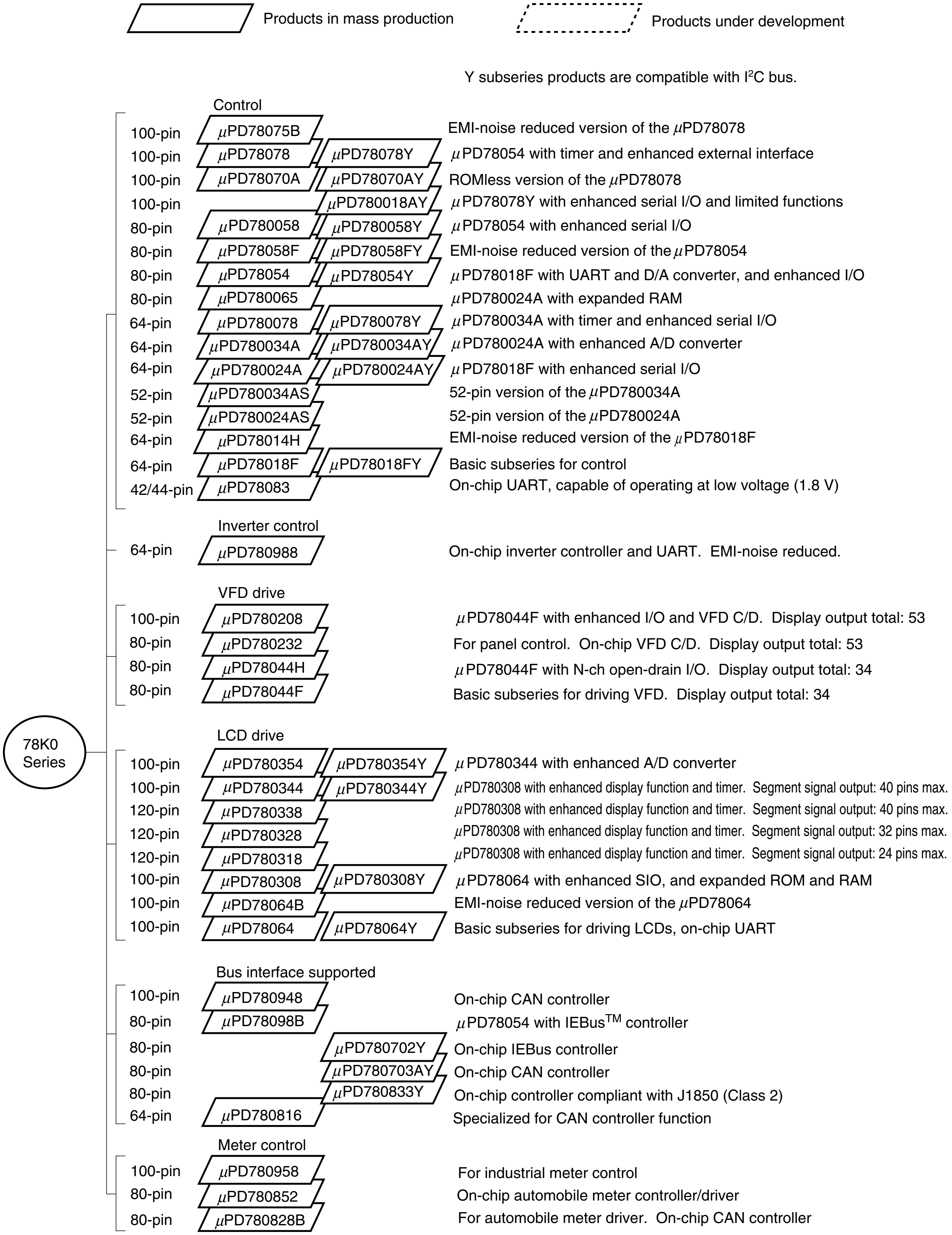
3. $\overline{\text{RESET}}$: Set to the low level.

4. Open: Do not connect anything.

A0 to A16:	Address bus	$\overline{\text{RESET}}$:	Reset
$\overline{\text{CE}}$:	Chip enable	V_{DD}:	Power supply
D0 to D7:	Data bus	V_{PP}:	Programming power supply
$\overline{\text{OE}}$:	Output enable	V_{SS}:	Ground
$\overline{\text{PGM}}$:	Program		

<R> 1.6 78K0 Series Lineup

The products in the 78K0 Series are listed below. The names enclosed in boxes are subseries name.



Products in mass production

Products under development

Y subseries products are compatible with I^2C bus.

78K0 Series

Control			
100-pin	μPD78075B		EMI-noise reduced version of the μPD78078
100-pin	μPD78078	μPD78078Y	μPD78054 with timer and enhanced external interface
100-pin	μPD78070A	μPD78070AY	ROMless version of the μPD78078
100-pin		μPD780018AY	μPD78078Y with enhanced serial I/O and limited functions
80-pin	μPD780058	μPD780058Y	μPD78054 with enhanced serial I/O
80-pin	μPD78058F	μPD78058FY	EMI-noise reduced version of the μPD78054
80-pin	μPD78054	μPD78054Y	μPD78018F with UART and D/A converter, and enhanced I/O
80-pin	μPD780065		μPD780024A with expanded RAM
64-pin	μPD780078	μPD780078Y	μPD780034A with timer and enhanced serial I/O
64-pin	μPD780034A	μPD780034AY	μPD780024A with enhanced A/D converter
64-pin	μPD780024A	μPD780024AY	μPD78018F with enhanced serial I/O
52-pin	μPD780034AS		52-pin version of the μPD780034A
52-pin	μPD780024AS		52-pin version of the μPD780024A
64-pin	μPD78014H		EMI-noise reduced version of the μPD78018F
64-pin	μPD78018F	μPD78018FY	Basic subseries for control
42/44-pin	μPD78083		On-chip UART, capable of operating at low voltage (1.8 V)
Inverter control			
64-pin	μPD780988		On-chip inverter controller and UART. EMI-noise reduced.
VFD drive			
100-pin	μPD780208		μPD78044F with enhanced I/O and VFD C/D. Display output total: 53
80-pin	μPD780232		For panel control. On-chip VFD C/D. Display output total: 53
80-pin	μPD78044H		μPD78044F with N-ch open-drain I/O. Display output total: 34
80-pin	μPD78044F		Basic subseries for driving VFD. Display output total: 34
LCD drive			
100-pin	μPD780354	μPD780354Y	μPD780344 with enhanced A/D converter
100-pin	μPD780344	μPD780344Y	μPD780308 with enhanced display function and timer. Segment signal output: 40 pins max.
120-pin	μPD780338		μPD780308 with enhanced display function and timer. Segment signal output: 40 pins max.
120-pin	μPD780328		μPD780308 with enhanced display function and timer. Segment signal output: 32 pins max.
120-pin	μPD780318		μPD780308 with enhanced display function and timer. Segment signal output: 24 pins max.
100-pin	μPD780308	μPD780308Y	μPD78064 with enhanced SIO, and expanded ROM and RAM
100-pin	μPD78064B		EMI-noise reduced version of the μPD78064
100-pin	μPD78064	μPD78064Y	Basic subseries for driving LCDs, on-chip UART
Bus interface supported			
100-pin	μPD780948		On-chip CAN controller
80-pin	μPD78098B		μPD78054 with IEBus™ controller
80-pin		μPD780702Y	On-chip IEBus controller
80-pin		μPD780703AY	On-chip CAN controller
80-pin		μPD780833Y	On-chip controller compliant with J1850 (Class 2)
64-pin	μPD780816		Specialized for CAN controller function
Meter control			
100-pin	μPD780958		For industrial meter control
80-pin	μPD780852		On-chip automobile meter controller/driver
80-pin	μPD780828B		For automobile meter driver. On-chip CAN controller

Remark VFD (Vacuum Fluorescent Display) is referred to as FIP™ (Fluorescent Indicator Panel) in some documents, but the functions of the two are the same.

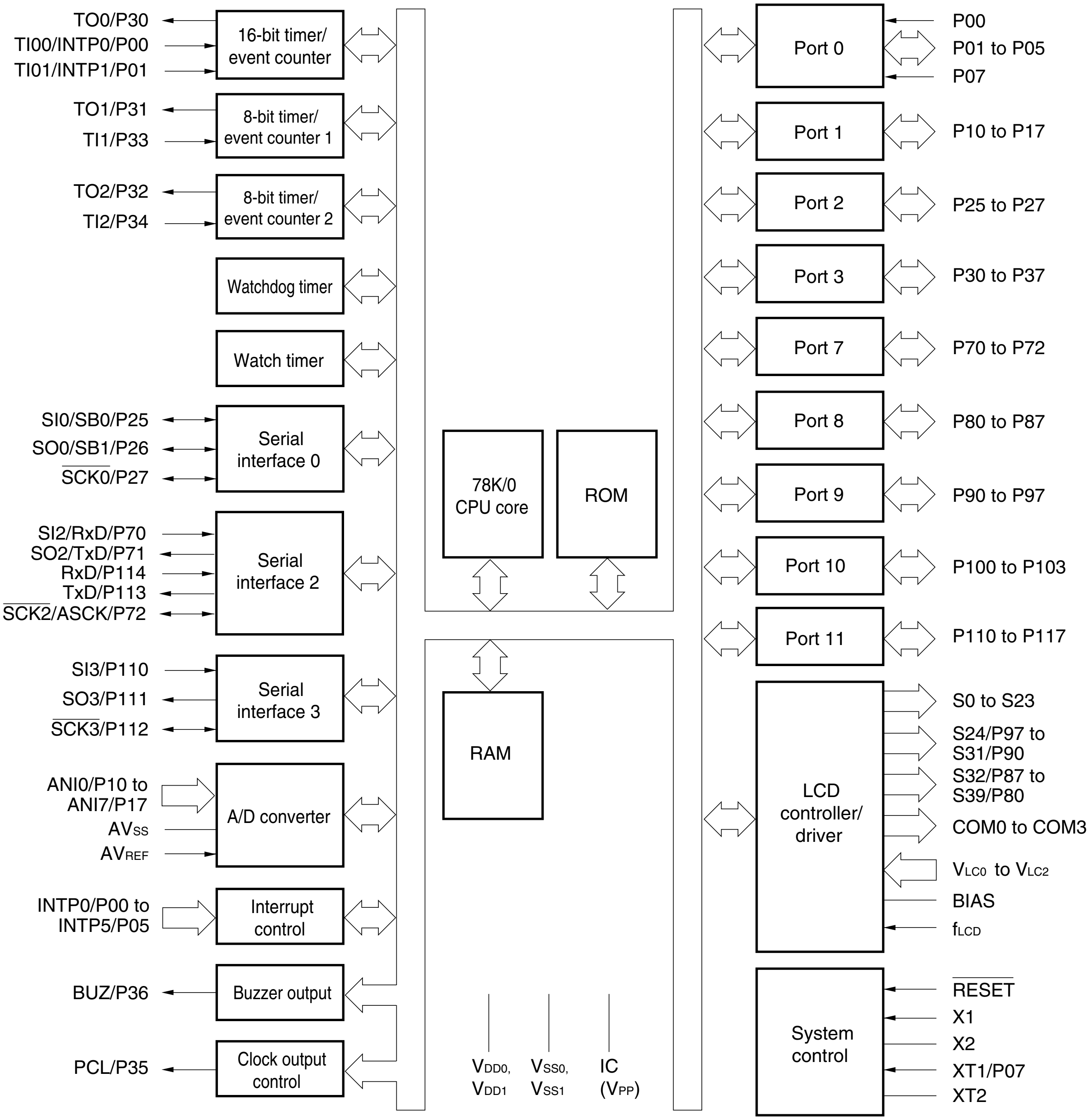
The major functional differences between the subseries are shown below.

- **Subseries without the suffix Y**

Function / Subseries Name		ROM Capacity	Timer				8-bit A/D	10-bit A/D	8-bit D/A	Serial Interface	I/O	V_{DD} MIN. Value	External Expansion
			8-bit	16-bit	Watch	WDT							
Control	μPD78075B	32 KB to 40 KB	4 ch	1 ch	1 ch	1 ch	8 ch	–	2 ch	3 ch (UART: 1 ch)	88	1.8 V	Yes
	μPD78078	48 KB to 60 KB											
	μPD78070A	–									61	2.7 V	
	μPD780058	24 KB to 60 KB	2 ch							3 ch (time-division UART: 1 ch)	68	1.8 V	
	μPD78058F	48 KB to 60 KB								3 ch (UART: 1 ch)	69	2.7 V	
	μPD78054	16 KB to 60 KB										2.0 V	
	μPD780065	40 KB to 48 KB							–	4 ch (UART: 1 ch)	60	2.7 V	
	μPD780078	48 KB to 60 KB		2 ch			–	8 ch		3 ch (UART: 2 ch)	52	1.8 V	
	μPD780034A	8 KB to 32 KB		1 ch						3 ch (UART: 1 ch)	51		
	μPD780024A						8 ch	–					
	μPD780034AS						–	4 ch			39		–
	μPD780024AS						4 ch	–					
	μPD78014H						8 ch			2 ch	53		Yes
	μPD78018F	8 KB to 60 KB											
	μPD78083	8 KB to 16 KB		–	–					1 ch (UART: 1 ch)	33		–
Inverter control	μPD780988	16 KB to 60 KB	3 ch	**Note**	–	1 ch	–	8 ch	–	3 ch (UART: 2 ch)	47	4.0 V	Yes
VFD drive	μPD780208	32 KB to 60 KB	2 ch	1 ch	1 ch	1 ch	8 ch	–	–	2 ch	74	2.7 V	–
	μPD780232	16 KB to 24 KB	3 ch	–	–		4 ch				40	4.5 V	
	μPD78044H	32 KB to 48 KB	2 ch	1 ch	1 ch		8 ch			1 ch	68	2.7 V	
	μPD78044F	16 KB to 40 KB								2 ch			
LCD drive	μPD780354	24 KB to 32 KB	4 ch	1 ch	1 ch	1 ch	–	8 ch	–	3 ch (UART: 1 ch)	66	1.8 V	–
	μPD780344						8 ch	–					
	μPD780338	48 KB to 60 KB	3 ch	2 ch			–	10 ch	1 ch	2 ch (UART: 1 ch)	54		
	μPD780328										62		
	μPD780318										70		
	μPD780308	48 KB to 60 KB	2 ch	1 ch			8 ch	–	–	3 ch (time-division UART: 1 ch)	57	2.0 V	
	μPD78064B	32 KB								2 ch (UART: 1 ch)			
	μPD78064	16 KB to 32 KB											
Bus interface supported	μPD780948	60 KB	2 ch	2 ch	1 ch	1 ch	8 ch	–	–	3 ch (UART: 1 ch)	79	4.0 V	Yes
	μPD78098B	40 KB to 60 KB		1 ch					2 ch		69	2.7 V	–
	μPD780816	32 KB to 60 KB		2 ch			12 ch		–	2 ch (UART: 1 ch)	46	4.0 V	
Meter control	μPD780958	48 KB to 60 KB	4 ch	2 ch	–	1 ch	–	–	–	2 ch (UART: 1 ch)	69	2.2 V	–
Dashboard control	μPD780852	32 KB to 40 KB	3 ch	1 ch	1 ch	1 ch	5 ch	–	–	3 ch (UART: 1 ch)	56	4.0 V	–
	μPD780828B	32 KB to 60 KB									59		

Note 16-bit timer: 2 channels
10-bit timer: 1 channel

1.7 Block Diagram



Remarks 1. The internal ROM capacity differs depending on the product.

2. Pin connection in parentheses is intended for the μPD78P0308.

1.8 Outline of Function

Part Number Item		μPD780306	μPD780308	μPD78P0308
Internal memory	ROM	Mask ROM		PROM
		48 KB	60 KB	60 KB[Note]
	High-speed RAM	1024 bytes		
	Expansion RAM	1024 bytes		
	LCD RAM	40 × 4 bits		
General-purpose register		8 bits × 8 × 4 banks		
Minimum instruction execution time	With main system clock selected	0.4 μs/0.8 μs/1.6 μs/3.2 μs/6.4 μs/12.8 μs (@ 5.0 MHz)		
	With subsystem clock selected	122 μs (@ 32.768 kHz)		
Instruction set		• 16-bit operation • Multiply/divide (8 bits × 8 bits, 16 bits ÷ 8 bits) • Bit manipulate (set, reset, test, and Boolean operation) • BCD adjust, etc.		
I/O port (including alternate-function pins for segment signal output)		• Total: 57 • CMOS input: 2 • CMOS I/O: 55		
A/D converter		8-bit resolution × 8 channels		
LCD controller/driver		• Segment signal output: Max. 40 • Common signal output: Max. 4 • Bias: 1/2, 1/3 bias switching possible		
Serial interface		• 3-wire serial I/O/SBI/2-wire serial I/O mode selection possible: 1 channel • 3-wire serial I/O mode/UART mode selection possible: 1 channel • 3-wire serial I/O mode: 1 channel		
Timer		• 16-bit timer/event counter: 1 channel • 8-bit timer/event counter: 2 channels • Watch timer: 1 channel • Watchdog timer: 1 channel		
Timer output		Three outputs (14-bit PWM output enable: 1)		
Clock output		19.5 kHz, 39.1 kHz, 78.1 kHz, 156 kHz, 313 kHz, 625 kHz, 1.25 MHz, 2.5 MHz, 5.0 MHz (@ 5.0 MHz with main system clock) 32.768 kHz (@ 32.768 kHz with subsystem clock)		
Buzzer output		1.2 kHz, 2.4 kHz, 4.9 kHz, 9.8 kHz (@ 5.0 MHz with main system clock)		

Note The capacity of the internal PROM can be changed using the internal memory size switching register (IMS).

Item	Part Number	μPD780306	μPD780308	μPD78P0308
Vectored interrupt source	Maskable	Internal: 13 External: 6		
	Non-maskable	Internal: 1		
	Software	1		
Test input		Internal: 1 External: 1		
Power supply voltage		V_{DD} = 2.0 to 5.5 V		
Operating ambient temperature		T_A = –40 to +85°C		
Package		• 100-pin plastic LQFP (Fine pitch) (14 × 14) • 100-pin plastic QFP (14 × 20)		

1.9 Mask Options

The mask ROM versions (μPD780306, 780308) provide mask options. By specifying this mask option at the time of ordering, split resistors which enable to generate LCD drive voltage suited to each bias method type can be incorporated. Using this mask option reduces the number of components to add to the device, resulting in board space saving.

The mask options provided in the μPD780308 Subseries are shown in Table 1-1.

Table 1-1. Mask Options of Mask ROM Versions

Pin Names	Mask Options
V_{LC0} to V_{LC2}	Split resistor can be incorporated.

CHAPTER 2 OUTLINE (μPD780308Y SUBSERIES)

2.1 Features

○ On-chip high-capacity ROM and RAM

Type / Part Number	Program Memory (ROM)	Data Memory		
		Internal High-Speed RAM	Internal Expansion RAM	LCD RAM
μPD780306Y	48 KB	1024 bytes	1024 bytes	40 × 4 bits
μPD780308Y	60 KB			
μPD78P0308Y	60 KB[Note]			

Note The capacity of internal PROM can be changed by means of the internal memory size switching register (IMS).

○ Minimum instruction execution time changeable from high speed (0.4 μs: @ 5.0 MHz operation with main system clock) to ultra-low speed (122 μs: @ 32.768 kHz operation with subsystem clock)

○ Instruction set suited to system control
- Bit manipulation possible in all address spaces
- Multiply and divide instructions

○ Fifty-seven I/O ports (including alternate-function pins for segment signal output)

○ LCD controller/driver

• Segment signal output:	Max. 40
• Common signal output:	Max. 4
• Bias:	1/2, 1/3 bias switching possible
• Power supply voltage:	V_{DD} = 2.0 to 5.5 V (can operate in all modes)

○ 8-bit resolution A/D converter: 8 channels

○ Serial interface: 3 channels
- 3-wire serial I/O/2-wire serial I/O/I^{2}C bus mode: 1 channel
- 3-wire serial I/O/UART mode: 1 channel
- 3-wire serial I/O mode: 1 channel

○ Timer: 5 channels
- 16-bit timer/event counter: 1 channel
- 8-bit timer/event counter: 2 channels
- Watch timer: 1 channel
- Watchdog timer: 1 channel

○ Twenty-one vectored interrupt sources

○ Two test inputs

○ Two types of on-chip clock oscillators (main system clock and subsystem clock)

○ Power supply voltage: V_{DD} = 2.0 to 5.5 V

2.2 Applications

Cellular phones, CD players, cameras, meters, audio equipment, etc.

2.3 Ordering Information

Part Number	Package	Internal ROM
μPD780306YGC-××× -8EU	100-pin plastic LQFP (Fine pitch) (14 × 14)	Mask ROM
μPD780306YGC-×××-8EU-A	100-pin plastic LQFP (Fine pitch) (14 × 14)	Mask ROM
μPD780306YGF-×××-3BA	100-pin plastic QFP (14 × 20)	Mask ROM
μPD780306YGF-×××-3BA-A	100-pin plastic QFP (14 × 20)	Mask ROM
μPD780308YGC-×××-8EU	100-pin plastic LQFP (Fine pitch) (14 × 14)	Mask ROM
μPD780308YGC-×××-8EU-A	100-pin plastic LQFP (Fine pitch) (14 × 14)	Mask ROM
μPD780308YGF-×××-3BA	100-pin plastic QFP (14 × 20)	Mask ROM
μPD780308YGF-×××-3BA-A	100-pin plastic QFP (14 × 20)	Mask ROM
μPD78P0308YGC-8EU	100-pin plastic LQFP (Fine pitch) (14 × 14)	One-time PROM
μPD78P0308YGC-8EU-A	100-pin plastic LQFP (Fine pitch) (14 × 14)	One-time PROM
μPD78P0308YGF-3BA	100-pin plastic QFP (14 × 20)	One-time PROM
μPD78P0308YGF-3BA-A	100-pin plastic QFP (14 × 20)	One-time PROM

Remarks 1. ××× indicates ROM code suffix.

2. Products with -A at the end of the part number are lead-free products.

2.4 Quality Grade

Part Number	Package	Quality Grade
μPD780306YGC-×××-8EU	100-pin plastic LQFP (Fine pitch) (14 × 14)	Standard
μPD780306YGC-×××-8EU-A	100-pin plastic LQFP (Fine pitch) (14 × 14)	Standard
μPD780306YGF-×××-3BA	100-pin plastic QFP (14 × 20)	Standard
μPD780306YGF-×××-3BA-A	100-pin plastic QFP (14 × 20)	Standard
μPD780308YGC-×××-8EU	100-pin plastic LQFP (Fine pitch) (14 × 14)	Standard
μPD780308YGC-×××-8EU-A	100-pin plastic LQFP (Fine pitch) (14 × 14)	Standard
μPD780308YGF-×××-3BA	100-pin plastic QFP (14 × 20)	Standard
μPD780308YGF-×××-3BA-A	100-pin plastic QFP (14 × 20)	Standard
μPD78P0308YGC-8EU	100-pin plastic LQFP (Fine pitch) (14 × 14)	Standard
μPD78P0308YGC-8EU-A	100-pin plastic LQFP (Fine pitch) (14 × 14)	Standard
μPD78P0308YGF-3BA	100-pin plastic QFP (14 × 20)	Standard
μPD78P0308YGF-3BA-A	100-pin plastic QFP (14 × 20)	Standard

Remarks 1. ××× indicates ROM code suffix.

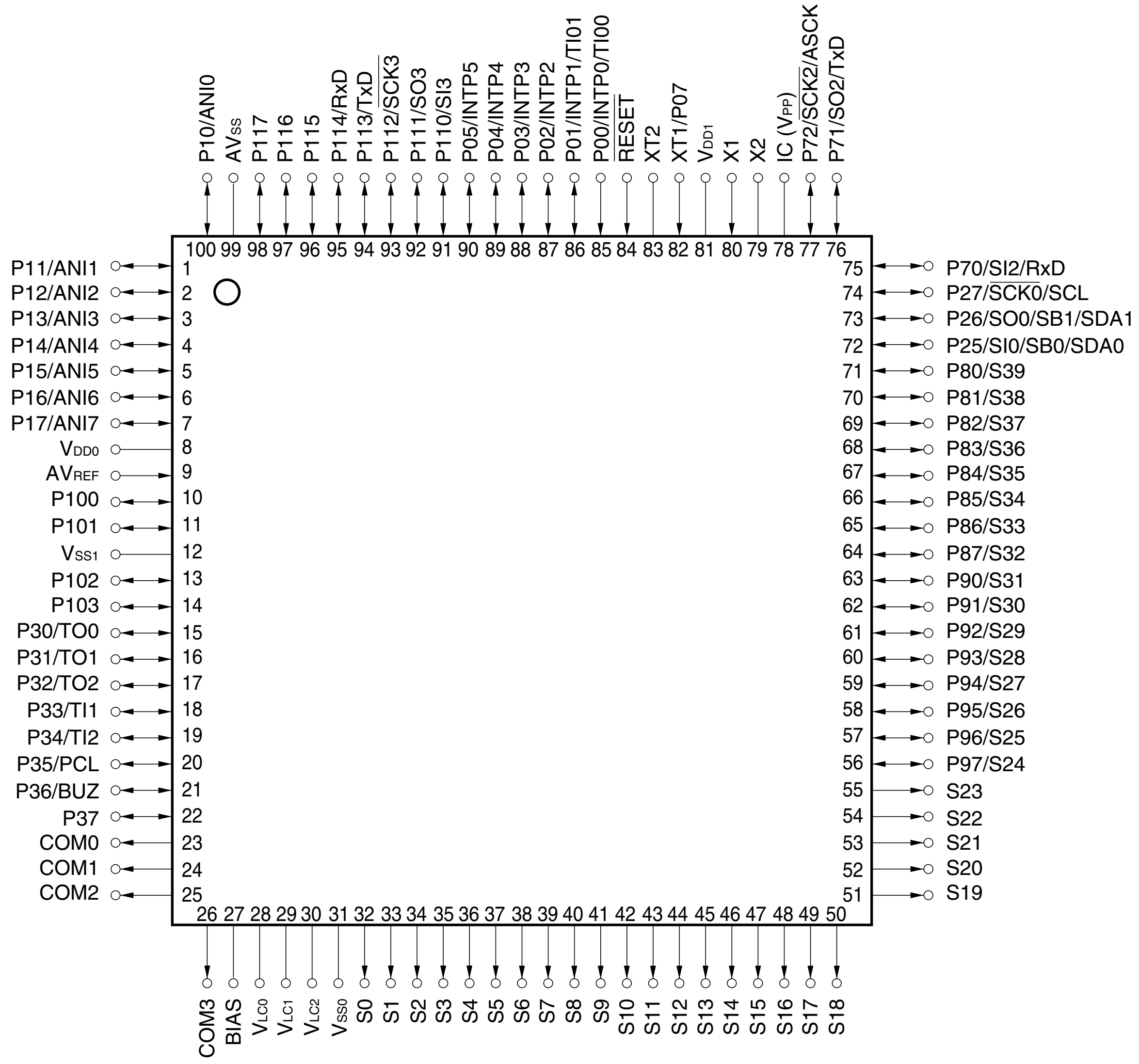
2. Products with -A at the end of the part number are lead-free products.

Please refer to "Quality Grades on NEC Semiconductor Devices" (Document No. C11531E) published by NEC Electronics Corporation to know the specification of the quality grade on the device and its recommended applications.

2.5 Pin Configuration (Top View)

(1) Normal operating mode

100-pin plastic LQFP (Fine pitch) (14 × 14)



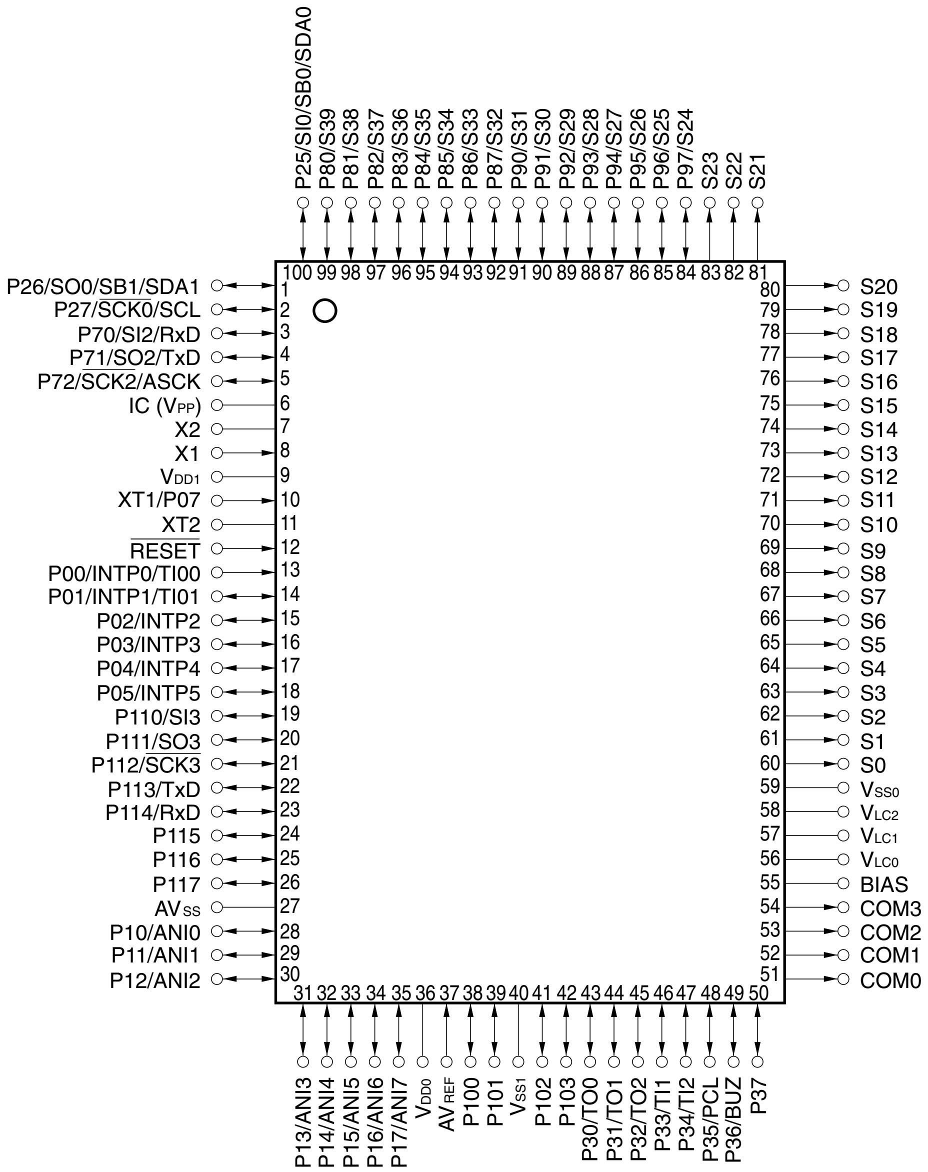
Cautions 1. Connect the IC (Internally Connected) pin directly to V_{SS0} or V_{SS1}.

2. Connect the AV_{SS} pin to V_{SS0}.

Remarks 1. Pin connection in parentheses is intended for the μPD78P0308Y.

2. When using the μPD780308Y Subseries in an application field where the noise generated from the microcontroller must be reduced, it is recommended to take noise reduction measures by supplying separate power to V_{DD0} and V_{DD1}, and connecting V_{SS0} and V_{SS1} to separate ground lines.

100-pin plastic QFP (14 × 20)



Cautions 1. Connect the IC (Internally Connected) pin directly to V_{SS0} or V_{SS1}.

2. Connect the AV_{SS} pin to V_{SS0}.

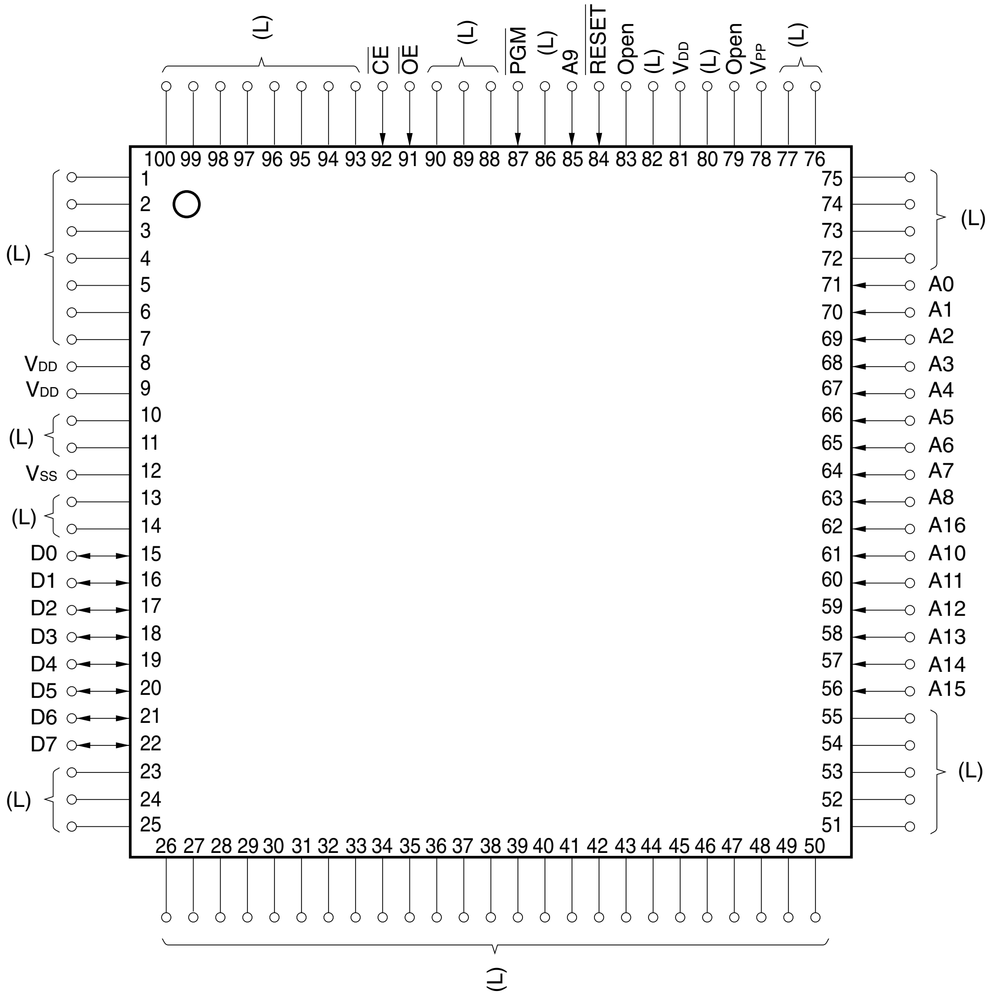
Remarks 1. Pin connection in parentheses is intended for the μPD78P0308Y.

2. When using the μPD780308Y Subseries in an application field where the noise generated from the microcontroller must be reduced, it is recommended to take noise reduction measures by supplying separate power to V_{DD0} and V_{DD1}, and connecting V_{SS0} and V_{SS1} to separate ground lines.

ANI0 to ANI7:	Analog input	$\overline{\text{RESET}}$:	Reset
ASCK:	Asynchronous serial clock	RxD:	Receive data
AV_{REF}:	Analog reference voltage	S0 to S39:	Segment output
AV_{SS}:	Analog ground	SB0, SB1:	Serial bus
BIAS:	LCD power supply bias control	$\overline{\text{SCK0}}$, $\overline{\text{SCK2}}$, $\overline{\text{SCK3}}$:	Serial clock
BUZ:	Buzzer clock	SCL:	Serial clock
COM0 to COM3:	Common output	SDA0, SDA1:	Serial data
INTP0 to INTP5:	Interrupt from peripherals	SI0, SI2, SI3:	Serial input
IC:	Internally connected	SO0, SO2, SO3:	Serial output
P00 to P05, P07:	Port 0	TI00, TI01:	Timer input
P10 to P17:	Port 1	TI1, TI2:	Timer input
P25 to P27:	Port 2	TO0 to TO2:	Timer output
P30 to P37:	Port 3	TxD:	Transmit data
P70 to P72:	Port 7	V_{DD0}, V_{DD1}:	Power supply
P80 to P87:	Port 8	V_{LC0} to V_{LC2}:	LCD power supply
P90 to P97:	Port 9	V_{PP}:	Programming power supply
P100 to P103:	Port 10	V_{SS0}, V_{SS1}:	Ground
P110 to P117:	Port 11	X1, X2:	Crystal (main system clock)
PCL:	Programmable clock	XT1, XT2:	Crystal (subsystem clock)

(2) PROM programming mode

100-pin plastic LQFP (Fine pitch) (14 × 14)



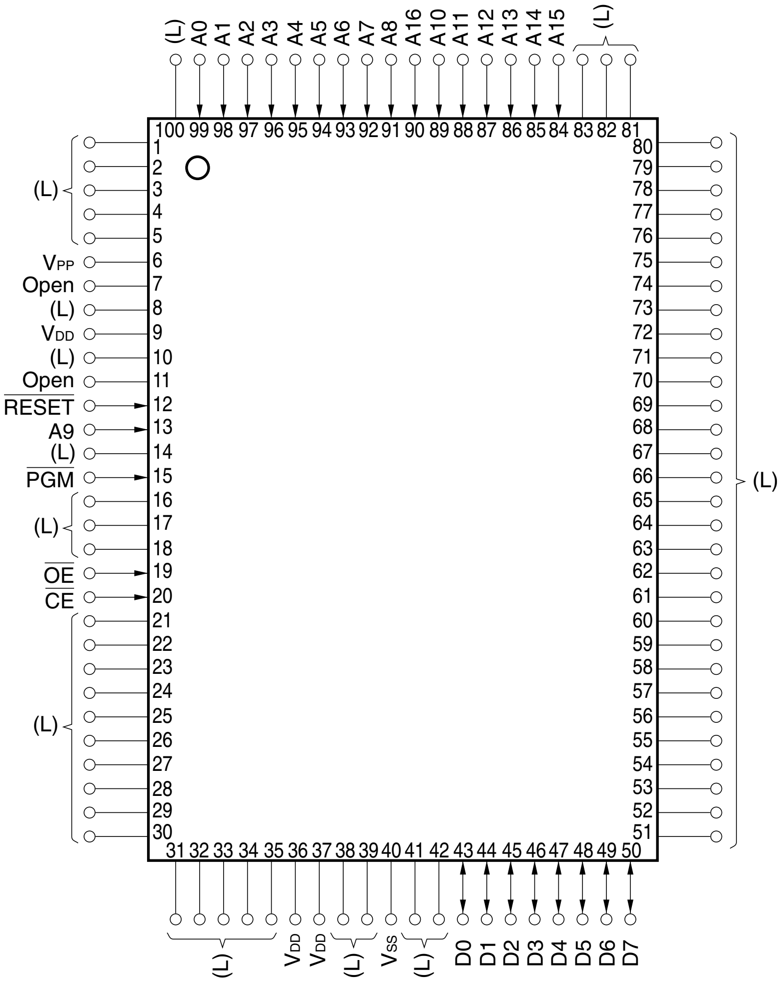
Cautions 1. (L): Independently connect to V_{SS} via a pull-down resistor.

2. V_{SS}: Connect to the ground.

3. $\overline{\text{RESET}}$: Set to the low level.

4. Open: Do not connect anything.

100-pin plastic QFP (14 × 20)



Cautions 1. (L): Independently connect to V_{SS} via a pull-down resistor.

2. V_{SS}: Connect to the ground.

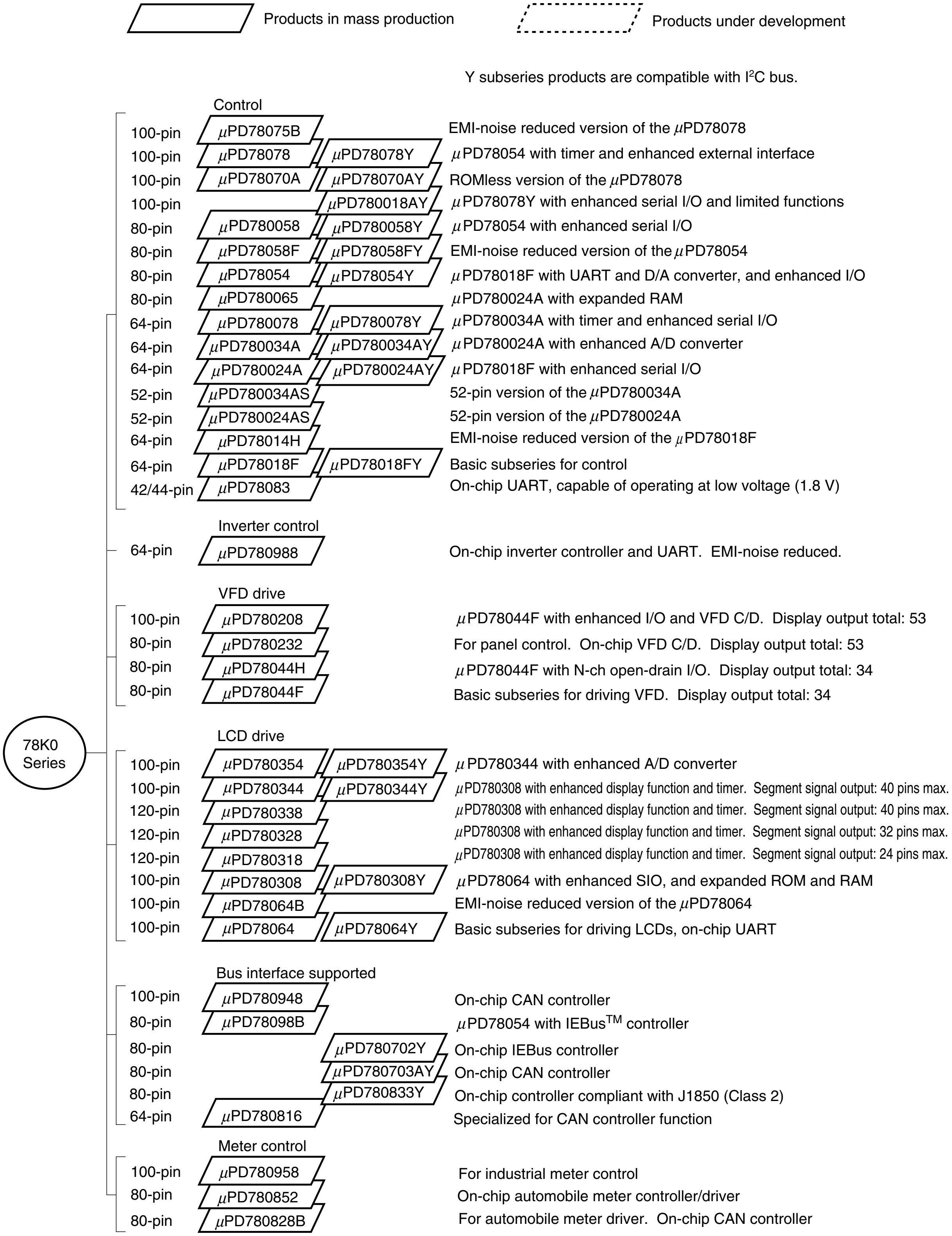
3. $\overline{\text{RESET}}$: Set to the low level.

4. Open: Do not connect anything.

A0 to A16:	Address bus	$\overline{\text{RESET}}$:	Reset
$\overline{\text{CE}}$:	Chip enable	V_{DD}:	Power supply
D0 to D7:	Data bus	V_{PP}:	Programming power supply
$\overline{\text{OE}}$:	Output enable	V_{SS}:	Ground
$\overline{\text{PGM}}$:	Program		

<R> 2.6 78K0 Series Lineup

The products in the 78K0 Series are listed below. The names enclosed in boxes are subseries name.



Products in mass production

Products under development

Y subseries products are compatible with I^{2}C bus.

78K0 Series

Control			
100-pin	μPD78075B		EMI-noise reduced version of the μPD78078
100-pin	μPD78078	μPD78078Y	μPD78054 with timer and enhanced external interface
100-pin	μPD78070A	μPD78070AY	ROMless version of the μPD78078
100-pin		μPD780018AY	μPD78078Y with enhanced serial I/O and limited functions
80-pin	μPD780058	μPD780058Y	μPD78054 with enhanced serial I/O
80-pin	μPD78058F	μPD78058FY	EMI-noise reduced version of the μPD78054
80-pin	μPD78054	μPD78054Y	μPD78018F with UART and D/A converter, and enhanced I/O
80-pin	μPD780065		μPD780024A with expanded RAM
64-pin	μPD780078	μPD780078Y	μPD780034A with timer and enhanced serial I/O
64-pin	μPD780034A	μPD780034AY	μPD780024A with enhanced A/D converter
64-pin	μPD780024A	μPD780024AY	μPD78018F with enhanced serial I/O
52-pin	μPD780034AS		52-pin version of the μPD780034A
52-pin	μPD780024AS		52-pin version of the μPD780024A
64-pin	μPD78014H		EMI-noise reduced version of the μPD78018F
64-pin	μPD78018F	μPD78018FY	Basic subseries for control
42/44-pin	μPD78083		On-chip UART, capable of operating at low voltage (1.8 V)
Inverter control			
64-pin	μPD780988		On-chip inverter controller and UART. EMI-noise reduced.
VFD drive			
100-pin	μPD780208		μPD78044F with enhanced I/O and VFD C/D. Display output total: 53
80-pin	μPD780232		For panel control. On-chip VFD C/D. Display output total: 53
80-pin	μPD78044H		μPD78044F with N-ch open-drain I/O. Display output total: 34
80-pin	μPD78044F		Basic subseries for driving VFD. Display output total: 34
LCD drive			
100-pin	μPD780354	μPD780354Y	μPD780344 with enhanced A/D converter
100-pin	μPD780344	μPD780344Y	μPD780308 with enhanced display function and timer. Segment signal output: 40 pins max.
120-pin	μPD780338		μPD780308 with enhanced display function and timer. Segment signal output: 40 pins max.
120-pin	μPD780328		μPD780308 with enhanced display function and timer. Segment signal output: 32 pins max.
120-pin	μPD780318		μPD780308 with enhanced display function and timer. Segment signal output: 24 pins max.
100-pin	μPD780308	μPD780308Y	μPD78064 with enhanced SIO, and expanded ROM and RAM
100-pin	μPD78064B		EMI-noise reduced version of the μPD78064
100-pin	μPD78064	μPD78064Y	Basic subseries for driving LCDs, on-chip UART
Bus interface supported			
100-pin	μPD780948		On-chip CAN controller
80-pin	μPD78098B		μPD78054 with IEBus™ controller
80-pin		μPD780702Y	On-chip IEBus controller
80-pin		μPD780703AY	On-chip CAN controller
80-pin		μPD780833Y	On-chip controller compliant with J1850 (Class 2)
64-pin	μPD780816		Specialized for CAN controller function
Meter control			
100-pin	μPD780958		For industrial meter control
80-pin	μPD780852		On-chip automobile meter controller/driver
80-pin	μPD780828B		For automobile meter driver. On-chip CAN controller

Remark VFD (Vacuum Fluorescent Display) is referred to as FIP™ (Fluorescent Indicator Panel) in some documents, but the functions of the two are the same.

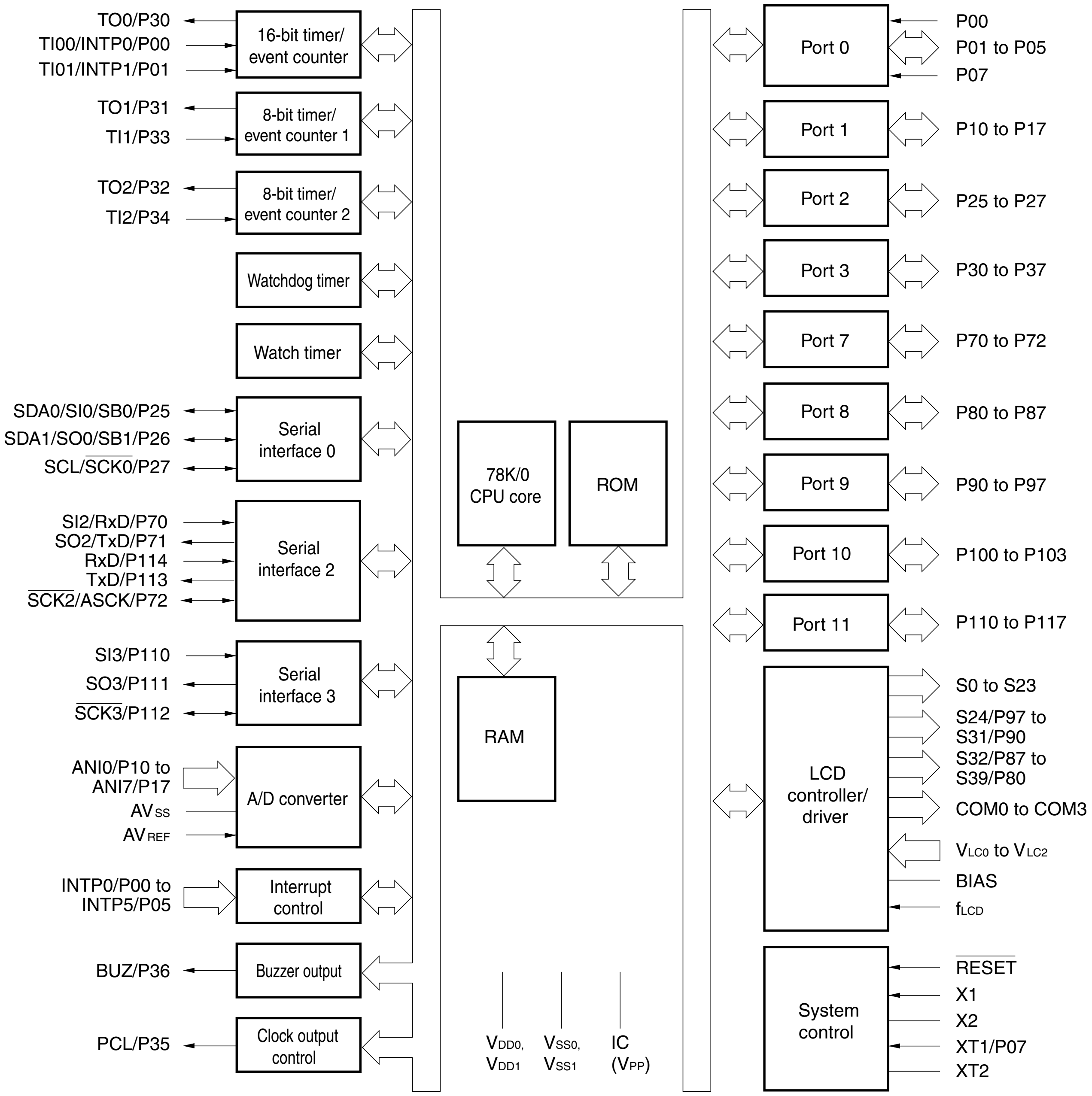
The major functional differences between the subseries are shown below.

- **Subseries with the suffix Y**

Function Subseries Name		ROM Capacity	Timer				8-bit A/D	10-bit A/D	8-bit D/A	Serial Interface	I/O	V_{DD}	External Expansion
			8-bit	16-bit	Watch	WDT						MIN. Value	
Control	µPD78078Y	48 KB to 60 KB	4 ch	1 ch	1 ch	1 ch	8 ch	–	2 ch	3 ch (UART: 1 ch, I^2C: 1 ch)	88	1.8 V	Yes
	µPD78070AY	–									61	2.7 V	
	µPD780018AY	48 KB to 60 KB							–	3 ch (I^2C: 1 ch)	88		
	µPD780058Y	24 KB to 60 KB	2 ch						2 ch	3 ch (time-division UART: 1 ch, I^2C: 1 ch)	68	1.8 V	
	µPD78058FY	48 KB to 60 KB								3 ch (UART: 1 ch, I^2C: 1 ch)	69	2.7 V	
	µPD78054Y	16 KB to 60 KB										2.0 V	
	µPD780078Y	48 KB to 60 KB		2 ch			–	8 ch	–	4 ch (UART: 2 ch, I^2C: 1 ch)	52	1.8 V	
	µPD780034AY	8 KB to 32 KB		1 ch						3 ch (UART: 1 ch, I^2C: 1 ch)	51		
	µPD780024AY						8 ch	–					
	µPD78018FY	8 KB to 60 KB								2 ch (I^2C: 1 ch)	53		
LCD drive	µPD780354Y	24 KB to 32 KB	4 ch	1 ch	1 ch	1 ch	–	8 ch	–	4 ch (UART: 1 ch, I^2C: 1 ch)	66	1.8 V	–
	µPD780344Y						8 ch	–					
	µPD780308Y	48 KB to 60 KB	2 ch							3 ch (time-division UART: 1 ch, I^2C: 1 ch)	57	2.0 V	
	µPD78064Y	16 KB to 32 KB								2 ch (UART: 1 ch, I^2C: 1 ch)			
Bus interface supported	µPD780702Y	60 KB	3 ch	2 ch	1 ch	1 ch	16 ch	–	–	4 ch (UART: 1 ch, I^2C: 1 ch)	67	3.5 V	–
	µPD780703AY	59.5 KB											
	µPD780833Y	60 KB									65	4.5 V	

Remark The functions of the subseries without the suffix Y and the subseries with the suffix Y are the same, except for the serial interface (if a subseries without the suffix Y is available).

2.7 Block Diagram



Remarks 1. The internal ROM capacity differs depending on the product.

2. Pin connection in parentheses is intended for the μPD78P0308Y.

2.8 Outline of Function

Part Number Item		μPD780306Y	μPD780308Y	μPD78P0308Y
Internal memory	ROM	Mask ROM		PROM
		48 KB	60 KB	60 KB[Note]
	High-speed RAM	1024 bytes		
	Expansion RAM	1024 bytes		
	LCD RAM	40 × 4 bits		
General-purpose register		8 bits × 8 × 4 banks		
Minimum instruction execution time	With main system clock selected	0.4 μs/0.8 μs/1.6 μs/3.2 μs/6.4 μs/12.8 μs (@ 5.0 MHz)		
	With subsystem clock selected	122 μs (@ 32.768 kHz)		
Instruction set		• 16-bit operation • Multiply/divide (8 bits × 8 bits, 16 bits ÷ 8 bits) • Bit manipulate (set, reset, test, and Boolean operation) • BCD adjust, etc.		
I/O port (including alternate-function pins for segment signal output)		• Total: 57 • CMOS input: 2 • CMOS I/O: 55		
A/D converter		8-bit resolution × 8 channels		
LCD controller/driver		• Segment signal output: Max. 40 • Common signal output: Max. 4 • Bias: 1/2, 1/3 bias switching possible		
Serial interface		• 3-wire serial I/O/2-wire serial I/O/I^{2}C bus mode selection possible: 1 channel • 3-wire serial I/O mode/UART mode selection possible: 1 channel • 3-wire serial I/O mode: 1 channel		
Timer		• 16-bit timer/event counter: 1 channel • 8-bit timer/event counter: 2 channels • Watch timer: 1 channel • Watchdog timer: 1 channel		
Timer output		Three outputs (14-bit PWM output enable: 1)		
Clock output		19.5 kHz, 39.1 kHz, 78.1 kHz, 156 kHz, 313 kHz, 625 kHz, 1.25 MHz, 2.5 MHz, 5.0 MHz (@ 5.0 MHz with main system clock) 32.768 kHz (@ 32.768 kHz with subsystem clock)		
Buzzer output		1.2 kHz, 2.4 kHz, 4.9 kHz, 9.8 kHz (@ 5.0 MHz with main system clock)		

Note The capacity of the internal PROM can be changed using the internal memory size switching register (IMS).

Part Number Item		µPD780306Y	µPD780308Y	µPD78P0308Y
Vectored interrupt source	Maskable	Internal: 13 External: 6		
	Non-maskable	Internal: 1		
	Software	1		
Test input		Internal: 1 External: 1		
Power supply voltage		V_{DD} = 2.0 to 5.5 V		
Operating ambient temperature		T_A = –40 to +85°C		
Package		• 100-pin plastic LQFP (14 × 14) • 100-pin plastic QFP (14 × 20)		

2.9 Mask Options

The mask ROM versions (µPD780306Y, 780308Y) provide mask options. By specifying this mask option at the time of ordering, split resistors which enable to generate LCD drive voltage suited to each bias method type can be incorporated. Using this mask option reduces the number of components to add to the device, resulting in board space saving.

The mask options provided in the µPD780308Y Subseries are shown in Table 2-1.

Table 2-1. Mask Options of Mask ROM Versions

Pin Names	Mask Options
V_{LC0} to V_{LC2}	Split resistor can be incorporated.

3.1 Pin Function List

3.1.1 Normal operating mode pins

(1) Port pins (1/2)

Pin Name	I/O	Function		After Reset	Alternate Function
P00	Input	Port 0. 7-bit I/O port.	Input only	Input	INTP0/TI00
P01	I/O		Input/output mode can be specified in 1-bit units. If used as an input port, an internal pull-up resistor can be used by software.	Input	INTP1/TI01
P02					INTP2
P03					INTP3
P04					INTP4
P05					INTP5
P07[Note 1]	Input		Input only	Input	XT1
P10 to P17	I/O	Port 1. 8-bit I/O port. Input/output mode can be specified in 1-bit units. If used as an input port, an internal pull-up resistor can be used by software[Note 2].		Input	ANI0 to ANI7
P25	I/O	Port 2. 3-bit I/O port. Input/output mode can be specified in 1-bit units. If used as an input port, an internal pull-up resistor can be used by software.		Input	SI0/SB0
P26					SO0/SB1
P27					$\overline{\text{SCK0}}$
P30	I/O	Port 3. 8-bit I/O port. Input/output mode can be specified in 1-bit units. If used as an input port, an internal pull-up resistor can be used by software.		Input	TO0
P31					TO1
P32					TO2
P33					TI1
P34					TI2
P35					PCL
P36					BUZ
P37					—

Notes **1.** When the P07/XT1 pin is used as an input port, set bit 6 (FRC) of the processor clock control register (PCC) to 1 (do not use the internal feedback resistor to the subsystem clock oscillator).

2. When pins P10/ANI0 to P17/ANI7 are used as an analog input of the A/D converter, the internal pull-up resistor is automatically disabled.

(1) Port pins (2/2)

Pin Name	I/O	Function	After Reset	Alternate Function
P70	I/O	Port 7. 3-bit I/O port. Input/output mode can be specified in 1-bit units. If used as an input port, an internal pull-up resistor can be used by software.	Input	SI2/RxD
P71				SO2/TxD
P72				$\overline{\text{SCK2}}$/ASCK
P80 to P87	I/O	Port 8. 8-bit I/O port. Input/output mode can be specified in 1-bit units. If used as an input port, an internal pull-up resistor can be used by software. I/O port/segment signal output can be specified in 2-bit units by LCD display control register (LCDC).	Input	S39 to S32
P90 to P97	I/O	Port 9. 8-bit I/O port. Input/output mode can be specified in 1-bit units. If used as an input port, an internal pull-up resistor can be used by software. I/O port/segment signal output can be specified in 2-bit units by LCD display control register (LCDC).	Input	S31 to S24
P100 to P103	I/O	Port 10. 4-bit I/O port. Input/output mode can be specified in 1-bit units. If used as an input port, an internal pull-up resistor can be used by software. LED can be driven directly.	Input	—
P110	I/O	Port 11. 8-bit I/O port. Input/output mode can be specified in 1-bit units. If used as an input port, an internal pull-up resistor can be used by software. Falling edge can be detected.	Input	SI3
P111				SO3
P112				$\overline{\text{SCK3}}$
P113				TxD
P114				RxD
P115 to P117				—

(2) Non-port pins (1/2)

Pin Name	I/O	Function	After Reset	Alternate Function
INTP0	Input	External interrupt request inputs with specifiable valid edges (rising edge, falling edge, both rising and falling edges).	Input	P00/TI00
INTP1				P01/TI01
INTP2				P02
INTP3				P03
INTP4				P04
INTP5				P05
SI0	Input	Serial interface serial data input.	Input	P25/SB0
SI2				P70/RxD
SI3				P110
SO0	Output	Serial interface serial data output.	Input	P26/SB1
SO2				P71/TxD
SO3				P111
SB0	I/O	Serial interface serial data input/output.	Input	P25/SI0
SB1				P26/SO0
$\overline{\text{SCK0}}$	I/O	Serial interface serial clock input/output.	Input	P27
$\overline{\text{SCK2}}$				P72/ASCK
$\overline{\text{SCK3}}$				P112
RxD	Input	Asynchronous serial interface serial data input.	Input	P70/SI2, P114
TxD	Output	Asynchronous serial interface serial data output.	Input	P71/SO2, P113
ASCK	Input	Asynchronous serial interface serial clock input.	Input	P72/$\overline{\text{SCK2}}$
TI00	Input	External count clock input to 16-bit timer (TM0).	Input	P00/INTP0
TI01		Capture trigger signal input to capture register (CR00).		P01/INTP1
TI1		External count clock input to 8-bit timer (TM1).		P33
TI2		External count clock input to 8-bit timer (TM2).		P34
TO0	Output	16-bit timer (TM0) output (also used for 14-bit PWM output).	Input	P30
TO1		8-bit timer (TM1) output.		P31
TO2		8-bit timer (TM2) output.		P32
PCL	Output	Clock output (for main system clock and subsystem clock trimming).	Input	P35
BUZ	Output	Buzzer output.	Input	P36
S0 to S23	Output	Segment signal output of LCD controller/driver.	Output	—
S24 to S31			Input	P97 to P90
S32 to S39				P87 to P80
COM0 to COM3	Output	Common signal output of LCD controller/driver	Output	—
V_{LC0} to V_{LC2}	—	LCD drive voltage (mask ROM versions can incorporate split resistor (mask option)).	—	—
BIAS	—	Power supply for LCD drive.	—	—
ANI0 to ANI7	Input	A/D converter analog input.	Input	P10 to P17
AV_{REF}	Input	A/D converter reference voltage input (also used for analog power).	—	—

(2) Non-port pins (2/2)

Pin Name	I/O	Function	After Reset	Alternate Function
AV_{SS}	—	A/D converter ground potential. Same potential as V_{SS0}.	—	—
$\overline{RESET}$	Input	System reset input.	—	—
X1	Input	Crystal connection for main system clock oscillation.	—	—
X2	—		—	—
XT1	Input	Crystal connection for subsystem clock oscillation.	Input	P07
XT2	—		—	—
V_{DD0}	—	Positive power supply to port.	—	—
V_{SS0}	—	Ground potential of port.	—	—
V_{DD1}	—	Positive power supply (except ports, analogs).	—	—
V_{SS1}	—	Ground potential (except ports, analogs).	—	—
V_{PP}	—	High-voltage application for program write/verify. Connect directly to V_{SS0} or V_{SS1} in normal operating mode.	—	—
IC	—	Internal connection. Connect directly to V_{SS0} or V_{SS1}.	—	—

3.1.2 PROM programming mode pins (μPD78P0308 only)

Pin Name	I/O	Function
$\overline{RESET}$	Input	PROM programming mode setting. When +5 V or +12.5 V is applied to the V_{PP} pin or a low level voltage is applied to the $\overline{RESET}$ pin, the PROM programming mode is set.
V_{PP}	Input	High-voltage application for PROM programming mode setting and program write/verify.
A0 to A16	Input	Address bus.
D0 to D7	I/O	Data bus.
$\overline{CE}$	Input	PROM enable input/program pulse input.
$\overline{OE}$	Input	Read strobe input to PROM.
$\overline{PGM}$	Input	Program/program inhibit input in PROM programming mode.
V_{DD}	—	Positive power supply.
V_{SS}	—	Ground potential.

3.2 Description of Pin Functions

3.2.1 P00 to P05, P07 (Port 0)

These are 7-bit I/O ports. Besides serving as I/O ports, they function as an external interrupt request input, an external count clock input to the timer, a capture trigger signal input, and crystal connection for subsystem clock oscillation.

The following operating modes can be specified in 1-bit units.

(1) Port mode

P00 and P07 function as input-only ports and P01 to P05 function as I/O ports.

P01 to P05 can be specified as input or output ports in 1-bit units with port mode register 0 (PM0). When they are used as input ports, internal pull-up resistors can be used by defining pull-up resistor option register L (PUOL).

(2) Control mode

These ports function as an external interrupt request input, an external count clock input to the timer, and crystal connection for subsystem clock oscillation.

(a) INTP0 to INTP5

INTP0 to INTP5 are external interrupt request input pins which can specify valid edges (rising edge, falling edge, and both rising and falling edges). INTP0 or INTP1 becomes a 16-bit timer/event counter capture trigger signal input pin with a valid edge input.

(b) TI00

Pin for external count clock input to 16-bit timer/event counter.

(c) TI01

Pin for capture trigger signal input to capture register (CR00) of 16-bit timer/event counter.

(d) XT1

Crystal connect pin for subsystem clock oscillation.

3.2.2 P10 to P17 (Port 1)

These are 8-bit I/O ports. Besides serving as I/O ports, they function as an A/D converter analog input.

The following operating modes can be specified in 1-bit units.

(1) Port mode

These ports function as 8-bit I/O ports. They can be specified as input or output ports in 1-bit units with port mode register 1 (PM1). When they are used as input ports, internal pull-up resistors can be used by defining pull-up resistor option register L (PUOL).

(2) Control mode

These ports function as A/D converter analog input pins (ANI0 to ANI7). The pull-up resistor is automatically disabled when the pins are specified for analog input.

3.2.3 P25 to P27 (Port 2)

These are 3-bit I/O ports. Besides serving as I/O ports, they function as data I/O and clock I/O of the serial interface.

The following operating modes can be specified in 1-bit units.

(1) Port mode

These ports function as 3-bit I/O ports. They can be specified as input or output ports in 1-bit units with port mode register 2 (PM2). When they are used as input ports, internal pull-up resistors can be used by defining pull-up resistor option register L (PUOL).

(2) Control mode

These ports function as serial interface data I/O and clock I/O.

(a) SI0, SO0

Serial interface serial data I/O pins.

(b) $\overline{\text{SCK0}}$

Serial interface serial clock I/O pins.

(c) SB0 and SB1

NEC Electronics standard serial bus interface I/O pins.

Caution When this port is used as a serial interface, the I/O and output latches must be set according to the function used. For the setting, refer to Figure 15-4 Serial Operating Mode Register 0 Format.

3.2.4 P30 to P37 (Port 3)

These are 8-bit I/O ports. Besides serving as I/O ports, they function as timer I/O, clock output, and buzzer output. The following operating modes can be specified in 1-bit units.

(1) Port mode

These ports function as 8-bit I/O ports. They can be specified as input or output ports in 1-bit units with port mode register 3 (PM3). When they are used as input ports, internal pull-up resistors can be used by defining pull-up resistor option register L (PUOL).

(2) Control mode

These ports function as timer I/O, clock output, and buzzer output.

(a) TI1 and TI2

Pins for external count clock input to the 8-bit timer/event counter.

(b) TO0 to TO2

Timer output pins.

(c) PCL

Clock output pin.

(d) BUZ

Buzzer output pin.

3.2.5 P70 to P72 (Port 7)

These are 3-bit I/O ports. Besides serving as I/O ports, they function as serial interface data I/O and clock I/O. The following operating modes can be specified in 1-bit units.

(1) Port mode

These ports function as 3-bit I/O ports. They can be specified as input or output ports in 1-bit units with port mode register 7 (PM7). When they are used as input ports, internal pull-up resistors can be used by defining pull-up resistor option register L (PUOL).

(2) Control mode

These ports function as serial interface data I/O and clock I/O.

(a) SI2, SO2

Serial interface serial data I/O pins.

(b) $\overline{\text{SCK2}}$

Serial interface serial clock I/O pin.

(c) RxD, TxD

Asynchronous serial interface serial data I/O pins.

(d) ASCK

Asynchronous serial interface serial clock input pin.

Caution When this port is used as a serial interface, the I/O and output latches must be set according to the function used. For the setting, refer to Table 17-2 Serial Interface Channel 2 Operating Mode Settings.

3.2.6 P80 to P87 (Port 8)

These are 8-bit I/O ports. Besides serving as I/O ports, they function as segment signal output of LCD controller/ driver.

The following operating modes can be specified in 1-bit units.

(1) Port mode

These ports function as 8-bit I/O ports. They can be specified as input or output ports in 1-bit units with port mode register 8 (PM8). When they are used as input ports, internal pull-up resistors can be used by defining pull-up resistor option register H (PUOH).

(2) Control mode

These ports function as segment signal output pins (S32 to S39) of LCD controller/driver.

3.2.7 P90 to P97 (Port 9)

These are 8-bit I/O ports. Besides serving as I/O ports, they function as segment signal output of LCD controller/ driver.

The following operating modes can be specified in 1-bit units.

(1) Port mode

These ports function as 8-bit I/O ports. They can be specified as input or output ports in 1-bit units with port mode register 9 (PM9). When they are used as input ports, internal pull-up resistors can be used by defining pull-up resistor option register H (PUOH).

(2) Control mode

These ports function as segment signal output pins (S24 to S31) of LCD controller/driver.

3.2.8 P100 to P103 (Port 10)

These are 4-bit I/O ports. They can be specified as input or output ports in 1-bit units with port mode register 10 (PM10). When they are used as input ports, internal pull-up resistors can be used by defining pull-up resistor option register H (PUOH).

LED can be driven directly.

3.2.9 P110 to P117 (Port 11)

These are 8-bit I/O ports. Besides serving as I/O ports, they function as serial interface data I/O and clock I/O. The following operating modes can be specified in 1-bit units.

(1) Port mode

These ports function as 8-bit I/O ports. They can be specified as input or output ports in 1-bit units with port mode register 11 (PM11). When they are used as input ports, internal pull-up resistors can be used by defining pull-up resistor option register H (PUOH).

When the falling edge is detected on a specified bit of this port, test input flag (KRIF) can be set to 1.

(2) Control mode

These ports function as serial interface data I/O and clock I/O.

(a) SI3, SO3

Serial interface serial data I/O pins.

(b) $\overline{\text{SCK3}}$

Serial interface serial clock I/O pin.

(c) RxD, TxD

Asynchronous serial interface serial data I/O pins.

Caution When this port is used as a serial interface, the I/O and output latches must be set according to the function used. For the setting, refer to Table 17-2 Serial Interface Channel 2 Operating Mode Settings, and Figure 18-3 Serial Operating Mode Register 3 Format.

3.2.10 COM0 to COM3

These are LCD controller/driver common signal output pins. They output common signals under either of the following conditions:

– when the static mode is selected (COM0 to COM3 outputs)

– when 2-time-division (COM0, COM1 outputs) or 3-time-division (COM0 to COM2 outputs) operation is performed in 1/2 bias mode

– when 3-time-division (COM0 to COM2 outputs) or 4-time-division (COM0 to COM3 outputs) operation is performed in 1/3 bias mode

3.2.11 V_{LC0} to V_{LC2}

These are LCD-driving voltage pins. The mask ROM versions can have split resistors by mask option so that LCD driving voltage can be supplied inside the V_{LC0} to V_{LC2} pins according to the required bias without connecting external split resistors.

3.2.12 BIAS

This is a LCD driving power supply pin. This pin should be connected to the V_{LC0} pin to realize user-desired LCD drive voltages to change resistance division ratios, or should be connected to external resistors together with the V_{LC0} to V_{LC2} pins and V_{SS1} pin to fine-adjust the LCD-driving power voltage.

3.2.13 AVREF

This pin inputs the reference voltage for the on-chip A/D converter. This pin also functions to supply power to the internal analog circuit. Supply power to this pin when using the A/D converter.

When not using the A/D converter, connect this pin to the V_{SS0} line.

3.2.14 AVSS

This is a ground potential pin of A/D converter. Always use the same voltage as that of the V_{SS0} pin even when A/D converter is not used.

3.2.15 $\overline{\text{RESET}}$

This is a low-level active system reset input pin.

3.2.16 X1 and X2

Crystal resonator connection pins for main system clock oscillation.

For external clock supply, input it to X1 and its inverted signal to X2.

3.2.17 XT1 and XT2

Crystal resonator connection pins for subsystem clock oscillation.

For external clock supply, input it to XT1 and its inverted signal to XT2.

3.2.18 V_{DD0}, V_{DD1}

V_{DD0} supplies positive power to the ports.

V_{DD1} supplies positive power to the circuits other than those of the ports.

3.2.19 V_{SS0}, V_{SS1}

V_{SS0} is the ground pin of the ports.

V_{SS1} is the ground pin of the circuits other than those of the ports.

3.2.20 V_{PP} (μPD78P0308 only)

High-voltage apply pin for PROM programming mode setting and program write/verify.

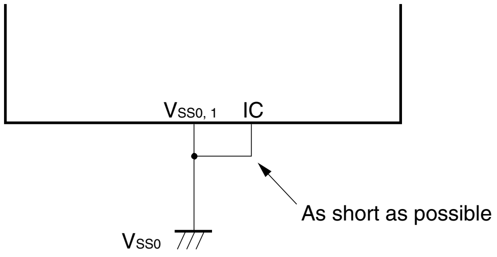
Connect directly to V_{SS0} or V_{SS1} in normal operating mode.

3.2.21 IC (Mask ROM version only)

The IC (Internally Connected) pin is provided to set the test mode to check the μPD780308 Subseries at delivery. Connect it directly to the V_{SS0} or V_{SS1} with the shortest possible wire in the normal operating mode.

When a voltage difference is produced between the IC pin and V_{SS0} or V_{SS1} pin because the wiring between those two pins is too long or an external noise is input to the IC pin, the user's program may not run normally.

○ Connect IC pins to V_{SS0} or V_{SS1} pins directly.



3.3 Pin I/O Circuits and Recommended Connection of Unused Pins

Table 3-1 shows the I/O circuit types of pins and the recommended connections of unused pins.
Refer to Figure 3-1 for the configuration of the I/O circuit of each type.

Table 3-1. Pin I/O Circuit Types (1/2)

Pin Name	I/O Circuit Type	I/O	Recommended Connection of Unused Pins
P00/INTP0/TI00	2	Input	Connect to V_{SS0}.
P01/INTP1/TI01	8-C	I/O	Independently connect to V_{SS0} via a resistor.
P02/INTP2			
P03/INTP3			
P04/INTP4			
P05/INTP5			
P07/XT1	16	Input	Connect to V_{DD0}.
P10/ANI0 to P17/ANI7	11-B	I/O	Independently connect to V_{DD0} or V_{SS0} via a resistor.
P25/SI0/SB0	10-B		
P26/SO0/SB1			
P27/$\overline{\text{SCK0}}$			
P30/TO0	5-H		
P31/TO1			
P32/TO2			
P33/TI1	8-C		
P34/TI2			
P35/PCL	5-H		
P36/BUZ			
P37			
P70/SI2/RxD	8-C		
P71/SO2/TxD	5-H		
P72/$\overline{\text{SCK2}}$/ASCK	8-C		
P80/S39 to P87/S32	17-C		
P90/S31 to P97/S24			
P100 to P103	5-H		
P110/SI3	8-C		Independently connect to V_{DD0} via a resistor.
P111/SO3			
P112/$\overline{\text{SCK3}}$			
P113/TxD			
P114/RxD			
P115 to P117			
S0 to S23	17-B	Output	Leave open.
COM0 to COM3	18-A		
V_{LC0} to V_{LC2}	—	—	
BIAS			

Table 3-1. Pin I/O Circuit Types (2/2)

Pin Name	I/O Circuit Type	I/O	Recommended Connection of Unused Pins
$\overline{\text{RESET}}$	2	Input	—
XT2	16	—	Leave open.
AV_{REF}	—	—	Connect to V_{SS0}.
AV_{SS}			Connect to V_{SS0}.
IC (mask ROM version)			Connect directly to V_{SS0} or V_{SS1}.
V_{PP} (μPD78P0308)			

Figure 3-1. Pin I/O Circuit List (1/2)

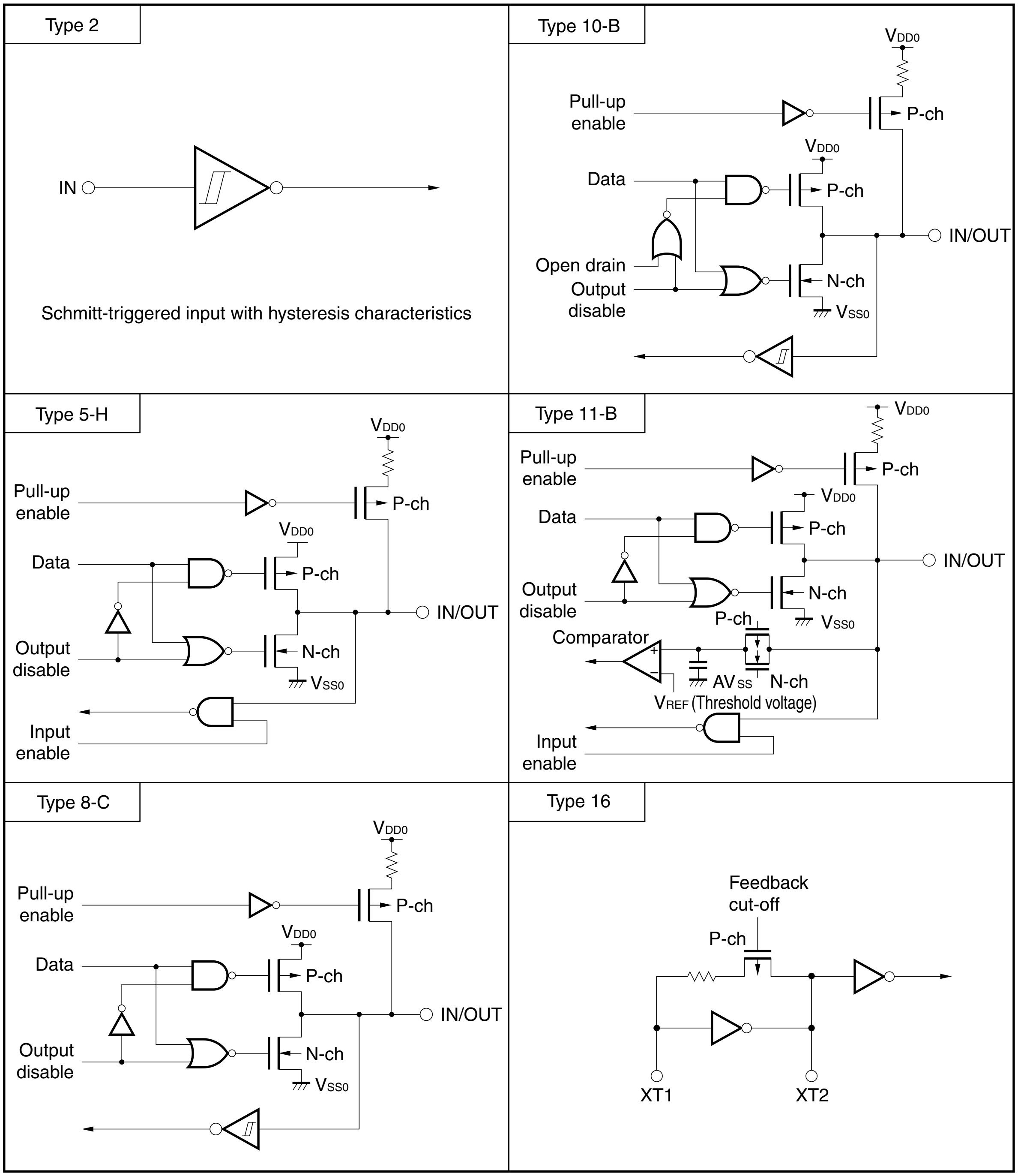
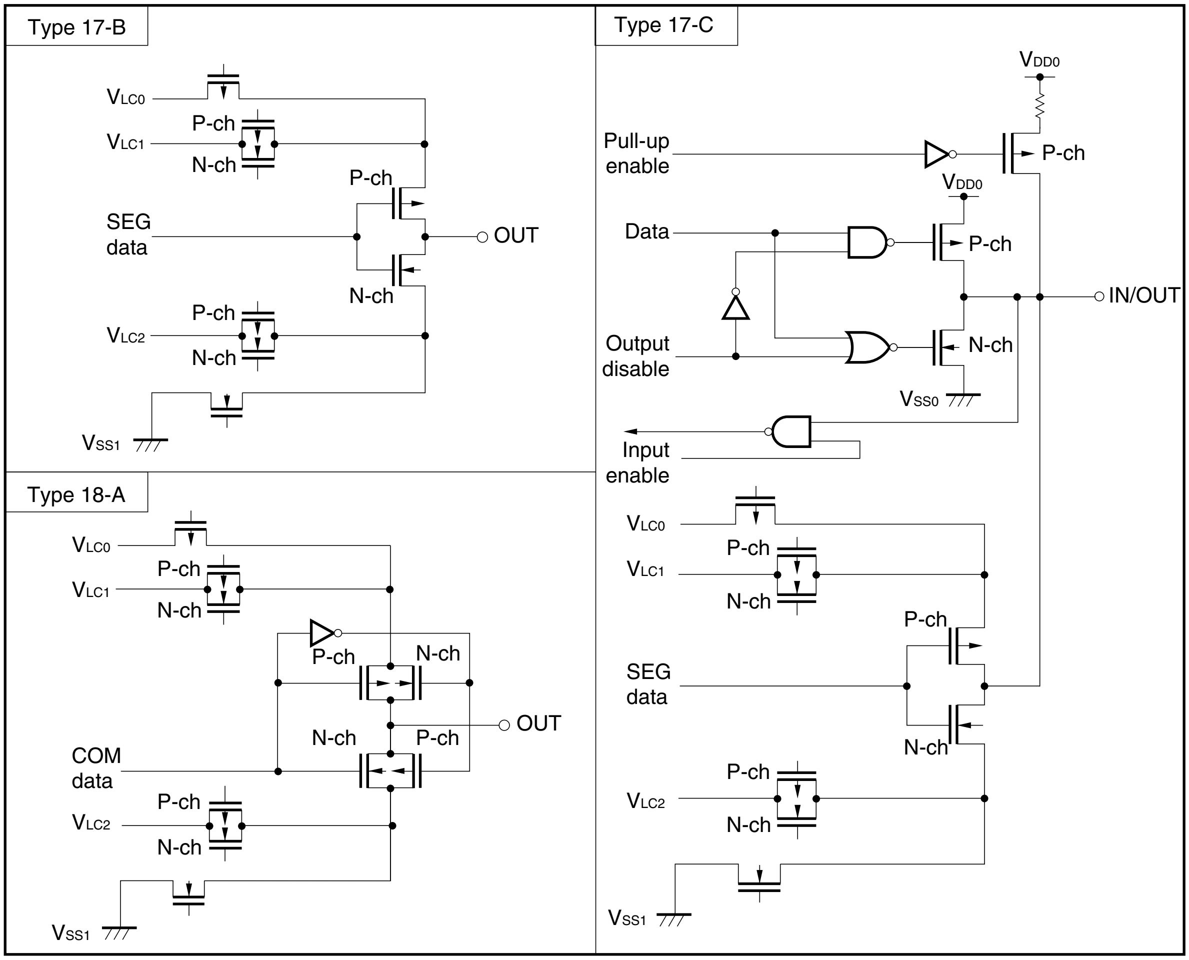


Figure 3-1. Pin I/O Circuit List (2/2)



CHAPTER 4 PIN FUNCTION (μPD780308Y SUBSERIES)

4.1 Pin Function List

4.1.1 Normal operating mode pins

(1) Port pins (1/2)

Pin Name	I/O	Function		After Reset	Alternate Function
P00	Input	Port 0. 7-bit I/O port.	Input only	Input	INTP0/TI00
P01	I/O		Input/output mode can be specified in 1-bit units. If used as an input port, an internal pull-up resistor can be used by software.	Input	INTP1/TI01
P02					INTP2
P03					INTP3
P04					INTP4
P05					INTP5
P07[Note 1]	Input		Input only	Input	XT1
P10 to P17	I/O	Port 1. 8-bit I/O port. Input/output mode can be specified in 1-bit units. If used as an input port, an internal pull-up resistor can be used by software[Note 2].		Input	ANI0 to ANI7
P25	I/O	Port 2. 3-bit I/O port. Input/output mode can be specified in 1-bit units. If used as an input port, an internal pull-up resistor can be used by software.		Input	SI0/SB0/SDA0
P26					SO0/SB1/SDA1
P27					$\overline{\text{SCK0}}$/SCL
P30	I/O	Port 3. 8-bit I/O port. Input/output mode can be specified in 1-bit units. If used as an input port, an internal pull-up resistor can be used by software.		Input	TO0
P31					TO1
P32					TO2
P33					TI1
P34					TI2
P35					PCL
P36					BUZ
P37					—

Notes

1. When the P07/XT1 pin is used as an input port, set bit 6 (FRC) of the processor clock control register (PCC) to 1 (do not use the internal feedback resistor to the subsystem clock oscillator).
2. When pins P10/ANI0 to P17/ANI7 are used as an analog input of the A/D converter, the internal pull-up resistor is automatically disabled.

(1) Port pins (2/2)

Pin Name	I/O	Function	After Reset	Alternate Function
P70	I/O	Port 7. 3-bit I/O port. Input/output mode can be specified in 1-bit units. If used as an input port, an internal pull-up resistor can be used by software.	Input	SI2/RxD
P71				SO2/TxD
P72				$\overline{\text{SCK2}}$/ASCK
P80 to P87	I/O	Port 8. 8-bit I/O port. Input/output mode can be specified in 1-bit units. If used as an input port, an internal pull-up resistor can be used by software. I/O port/segment signal output can be specified in 2-bit units by LCD display control register (LCDC).	Input	S39 to S32
P90 to P97	I/O	Port 9. 8-bit I/O port. Input/output mode can be specified in 1-bit units. If used as an input port, an internal pull-up resistor can be used by software. I/O port/segment signal output can be specified in 2-bit units by LCD display control register (LCDC).	Input	S31 to S24
P100 to P103	I/O	Port 10. 4-bit I/O port. Input/output mode can be specified in 1-bit units. If used as an input port, an internal pull-up resistor can be used by software. LED can be driven directly.	Input	—
P110	I/O	Port 11. 8-bit I/O port. Input/output mode can be specified in 1-bit units. If used as an input port, an internal pull-up resistor can be used by software. Falling edge can be detected.	Input	SI3
P111				SO3
P112				$\overline{\text{SCK3}}$
P113				TxD
P114				RxD
P115 to P117				—

(2) Non-port pins (1/2)

Pin Name	I/O	Function	After Reset	Alternate Function
INTP0	Input	External interrupt request inputs with specifiable valid edges (rising edge, falling edge, both rising and falling edges).	Input	P00/TI00
INTP1				P01/TI01
INTP2				P02
INTP3				P03
INTP4				P04
INTP5				P05
SI0	Input	Serial interface serial data input.	Input	P25/SB0/SDA0
SI2				P70/RxD
SI3				P110
SO0	Output	Serial interface serial data output.	Input	P26/SB1/SDA1
SO2				P71/TxD
SO3				P111
SB0	I/O	Serial interface serial data input/output.	Input	P25/SI0/SDA0
SB1				P26/SO0/SDA1
SDA0				P25/SI0/SB0
SDA1				P26/SO0/SB1
$\overline{\text{SCK0}}$	I/O	Serial interface serial clock input/output.	Input	P27/SCL
$\overline{\text{SCK2}}$				P72/ASCK
$\overline{\text{SCK3}}$				P112
SCL				P27/$\overline{\text{SCK0}}$
RxD	Input	Asynchronous serial interface serial data input.	Input	P70/SI2, P114
TxD	Output	Asynchronous serial interface serial data output.	Input	P71/SO2, P113
ASCK	Input	Asynchronous serial interface serial clock input.	Input	P72/$\overline{\text{SCK2}}$
TI00	Input	External count clock input to 16-bit timer (TM0).	Input	P00/INTP0
TI01		Capture trigger signal input to capture register (CR00).		P01/INTP1
TI1		External count clock input to 8-bit timer (TM1).		P33
TI2		External count clock input to 8-bit timer (TM2).		P34
TO0	Output	16-bit timer (TM0) output (also used for 14-bit PWM output).	Input	P30
TO1		8-bit timer (TM1) output.		P31
TO2		8-bit timer (TM2) output.		P32
PCL	Output	Clock output (for main system clock and subsystem clock trimming).	Input	P35
BUZ	Output	Buzzer output.	Input	P36
S0 to S23	Output	Segment signal output of LCD controller/driver.	Output	—
S24 to S31			Input	P97 to P90
S32 to S39				P87 to P80
COM0 to COM3	Output	Common signal output of LCD controller/driver.	Output	—
V$_{LC0}$ to V$_{LC2}$	—	LCD drive voltage (mask ROM versions can incorporate split resistor (mask option)).	—	—
BIAS	—	Power supply for LCD drive.	—	—
ANI0 to ANI7	Input	A/D converter analog input.	Input	P10 to P17
AV$_{REF}$	Input	A/D converter reference voltage input (also used for analog power).	—	—

(2) Non-port pins (2/2)

Pin Name	I/O	Function	After Reset	Alternate Function
AV_{SS}	—	A/D converter ground potential. Same potential as V_{SS0}.	—	—
$\overline{\text{RESET}}$	Input	System reset input.	—	—
X1	Input	Crystal connection for main system clock oscillation.	—	—
X2	—		—	—
XT1	Input	Crystal connection for subsystem clock oscillation.	Input	P07
XT2	—		—	—
V_{DD0}	—	Positive power supply to port.	—	—
V_{SS0}	—	Ground potential of port.	—	—
V_{DD1}	—	Positive power supply (except ports, analogs).	—	—
V_{SS1}	—	Ground potential (except ports, analogs).	—	—
V_{PP}	—	High-voltage application for program write/verify. Connect directly to V_{SS0} or V_{SS1} in normal operating mode.	—	—
IC	—	Internal connection. Connect directly to V_{SS0} or V_{SS1}.	—	—

4.1.2 PROM programming mode pins (μPD78P0308Y only)

Pin Name	I/O	Function
$\overline{\text{RESET}}$	Input	PROM programming mode setting. When +5 V or +12.5 V is applied to the V_{PP} pin or a low level voltage is applied to the $\overline{\text{RESET}}$ pin, the PROM programming mode is set.
V_{PP}	Input	High-voltage application for PROM programming mode setting and program write/verify.
A0 to A16	Input	Address bus.
D0 to D7	I/O	Data bus.
$\overline{\text{CE}}$	Input	PROM enable input/program pulse input.
$\overline{\text{OE}}$	Input	Read strobe input to PROM.
$\overline{\text{PGM}}$	Input	Program/program inhibit input in PROM programming mode.
V_{DD}	—	Positive power supply.
V_{SS}	—	Ground potential.

4.2 Description of Pin Functions

4.2.1 P00 to P05, P07 (Port 0)

These are 7-bit I/O ports. Besides serving as I/O ports, they function as an external interrupt request input, an external count clock input to the timer, a capture trigger signal input, and crystal connection for subsystem clock oscillation.

The following operating modes can be specified in 1-bit units.

(1) Port mode

P00 and P07 function as input-only ports and P01 to P05 function as I/O ports.

P01 to P05 can be specified as input or output ports in 1-bit units with port mode register 0 (PM0). When they are used as input ports, internal pull-up resistors can be used by defining pull-up resistor option register L (PUOL).

(2) Control mode

These ports function as an external interrupt request input, an external count clock input to the timer, and crystal connection for subsystem clock oscillation.

(a) INTP0 to INTP5

INTP0 to INTP5 are external interrupt request input pins which can specify valid edges (rising edge, falling edge, and both rising and falling edges). INTP0 or INTP1 becomes a 16-bit timer/event counter capture trigger signal input pin with a valid edge input.

(b) TI00

Pin for external count clock input to 16-bit timer/event counter.

(c) TI01

Pin for capture trigger signal input to capture register (CR00) of 16-bit timer/event counter.

(d) XT1

Crystal connect pin for subsystem clock oscillation.

4.2.2 P10 to P17 (Port 1)

These are 8-bit I/O ports. Besides serving as I/O ports, they function as an A/D converter analog input.
The following operating modes can be specified in 1-bit units.

(1) Port mode

These ports function as 8-bit I/O ports. They can be specified as input or output ports in 1-bit units with port mode register 1 (PM1). When they are used as input ports, internal pull-up resistors can be used by defining pull-up resistor option register L (PUOL).

(2) Control mode

These ports function as A/D converter analog input pins (ANI0 to ANI7). The pull-up resistor is automatically disabled when the pins are specified for analog input.

4.2.3 P25 to P27 (Port 2)

These are 3-bit I/O ports. Besides serving as I/O ports, they function as data I/O and clock I/O of the serial interface.
The following operating modes can be specified in 1-bit units.

(1) Port mode

These ports function as 3-bit I/O ports. They can be specified as input or output ports in 1-bit units with port mode register 2 (PM2). When they are used as input ports, internal pull-up resistors can be used by defining pull-up resistor option register L (PUOL).

(2) Control mode

These ports function as serial interface data I/O and clock I/O.

(a) SI0, SO0, SB0, SB1, SDA0, SDA1

Serial interface serial data I/O pins.

(b) $\overline{\text{SCK0}}$, SCL

Serial interface serial clock I/O pins.

Caution When this port is used as a serial interface, the I/O and output latches must be set according to the function used. For the setting, refer to Figure 16-4 Serial Operating Mode Register 0 Format.

4.2.4 P30 to P37 (Port 3)

These are 8-bit I/O ports. Besides serving as I/O ports, they function as timer I/O, clock output, and buzzer output. The following operating modes can be specified in 1-bit units.

(1) Port mode

These ports function as 8-bit I/O ports. They can be specified as input or output ports in 1-bit units with port mode register 3 (PM3). When they are used as input ports, internal pull-up resistors can be used by defining pull-up resistor option register L (PUOL).

(2) Control mode

These ports function as timer I/O, clock output, and buzzer output.

(a) TI1 and TI2

Pins for external clock input to the 8-bit timer/event counter.

(b) TO0 to TO2

Timer output pins.

(c) PCL

Clock output pin.

(d) BUZ

Buzzer output pin.

4.2.5 P70 to P72 (Port 7)

These are 3-bit I/O ports. Besides serving as I/O ports, they function as serial interface data I/O and clock I/O. The following operating modes can be specified in 1-bit units.

(1) Port mode

These ports function as 3-bit I/O ports. They can be specified as input or output ports in 1-bit units with port mode register 7 (PM7). When they are used as input ports, internal pull-up resistors can be used by defining pull-up resistor option register L (PUOL).

(2) Control mode

These ports function as serial interface data I/O and clock I/O.

(a) SI2, SO2

Serial interface serial data I/O pins.

(b) $\overline{\text{SCK2}}$

Serial interface serial clock I/O pin.

(c) RxD, TxD

Asynchronous serial interface serial data I/O pins.

(d) ASCK

Asynchronous serial interface serial clock input pin.

Caution When this port is used as a serial interface, the I/O and output latches must be set according to the function used. For the setting, refer to Table 17-2 Serial Interface Channel 2 Operating Mode Settings.

4.2.6 P80 to P87 (Port 8)

These are 8-bit I/O ports. Besides serving as I/O ports, they function as segment signal output of LCD controller/driver.

The following operating modes can be specified in 1-bit units.

(1) Port mode

These ports function as 8-bit I/O ports. They can be specified as input or output ports in 1-bit units with port mode register 8 (PM8). When they are used as input ports, internal pull-up resistors can be used by defining pull-up resistor option register H (PUOH).

(2) Control mode

These ports function as segment signal output pins (S32 to S39) of LCD controller/driver.

4.2.7 P90 to P97 (Port 9)

These are 8-bit I/O ports. Besides serving as I/O ports, they function as segment signal output of LCD controller/driver.

The following operating modes can be specified in 1-bit units.

(1) Port mode

These ports function as 8-bit I/O ports. They can be specified as input or output ports in 1-bit units with port mode register 9 (PM9). When they are used as input ports, internal pull-up resistors can be used by defining pull-up resistor option register H (PUOH).

(2) Control mode

These ports function as segment signal output pins (S24 to S31) of LCD controller/driver.

4.2.8 P100 to P103 (Port 10)

These are 4-bit I/O ports. They can be specified as input or output ports in 1-bit units with port mode register 10 (PM10). When they are used as input ports, internal pull-up resistors can be used by defining pull-up resistor option register H (PUOH).

LED can be driven directly.

4.2.9 P110 to P117 (Port 11)

These are 8-bit I/O ports. Besides serving as I/O ports, they function as serial interface data I/O and clock I/O. The following operating modes can be specified in 1-bit units.

(1) Port mode

These ports function as 8-bit I/O ports. They can be specified as input or output ports in 1-bit units with port mode register 11 (PM11). When they are used as input ports, internal pull-up resistors can be used by defining pull-up resistor option register H (PUOH).

When the falling edge is detected on a specified bit of this port, test input flag (KRIF) can be set to 1.

(2) Control mode

These ports function as serial interface data I/O and clock I/O.

(a) SI3, SO3

Serial interface serial data I/O pins.

(b) $\overline{\text{SCK3}}$

Serial interface serial clock I/O pin.

(c) RxD, TxD

Asynchronous serial interface serial data I/O pins.

Caution When this port is used as a serial interface, the I/O and output latches must be set according to the function used. For the setting, refer to Table 17-2 Serial Interface Channel 2 Operating Mode Settings, and Figure 18-3 Serial Operating Mode Register 3 Format.

4.2.10 COM0 to COM3

These are LCD controller/driver common signal output pins. They output common signals under either of the following conditions:

– when the static mode is selected (COM0 to COM3 outputs)

– when 2-time-division (COM0, COM1 outputs) or 3-time-division (COM0 to COM2 outputs) operation is performed in 1/2 bias mode

– when 3-time-division (COM0 to COM2 outputs) or 4-time-division (COM0 to COM3 outputs) operation is performed in 1/3 bias mode

4.2.11 V_{LC0} to V_{LC2}

These are LCD-driving voltage pins. The mask ROM versions can have split resistors by mask option so that LCD driving voltage can be supplied inside the V_{LC0} to V_{LC2} pins according to the required bias without connecting external split resistors.

4.2.12 BIAS

This is a LCD driving power supply pin. This pin should be connected to the V_{LC0} pin to realize user-desired LCD drive voltages to change resistance division ratios, or should be connected to external resistors together with the V_{LC0} to V_{LC2} pins and V_{SS1} pin to fine-adjust the LCD-driving power voltage.

4.2.13 AVREF

This pin inputs the reference voltage for the on-chip A/D converter. This pin also functions to supply power to the internal analog circuit. Supply power to this pin when using the A/D converter.

When not using the A/D converter, connect this pin to the V_{SS0} line.

4.2.14 AVSS

This is a ground potential pin of A/D converter. Always use the same voltage as that of the V_{SS0} pin even when A/D converter is not used.

4.2.15 $\overline{\text{RESET}}$

This is a low-level active system reset input pin.

4.2.16 X1 and X2

Crystal resonator connection pins for main system clock oscillation.

For external clock supply, input it to X1 and its inverted signal to X2.

4.2.17 XT1 and XT2

Crystal resonator connection pins for subsystem clock oscillation.

For external clock supply, input it to XT1 and its inverted signal to XT2.

4.2.18 V_{DD0}, V_{DD1}

V_{DD0} supplies positive power to the ports.

V_{DD1} supplies positive power to the circuits other than those of the ports.

4.2.19 V_{SS0}, V_{SS1}

V_{SS0} is the ground pin of the ports.

V_{SS1} is the ground pin of the circuits other than those of the ports.

4.2.20 V_{PP} (μPD78P0308Y only)

High-voltage apply pin for PROM programming mode setting and program write/verify.

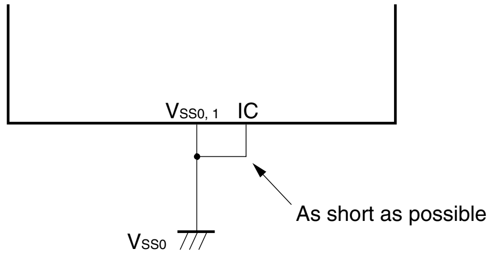
Connect directly to V_{SS0} or V_{SS1} in normal operating mode.

4.2.21 IC (Mask ROM version only)

The IC (Internally Connected) pin is provided to set the test mode to check the μPD780308Y Subseries at delivery. Connect it directly to the V_{SS0} or V_{SS1} with the shortest possible wire in the normal operating mode.

When a voltage difference is produced between the IC pin and V_{SS0} or V_{SS1} pin because the wiring between those two pins is too long or an external noise is input to the IC pin, the user's program may not run normally.

○ Connect IC pins to V_{SS0} or V_{SS1} pins directly.



4.3 Pin I/O Circuits and Recommended Connection of Unused Pins

Table 4-1 shows the I/O circuit types of pins and the recommended connections of unused pins.
Refer to Figure 4-1 for the configuration of the I/O circuit of each type.

Table 4-1. Pin I/O Circuit Types (1/2)

Pin Name	I/O Circuit Type	I/O	Recommended Connection of Unused Pins
P00/INTP0/TI00	2	Input	Connect to V_{SS0}.
P01/INTP1/TI01	8-C	I/O	Independently connect to V_{SS0} via a resistor.
P02/INTP2			
P03/INTP3			
P04/INTP4			
P05/INTP5			
P07/XT1	16	Input	Connect to V_{DD0}.
P10/ANI0 to P17/ANI7	11-B	I/O	Independently connect to V_{DD0} or V_{SS0} via a resistor.
P25/SI0/SB0/SDA0	10-B		
P26/SO0/SB1/SDA1			
P27/$\overline{\text{SCK0}}$/SCL			
P30/TO0	5-H		
P31/TO1			
P32/TO2			
P33/TI1	8-C		
P34/TI2			
P35/PCL	5-H		
P36/BUZ			
P37			
P70/SI2/RxD	8-C		
P71/SO2/TxD	5-H		
P72/$\overline{\text{SCK2}}$/ASCK	8-C		
P80/S39 to P87/S32	17-C		
P90/S31 to P97/S24			
P100 to P103	5-H		
P110/SI3	8-C		Independently connect to V_{DD0} via a resistor.
P111/SO3			
P112/$\overline{\text{SCK3}}$			
P113/TxD			
P114/RxD			
P115 to P117			
S0 to S23	17-B	Output	Leave open.
COM0 to COM3	18-A		
V_{LC0} to V_{LC2}	—	—	
BIAS			

Table 4-1. Pin I/O Circuit Types (2/2)

Pin Name	I/O Circuit Type	I/O	Recommended Connection of Unused Pins
$\overline{\text{RESET}}$	2	Input	—
XT2	16	—	Leave open.
AV_{REF}	—	—	Connect to V_{SS0}.
AV_{SS}			Connect to V_{SS0}.
IC (mask ROM version)			Connect directly to V_{SS0} or V_{SS1}.
V_{PP} (μPD78P0308Y)			

Figure 4-1. Pin I/O Circuit List (1/2)

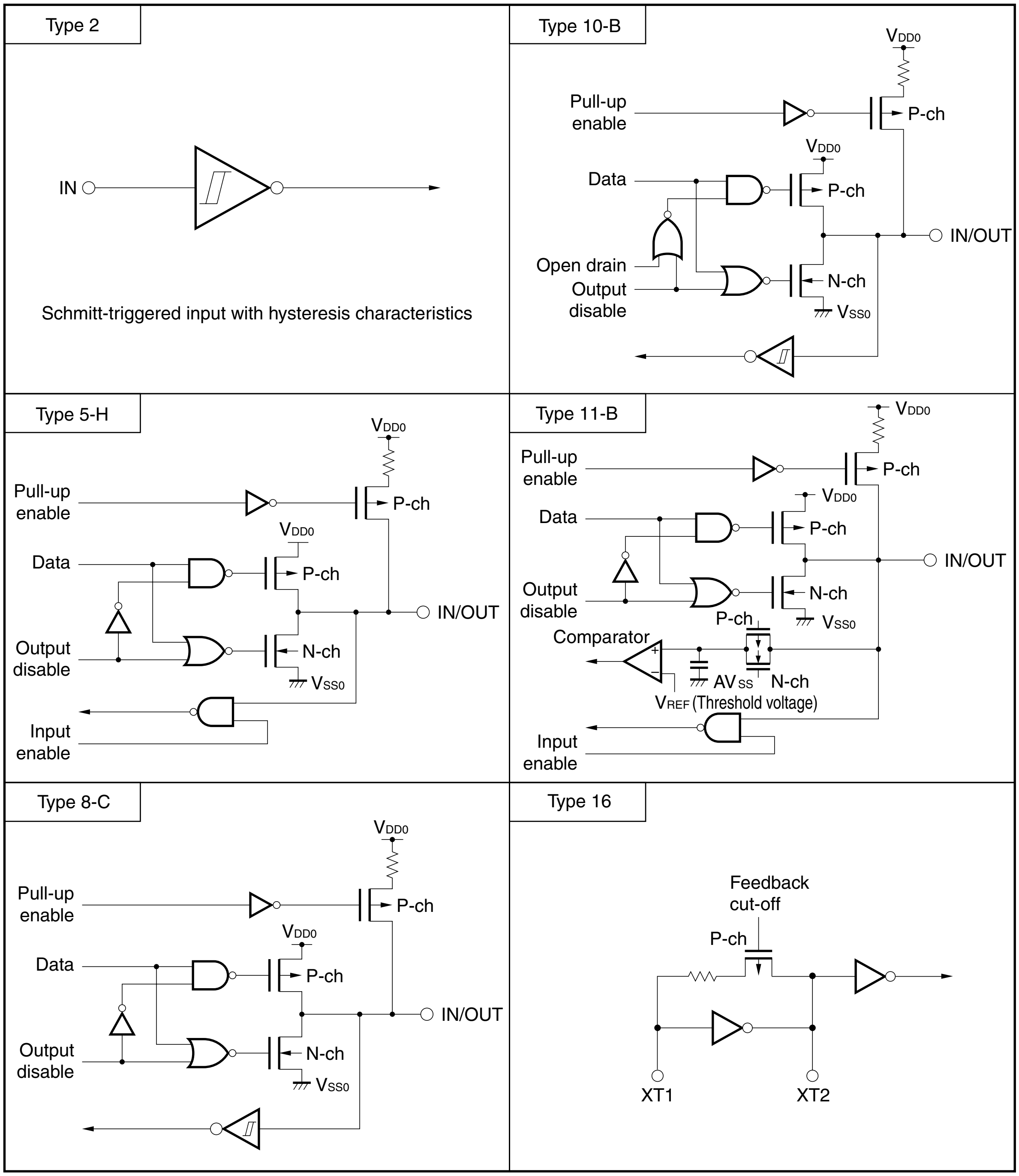
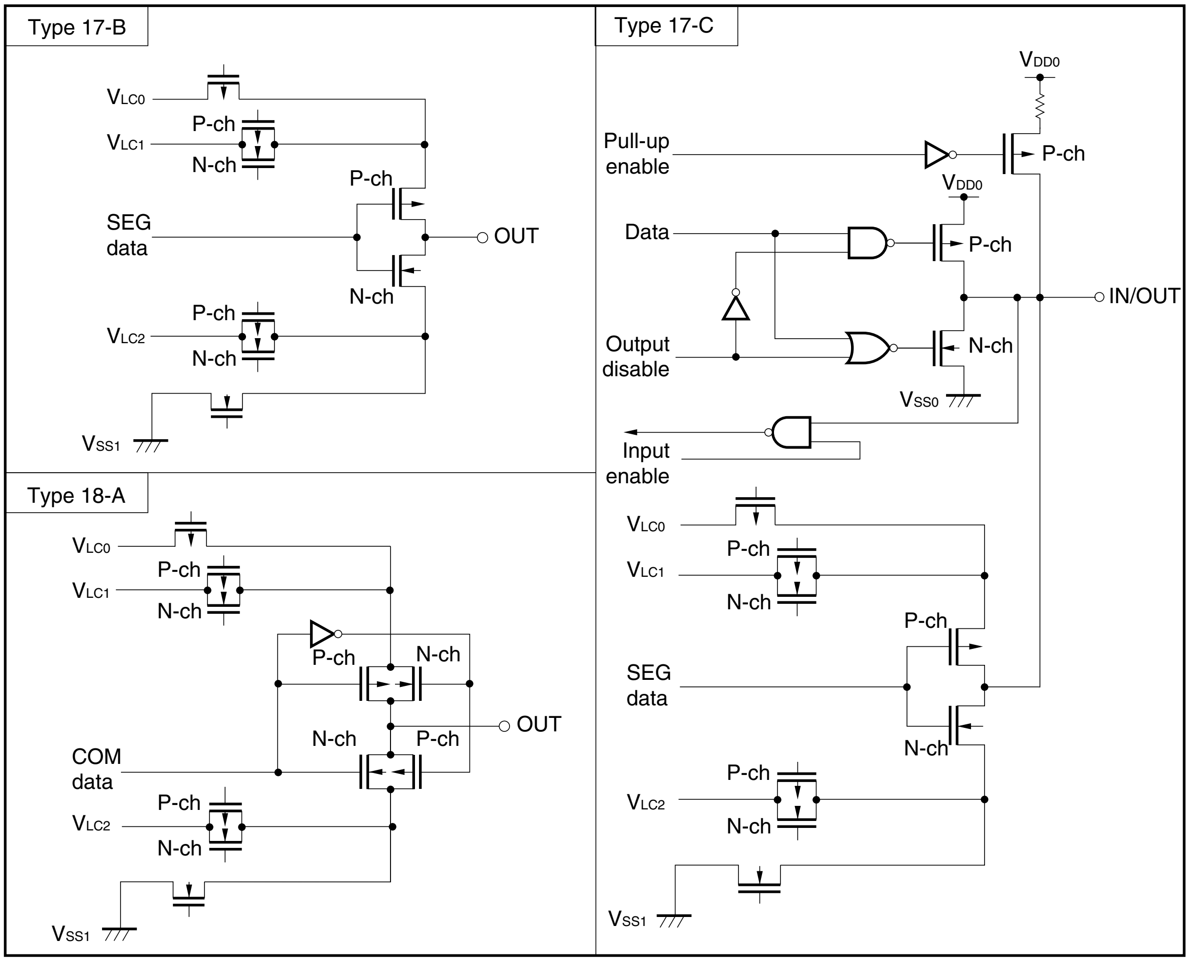


Figure 4-1. Pin I/O Circuit List (2/2)



5.1 Memory Spaces

The μPD780308 and 780308Y Subseries can access a 64 KB memory space. Figures 5-1 to 5-3 show memory maps.

Figure 5-1. Memory Map (μPD780306, 780306Y)

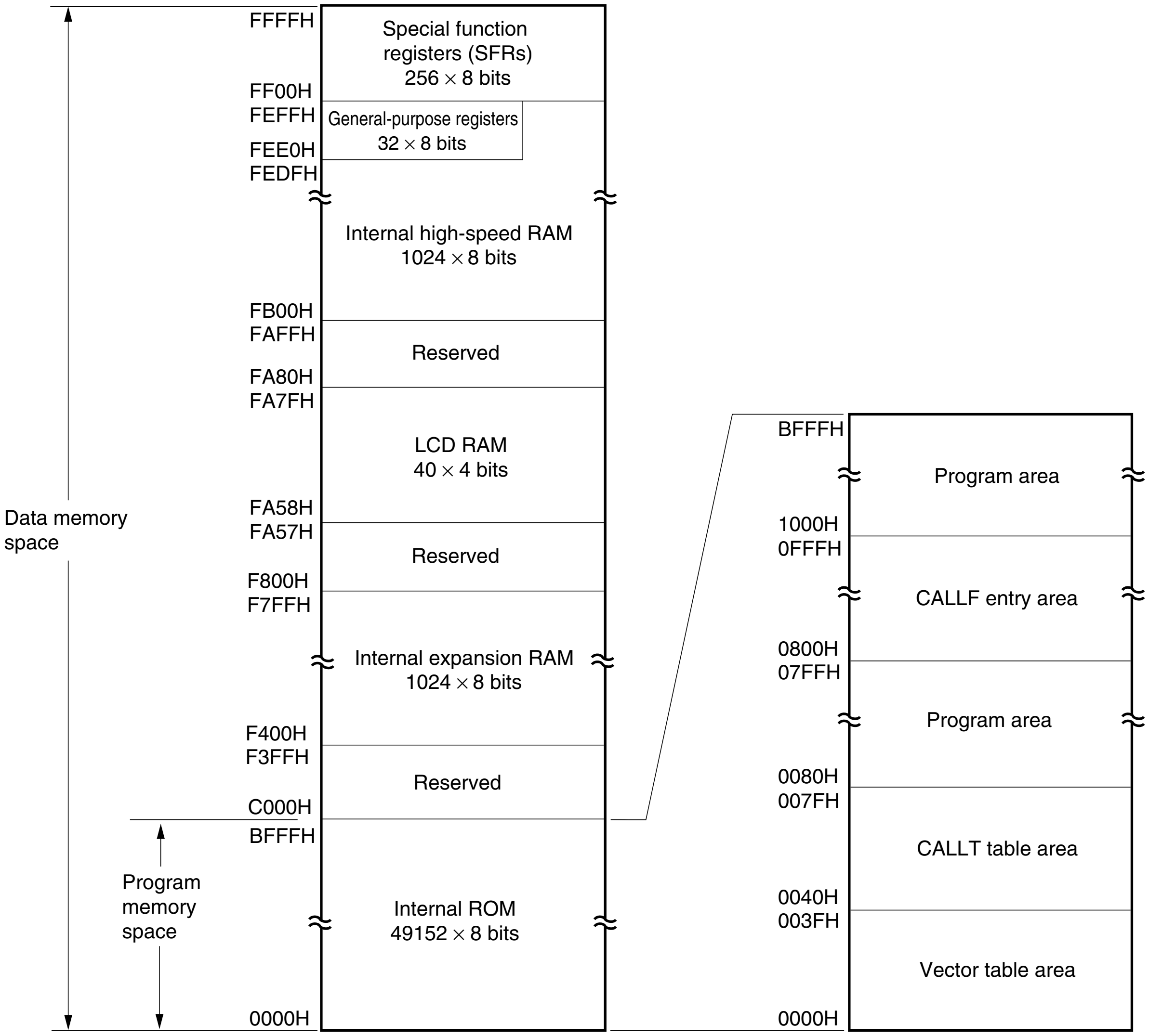


Figure 5-2. Memory Map (μPD780308, 780308Y)

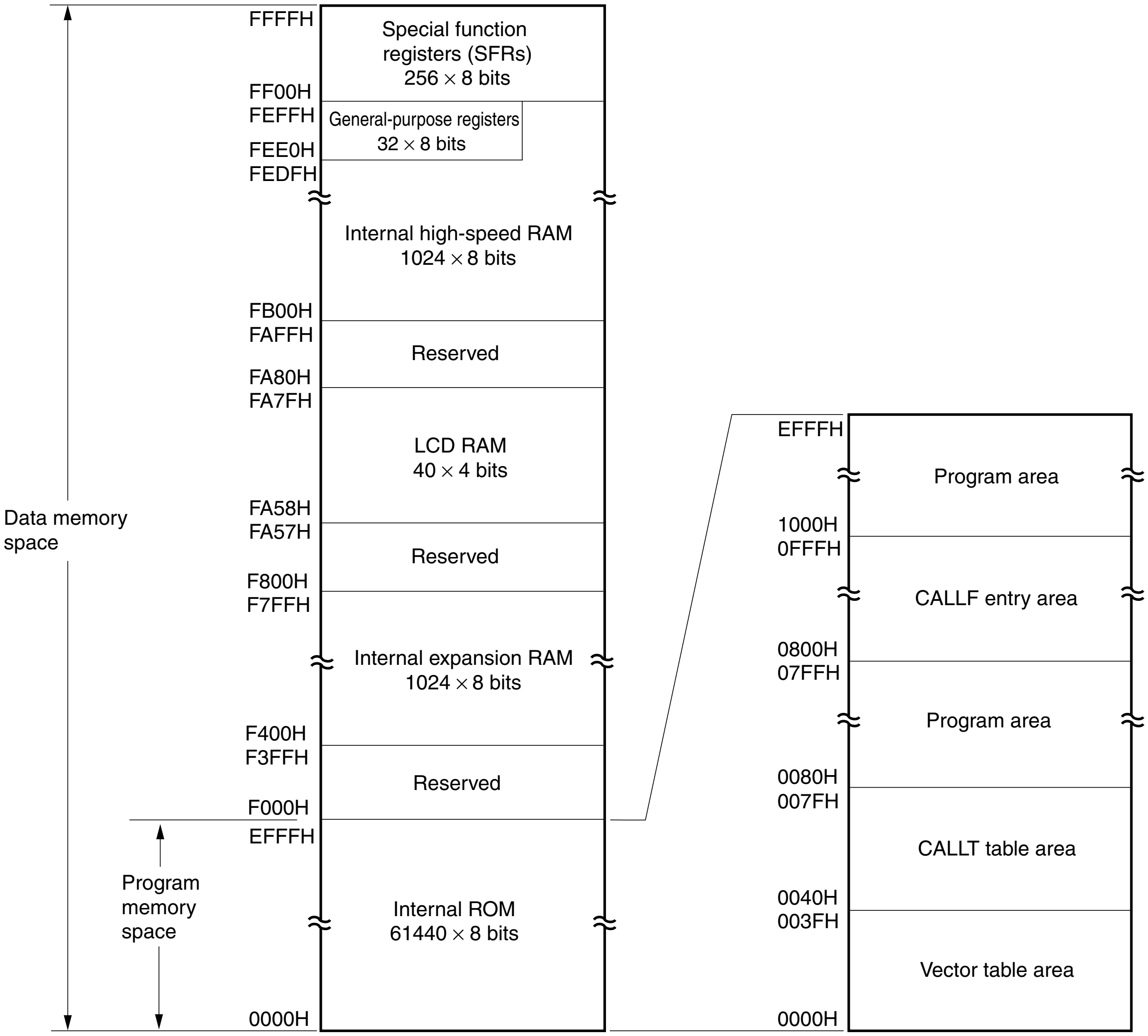
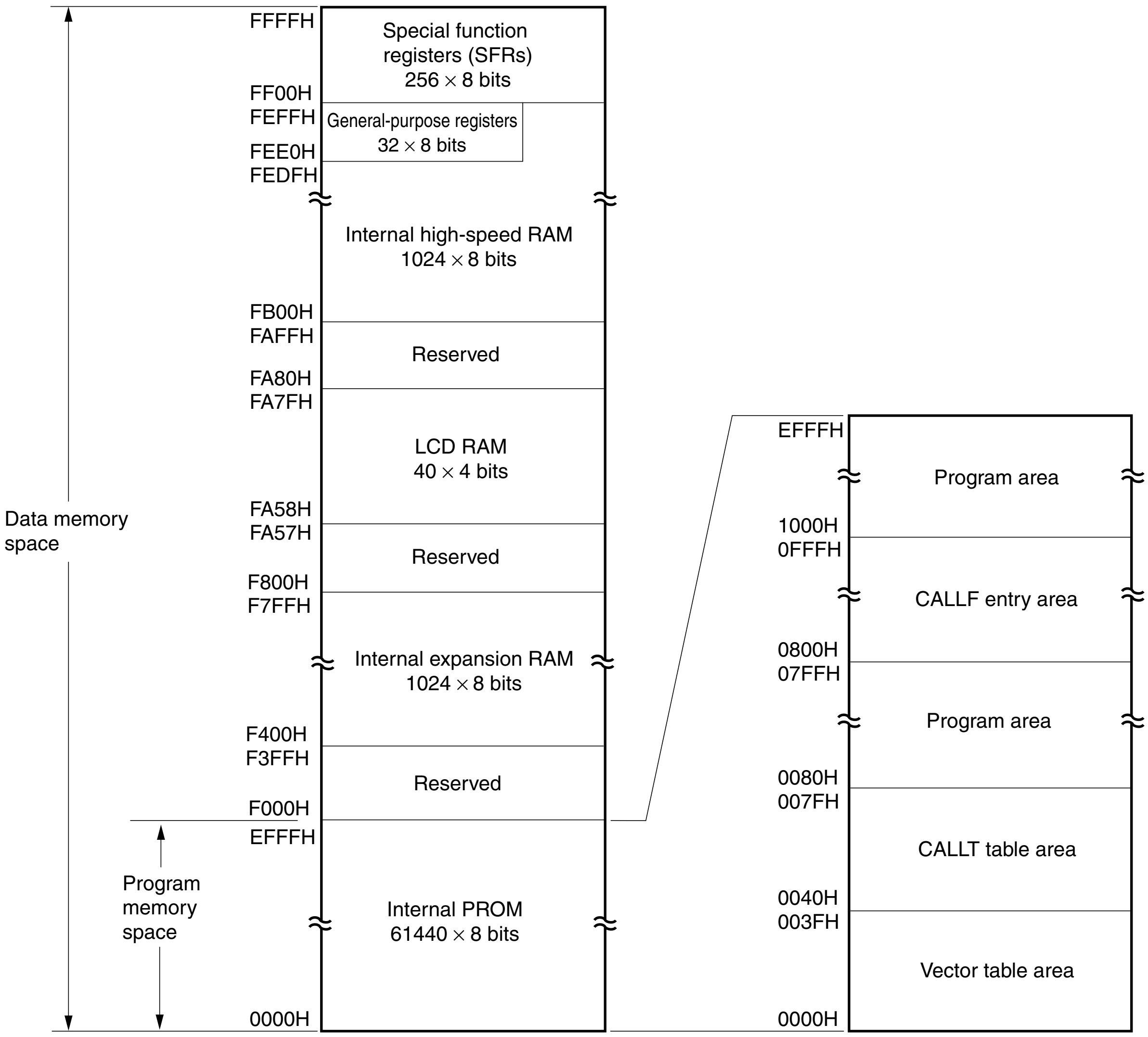


Figure 5-3. Memory Map (μPD78P0308, 78P0308Y)



5.1.1 Internal program memory space

The internal program memory space stores program data and table data. This space is generally accessed with program counter (PC).

The µPD780308, 780308Y Subseries has internal ROM (or PROM) and the capacity of the memory varies depending on the part number.

Table 5-1. Internal ROM Capacity

Part Number	Internal ROM	
	Type	Capacity
µPD780306, 780306Y	Mask ROM	49152 × 8 bits
µPD780308, 780308Y		61440 × 8 bits
µPD78P0308, 78P0308Y	PROM	

The internal program memory is divided into the following three areas.

(1) Vector table area

The 64-byte area 0000H to 003FH is reserved as a vector table area. The $\overline{\text{RESET}}$ input and program start addresses for branch upon generation of each interrupt request are stored in the vector table area. Of the 16-bit address, lower 8 bits are stored at even addresses and higher 8 bits are stored at odd addresses.

Table 5-2. Vector Table

Vector Table Address	Interrupt Source
0000H	$\overline{\text{RESET}}$ input
0004H	INTWDT
0006H	INTP0
0008H	INTP1
000AH	INTP2
000CH	INTP3
000EH	INTP4
0010H	INTP5
0014H	INTCSI0
0018H	INTSER
001AH	INTSR/INTCSI2
001CH	INTST
001EH	INTTM3
0020H	INTTM00
0022H	INTTM01
0024H	INTTM1
0026H	INTTM2
0028H	INTAD
002AH	INTCSI1
003EH	BRK

(2) CALLT instruction table area

The 64-byte area 0040H to 007FH can store the subroutine entry address of a 1-byte call instruction (CALLT).

(3) CALLF instruction entry area

The area 0800H to 0FFFH can perform a direct subroutine call with a 2-byte call instruction (CALLF).

5.1.2 Internal data memory space

The μPD780308 and 780308Y Subseries units incorporate the following RAMs.

(1) Internal high-speed RAM

The internal high-speed RAM space consists of 1024 × 8 bits, or addresses FB00H to FEFFH. In this area, four banks of general-purpose registers, each bank consisting of eight 8-bit registers, are allocated in the 32-byte area FEE0H to FEFFH.

The internal high-speed RAM can also be used as a stack.

(2) Internal expansion RAM

Internal expansion RAM is allocated to the 1024-byte area of addresses F400H to F7FFH.

(3) LCD display RAM

Addresses FA58H to FA7FH of 40 × 4 bits are allocated for LCD display RAM, However, this area can also be used as general-purpose RAM.

5.1.3 Special-function register (SFR) area

An on-chip peripheral hardware special-function register (SFR) is allocated in the area FF00H to FFFFH (refer to **Table 5-3**).

Caution Do not access addresses where the SFR is not assigned.

5.1.4 Data memory addressing

Addressing is to specify the address of the instruction to be executed next or the address of a register or memory to be manipulated when an instruction is executed.

The address of the instruction to be executed next is specified by the program counter (for details, refer to **5.3 Instruction Address Addressing**).

To specify the address of the memory to be manipulated when an instruction is executed, the μPD780308 and 780308Y Subseries are provided with many addressing modes to improve operability. Especially at addresses corresponding to data memory area (FB00H to FFFFH), particular addressing modes are possible to meet the functions of the special function registers (SFRs) and general-purpose registers. This area is between FB00H and FFFFH. Figures 5-4 to 5-6 show the data memory addressing modes.

For details of each addressing, refer to **5.4 Operand Address Addressing**.

Figure 5-4. Data Memory Addressing (μPD780306, 780306Y)

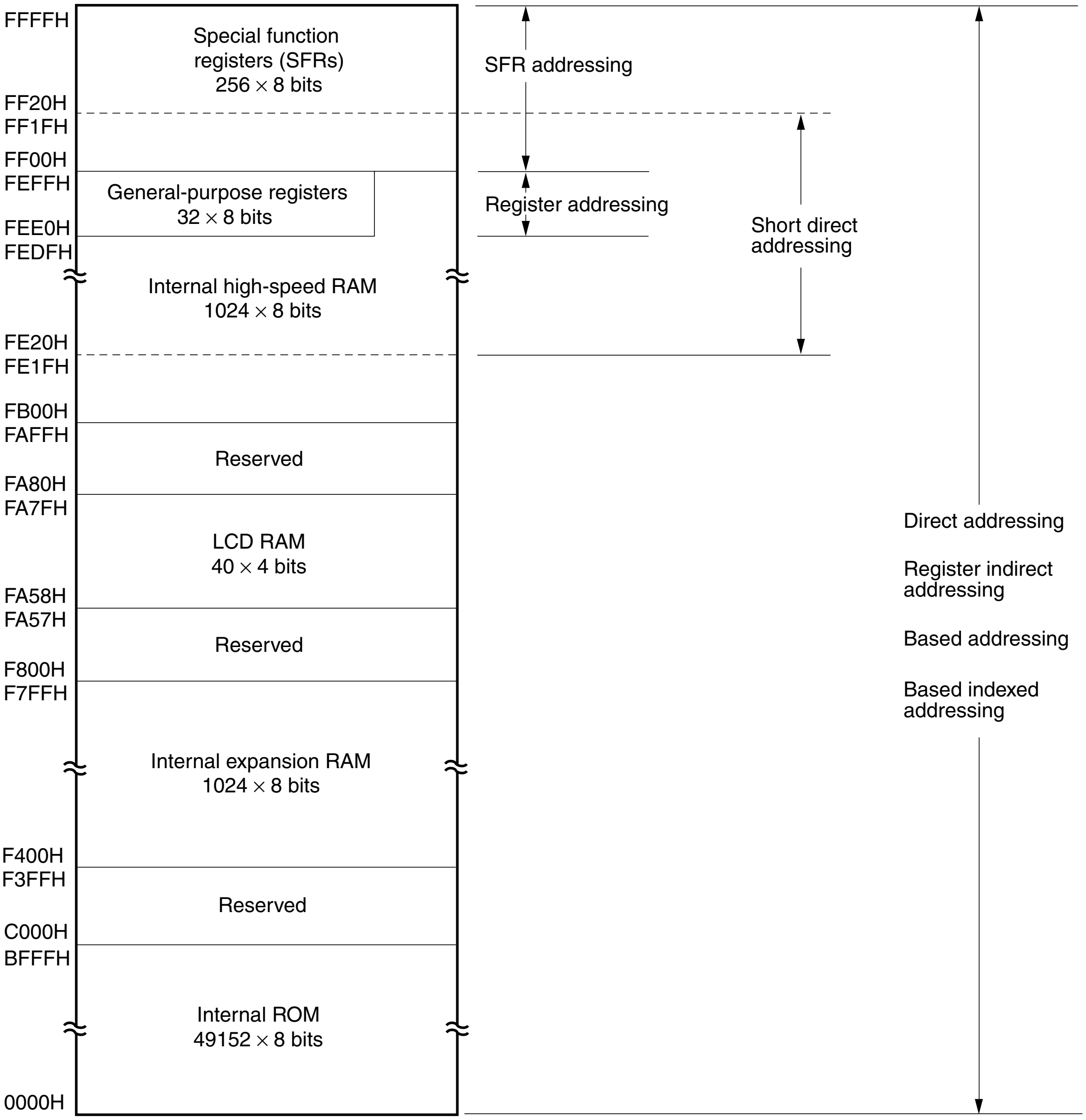


Figure 5-5. Data Memory Addressing (μPD780308, 780308Y)

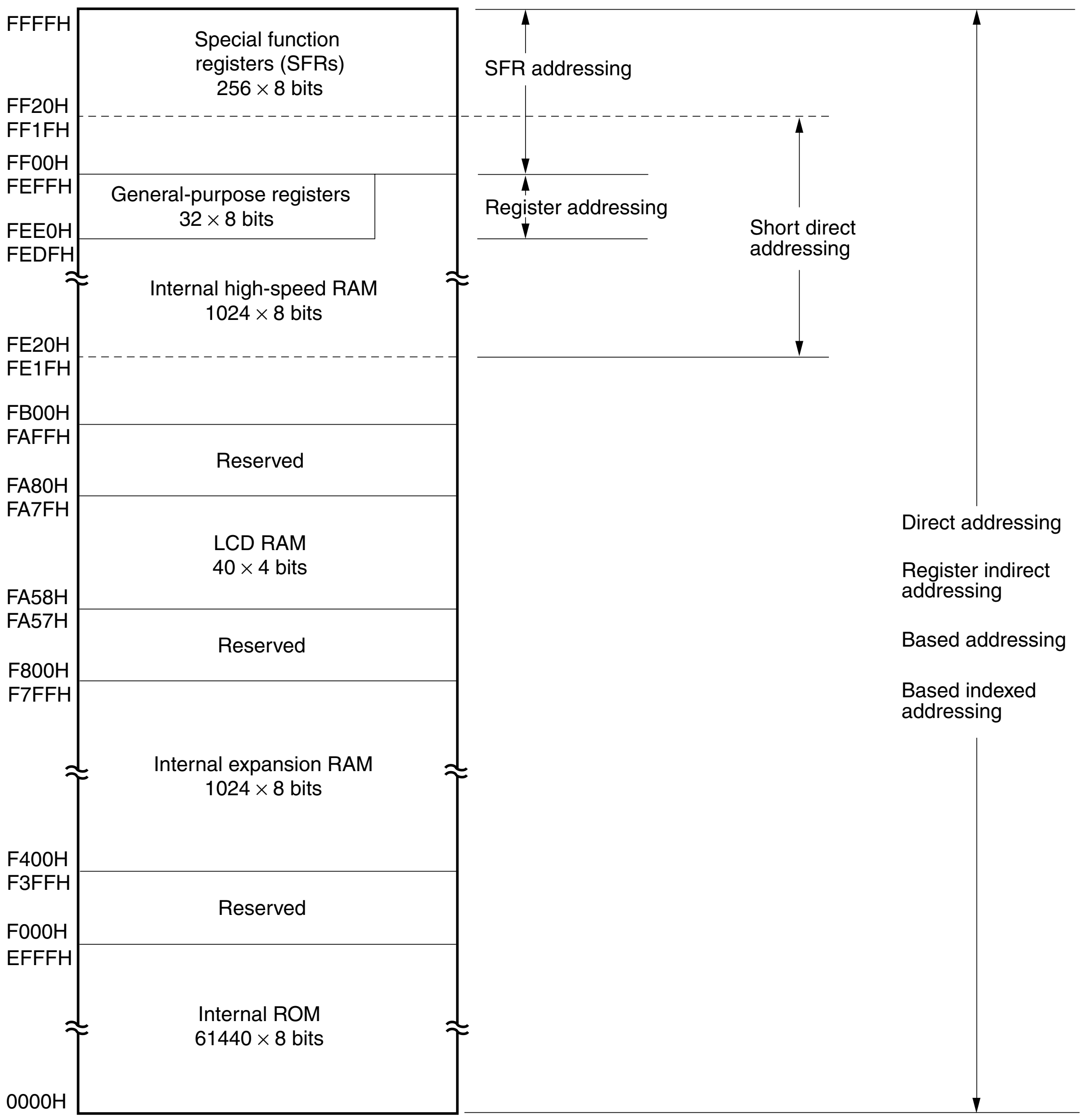
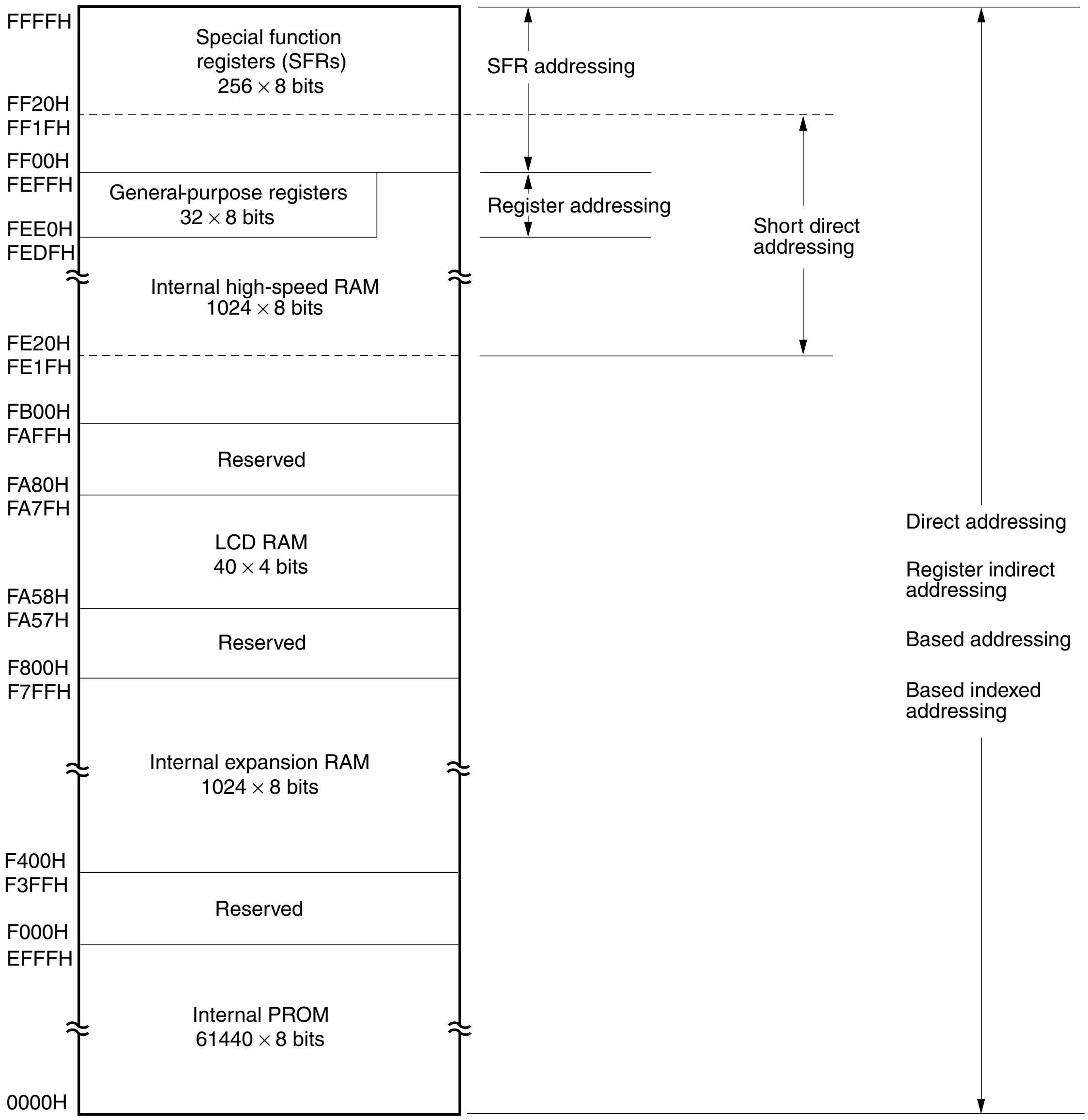


Figure 5-6. Data Memory Addressing (μPD78P0308, 78P0308Y)



5.2 Processor Registers

The μPD780308 and 780308Y Subseries units incorporate the following processor registers.

5.2.1 Control registers

The control registers control the program sequence, statuses and stack memory. The control registers consist of a program counter (PC), a program status word (PSW), and a stack pointer (SP).

(1) Program counter (PC)

The program counter is a 16-bit register which holds the address information of the next program to be executed.

In normal operation, the PC is automatically incremented according to the number of bytes of the instruction to be fetched. When a branch instruction is executed, immediate data and register contents are set.

$\overline{\text{RESET}}$ input sets the reset vector table values at addresses 0000H and 0001H to the program counter.

Figure 5-7. Program Counter Configuration

	15															0
PC	PC15	PC14	PC13	PC12	PC11	PC10	PC9	PC8	PC7	PC6	PC5	PC4	PC3	PC2	PC1	PC0

(2) Program status word (PSW)

The program status word is an 8-bit register consisting of various flags to be set/reset by instruction execution.

Program status word contents are automatically stacked upon interrupt request generation or PUSH PSW instruction execution and are automatically reset upon execution of the RETB, RETI and POP PSW instructions.

$\overline{\text{RESET}}$ input sets the PSW to 02H.

Figure 5-8. Program Status Word Configuration

	7							0
PSW	IE	Z	RBS1	AC	RBS0	0	ISP	CY

(a) Interrupt enable flag (IE)

This flag controls the interrupt request acknowledge operations of the CPU.

When IE = 0, interrupt requests are disabled (DI), and all interrupts except the non-maskable interrupt are disabled.

When IE = 1, the interrupts are enabled. At this time, acknowledging interrupt requests is controlled with an in-service priority flag (ISP), an interrupt mask flag for various interrupt sources, and a priority specification flag.

This flag is reset (to 0) upon DI instruction execution or interrupt acknowledgment and is set (to 1) upon EI instruction execution.

(b) Zero flag (Z)

When the operation result is zero, this flag is set (to 1). It is reset (to 0) in all other cases.

(c) Register bank select flags (RBS0 and RBS1)

These are 2-bit flags to select one of the four register banks.

In these flags, the 2-bit information which indicates the register bank selected by SEL RBn instruction execution is stored.

(d) Auxiliary carry flag (AC)

If the operation result has a carry from bit 3 or a borrow at bit 3, this flag is set (to 1). It is reset (to 0) in all other cases.

(e) In-service priority flag (ISP)

This flag manages the priority of acknowledgeable maskable vectored interrupt requests.

When ISP= 0, the vector interrupts assigned a low priority with the priority specify flag registers (PR0L, PR0H, and PR1L) (refer to **20.3 (3) Priority specify flag registers (PR0L, PR0H, and PR1L)**) are acknowledge disabled. Actual acknowledgment is controlled with the interrupt enable flag (IE).

(f) Carry flag (CY)

This flag stores overflow and underflow upon add/subtract instruction execution. It stores the shift-out value upon rotate instruction execution and functions as a bit accumulator during bit manipulation instruction execution.

(3) Stack pointer (SP)

This is a 16-bit register to hold the start address of the memory stack area. Only the internal high-speed RAM area (FB00H to FEFFH) can be set as the stack area.

Figure 5-9. Stack Pointer Configuration

	15															0
SP	SP15	SP14	SP13	SP12	SP11	SP10	SP9	SP8	SP7	SP6	SP5	SP4	SP3	SP2	SP1	SP0

The SP is decremented ahead of write (save) to the stack memory and is incremented after read (reset) from the stack memory.

Each stack operation saves/resets data as shown in Figures 5-10 and 5-11.

<R> **Caution Since $\overline{\text{RESET}}$ input makes SP contents indeterminate, be sure to initialize the SP before using the stack.**

Figure 5-10. Data to Be Saved to Stack Memory

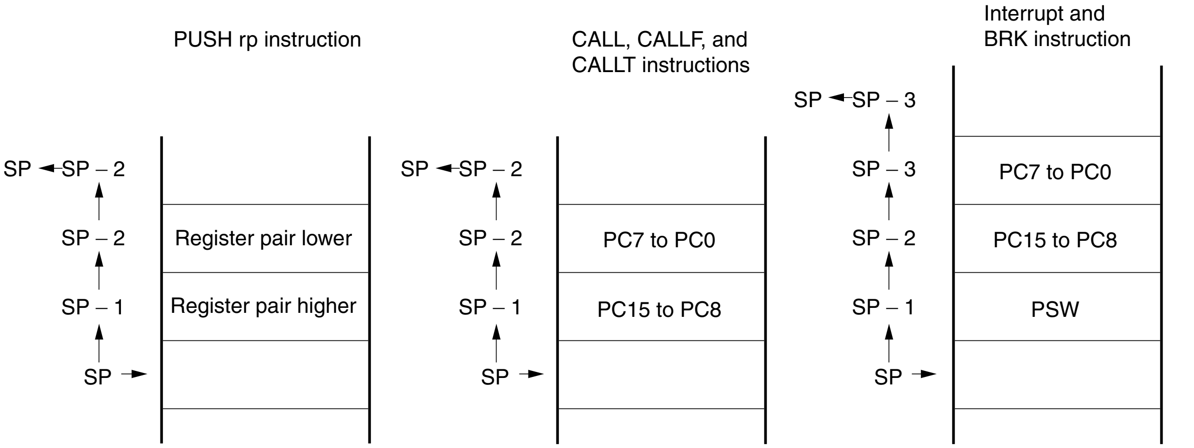
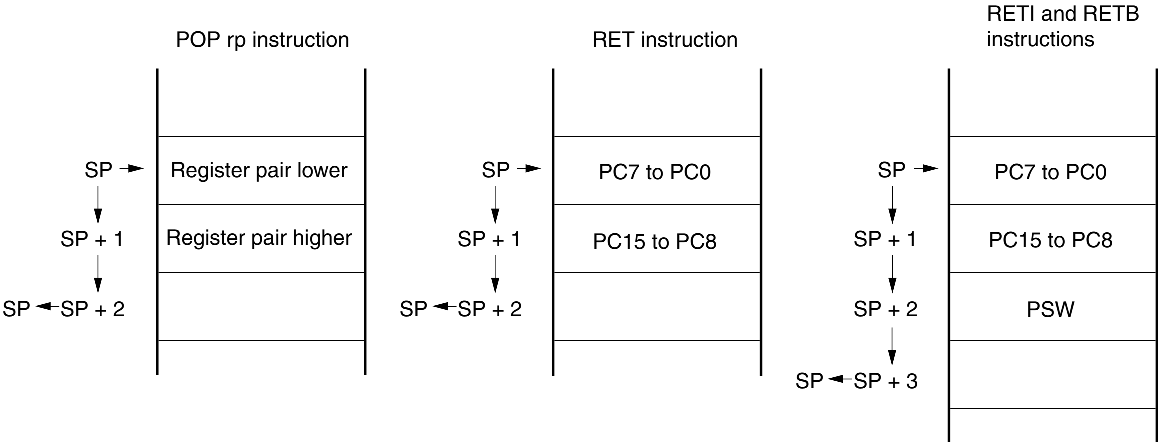


Figure 5-11. Data to Be Reset from Stack Memory



5.2.2 General-purpose registers

A general-purpose register is mapped at particular addresses (FEE0H to FEFFH) of the data memory. It consists of 4 banks, each bank consisting of eight 8-bit registers (X, A, C, B, E, D, L and H).

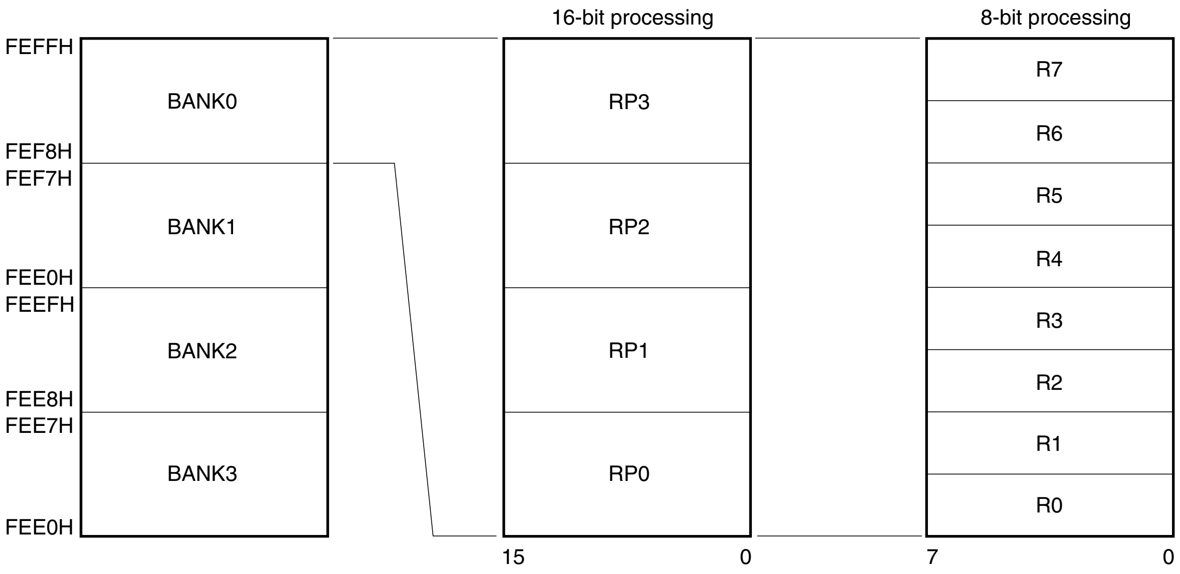
Each register can also be used as an 8-bit register. Two 8-bit registers can be used in pairs as a 16-bit register (AX, BC, DE and HL).

They can be described in terms of function names (X, A, C, B, E, D, L, H, AX, BC, DE and HL) and absolute names (R0 to R7 and RP0 to RP3).

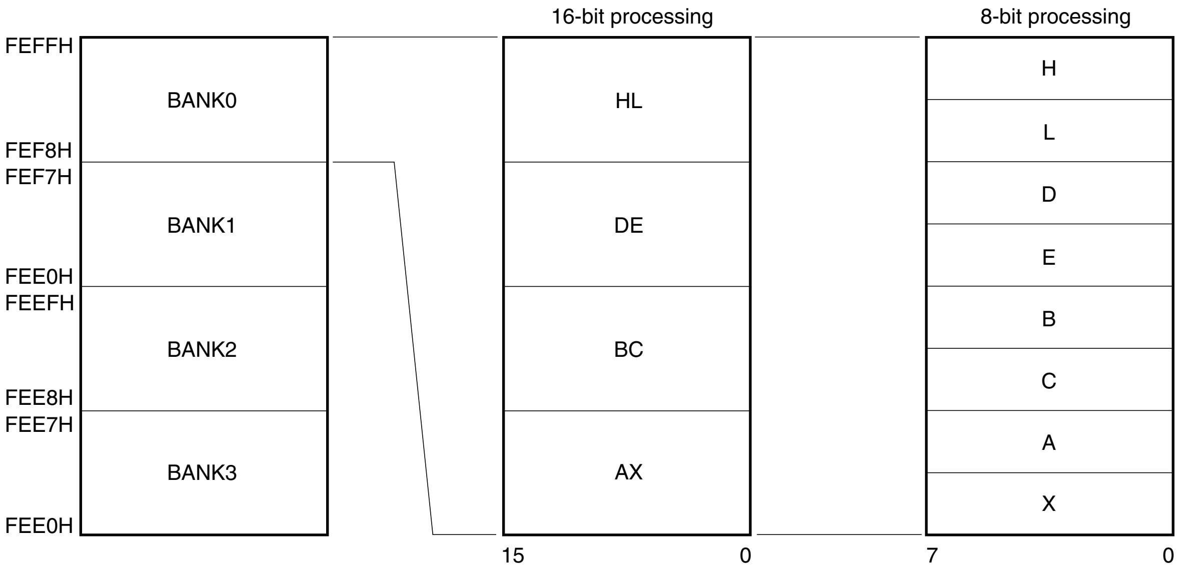
Register banks to be used for instruction execution are set with the CPU control instruction (SEL RBn). Because of the 4-register bank configuration, an efficient program can be created by switching between a register for normal processing and a register for interruption for each bank.

Figure 5-12. General-Purpose Register Configuration

(a) Absolute name



(b) Function name



5.2.3 Special-function register (SFR)

Unlike a general-purpose register, each special-function register has special functions.

It is allocated in the FF00H to FFFFH area.

The special-function register can be manipulated like the general-purpose register, with the operation, transfer and bit manipulation instructions. Manipulatable bit units, 1, 8 and 16, depend on the special-function register type.

Each manipulation bit unit can be specified as follows.

- 1-bit manipulation
  Describe the symbol reserved with assembler for the 1-bit manipulation instruction operand (sfr.bit).
  This manipulation can also be specified with an address.
- 8-bit manipulation
  Describe the symbol reserved with assembler for the 8-bit manipulation instruction operand (sfr).
  This manipulation can also be specified with an address.
- 16-bit manipulation
  Describe the symbol reserved with assembler for the 16-bit manipulation instruction operand (sfrp).
  When addressing an address, describe an even address.

Table 5-3 gives a list of special-function registers. The meaning of items in the table is as follows.

- Symbol
  There are symbols indicating the addresses of the special function registers.
  These symbols are reserved words in the RA78K0, and are defined as an sfr variable using the #pragma sfr directive in the CC78K0. When using the RA78K0, ID78K0, or SD78K0, symbols can be written as an instruction operand.
- R/W
  Indicates whether the corresponding special-function register can be read or written.
  R/W: Read/write enable
  R: Read only
  W: Write only
- Manipulatable bit units
  "√" indicates the manipulatable bit unit (1, 8, or 16). "—" indicates a bit unit for which manipulation is not possible.
- After reset
  Indicates each register status upon $\overline{\text{RESET}}$ input.

Table 5-3. Special-Function Register List (1/3)

Address	Special-Function Register (SFR) Name	Symbol		R/W	Manipulatable Bit Unit			After Reset
					1 Bit	8 Bits	16 Bits	
FF00H	Port 0	P0		R/W	√	√	—	00H
FF01H	Port 1	P1			√	√	—	
FF02H	Port 2	P2			√	√	—	
FF03H	Port 3	P3			√	√	—	
FF07H	Port 7	P7			√	√	—	
FF08H	Port 8	P8			√	√	—	
FF09H	Port 9	P9			√	√	—	
FF0AH	Port 10	P10			√	√	—	
FF0BH	Port 11	P11			√	√	—	
FF10H FF11H	Capture/compare register 00	CR00			—	—	√	Undefined
FF12H FF13H	Capture/compare register 01	CR01			—	—	√	
FF14H FF15H	16-bit timer register	TM0		R	—	—	√	0000H
FF16H	Compare register 10	CR10		R/W	—	√	—	Undefined
FF17H	Compare register 20	CR20			—	√	—	
FF18H	8-bit timer register 1	TMS	TM1	R	—	√	√	0000H
FF19H	8-bit timer register 2		TM2		—	√		
FF1AH	Serial I/O shift register 0	SIO0		R/W	—	√	—	Undefined
FF1FH	A/D conversion result register	ADCR		R	—	√	—	
FF20H	Port mode register 0	PM0		R/W	√	√	—	FFH
FF21H	Port mode register 1	PM1			√	√	—	
FF22H	Port mode register 2	PM2			√	√	—	
FF23H	Port mode register 3	PM3			√	√	—	
FF27H	Port mode register 7	PM7			√	√	—	
FF28H	Port mode register 8	PM8			√	√	—	
FF29H	Port mode register 9	PM9			√	√	—	
FF2AH	Port mode register 10	PM10			√	√	—	
FF2BH	Port mode register 11	PM11			√	√	—	
FF40H	Timer clock select register 0	TCL0			√	√	—	00H
FF41H	Timer clock select register 1	TCL1			—	√	—	
FF42H	Timer clock select register 2	TCL2			—	√	—	
FF43H	Timer clock select register 3	TCL3			—	√	—	88H
FF44H	Timer clock select register 4	TCL4			—	√	—	
FF47H	Sampling clock select register	SCS			—	√	—	00H
FF48H	16-bit timer mode control register	TMC0			√	√	—	

Table 5-3. Special-Function Register List (2/3)

Address	Special-Function Register (SFR) Name	Symbol		R/W	Manipulatable Bit Unit			After Reset
					1 Bit	8 Bits	16 Bits	
FF49H	8-bit timer mode control register	TMC1		R/W	√	√	—	00H
FF4AH	Watch timer mode control register	TMC2			√	√	—	
FF4CH	Capture/compare control register 0	CRC0			√	√	—	04H
FF4EH	16-bit timer output control register	TOC0			√	√	—	00H
FF4FH	8-bit timer output control register	TOC1			√	√	—	
FF60H	Serial operating mode register 0	CSIM0			√	√	—	
FF61H	Serial bus interface control register	SBIC			√	√	—	
FF62H	Slave address register	SVA			—	√	—	Undefined
FF63H	Interrupt timing specify register	SINT			√	√	—	00H
FF6CH	Serial operating mode register 3	CSIM3			√	√	—	
FF6DH	Serial I/O shift register 3	SIO3			—	√	—	Undefined
FF70H	Asynchronous serial interface mode register	ASIM			√	√	—	00H
FF71H	Asynchronous serial interface status register	ASIS		R	—	√	—	
FF72H	Serial operating mode register 2	CSIM2		R/W	√	√	—	
FF73H	Baud rate generator control register	BRGC			—	√	—	
FF74H	Transmit shift register	TXS	SIO2	W	—	√	—	FFH
	Receive buffer register	RXB		R				
FF75H	Serial interface pin select register	SIPS		R/W	√	√	—	00H
FF80H	A/D converter mode register	ADM			√	√	—	01H
FF84H	A/D converter input select register	ADIS			—	√	—	00H
FFB0H	LCD display mode register	LCDM			√	√	—	
FFB2H	LCD display control register	LCDC			√	√	—	
FFB8H	Key return mode register	KRM			√	√	—	02H
FFE0H	Interrupt request flag register 0L	IF0	IF0L		√	√	√	00H
FFE1H	Interrupt request flag register 0H		IF0H		√	√		
FFE2H	Interrupt request flag register 1L	IF1L			√	√	—	
FFE4H	Interrupt mask flag register 0L	MK0	MK0L		√	√	√	FFH
FFE5H	Interrupt mask flag register 0H		MK0H		√	√		
FFE6H	Interrupt mask flag register 1L	MK1L			√	√	—	
FFE8H	Priority order specify flag register 0L	PR0	PR0L		√	√	√	
FFE9H	Priority order specify flag register 0H		PR0H		√	√		
FFEAH	Priority order specify flag register 1L	PR1L			√	√	—	
FFECH	External interrupt mode register 0	INTM0			—	√	—	00H
FFEDH	External interrupt mode register 1	INTM1			—	√	—	
FFF0H	Internal memory size switching register	IMS			—	√	—	Note
FFF2H	Oscillation mode select register	OSMS		W	—	√	—	00H
FFF3H	Pull-up resistor option register H	PUOH		R/W	√	√	—	

Note The value after reset depends on products.
μPD780306, 780306Y: CCH, μPD780308, 780308Y: CFH, μPD78P0308, 78P0308Y: CFH

Table 5-3. Special-Function Register List (3/3)

Address	Special-Function Register (SFR) Name	Symbol	R/W	Manipulatable Bit Unit			After Reset
				1 Bit	8 Bits	16 Bits	
FFF4H	Internal expansion RAM size switching register	IXS	W	—	√	—	0AH
FFF7H	Pull-up resistor option register L	PUOL	R/W	√	√	—	00H
FFF9H	Watchdog timer mode register	WDTM		√	√	—	
FFFAH	Oscillation stabilization time select register	OSTS		—	√	—	04H
FFFBH	Processor clock control register	PCC		√	√	—	

5.3 Instruction Address Addressing

An instruction address is determined by program counter (PC) contents and is normally incremented (+1 for each byte) automatically according to the number of bytes of an instruction to be fetched each time another instruction is executed. When a branch instruction is executed, the branch destination information is set to the PC and branched by the following addressing (for details of instructions, refer to **78K/0 Series Instructions User's Manual (U12326E)**).

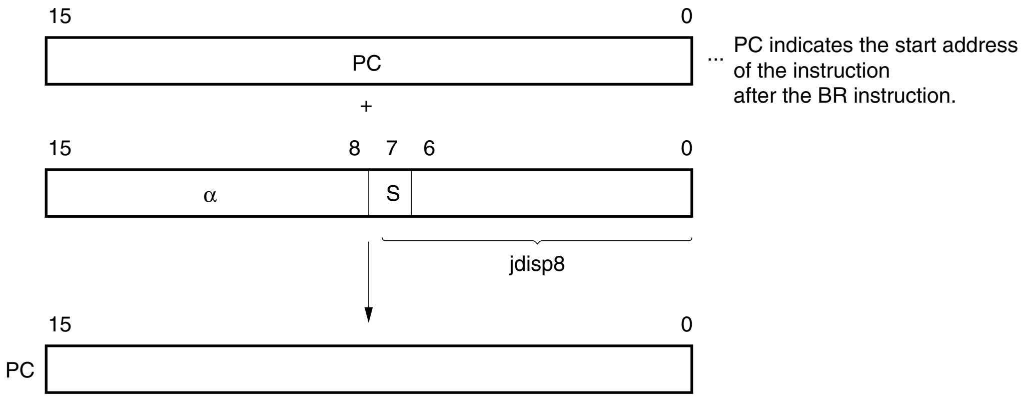
5.3.1 Relative addressing

[Function]

The value obtained by adding 8-bit immediate data (displacement value: jdisp8) of an instruction code to the start address of the following instruction is transferred to the program counter (PC) and branched. The displacement value is treated as signed two's complement data (–128 to +127) and bit 7 becomes a sign bit. In other words, relative addressing consists in relative branching from the start address of the following instruction to the –128 to +127 range.

This function is carried out when the BR $addr16 instruction or a conditional branch instruction is executed.

[Illustration]


15
0
PC
... PC indicates the start address of the instruction after the BR instruction.
+
15
8
7
6
0
α
S
jdisp8
15
0
PC

When S = 0, all bits of α are 0.
When S = 1, all bits of α are 1.

5.3.2 Immediate addressing

[Function]

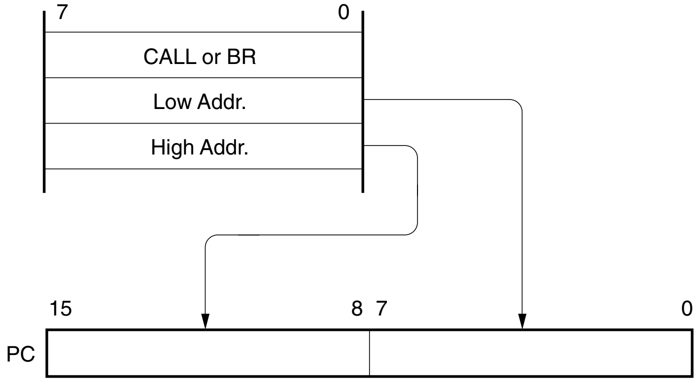
Immediate data in the instruction word is transferred to the program counter (PC) and branched.

This function is carried out when the CALL !addr16 or BR !addr16 or CALLF !addr11 instruction is executed.

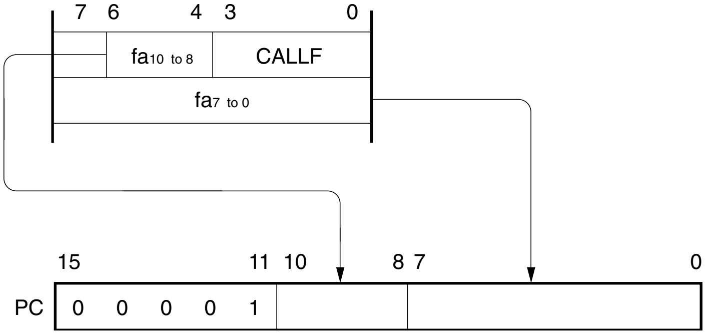
CALL !addr16 and BR !addr16 instructions can be branched to the entire memory space. The CALLF !addr11 instruction is branched to the 0800H to 0FFFH area.

[Illustration]

In the case of CALL !addr16 and BR !addr16 instructions



In the case of CALLF !addr11 instruction



5.3.3 Table indirect addressing

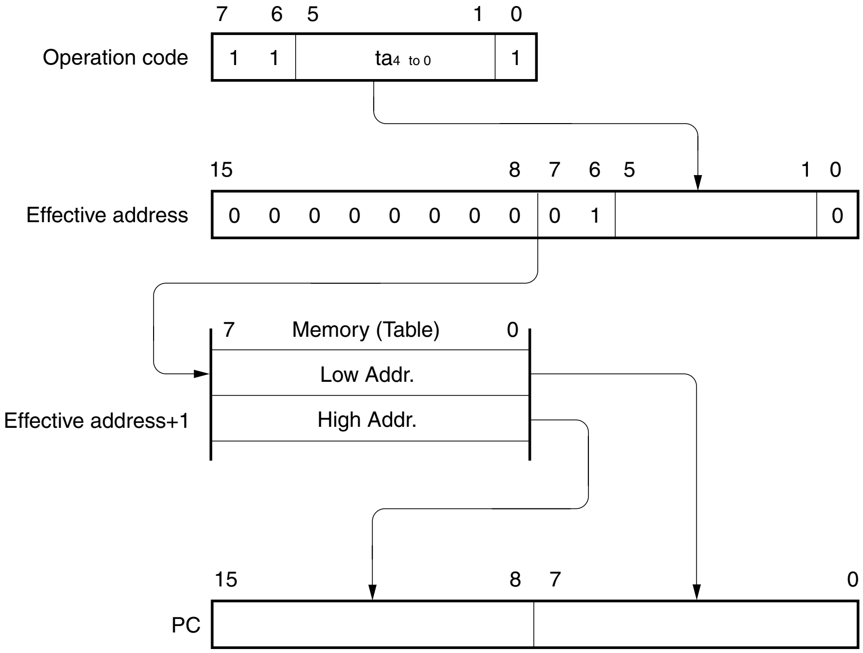
[Function]

Table contents (branch destination address) of the particular location to be addressed by bits 1 to 5 of the immediate data of an operation code are transferred to the program counter (PC) and branched.

Table indirect addressing is carried out when the CALLT [addr5] instruction is executed.

This instruction references the address stored in the memory table from 40H to 7FH, and allows branching to the entire memory space.

[Illustration]



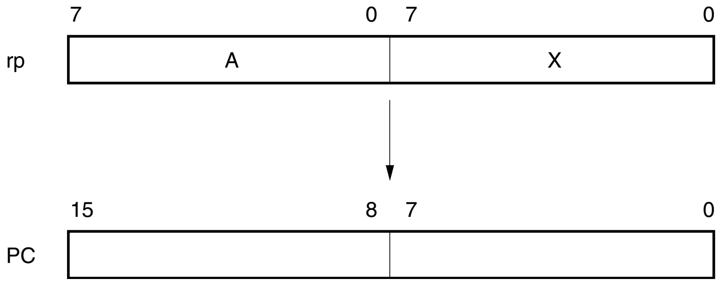
5.3.4 Register addressing

[Function]

Register pair (AX) contents to be specified with an instruction word are transferred to the program counter (PC) and branched.

This function is carried out when the BR AX instruction is executed.

[Illustration]



5.4 Operand Address Addressing

The following various methods are available to specify the register and memory (addressing) which undergo manipulation during instruction execution.

5.4.1 Implied addressing

[Function]

The register which functions as an accumulator (A and AX) in the general-purpose register is automatically (implicity) addressed.

Of the μPD780308 and 780308Y Subseries instruction words, the following instructions employ implied addressing.

Instruction	Register to Be Specified by Implied Addressing
MULU	A register for multiplicand and AX register for product storage
DIVUW	AX register for dividend and quotient storage
ADJBA/ADJBS	A register for storage of numeric values which become decimal correction targets
ROR4/ROL4	A register for storage of digit data which undergoes digit rotation

[Operand format]

Because implied addressing can be automatically employed with an instruction, no particular operand format is necessary.

[Description example]

In the case of MULU X

With an 8-bit × 8-bit multiply instruction, the product of A register and X register is stored in AX. In this example, the A and AX registers are specified by implied addressing.

5.4.2 Register addressing

[Function]

The general-purpose register to be specified is accessed as an operand with the register bank select flag (RBS0 and RBS1) and with the register specify code (Rn and RPn) in an operation code.

Register addressing is carried out when an instruction with the following operand format is executed. When an 8-bit register is specified, one of the eight registers is specified with 3 bits in the operation code.

[Operand format]

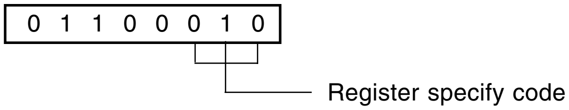
Identifier	Description
r	X, A, C, B, E, D, L, H
rp	AX, BC, DE, HL

'r' and 'rp' can be described with function names (X, A, C, B, E, D, L, H, AX, BC, DE and HL) as well as absolute names (R0 to R7 and RP0 to RP3).

[Description example]

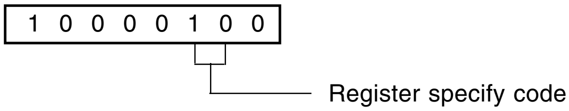
MOV A, C; when selecting C register as r

Operation code



INCW DE; when selecting DE register pair as rp

Operation code



5.4.3 Direct addressing

[Function]

This addressing is to directly address the memory indicated by the immediate data in the instruction word.

[Operand format]

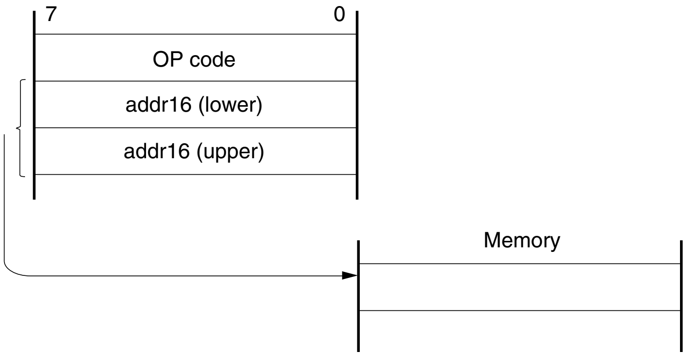
Identifier	Description
addr16	Label or 16-bit immediate data

[Description example]

MOV A, !FE00H; when setting !addr16 to FE00H

Operation code	1 0 0 0 1 1 1 0	OP code
	0 0 0 0 0 0 0 0	00H
	1 1 1 1 1 1 1 0	FFH

[Illustration]



5.4.4 Short direct addressing

[Function]

The memory to be manipulated in the fixed space is directly addressed with 8-bit data in an instruction word. This addressing is applied to a fixed 256-byte space of FE20H to FF1FH. An internal high-speed RAM and a special-function register (SFR) are mapped at FE20H to FEFFH and FF00H to FF1FH, respectively.

The SFR area (FF00H to FF1FH) to which short direct addressing is applied, is a part of the entire SFR area. Ports which are frequency accessed in a program, and compare registers and capture registers of timers/event counters are mapped to this area and these SFRs can be manipulated with a small number of bytes and clocks.

When 8-bit immediate data is at 20H to FFH, bit 8 of an effective address is set to 0. When it is at 00H to 1FH, bit 8 is set to 1. Refer to the **[Illustration]** on the next page.

[Operand format]

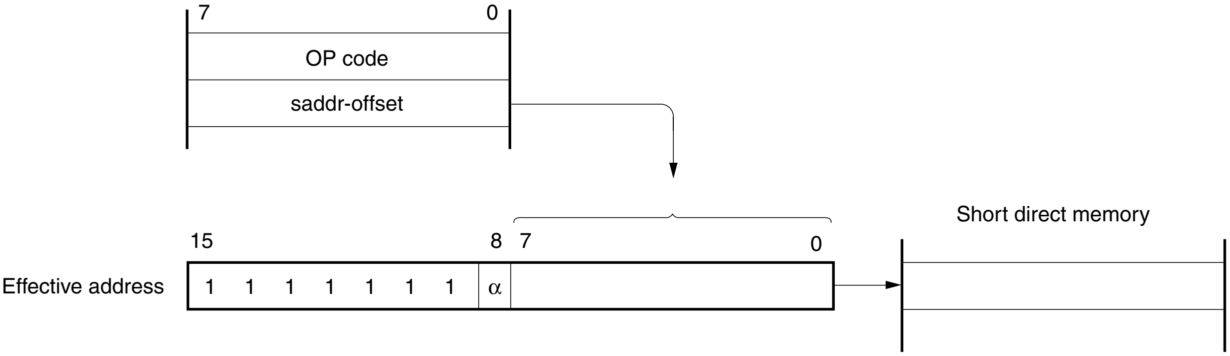
Identifier	Description
saddr	Label or FE20H to FF1FH immediate data
saddrp	Label or FE20H to FF1FH immediate data (even address only)

[Description example]

MOV0 FE30H, #50H; when setting saddr to FE30H and immediate data to 50H

Operation code	0 0 0 1 0 0 0 1	OP code
	0 0 1 1 0 0 0 0	30H (saddr-offset)
	0 1 0 1 0 0 0 0	50H (immediate data)

[Illustration]



When 8-bit immediate data is 20H to FFH, α = 0

When 8-bit immediate data is 00H to 1FH, α = 1

5.4.5 Special-function register (SFR) addressing

[Function]

The memory-mapped special-function register (SFR) is addressed with 8-bit immediate data in an instruction word.

This addressing is applied to the 240-byte spaces FF00H to FFCFH and FFE0H to FFFFH. However, the SFR mapped at FF00H to FF1FH can be accessed with short direct addressing.

[Operand format]

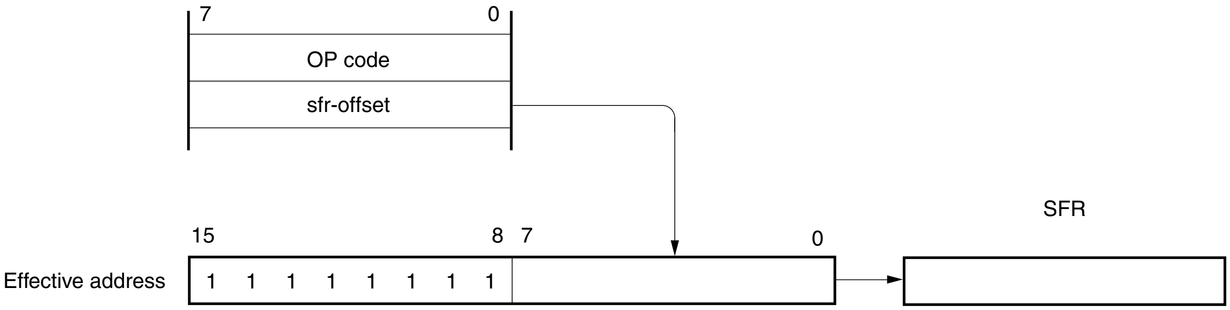
Identifier	Description
sfr	Special-function register name
sfrp	16-bit manipulatable special-function register name (even address only)

[Description example]

MOV PM0, A; when selecting PM0 (FF20H) as sfr

Operation code	1 1 1 1 0 1 1 0	OP code
	0 0 1 0 0 0 0 0	20H (sfr-offset)

[Illustration]



5.4.6 Register indirect addressing

[Function]

This addressing is to address a memory area to be manipulated by using as an operand address the contents of a register pair specified by the register bank select flags (RBS0 and RBS1) and the register pair specification code in the operation code.

[Operand format]

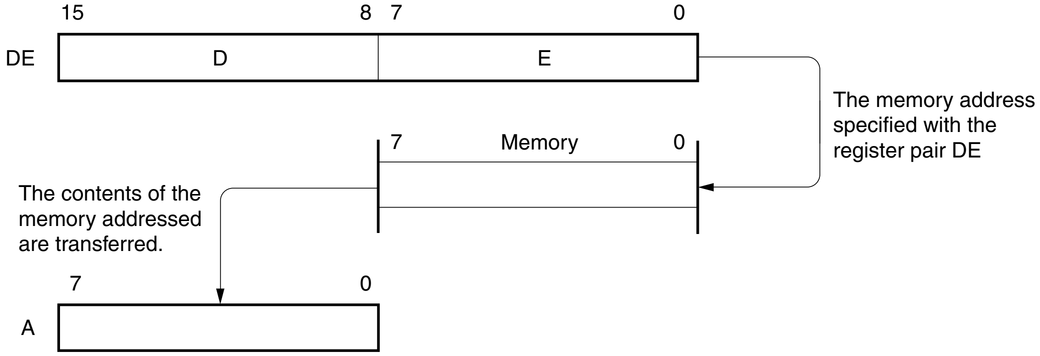
Identifier	Description
—	[DE], [HL]

[Description example]

MOV A, [DE]; when selecting [DE] as register pair

Operation code 1 0 0 0 0 1 0 1

[Illustration]


15
8
7
0
DE
D
E
The memory address specified with the register pair DE
7
Memory
0
The contents of the memory addressed are transferred.
7
0
A

5.4.7 Based addressing

[Function]

This addressing is to address the memory by using the result of adding 8-bit immediate data to the contents of the HL register pair that is used as a base register. The HL register pair to be accessed is in the register bank specified with the register bank select flags (RBS0 and RBS1). Addition is performed by expanding the offset data as a positive number to 16 bits. A carry from the 16th bit is ignored. This addressing can be carried out for all the memory spaces.

[Operand format]

Identifier	Description
—	[HL + byte]

[Description example]

MOV A, [HL + 10H]; when setting byte to 10H

```
Operation code   1 0 1 0 1 1 1 0

                 0 0 0 1 0 0 0 0
```

5.4.8 Based indexed addressing

[Function]

This addressing is to address the memory by using the result of adding the contents of the B or C register specified in the instruction word to the contents of the HL register that is used as a base register. The H, B, and C registers accessed are in the register bank specified by the register bank select flags (RBS0 and RBS1). The addition is executed with the contents of the B or C register extended to 16 bits as a positive number. A carry from the 16th bit is ignored. This addressing can be carried out for all the memory spaces.

[Operand format]

Identifier	Description
—	[HL + B], [HL + C]

[Description example]

In the case of MOV A, [HL + B]

Operation code 1 0 1 0 1 0 1 1

5.4.9 Stack addressing

[Function]

The stack area is indirectly addressed with the stack pointer (SP) contents.

This addressing method is automatically employed when the PUSH, POP, subroutine call and return instructions are executed or the register is saved/reset upon generation of an interrupt request.

Stack addressing enables to address the internal high-speed RAM area only.

[Description example]

In the case of PUSH DE

Operation code 1 0 1 1 0 1 0 1

CHAPTER 6 PORT FUNCTIONS

6.1 Port Functions

The μPD780308 and 780308Y Subseries units incorporate two input ports and 55 I/O ports. Figure 6-1 shows the port configuration. Every port is capable of 1-bit and 8-bit manipulations and can carry out considerably varied control operations. Besides port functions, the ports can also serve as on-chip hardware I/O pins.

Figure 6-1. Port Types

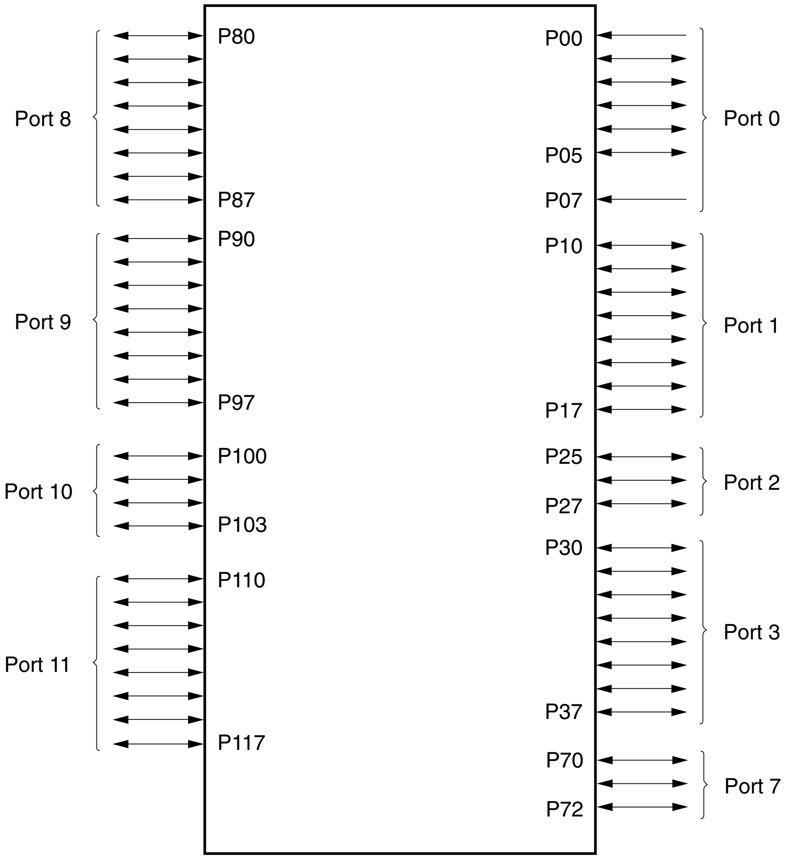


Table 6-1. Port Functions (μPD780308 Subseries)

Pin Name	Function		Alternate Function
P00	Port 0. 7-bit I/O port.	Input only	INTP0/TI00
P01		Input/output mode can be specified in 1-bit units. If used as an input port, an internal pull-up resistor can be used by software.	INTP1/TI01
P02			INTP2
P03			INTP3
P04			INTP4
P05			INTP5
P07		Input only	XT1
P10 to P17	Port 1. 8-bit I/O port. Input/output mode can be specified in 1-bit units. If used as an input port, an internal pull-up resistor can be used by software.		ANI0 to ANI7
P25	Port 2. 3-bit I/O port. Input/output mode can be specified in 1-bit units. If used as an input port, an internal pull-up resistor can be used by software.		SI0/SB0
P26			SO0/SB1
P27			$\overline{\text{SCK0}}$
P30	Port 3. 8-bit I/O port. Input/output mode can be specified in 1-bit units. If used as an input port, an internal pull-up resistor can be used by software.		TO0
P31			TO1
P32			TO2
P33			TI1
P34			TI2
P35			PCL
P36			BUZ
P37			—
P70	Port 7. 3-bit I/O port. Input/output mode can be specified in 1-bit units. If used as an input port, an internal pull-up resistor can be used by software.		SI2/RxD
P71			SO2/TxD
P72			$\overline{\text{SCK2}}$/ASCK
P80 to P87	Port 8. 8-bit I/O port. Input/output mode can be specified in 1-bit units. If used as an input port, an internal pull-up resistor can be used by software. This port can be used as a segment signal output port or an I/O port in 2-bit units by setting LCD display control register (LCDC).		S39 to S32
P90 to P97	Port 9. 8-bit I/O port. Input/output mode can be specified in 1-bit units. If used as an input port, an internal pull-up resistor can be used by software. This port can be used as a segment signal output port or an I/O port in 2-bit units by setting LCD display control register (LCDC).		S31 to S24
P100 to P103	Port 10. 4-bit I/O port. Input/output mode can be specified in 1-bit units. If used as an input port, an internal pull-up resistor can be used by software. This port can directly drive LEDs.		—
P110	Port 11. 8-bit I/O port. Input/output mode can be specified in 1-bit units. If used as an input port, an internal pull-up resistor can be used by software. Falling edge detection is possible.		SI3
P111			SO3
P112			$\overline{\text{SCK3}}$
P113			TxD
P114			RxD
P115 to P117			—

Table 6-2. Port Functions (μPD780308Y Subseries)

Pin Name	Function		Alternate Function
P00	Port 0. 7-bit I/O port.	Input only	INTP0/TI00
P01		Input/output mode can be specified in 1-bit units. If used as an input port, an internal pull-up resistor can be used by software.	INTP1/TI01
P02			INTP2
P03			INTP3
P04			INTP4
P05			INTP5
P07		Input only	XT1
P10 to P17	Port 1. 8-bit I/O port. Input/output mode can be specified in 1-bit units. If used as an input port, an internal pull-up resistor can be used by software.		ANI0 to ANI7
P25	Port 2. 3-bit I/O port. Input/output mode can be specified in 1-bit units. If used as an input port, an internal pull-up resistor can be used by software.		SI0/SB0/SDA0
P26			SO0/SB1/SDA1
P27			$\overline{\text{SCK0}}$/SCL
P30	Port 3. 8-bit I/O port. Input/output mode can be specified in 1-bit units. If used as an input port, an internal pull-up resistor can be used by software.		TO0
P31			TO1
P32			TO2
P33			TI1
P34			TI2
P35			PCL
P36			BUZ
P37			—
P70	Port 7. 3-bit I/O port. Input/output mode can be specified in 1-bit units. If used as an input port, an internal pull-up resistor can be used by software.		SI2/RxD
P71			SO2/TxD
P72			$\overline{\text{SCK2}}$/ASCK
P80 to P87	Port 8. 8-bit I/O port. Input/output mode can be specified in 1-bit units. If used as an input port, an internal pull-up resistor can be used by software. This port can be used as a segment signal output port or an I/O port in 2-bit units by setting LCD display control register (LCDC).		S39 to S32
P90 to P97	Port 9. 8-bit I/O port. Input/output mode can be specified in 1-bit units. If used as an input port, an internal pull-up resistor can be used by software. This port can be used as a segment signal output port or an I/O port in 2-bit units by setting LCD display control register (LCDC).		S31 to S24
P100 to P103	Port 10. 4-bit I/O port. Input/output mode can be specified in 1-bit units. If used as an input port, an internal pull-up resistor can be used by software. This port can directly drive LEDs.		—
P110	Port 11. 8-bit I/O port. Input/output mode can be specified in 1-bit units. If used as an input port, an internal pull-up resistor can be used by software. Falling edge detection is possible.		SI3
P111			SO3
P112			$\overline{\text{SCK3}}$
P113			TxD
P114			RxD
P115 to P117			—

6.2 Port Configuration

A port consists of the following hardware.

Table 6-3. Port Configuration

Item	Configuration
Control register	Port mode register (PMm: m = 0 to 3, 7 to 11) Pull-up resistor option register (PUOH, PUOL) Key return mode register (KRM)
Port	Total: 57 ports (2 inputs, 55 inputs/outputs)
Pull-up resistor	Total: 55 (software specifiable: 55)

6.2.1 Port 0

Port 0 is a 7-bit I/O port with output latch. P01 to P05 pins can specify the input mode/output mode in 1-bit units with port mode register 0. P00 and P07 pins are input-only ports. When P01 to P05 pins are used as input ports, an internal pull-up resistor can be used to them in 5-bit units with pull-up resistor option register L.

Alternate functions include external interrupt request input, external count clock input to the timer and crystal connection for subsystem clock oscillation.

$\overline{\text{RESET}}$ input sets port 0 to input mode.

Figures 6-2 and 6-3 show block diagrams of port 0.

Caution Because port 0 also serves for external interrupt request input, when the port function output mode is specified and the output level is changed, the interrupt request flag is set. Thus, when the output mode is used, set the interrupt mask flag to 1.

Figure 6-2. P00 and P07 Block Diagram

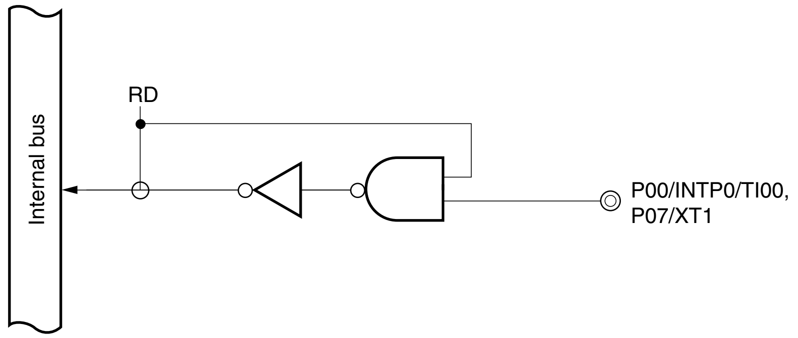
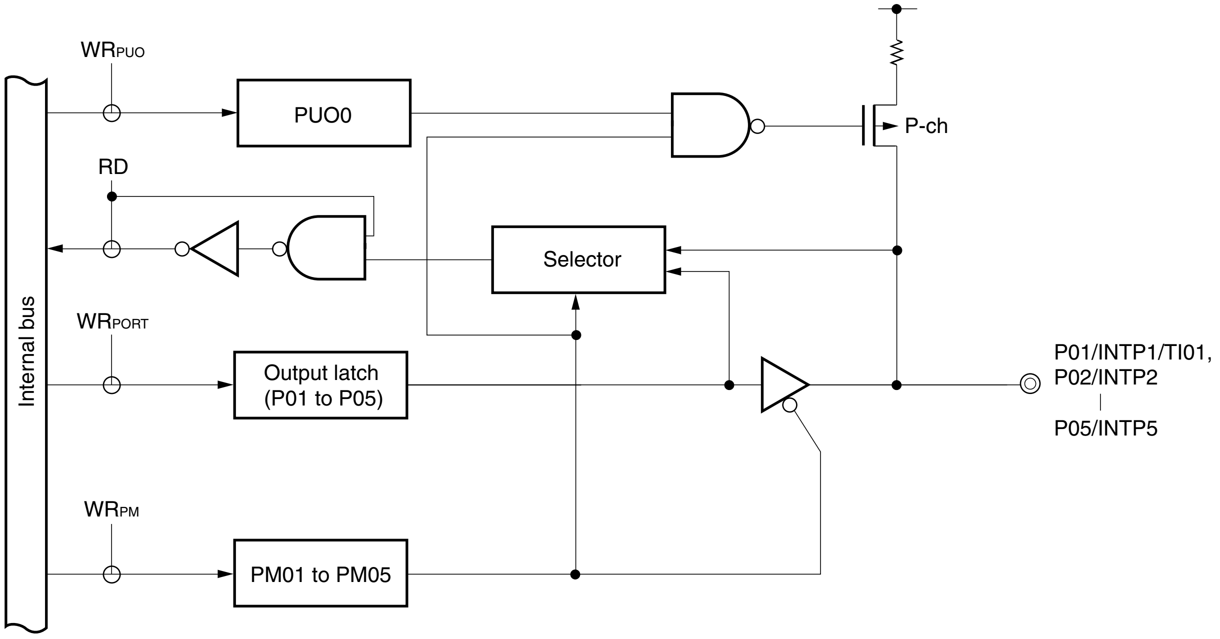


Figure 6-3. P01 to P05 Block Diagram



PUO: Pull-up resistor option register

PM: Port mode register

RD: Port 0 read signal

WR: Port 0 write signal

6.2.2 Port 1

Port 1 is an 8-bit I/O port with output latch. P10 to P17 pins can specify the input mode/output mode in 1-bit units with port mode register 1. When P10 to P17 pins are used as input ports, an internal pull-up resistor can be used to them in 8-bit units with pull-up resistor option register L.

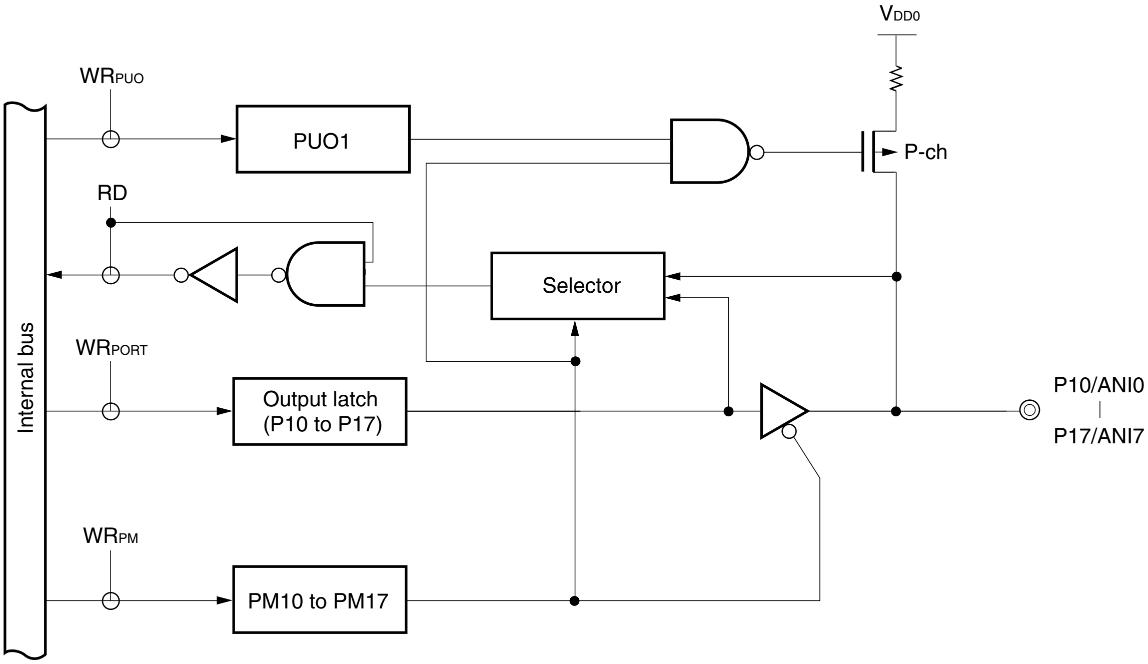
Alternate functions include an A/D converter analog input.

$\overline{\text{RESET}}$ input sets port 1 to input mode.

Figure 6-4 shows a block diagram of port 1.

Caution A pull-up resistor cannot be used for pins used as A/D converter analog input.

Figure 6-4. P10 to P17 Block Diagram



PUO: Pull-up resistor option register
PM: Port mode register
RD: Port 1 read signal
WR: Port 1 write signal

6.2.3 Port 2 (μPD780308 Subseries)

Port 2 is a 3-bit I/O port with output latch. P25 to P27 pins can specify the input mode/output mode in 1-bit units with port mode register 2. When P25 to P27 pins are used as input ports, an internal pull-up resistor can be used to them in 3-bit units with pull-up resistor option register L.

Alternate functions include serial interface data I/O and clock I/O.

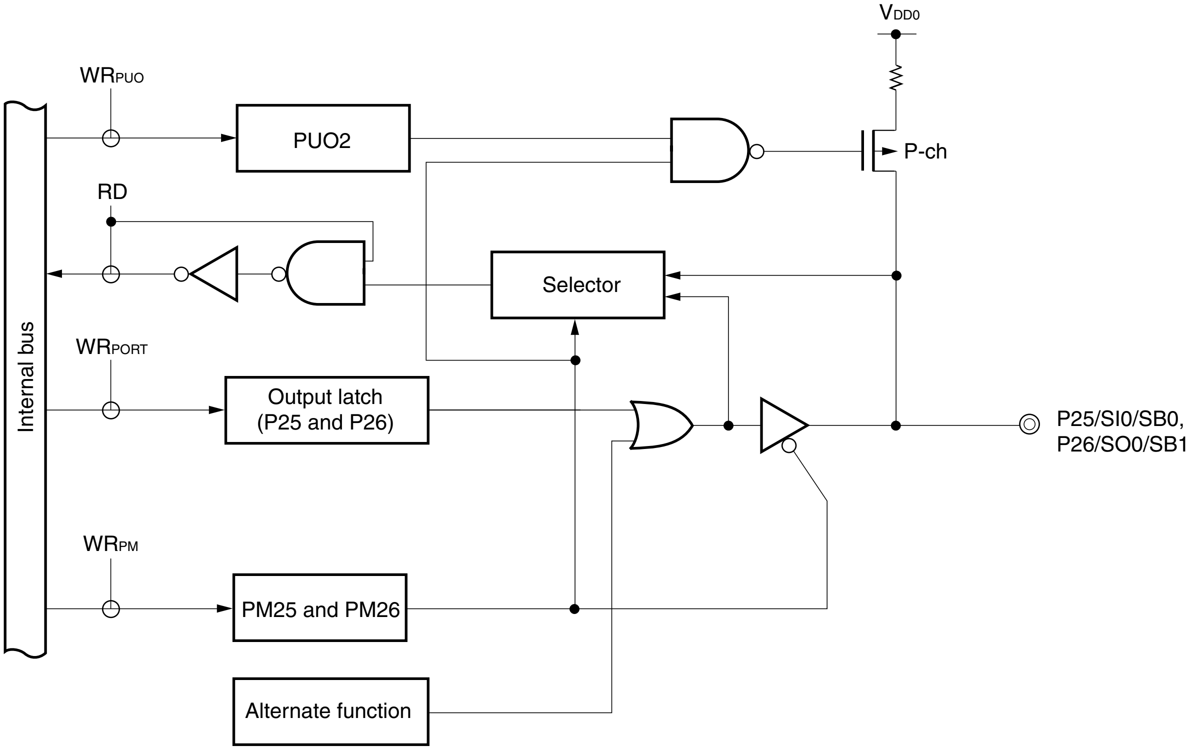
$\overline{\text{RESET}}$ input sets port 2 to input mode.

Figures 6-5 and 6-6 show the block diagrams of port 2.

Cautions 1. When used as a serial interface, set the I/O and output latch according to its functions. For the setting method, refer to Figure 15-4 Serial Operating Mode Register 0 Format.

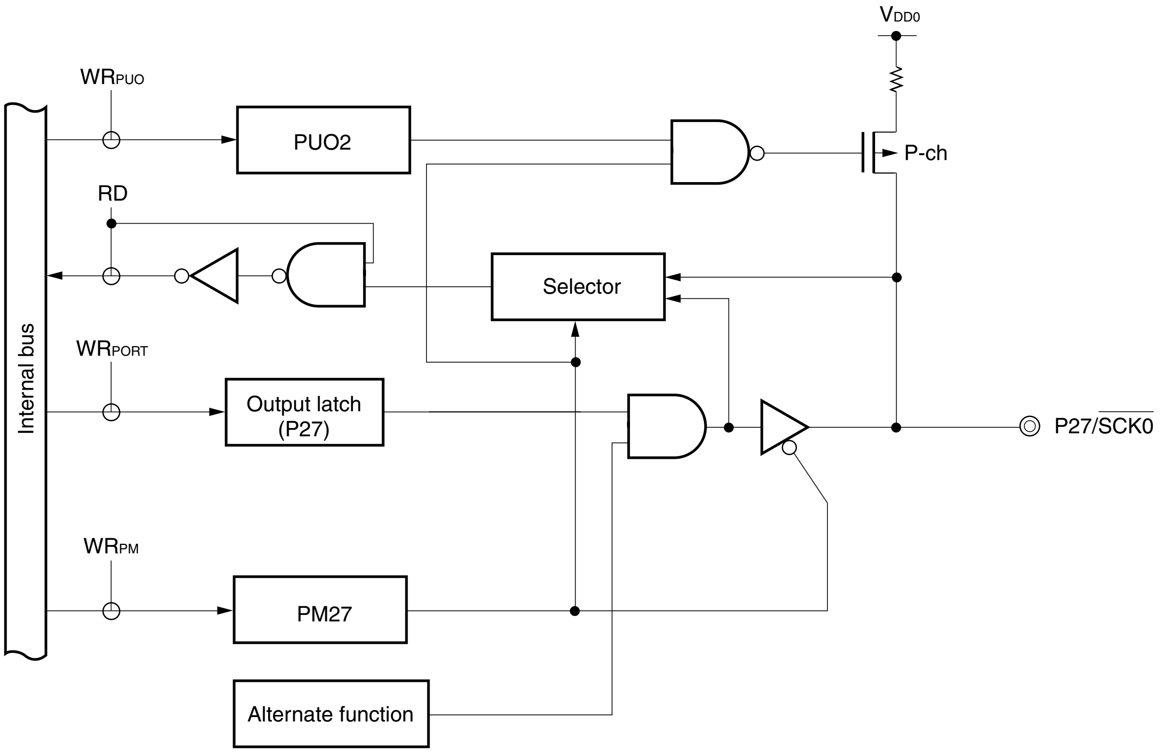
2. When reading the pin state in SBI mode, set PM2n to 1 (n = 5, 6) (refer to the description of (10) Identifying busy status of slave in 15.4.3 SBI mode operation).

Figure 6-5. P25, P26 Block Diagram (μPD780308 Subseries)



PUO: Pull-up resistor option register
PM: Port mode register
RD: Port 2 read signal
WR: Port 2 write signal

Figure 6-6. P27 Block Diagram (μPD780308 Subseries)



PUO: Pull-up resistor option register
PM: Port mode register
RD: Port 2 read signal
WR: Port 2 write signal

6.2.4 Port 2 (μPD780308Y Subseries)

Port 2 is a 3-bit I/O port with output latch. P25 to P27 pins can specify the input mode/output mode in 1-bit units with port mode register 2. When P25 to P27 pins are used as input ports, an internal pull-up resistor can be used to them in 3-bit units with pull-up resistor option register L.

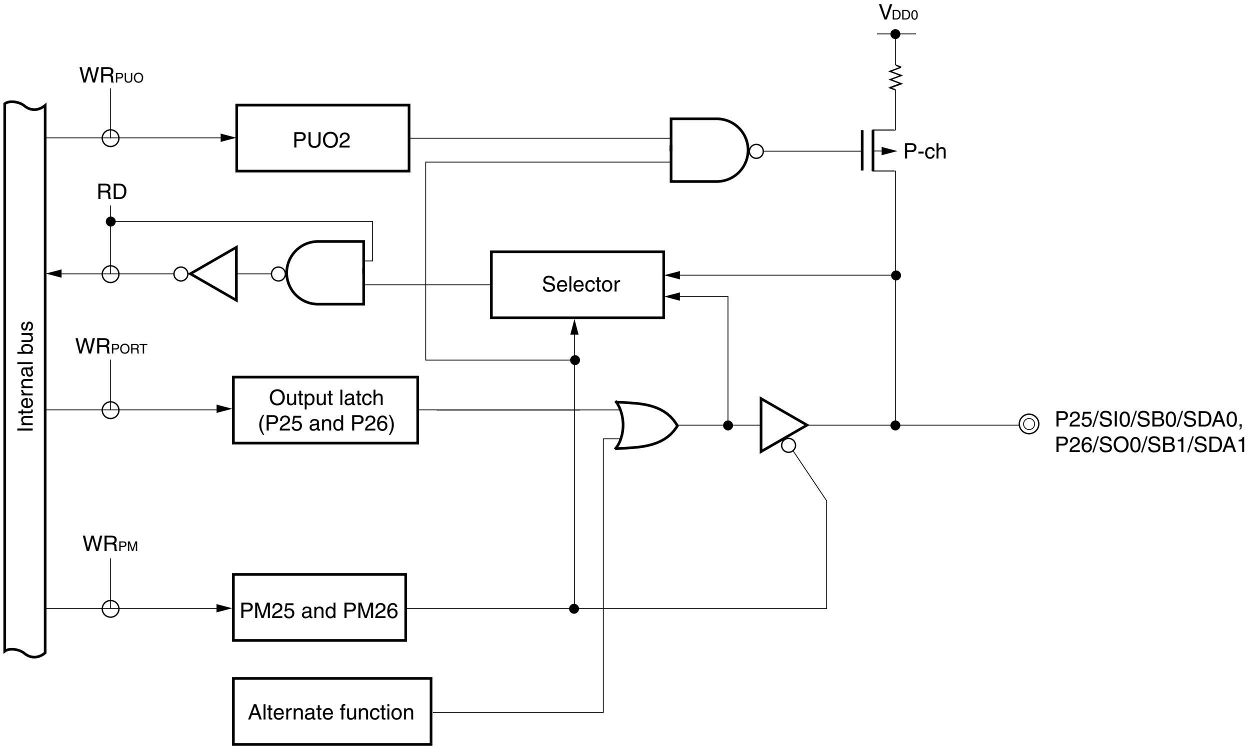
Alternate functions include serial interface data I/O and clock I/O.

$\overline{\text{RESET}}$ input sets port 2 to input mode.

Figures 6-7 and 6-8 show the block diagrams of port 2.

Caution When used as a serial interface, set the I/O and output latch according to its functions. For the setting method, refer to Figure 16-4 Serial Operating Mode Register 0 Format.

Figure 6-7. P25, P26 Block Diagram (μPD780308Y Subseries)



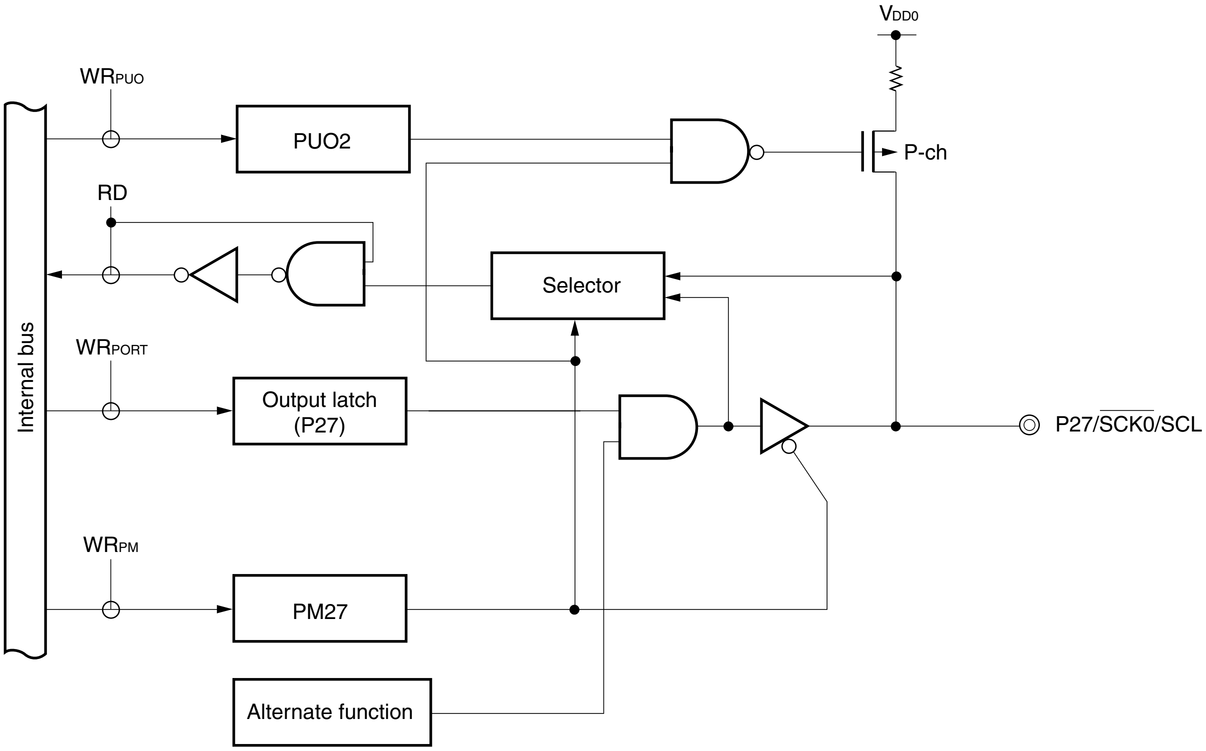
PUO: Pull-up resistor option register

PM: Port mode register

RD: Port 2 read signal

WR: Port 2 write signal

Figure 6-8. P27 Block Diagram (μPD780308Y Subseries)



PUO: Pull-up resistor option register
PM: Port mode register
RD: Port 2 read signal
WR: Port 2 write signal

6.2.5 Port 3

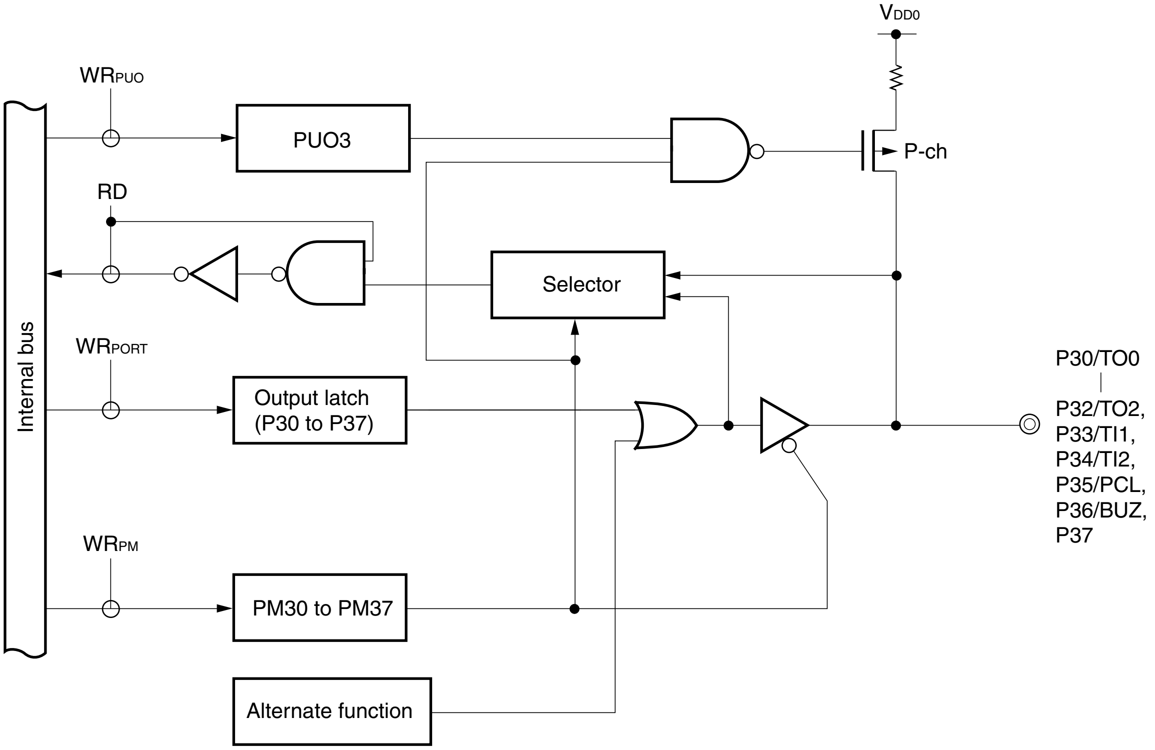
Port 3 is an 8-bit I/O port with output latch. P30 to P37 pins can specify the input mode/output mode in 1-bit units with port mode register 3. When P30 to P37 pins are used as input ports, an internal pull-up resistor can be used to them in 8-bit units with pull-up resistor option register L.

Alternate functions include timer I/O, clock output and buzzer output.

$\overline{\text{RESET}}$ input sets port 3 to input mode.

Figure 6-9 shows a block diagram of port 3.

Figure 6-9. P30 to P37 Block Diagram



PUO: Pull-up resistor option register
PM: Port mode register
RD: Port 3 read signal
WR: Port 3 write signal

6.2.6 Port 7

Port 7 is a 3-bit I/O port with output latch. P70 to P72 pins can specify the input mode/output mode in 1-bit units with port mode register 7. When P70 to P72 pins are used as input ports, an internal pull-up resistor can be used to them in 3-bit units with pull-up resistor option register L.

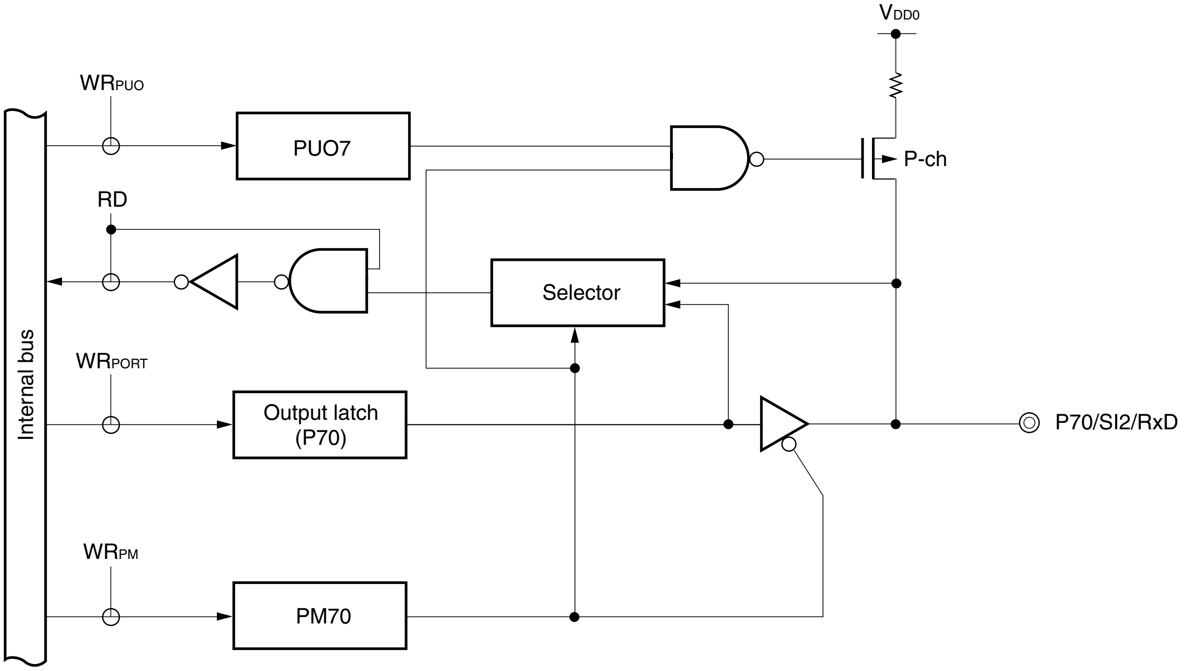
Alternate functions include serial interface channel 2 data I/O and clock I/O.

$\overline{\text{RESET}}$ input sets port 7 to input mode.

Figures 6-10 and 6-11 show the block diagrams of port 7.

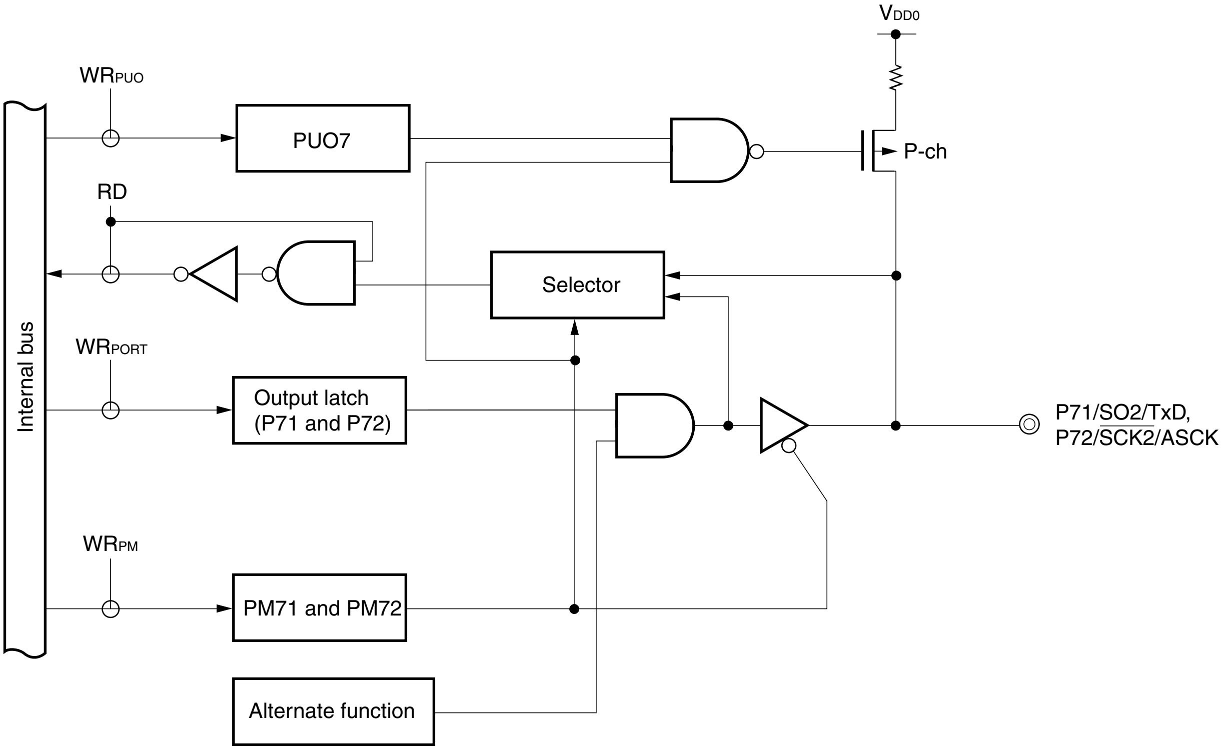
Caution When used as a serial interface, set the I/O and output latch according to its functions. For the setting method, refer to Table 17-2 Serial Interface Channel 2 Operating Mode Settings.

Figure 6-10. P70 Block Diagram



PUO: Pull-up resistor option register
PM: Port mode register
RD: Port 7 read signal
WR: Port 7 write signal

Figure 6-11. P71 and P72 Block Diagram



PUO: Pull-up resistor option register
PM: Port mode register
RD: Port 7 read signal
WR: Port 7 write signal

6.2.7 Port 8

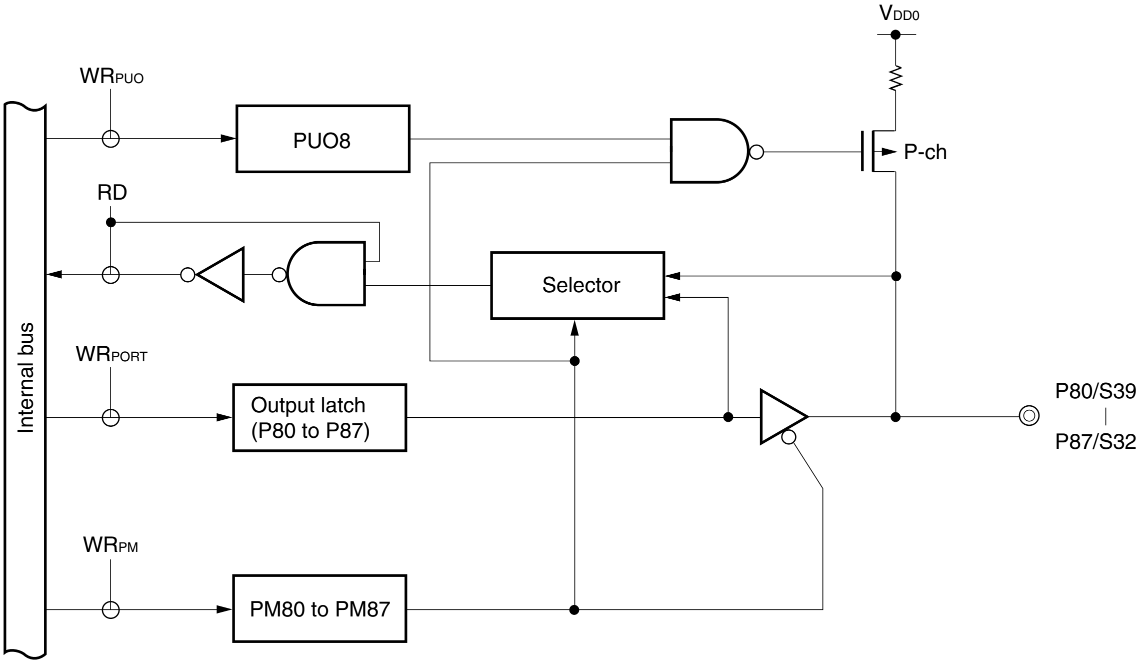
Port 8 is an 8-bit I/O port with output latch. P80 to P87 pins can specify the input mode/output mode in 1-bit units with port mode register 8. When P80 to P87 pins are used as input ports, an internal pull-up resistor can be used to them in 8-bit units with pull-up resistor option register H.

Alternate functions include LCD controller/driver segment signal output.

$\overline{\text{RESET}}$ input sets port 8 to input mode.

Figure 6-12 shows a block diagram of port 8.

Figure 6-12. P80 to P87 Block Diagram



PUO: Pull-up resistor option register

PM: Port mode register

RD: Port 8 read signal

WR: Port 8 write signal

6.2.8 Port 9

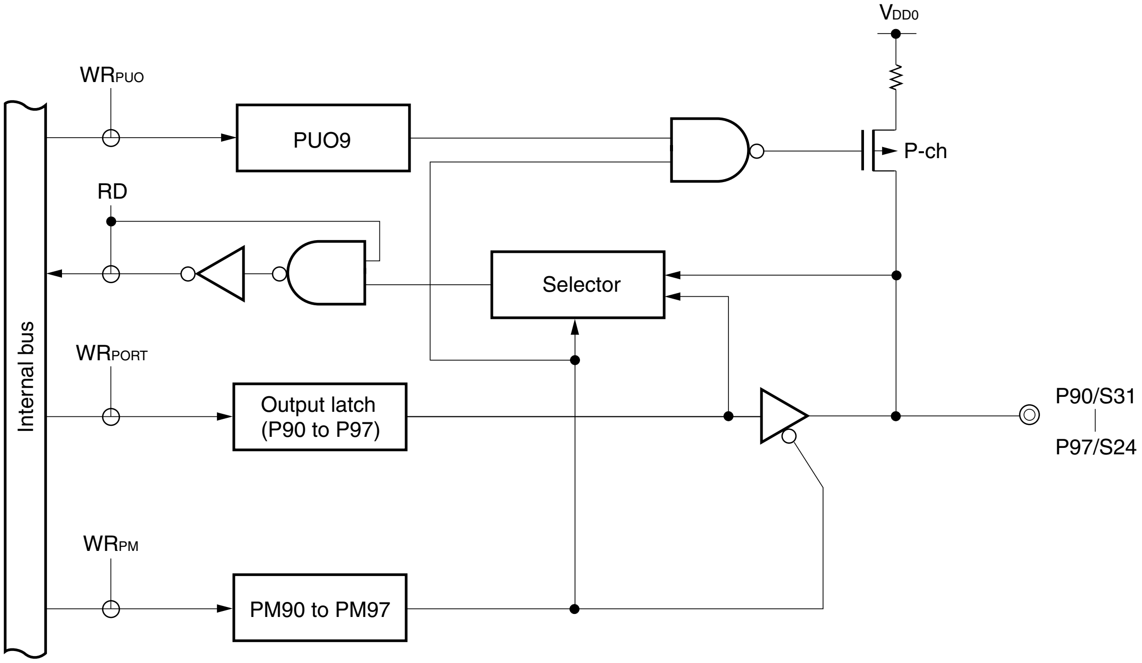
Port 9 is an 8-bit I/O port with output latch. P90 to P97 pins can specify the input mode/output mode in 1-bit units with port mode register 9. When P90 to P97 pins are used as input ports, an internal pull-up resistor can be used to them in 8-bit units with pull-up resistor option register H.

Alternate functions include LCD controller/driver segment signal output.

$\overline{\text{RESET}}$ input sets port 9 to input mode.

Figure 6-13 shows a block diagram of port 9.

Figure 6-13. P90 to P97 Block Diagram



PUO: Pull-up resistor option register
PM: Port mode register
RD: Port 9 read signal
WR: Port 9 write signal

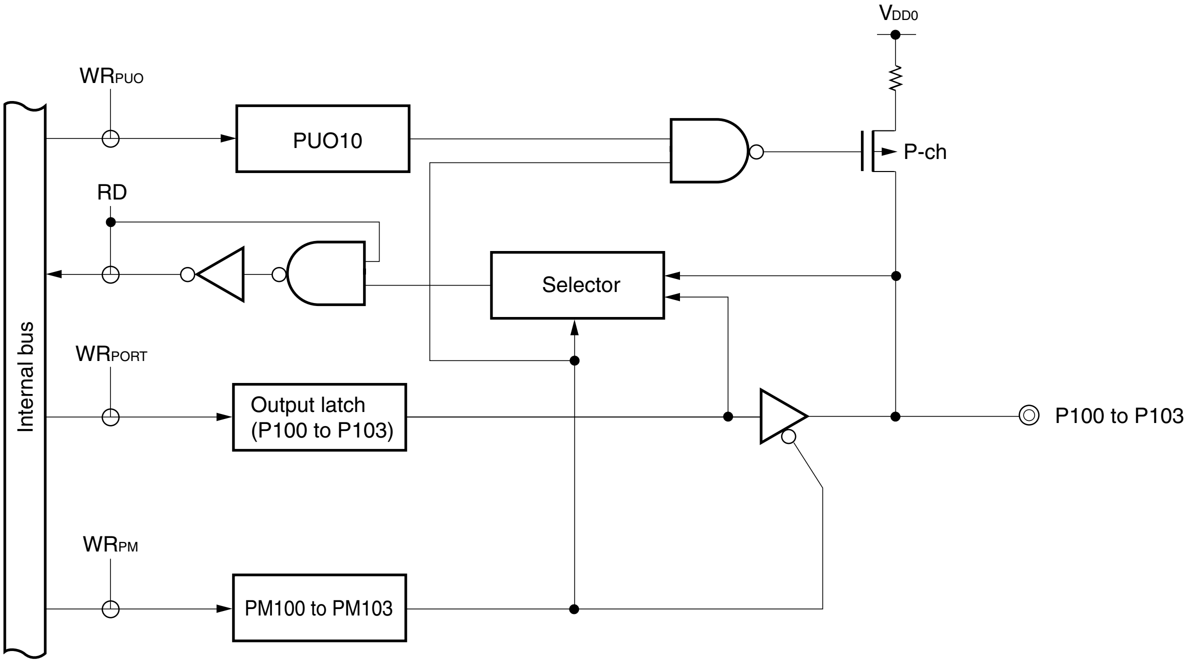
6.2.9 Port 10

Port 10 is a 4-bit I/O port with output latch. P100 to P103 pins can specify the input mode/output mode in 1-bit units with port mode register 10. When P100 to P103 pins are used as input ports, an internal pull-up resistor can be used to them in 4-bit units with pull-up resistor option register H.

$\overline{\text{RESET}}$ input sets port 10 to input mode.

Figure 6-14 shows a block diagram of port 10.

Figure 6-14. P100 to P103 Block Diagram



PUO: Pull-up resistor option register
PM: Port mode register
RD: Port 10 read signal
WR: Port 10 write signal

6.2.10 Port 11

Port 11 is an 8-bit I/O port with output latch. P110 to P117 pins can specify the input mode/output mode in 1-bit units with port mode register 11. When P110 to P117 pins are used as input ports, an internal pull-up resistor can be used to them in 8-bit units with pull-up resistor option register H.

Alternate functions include serial interface data I/O and clock I/O.

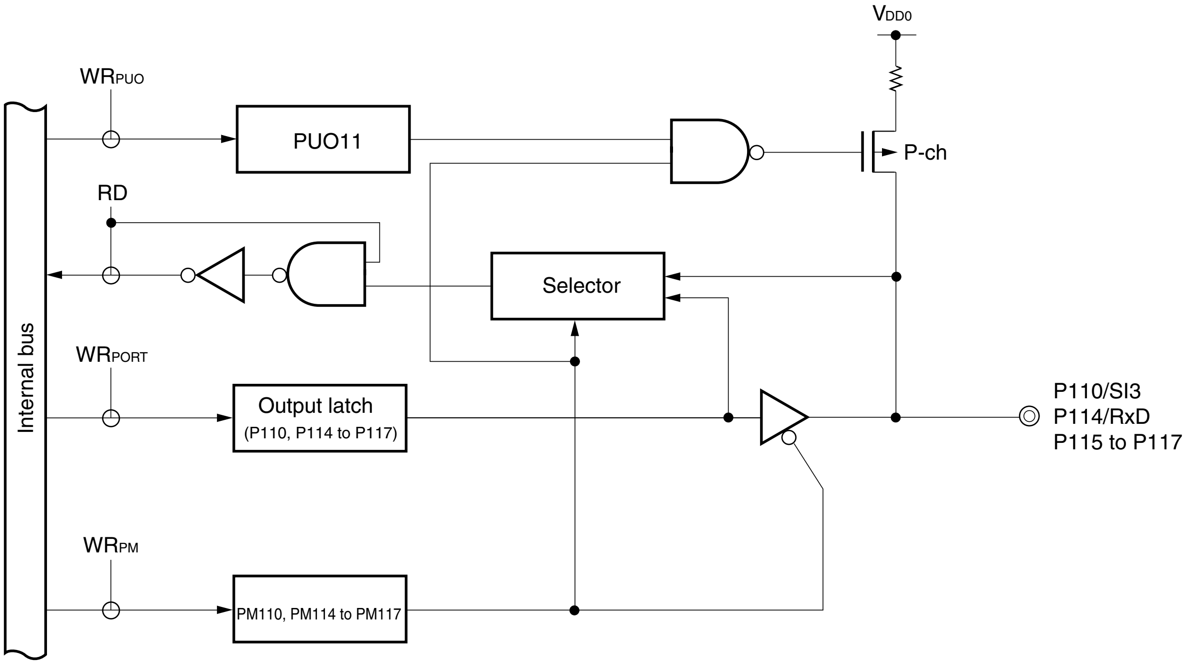
When this function is not used, the test input flag (KRIF) can be set to 1 when the falling edge is detected on this port.

$\overline{\text{RESET}}$ input sets port 11 to input mode.

Figures 6-15 to 6-17 show the block diagrams of port 11, and Figure 6-18 shows the falling edge detector, respectively.

Caution When used as a serial interface, set the I/O and output latch according to its functions. For the setting method, refer to Table 17-2 Serial Interface Channel 2 Operating Mode Settings and Figure 18-3 Serial Operating Mode Register 3 Format.

Figure 6-15. P110, P114 to P117 Block Diagram



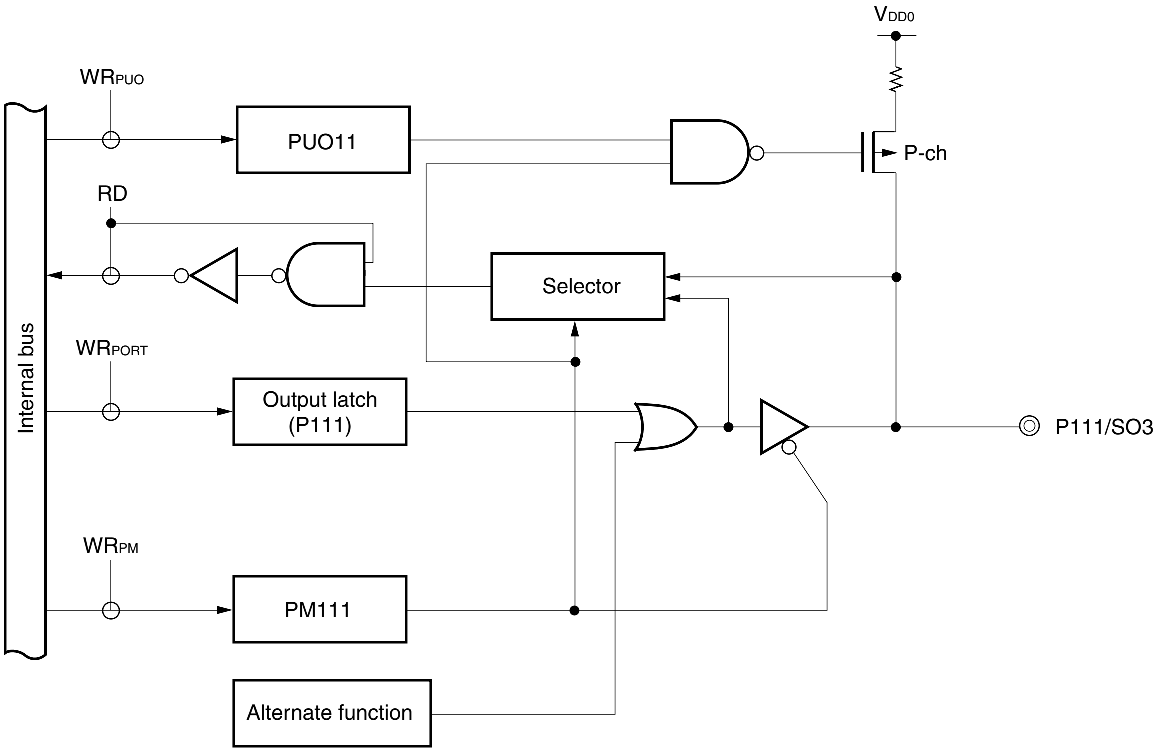
PUO: Pull-up resistor option register

PM: Port mode register

RD: Port 11 read signal

WR: Port 11 write signal

Figure 6-16. P111 Block Diagram



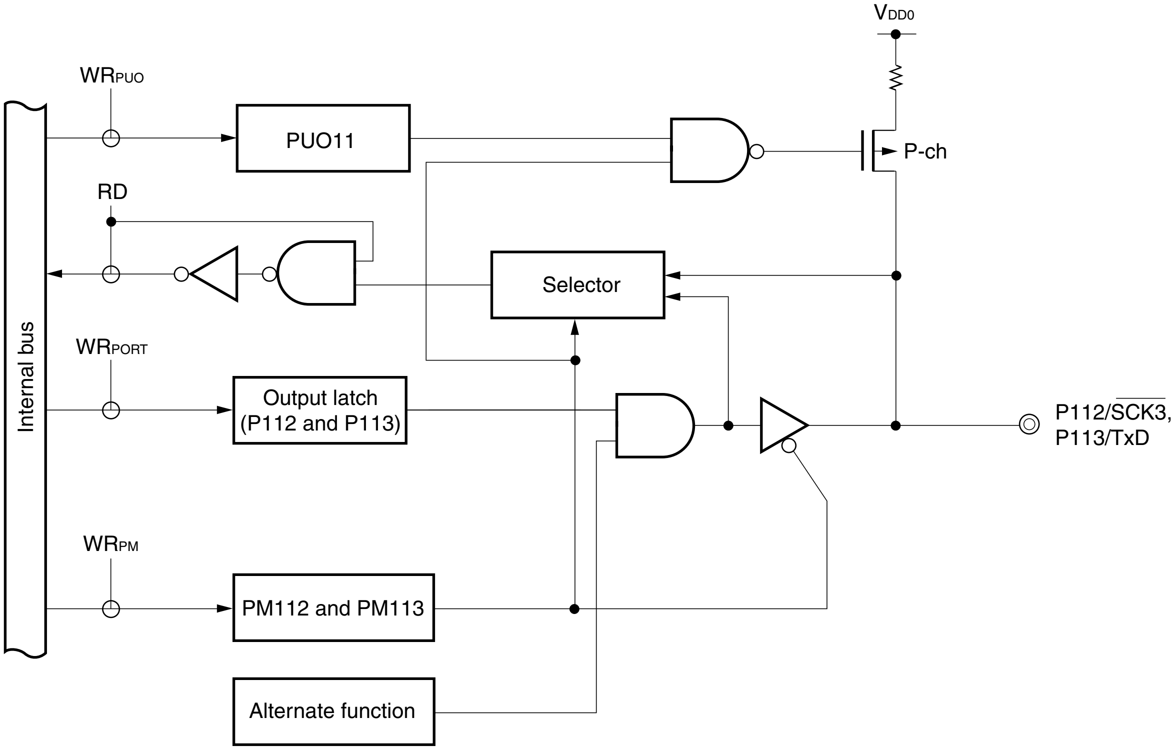
PUO: Pull-up resistor option register

PM: Port mode register

RD: Port 11 read signal

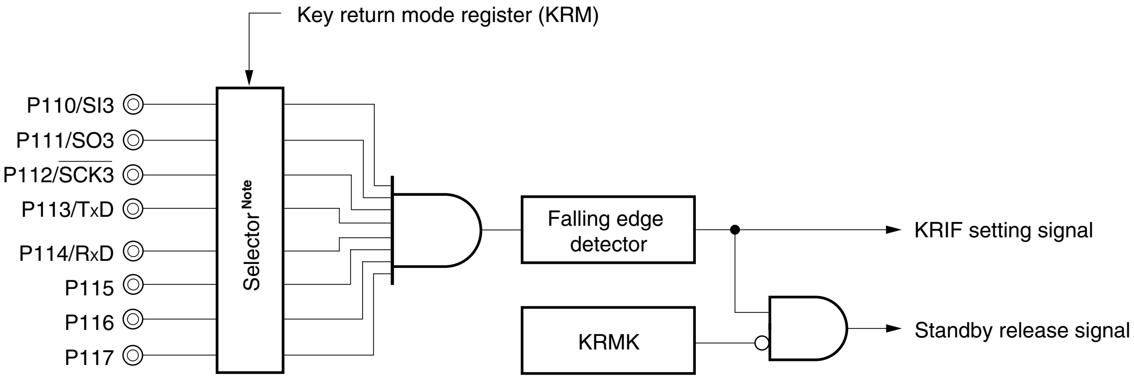
WR: Port 11 write signal

Figure 6-17. P112 and P113 Block Diagram



PUO: Pull-up resistor option register
PM: Port mode register
RD: Port 11 read signal
WR: Port 11 write signal

Figure 6-18. Block Diagram of Falling Edge Detector



Note Selector that selects a pin used to input the falling edge

6.3 Port Function Control Registers

The following three types of registers control the ports.

- Port mode registers (PM0 to PM3, PM7 to PM11)
- Pull-up resistor option register (PUOH, PUOL)
- Key return mode register (KRM)

(1) Port mode registers (PM0 to PM3, PM7 to PM11)

These registers are used to set port input/output in 1-bit units.

PM0 to PM3 and PM7 to PM11 are independently set with a 1-bit or 8-bit memory manipulation instruction.

$\overline{\text{RESET}}$ input sets registers to FFH.

When port pins are used as the alternate-function pins, set the port mode register and output latch according to Table 6-4.

Cautions 1. Pins P00 and P07 are input-only pins.

2. As port 0 has an alternate function as external interrupt request input, when the port function output mode is specified and the output level is changed, the interrupt request flag is set. When the output mode is used, therefore, the interrupt mask flag should be set to 1 beforehand.

3. Port 11 has a falling edge detection function. Do not specify the pin of this port to input the falling edge in a mode other than port mode. For how to set this port to falling edge input, refer to Figure 6-21 Key Return Mode Register Format.

Table 6-4. Port Mode Register and Output Latch Settings When Using Alternate Functions

Pin Name	Alternate Functions		PM××	P××
	Name	I/O		
P00	INTP0	Input	1 (Fixed)	None
	TI00	Input	1 (Fixed)	None
P01	INTP1	Input	1	×
	TI01	Input	1	×
P02 to P05	INTP2 to INTP5	Input	1	×
P07Note 1	XT1	Input	1 (Fixed)	None
P10 to P17Note 1	ANI0 to ANI7	Input	1	×
P30 to P32	TO0 to TO2	Output	0	0
P33, P34	TI1, TI2	Input	1	×
P35	PCL	Output	0	0
P36	BUZ	Output	0	0
P80 to P87	S39 to S32	Output	×Note 2	
P90 to P97	S31 to S24	Output	×Note 2	

Notes 1. If these ports are read out when these pins are used in the alternate function mode, undefined values are read.

2. When the P80 to P87 and P90 to P97 pins are used for alternate functions, set the function by the LCD display control register (LCDC).

Caution When port 2, port 7, and port 11 are used for serial interface, the I/O latch or output latch must be set according to their function. For the setting methods, see Figure 15-4 Serial Operating Mode Register 0 Format, Figure 16-4 Serial Operating Mode Register 0 Format, Table 17-2 Serial Interface Channel 2 Operating Mode Settings, and Figure 18-3 Serial Operating Mode Register 3 Format.

Remark	
×:	don't care
PM××:	port mode register
P××:	port output latch

Figure 6-19. Port Mode Register Format

Symbol	7	6	5	4	3	2	1	0	Address	After Reset	R/W
PM0	1	1	PM05	PM04	PM03	PM02	PM01	1	FF20H	FFH	R/W
PM1	PM17	PM16	PM15	PM14	PM13	PM12	PM11	PM10	FF21H	FFH	R/W
PM2	PM27	PM26	PM25	1	1	1	1	1	FF22H	FFH	R/W
PM3	PM37	PM36	PM35	PM34	PM33	PM32	PM31	PM30	FF23H	FFH	R/W
PM7	1	1	1	1	1	PM72	PM71	PM70	FF27H	FFH	R/W
PM8	PM87	PM86	PM85	PM84	PM83	PM82	PM81	PM80	FF28H	FFH	R/W
PM9	PM97	PM96	PM95	PM94	PM93	PM92	PM91	PM90	FF29H	FFH	R/W
PM10	1	1	1	1	PM103	PM102	PM101	PM100	FF2AH	FFH	R/W
PM11	PM117	PM116	PM115	PM114	PM113	PM112	PM111	PM110	FF2BH	FFH	R/W

PMmn	Pmn Pin I/O Mode Selection (m = 0 to 3, 7 to 11 : n = 0 to 7)
0	Output mode (output buffer ON)
1	Input mode (output buffer OFF)

(2) Pull-up resistor option register (PUOH, PUOL)

This register is used to set whether to use an internal pull-up resistor at each port or not. A pull-up resistor is internally used at bits which are set to the input mode at a port where internal pull-up resistor use has been specified with PUOH, PUOL. No internal pull-up resistors can be used to the bits set to the output mode or to the bits used as an analog input pin, irrespective of PUOH or PUOL setting.

PUOH and PUOL are set with a 1-bit or 8-bit memory manipulation instruction.

$\overline{\text{RESET}}$ input clears this register to 00H.

Cautions 1. P00 and P07 pins do not incorporate a pull-up resistor.

2. When ports 1, 8, and 9 are used as alternate-function pins, an internal pull-up resistor cannot be used even if 1 is set in PUOm (m = 1, 8, 9).

Figure 6-20. Pull-Up Resistor Option Register Format

Symbol	7	6	5	4	③	②	①	⓪	Address	After Reset	R/W
PUOH	0	0	0	0	PUO11	PUO10	PUO9	PUO8	FFF3H	00H	R/W
	⑦	6	5	4	③	②	①	⓪			
PUOL	PUO7	0	0	0	PUO3	PUO2	PUO1	PUO0	FFF7H	00H	R/W

PUOm	Pm Internal Pull-up Resistor Selection (m = 0 to 3, 7 to 11)
0	Internal pull-up resistor not used
1	Internal pull-up resistor used

Caution Zeros must be set to bits 4 to 7 of PUOH and bits 4 to 6 of PUOL.

(3) Key return mode register (KRM)

This register sets enabling/disabling of standby function release by a key return signal (falling edge detection of port 11), and selects the port 11 falling edge input.

KRM is set with a 1-bit or 8-bit memory manipulation instruction.

$\overline{\text{RESET}}$ input sets KRM to 02H.

Figure 6-21. Key Return Mode Register Format

Symbol	7	6	5	4	3	2	①	⓪	Address	After Reset	R/W
KRM	0	0	0	0	KRM3	KRM2	KRMK	KRIF	FFB8H	02H	R/W

KRM3	KRM2	Selection of Port 11 Falling Edge Input
0	0	P117
0	1	P114 to P117
1	0	P112 to P117
1	1	P110 to P117

KRMK	Standby Mode Control by Key Return Signal
0	Standby mode release enabled
1	Standby mode release disabled

KRIF	Key Return Signal Detection Flag
0	Not detected
1	Detected (falling edge detection of port 11)

Caution When falling edge detection of port 11 is used, KRIF should be cleared to 0 (not cleared to 0 automatically).

6.4 Port Function Operations

Port operations differ depending on whether the input or output mode is set, as shown below.

6.4.1 Writing to I/O port

(1) Output mode

A value is written to the output latch by a transfer instruction, and the output latch contents are output from the pin.

Once data is written to the output latch, it is retained until data is written to the output latch again.

(2) Input mode

A value is written to the output latch by a transfer instruction, but since the output buffer is OFF, the pin status does not change.

Once data is written to the output latch, it is retained until data is written to the output latch again.

Caution In the case of 1-bit memory manipulation instruction, although a single bit is manipulated the port is accessed as an 8-bit unit. Therefore, on a port with a mixture of input and output pins, the output latch contents for pins specified as input are undefined, even for bits other than the manipulated bit.

6.4.2 Reading from I/O port

(1) Output mode

The output latch contents are read by a transfer instruction. The output latch contents do not change.

(2) Input mode

The pin status is read by a transfer instruction. The output latch contents do not change.

6.4.3 Operations on I/O port

(1) Output mode

An operation is performed on the output latch contents, and the result is written to the output latch. The output latch contents are output from the pins.

Once data is written to the output latch, it is retained until data is written to the output latch again.

(2) Input mode

The output latch contents are undefined, but since the output buffer is OFF, the pin status does not change.

Caution In the case of 1-bit memory manipulation instruction, although a single bit is manipulated the port is accessed as an 8-bit unit. Therefore, on a port with a mixture of input and output pins, the output latch contents for pins specified as input are undefined, even for bits other than the manipulated bit.

CHAPTER 7 CLOCK GENERATOR

7.1 Clock Generator Functions

The clock generator generates the clock to be supplied to the CPU and peripheral hardware. The following two types of system clock oscillators are available.

(1) Main system clock oscillator
This circuit oscillates at frequencies of 1 to 5.0 MHz. Oscillation can be stopped by executing the STOP instruction or setting the processor clock control register (PCC).

(2) Subsystem clock oscillator
The circuit oscillates at a frequency of 32.768 kHz. Oscillation cannot be stopped. If the subsystem clock oscillator is not used, not using the internal feedback resistance can be set by the processor clock control register (PCC). This enables to decrease power consumption in the STOP mode.

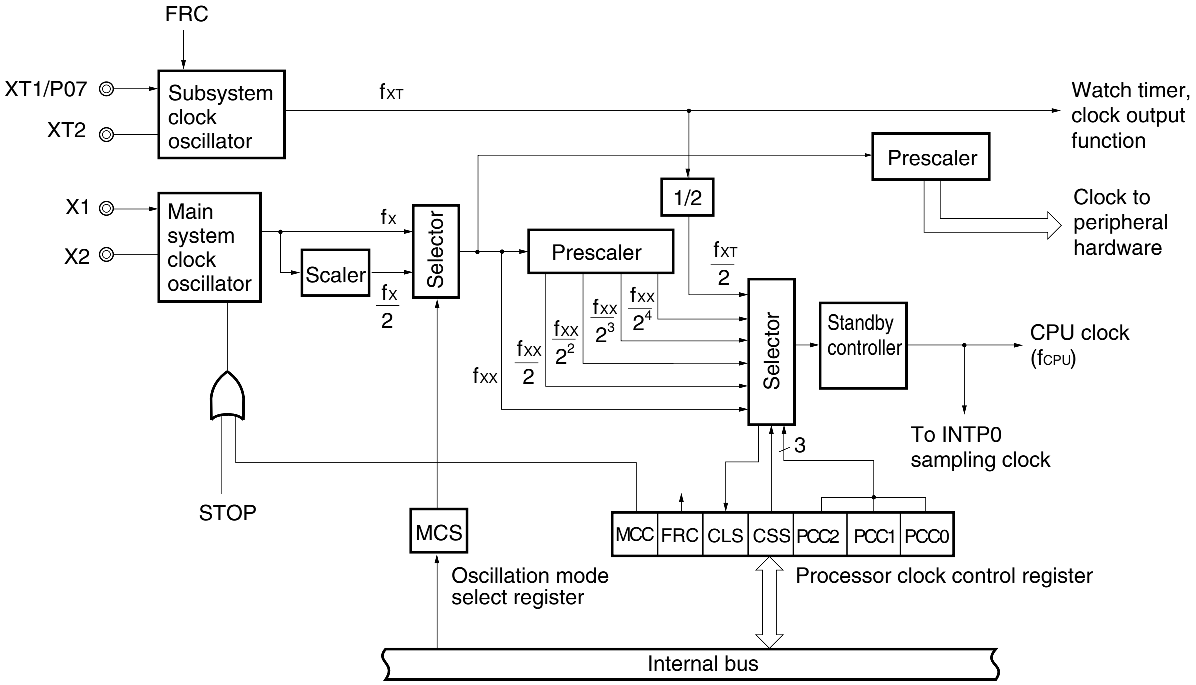
7.2 Clock Generator Configuration

The clock generator consists of the following hardware.

Table 7-1. Clock Generator Configuration

Item	Configuration
Control register	Processor clock control register (PCC) Oscillation mode select register (OSMS)
Oscillator	Main system clock oscillator Subsystem clock oscillator

Figure 7-1. Clock Generator Block Diagram



7.3 Clock Generator Control Register

The clock generator is controlled by the following two registers:

- Processor clock control register (PCC)
- Oscillation mode select register (OSMS)

(1) Processor clock control register (PCC)

The PCC sets whether to use CPU clock selection, the ratio of division, main system clock oscillator operation/ stop and subsystem clock oscillator internal feedback resistor.

The PCC is set with a 1-bit or 8-bit memory manipulation instruction.

$\overline{\text{RESET}}$ input sets the PCC to 04H.

Figure 7-2. Subsystem Clock Feedback Resistor

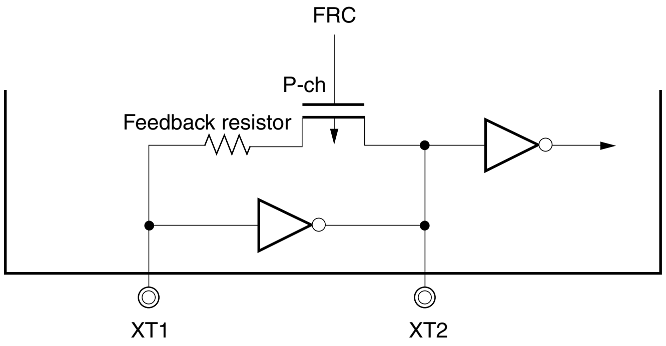


Figure 7-3. Processor Clock Control Register Format

Symbol	⑦	⑥	⑤	④	3	2	1	0	Address	After Reset	R/W
PCC	MCC	FRC	CLS	CSS	0	PCC2	PCC1	PCC0	FFFBH	04H	R/W[Note 1]

R/W	MCC	Main System Clock Oscillation Control[Note 2]
	0	Oscillation possible
	1	Oscillation stopped

R/W	FRC	Subsystem Clock Feedback Resistor Selection
	0	Internal feedback resistor used
	1	Internal feedback resistor not used

R	CLS	CPU Clock Status
	0	Main system clock
	1	Subsystem clock

R/W	CSS	PCC2	PCC1	PCC0	CPU Clock (f_{CPU}) Selection	
					MCS = 1	MCS = 0
	0	0	0	0	f_X	$f_X/2$
		0	0	1	$f_X/2$	$f_X/2^2$
		0	1	0	$f_X/2^2$	$f_X/2^3$
		0	1	1	$f_X/2^3$	$f_X/2^4$
		1	0	0	$f_X/2^4$	$f_X/2^5$
	1	0	0	0	$f_{XT}/2$	
		0	0	1		
		0	1	0		
		0	1	1		
		1	0	0		
	Other than above				Setting prohibited	

Notes 1. Bit 5 is a read-only bit.

2. When the CPU is operating on the subsystem clock, MCC should be used to stop the main system clock oscillation. A STOP instruction should not be used.

Caution Bit 3 must be set to 0.

Remarks 1. f_X: Main system clock oscillation frequency

2. f_{XT}: Subsystem clock oscillation frequency

3. MCS: Bit 0 of oscillation mode select register

The fastest instruction of the μPD780308 and 780308Y Subseries is executed with two CPU clocks. Therefore, the relation between the CPU clock (f_{CPU}) and the minimum instruction execution time is as shown in Table 7-2.

Table 7-2. Relation Between CPU Clock and Minimum Instruction Execution Time

CPU Clock (f_{CPU})	Minimum Instruction Execution Time: $2/f_{CPU}$
f_X	0.4 μs
$f_X/2$	0.8 μs
$f_X/2^2$	1.6 μs
$f_X/2^3$	3.2 μs
$f_X/2^4$	6.4 μs
$f_X/2^5$	12.8 μs
$f_{XT}/2$	122 μs

f_X = 5.0 MHz, f_{XT} = 32.768 kHz

f_X: Main system clock oscillation frequency

f_{XT}: Subsystem clock oscillation frequency

(2) Oscillation mode select register (OSMS)

This register specifies whether the clock output from the main system clock oscillator without passing through the scaler is used as the main system clock, or the clock output via the scaler is used as the main system clock.

OSMS is set with an 8-bit memory manipulation instruction.

$\overline{\text{RESET}}$ input clears OSMS to 00H.

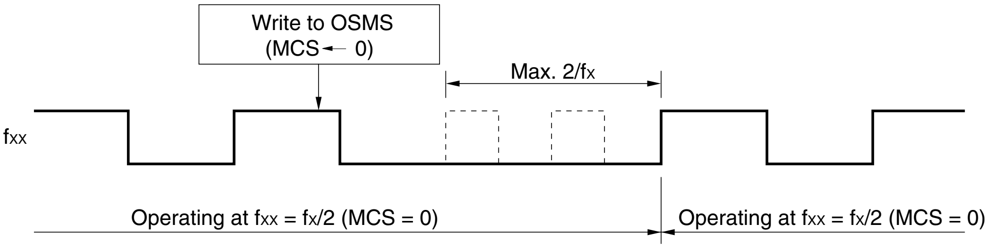
Figure 7-4. Oscillation Mode Select Register Format

Symbol	7	6	5	4	3	2	1	0	Address	After Reset	R/W
OSMS	0	0	0	0	0	0	0	MCS	FFF2H	00H	W

MCS	Main System Clock Scaler Control
0	Scaler used
1	Scaler not used

Cautions 1. As shown in Figure 7-5 below, writing data (including same data as previous) to OSMS cause delay of main system clock cycle up to 2/f_X during the write operation. Therefore, if this register is written during the operation, in peripheral hardware which operates with the main system clock, a temporary error occurs in the count clock cycle of timer, etc. In addition, because the oscillation mode is changed by this register, the clocks for peripheral hardware as well as that for the CPU are switched.

Figure 7-5. Main System Clock Waveform due to Writing to OSMS


Write to OSMS
(MCS ← 0)
Max. 2/fX
fXX
Operating at fXX = fX/2 (MCS = 0)
Operating at fXX = fX/2 (MCS = 0)

2. When writing "1" to MCS, V_{DD} must be 2.7 V or higher before the write execution.

Remark f_{XX}: Main system clock frequency (f_X or f_X/2)

f_X: Main system clock oscillation frequency

7.4 System Clock Oscillator

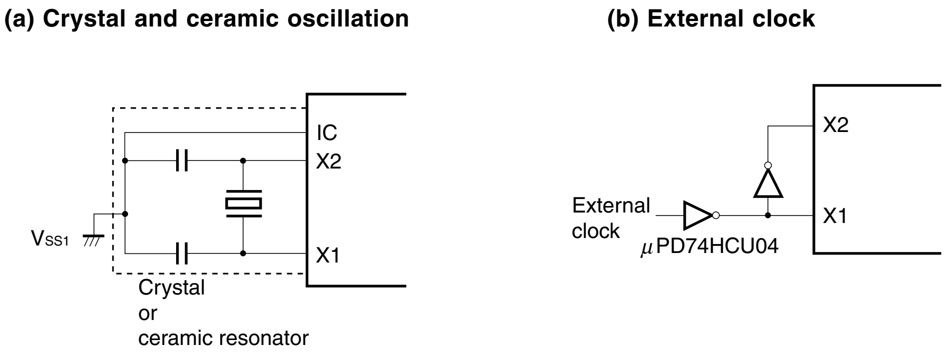
7.4.1 Main system clock oscillator

The main system clock oscillator oscillates with a crystal resonator or a ceramic resonator (standard: 5.0 MHz) connected to the X1 and X2 pins.

External clocks can be input to the main system clock oscillator. In this case, input a clock signal to the X1 pin and an antiphase clock signal to the X2 pin.

Figure 7-6 shows an external circuit of the main system clock oscillator.

Figure 7-6. External Circuit of Main System Clock Oscillator



(a) Crystal and ceramic oscillation

(b) External clock

Caution Do not execute the STOP instruction and do not set MCC to 1 if an external clock is used. This is because the X2 pin is connected to V_{DD1} via a pull-up resistor.

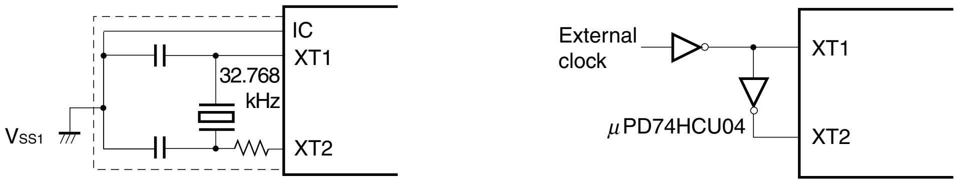
7.4.2 Subsystem clock oscillator

The subsystem clock oscillator oscillates with a crystal resonator (standard: 32.768 kHz) connected to the XT1 and XT2 pins.

External clocks can be input to the subsystem clock oscillator. In this case, input a clock signal to the XT1 pin and an antiphase clock signal to the XT2 pin.

Figure 7-7 shows an external circuit of the subsystem clock oscillator.

Figure 7-7. External Circuit of Subsystem Clock Oscillator



(a) Crystal oscillation

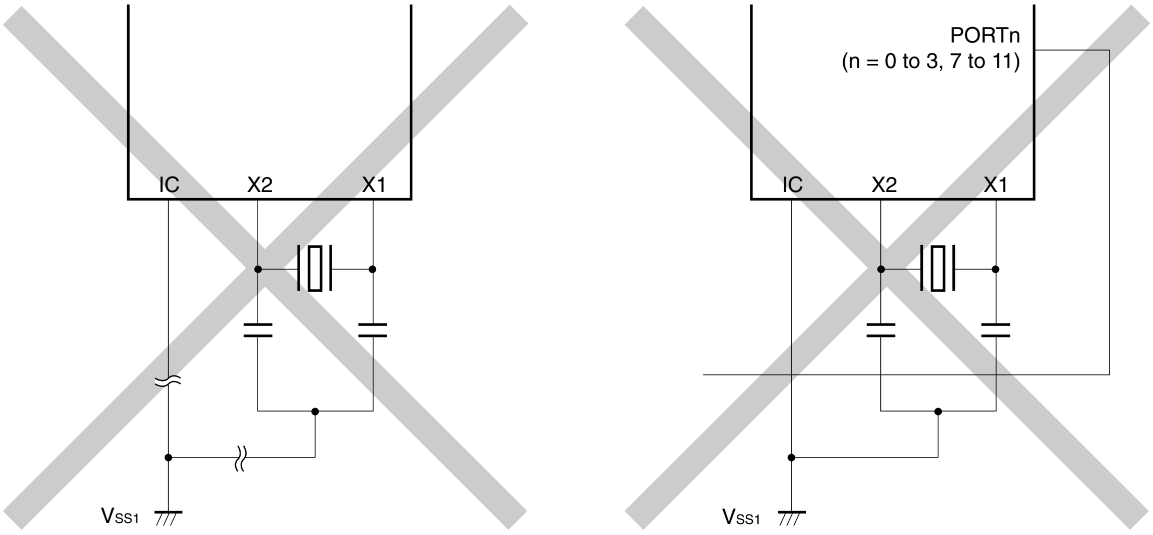
(b) External clock

Caution 1. When using the main system clock oscillator and subsystem clock oscillator, wire as follows in the area enclosed by broken lines in Figures 7-6 and 7-7 to avoid an adverse effect from wiring capacitance.

- **Keep the wiring length as short as possible.**
- **Do not cross the wiring with other signal lines. Do not route the wiring near a signal line through which a high fluctuating current flows.**
- **Always make the ground point of the oscillator capacitor the same potential as VSS1. Do not ground the capacitor to a ground pattern through which a high current flows.**
- **Do not fetch signals from the oscillator.**

Note that the subsystem clock oscillator is designed as a low-amplitude circuit for reducing power consumption.
Figure 7-8 shows examples of incorrect resonator connection.

Figure 7-8. Examples of Incorrect Resonator Connection (1/2)

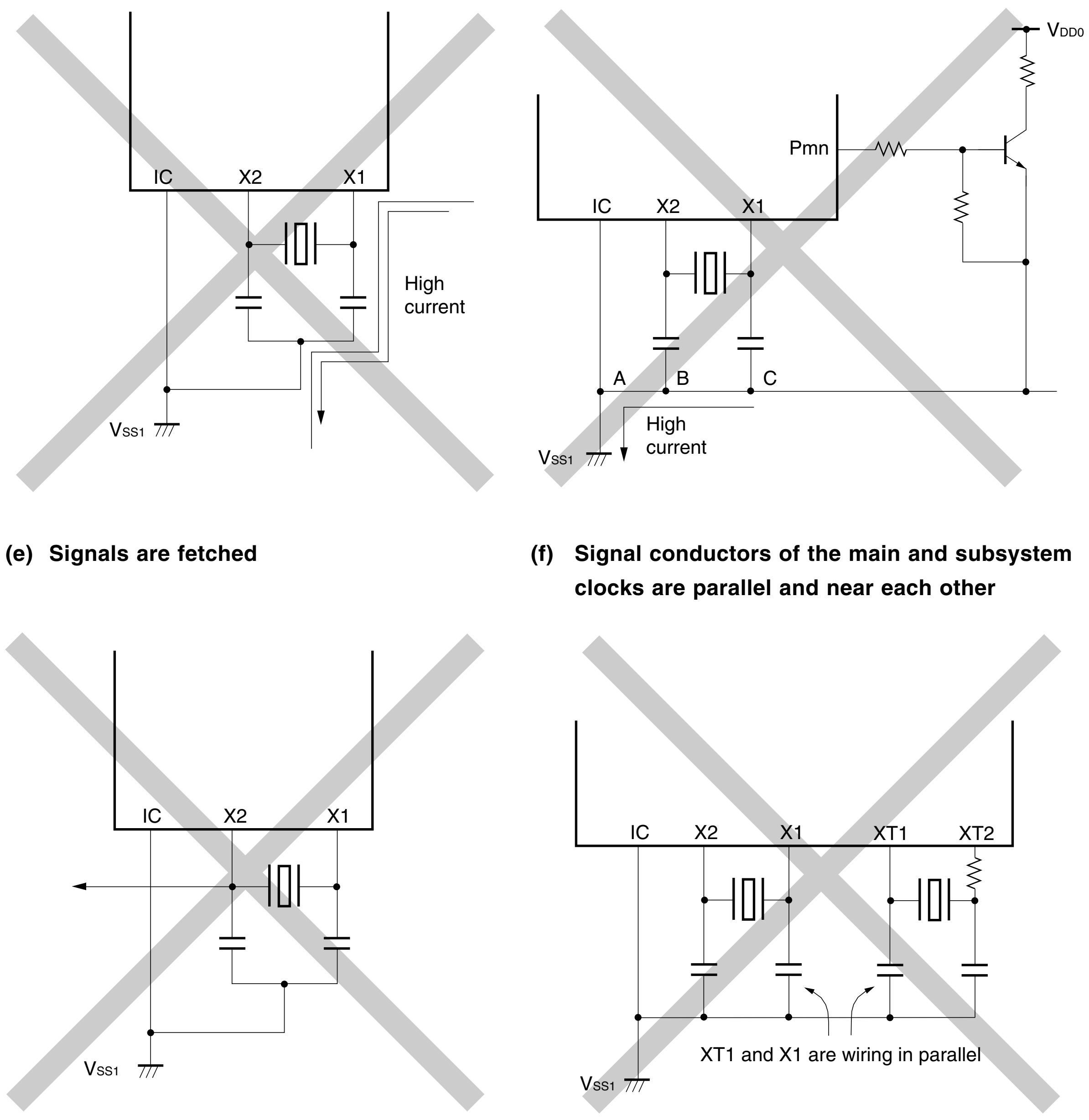


(a) Wiring of connection circuits is too long

(b) Signal conductors intersect each other

Remark When using a subsystem clock, replace X1 and X2 with XT1 and XT2, respectively. Also, insert resistors in series on the XT2 side.

Figure 7-8. Examples of Incorrect Resonator Connection (2/2)



(c) High fluctuating current is too near a signal conductor

(d) Current flows through the ground line of the oscillator (potential at points A, B, and C fluctuates)

(e) Signals are fetched

(f) Signal conductors of the main and subsystem clocks are parallel and near each other

Remark When using a subsystem clock, replace X1 and X2 with XT1 and XT2, respectively. Also, insert resistors in series on the XT2 side.

Caution 2. In Figure 7-8 (f), XT1 and X1 are wired in parallel. Thus, the crosstalk noise of X1 may increase with XT1, resulting in malfunctioning. To prevent that from occurring, it is recommended to wire XT1 and X1 so that they are not in parallel.

7.4.3 Scaler

The scaler divides the main system clock oscillator output (f_{XX}) and generates various clocks.

7.4.4 When no subsystem clocks are used

If it is not necessary to use subsystem clocks for low power consumption operations and clock operations, connect the XT1 and XT2 pins as follows.

XT1 : Connect to V_{DD0}
XT2 : Leave open

In this state, however, some current may leak via the internal feedback resistor of the subsystem clock oscillator when the main system clock stops. To minimize leakage current, the above internal feedback resistance can be removed with bit 6 (FRC) of the processor clock control register (PCC). In this case also, connect the XT1 and XT2 pins as described above.

7.5 Clock Generator Operations

The clock generator generates the following types of clocks and controls the CPU operating mode including the standby mode.

- Main system clock f_{XX}
- Subsystem clock f_{XT}
- CPU clock f_{CPU}
- Clock to peripheral hardware

The following clock generator functions and operations are determined with the processor clock control register (PCC) and the oscillation mode select register (OSMS).

(a) Upon generation of the $\overline{\text{RESET}}$ signal, the lowest speed mode of the main system clock (12.8 μs when operated at 5.0 MHz) is selected (PCC = 04H, OSMS = 00H). Main system clock oscillation stops while low level is applied to the $\overline{\text{RESET}}$ pin.

(b) With the main system clock selected, one of the six CPU clock types (0.4 μs. 0.8 μs, 1.6 μs, 3.2 μs, 6.4 μs, 12.8 μs @ 5.0 MHz) can be selected by setting the PCC and OSMS.

(c) With the main system clock selected, two standby modes, the STOP and HALT modes, are available. In a system that does not use the subsystem clock, the power consumption in the STOP mode can be further reduced by specifying not to use the internal feedback resistor by using bit 6 (FRC) of PCC.

(d) The PCC can be used to select the subsystem clock and to operate the system with low current consumption (122 μs when operated at 32.768 kHz).

(e) With the subsystem clock selected, main system clock oscillation can be stopped with the PCC. The HALT mode can be used. However, the STOP mode cannot be used. (Subsystem clock oscillation cannot be stopped.)

(f) The main system clock is divided and supplied to the peripheral hardware. The subsystem clock is supplied to the 16-bit timer/event counter, the watch timer, and clock output functions only. Thus, 16-bit timer/event counter (when selecting watch timer output for count clock operating with subsystem clock), the watch function, and the clock output function can also be continued in the standby state. However, since all other peripheral hardware operates with the main system clock, the peripheral hardware also stops if the main system clock is stopped (except external input clock operation).

7.5.1 Main system clock operations

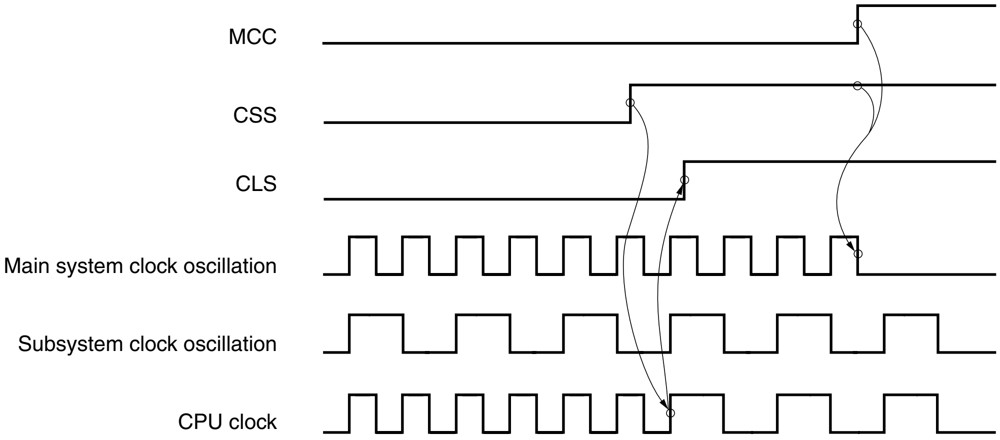
When operated with the main system clock (with bit 5 (CLS) of the processor clock control register (PCC) set to 0), the following operations are carried out by PCC setting.

(a) Because the operation guarantee instruction execution speed depends on the power supply voltage, the minimum instruction execution time can be changed by bit 0 to bit 2 (PCC0 to PCC2) of the PCC.

(b) If bit 7 (MCC) of the PCC is set to 1 when operated with the main system clock, the main system clock oscillation does not stop. When bit 4 (CSS) of the PCC is set to 1 and the operation is switched to subsystem clock operation (CLS = 1) after that, the main system clock oscillation stops (see **Figure 7-9**).

Figure 7-9. Main System Clock Stop Function (1/2)

(a) Operation when MCC is set after setting CSS with main system clock operation



(b) Operation when MCC is set in case of main system clock operation

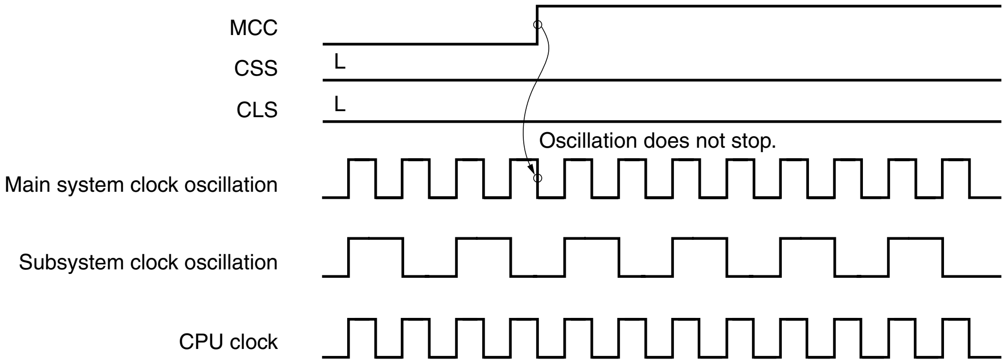
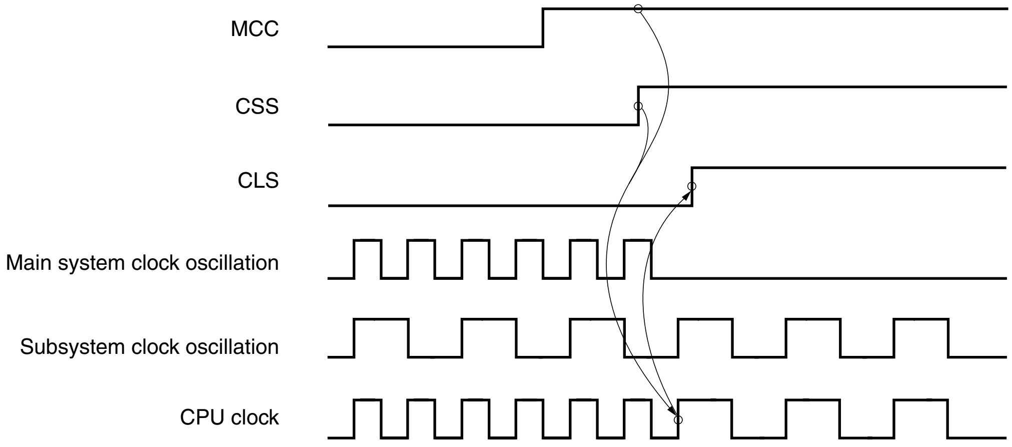

MCC
CSS
L
CLS
L
Oscillation does not stop.
Main system clock oscillation
Subsystem clock oscillation
CPU clock

Figure 7-9. Main System Clock Stop Function (2/2)

(c) Operation when CSS is set after setting MCC with main system clock operation



7.5.2 Subsystem clock operations

When operated with the subsystem clock (with bit 5 (CLS) of the processor clock control register (PCC) set to 1), the following operations are carried out.

(a) The minimum instruction execution time remains constant (122 μs when operated at 32.768 kHz) irrespective of bit 0 to bit 2 (PCC0 to PCC2) of the PCC.

(b) Watchdog timer counting stops.

Caution Do not execute the STOP instruction while the subsystem clock is in operation.

7.6 Changing System Clock and CPU Clock Settings

7.6.1 Time required for switchover between system clock and CPU clock

The system clock and CPU clock can be switched over by means of bit 0 to bit 2 (PCC0 to PCC2) and bit 4 (CSS) of the processor clock control register (PCC).

The actual switchover operation is not performed directly after writing to the PCC, but operation continues on the pre-switchover clock for several instructions (see **Table 7-3**).

Determination as to whether the system is operating on the main system clock or the subsystem clock is performed by bit 5 (CLS) of the PCC register.

Table 7-3. Maximum Time Required for CPU Clock Switchover

Set Values after Switchover					Set Values before Switchover					
MCS	CSS	PCC2	PCC1	PCC0	CSS PCC2 PCC1 PCC0	CSS PCC2 PCC1 PCC0	CSS PCC2 PCC1 PCC0	CSS PCC2 PCC1 PCC0	CSS PCC2 PCC1 PCC0	CSS PCC2 PCC1 PCC0
					0 0 0 0	0 0 0 1	0 0 1 0	0 0 1 1	0 1 0 0	1 × × ×
×	0	0	0	0		8 instructions	4 instructions	2 instructions	1 instruction	1 instruction
		0	0	1	16 instructions		4 instructions	2 instructions	1 instruction	1 instruction
		0	1	0	16 instructions	8 instructions		2 instructions	1 instruction	1 instruction
		0	1	1	16 instructions	8 instructions	4 instructions		1 instruction	1 instruction
		1	0	0	16 instructions	8 instructions	4 instructions	2 instructions		1 instruction
1	1	×	×	×	$f_X/2f_{XT}$ instruction (77 instructions)	$f_X/4f_{XT}$ instruction (39 instructions)	$f_X/8f_{XT}$ instruction (20 instructions)	$f_X/16f_{XT}$ instruction (10 instructions)	$f_X/32f_{XT}$ instruction (5 instructions)	
0					$f_X/4f_{XT}$ instruction (39 instructions)	$f_X/8f_{XT}$ instruction (20 instructions)	$f_X/16f_{XT}$ instruction (10 instructions)	$f_X/32f_{XT}$ instruction (5 instructions)	$f_X/64f_{XT}$ instruction (3 instructions)	

Caution Selection of the CPU clock cycle scaling factor (PCC0 to PCC2) and switchover from the main system clock to the subsystem clock (changing CSS from 0 to 1) should not be performed simultaneously. Simultaneous setting is possible, however, for selection of the CPU clock cycle scaling factor (PCC0 to PCC2) and switchover from the subsystem clock to the main system clock (changing CSS from 1 to 0).

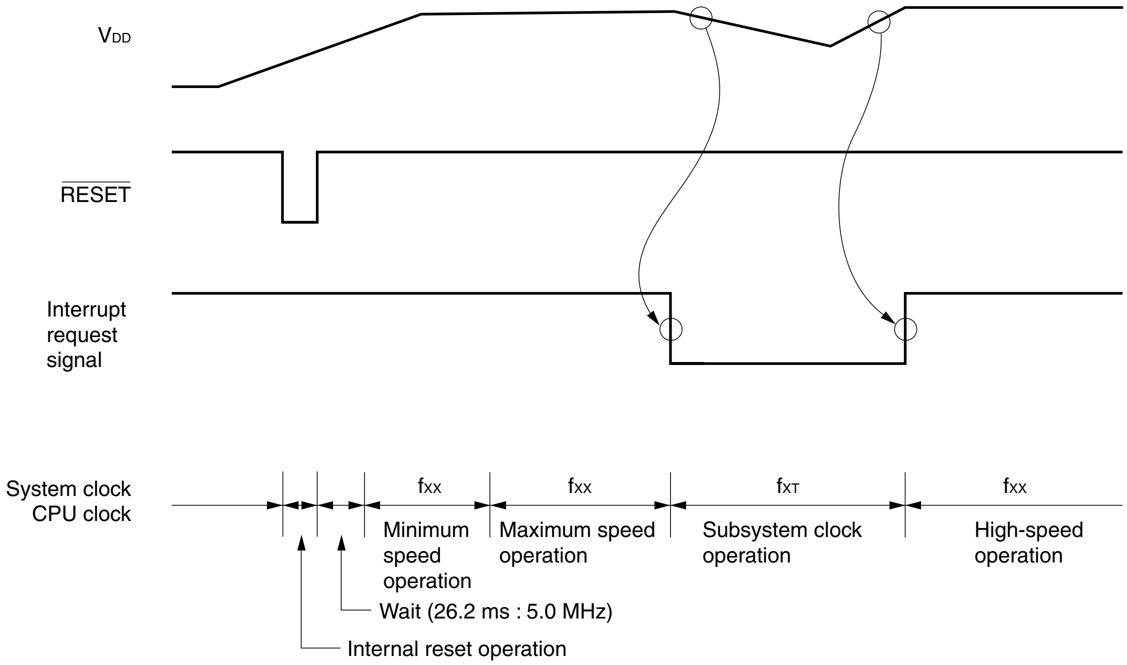
Remarks 1. One instruction is the minimum instruction execution time with the pre-switchover CPU clock.

2. Figures in parentheses apply to operation with f_X = 5.0 MHz and f_{XT} = 32.768 kHz.

7.6.2 System clock and CPU clock switching procedure

This section describes switching procedure between system clock and CPU clock.

Figure 7-10. System Clock and CPU Clock Switching



(1) The CPU is reset by setting the $\overline{\text{RESET}}$ signal to low level after power-on. After that, when reset is released by setting the $\overline{\text{RESET}}$ signal to high level, main system clock starts oscillation. At this time, oscillation stabilization time ($2^{17}/f_X$) is secured automatically.
After that, the CPU starts executing the instruction at the minimum speed of the main system clock (12.8 μs when operated at 5.0 MHz).

(2) After the lapse of a sufficient time for the V_{DD} voltage to increase to enable operation at maximum speeds, the processor clock control register (PCC) and oscillation mode select register (OSMS) are rewritten and the maximum-speed operation is carried out.

(3) Upon detection of a decrease of the V_{DD} voltage due to an interrupt request signal, the main system clock is switched to the subsystem clock (which must be in an oscillation stable state).

(4) Upon detection of V_{DD} voltage reset due to an interrupt request signal, 0 is set to bit 7 of PCC (MCC) and oscillation of the main system clock is started. After the lapse of time required for stabilization of oscillation, the PCC and OSMS are rewritten and the maximum-speed operation is resumed.

Caution When subsystem clock is being operated while main system clock was stopped, if switching to the main system clock is made again, be sure to switch after securing oscillation stable time by software.

CHAPTER 8 16-BIT TIMER/EVENT COUNTER

8.1 Outline of Internal Timer of μPD780308 and 780308Y Subseries

This chapter explains the 16-bit timer/event counter. Before that, the internal timer of the μPD780308 and 780308Y Subseries and related functions are briefly explained below.

(1) 16-bit timer/event counter (TM0)

TM0 can be used for an interval timer, PWM output, pulse widths measurement (infrared ray remote control receive function), external event counter, square wave output of any frequency or one-shot pulse output.

(2) 8-bit timers/event counters 1 and 2 (TM1 and TM2)

TM1 and TM2 can be used to serve as an interval timer and an external event counter and to output square waves with any selected frequency. Two 8-bit timer/event counters can be used as one 16-bit timer/event counter (see **CHAPTER 9 8-BIT TIMER/EVENT COUNTER**).

(3) Watch timer (TM3)

This timer can set a flag every 0.5 sec. and simultaneously generates interrupt requests at the preset time intervals (see **CHAPTER 10 WATCH TIMER**).

(4) Watchdog timer (WDTM)

WDTM can perform the watchdog timer function or generate non-maskable interrupt requests, maskable interrupt requests and $\overline{\text{RESET}}$ at the preset time intervals (see **CHAPTER 11 WATCHDOG TIMER**).

(5) Clock output controller

This circuit supplies other devices with the divided main system clock and the subsystem clock (see **CHAPTER 12 CLOCK OUTPUT CONTROLLER**).

(6) Buzzer output controller

This circuit outputs the buzzer frequency obtained by dividing the main system clock (see **CHAPTER 13 BUZZER OUTPUT CONTROLLER**).

Table 8-1. Timer/Event Counter Types and Functions

		16-bit Timer/ Event Counter	8-bit Timer/Event Counter	Watch Timer	Watchdog Timer
Type	Interval timer	2 channels[Note 1]	2 channels	1 channel[Note 2]	1 channel[Note 3]
	External event counter	√	√	—	—
Function	Timer output	√	√	—	—
	PWM output	√	—	—	—
	Pulse width measurement	√	—	—	—
	Square-wave output	√	√	—	—
	One-shot pulse output	√	—	—	—
	Interrupt request	√	√	√	√
	Test input	—	—	√	—

Notes 1. When capture/compare registers 00 and 01 (CR00, CR01) are specified as compare registers.

2. Watch timer can perform both watch timer and interval timer functions at the same time.

3. Watchdog timer can perform either the watchdog timer function or the interval timer function.

8.2 16-bit Timer/Event Counter Functions

The 16-bit timer/event counter (TM0) has the following functions.

- Interval timer
- PWM output
- Pulse width measurement
- External event counter
- Square-wave output
- One-shot pulse output

(1) Interval timer

TM0 generates interrupt requests at the preset time interval.

Table 8-2. 16-bit Timer/Event Counter Interval Times

Minimum Interval Time		Maximum Interval Time		Resolution	
MCS = 1	MCS = 0	MCS = 1	MCS = 0	MCS = 1	MCS = 0
2 × TI00 input cycle		2^{16} × TI00 input cycle		TI00 input edge cycle	
—	$2 \times 1/f_X$ (400 ns)	—	$2^{16} \times 1/f_X$ (13.1 ms)	—	$1/f_X$ (200 ns)
$2 \times 1/f_X$ (400 ns)	$2^2 \times 1/f_X$ (800 ns)	$2^{16} \times 1/f_X$ (13.1 ms)	$2^{17} \times 1/f_X$ (26.2 ms)	$1/f_X$ (200 ns)	$2 \times 1/f_X$ (400 ns)
$2^2 \times 1/f_X$ (800 ns)	$2^3 \times 1/f_X$ (1.6 μs)	$2^{17} \times 1/f_X$ (26.2 ms)	$2^{18} \times 1/f_X$ (52.4 ms)	$2 \times 1/f_X$ (400 ns)	$2^2 \times 1/f_X$ (800 ns)
$2^3 \times 1/f_X$ (1.6 μs)	$2^4 \times 1/f_X$ (3.2 μs)	$2^{18} \times 1/f_X$ (52.4 ms)	$2^{19} \times 1/f_X$ (104.9 ms)	$2^2 \times 1/f_X$ (800 ns)	$2^3 \times 1/f_X$ (1.6 μs)
2 × watch timer output cycle		2^{16} × watch timer output cycle		Watch timer output edge cycle	

Remarks 1. f_X: Main system clock oscillation frequency

2. MCS: Oscillation mode select register bit 0

3. Values in parentheses when operated at f_X = 5.0 MHz

(2) PWM output

TM0 can generate 14-bit resolution PWM output.

(3) Pulse width measurement

TM0 can measure the pulse width of an externally input signal.

(4) External event counter

TM0 can measure the number of pulses of an externally input signal.

(5) Square-wave output

TM0 can output a square wave with any selected frequency.

Table 8-3. 16-bit Timer/Event Counter Square-Wave Output Ranges

Minimum Pulse Width		Maximum Pulse Width		Resolution	
MCS = 1	MCS = 0	MCS = 1	MCS = 0	MCS = 1	MCS = 0
2 × TI00 input cycle		2^{16} × TI00 input cycle		TI00 input edge cycle	
—	$2 \times 1/f_X$ (400 ns)	—	$2^{16} \times 1/f_X$ (13.1 ms)	—	$1/f_X$ (200 ns)
$2 \times 1/f_X$ (400 ns)	$2^2 \times 1/f_X$ (800 ns)	$2^{16} \times 1/f_X$ (13.1 ms)	$2^{17} \times 1/f_X$ (26.2 ms)	$1/f_X$ (200 ns)	$2 \times 1/f_X$ (400 ns)
$2^2 \times 1/f_X$ (800 ns)	$2^3 \times 1/f_X$ (1.6 μs)	$2^{17} \times 1/f_X$ (26.2 ms)	$2^{18} \times 1/f_X$ (52.4 ms)	$2 \times 1/f_X$ (400 ns)	$2^2 \times 1/f_X$ (800 ns)
$2^3 \times 1/f_X$ (1.6 μs)	$2^4 \times 1/f_X$ (3.2 μs)	$2^{18} \times 1/f_X$ (52.4 ms)	$2^{19} \times 1/f_X$ (104.9 ms)	$2^2 \times 1/f_X$ (800 ns)	$2^3 \times 1/f_X$ (1.6 μs)
2 × watch timer output cycle		2^{16} × watch timer output cycle		Watch timer output edge cycle	

Remarks 1. f_X: Main system clock oscillation frequency

2. MCS: Oscillation mode select register bit 0

3. Values in parentheses when operated at f_X = 5.0 MHz

(6) One-shot pulse output

TM0 is able to output one-shot pulse which can set any width of output pulse.

8.3 16-bit Timer/Event Counter Configuration

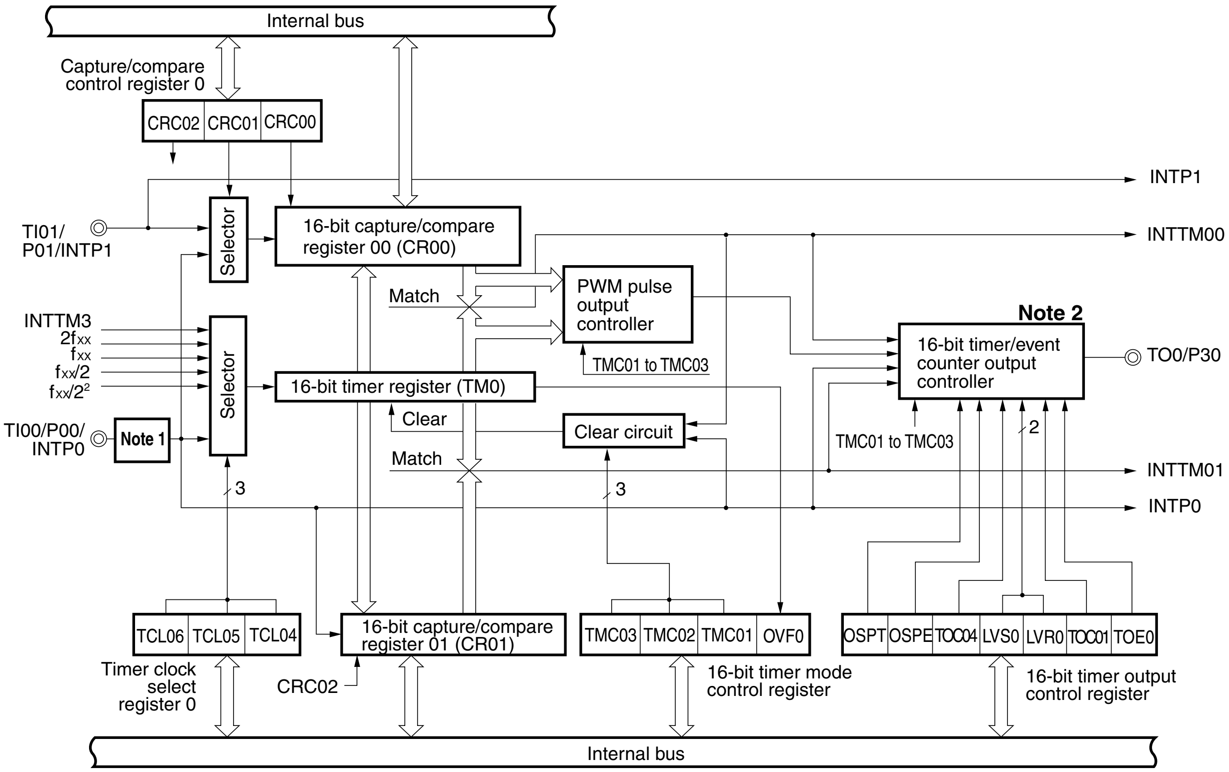
The 16-bit timer/event counter consists of the following hardware.

Table 8-4. 16-bit Timer/Event Counter Configuration

Item	Configuration
Timer register	16 bits × 1 (TM0)
Register	Capture/compare register: 16 bits × 2 (CR00, CR01)
Timer output	1 (TO0)
Control register	Timer clock select register 0 (TCL0) 16-bit timer mode control register (TMC0) Capture/compare control register 0 (CRC0) 16-bit timer output control register (TOC0) Port mode register 3 (PM3) External interrupt mode register 0 (INTM0) Sampling clock select register (SCS)[Note]

Note For details, refer to **Figure 20-1 Basic Configuration of Interrupt Function**.

Figure 8-1. 16-bit Timer/Event Counter Block Diagram

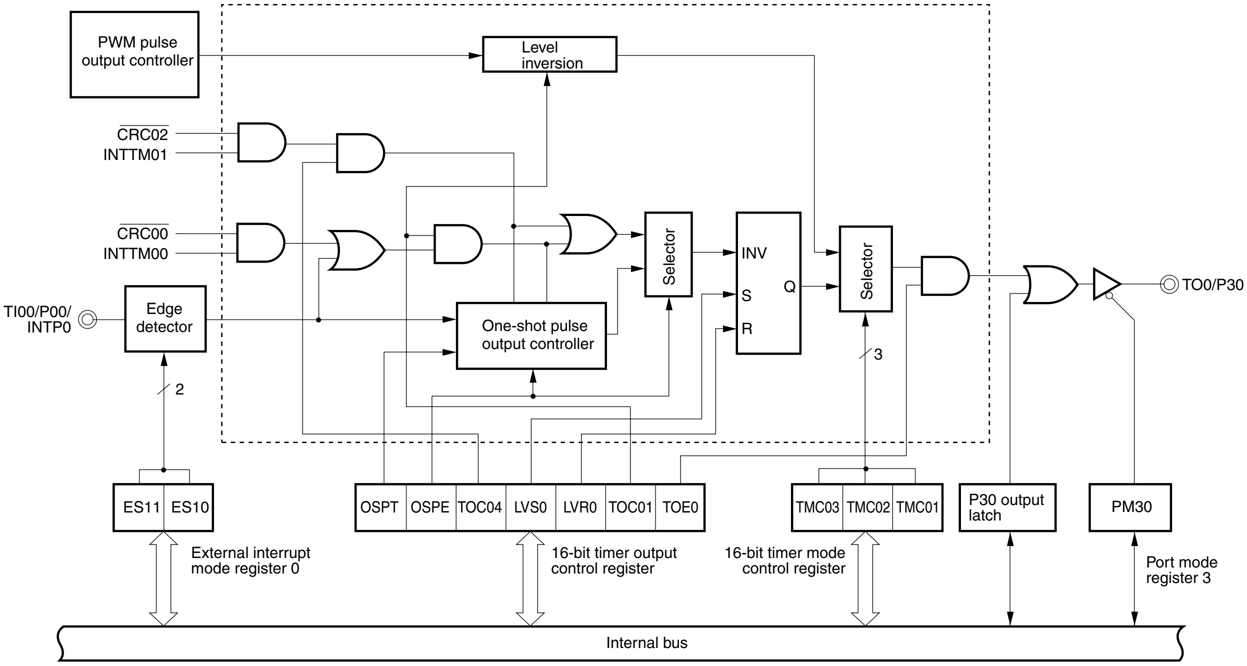

Internal bus
Capture/compare control register 0
CRC02
CRC01
CRC00
INTP1
TI01/P01/INTP1
Selector
16-bit capture/compare register 00 (CR00)
INTTM00
Match
PWM pulse output controller
TMC01 to TMC03
Note 2
16-bit timer/event counter output controller
TO0/P30
INTTM3
2fxx
fxx
fxx/2
fxx/2²
Selector
16-bit timer register (TM0)
Clear
Clear circuit
TI00/P00/INTP0
Note 1
TMC01 to TMC03
2
Match
INTTM01
3
3
INTP0
TCL06
TCL05
TCL04
16-bit capture/compare register 01 (CR01)
TMC03
TMC02
TMC01
OVF0
OSPT
OSPE
TOC04
LVS0
LVR0
TOC01
TOE0
Timer clock select register 0
CRC02
16-bit timer mode control register
16-bit timer output control register
Internal bus

Notes 1. Edge detector

2. The configuration of the 16-bit timer/event counter output controller is shown in Figure 8-2.

Remark fxx = fx/2 (MCS = 0), fxx = fx (MCS = 1)

Figure 8-2. 16-bit Timer/Event Counter Output Controller Block Diagram


PWM pulse output controller
Level inversion
$\overline{\text{CRC02}}$
INTTM01
$\overline{\text{CRC00}}$
INTTM00
TI00/P00/ INTP0
Edge detector
2
One-shot pulse output controller
Selector
INV
S
R
Q
Selector
3
TO0/P30
ES11
ES10
OSPT
OSPE
TOC04
LVS0
LVR0
TOC01
TOE0
TMC03
TMC02
TMC01
P30 output latch
PM30
External interrupt mode register 0
16-bit timer output control register
16-bit timer mode control register
Port mode register 3
Internal bus

Remark The circuitry enclosed by the dotted line is the output controller.

(1) Capture/compare register 00 (CR00)

CR00 is a 16-bit register which has the functions of both a capture register and a compare register. Whether it is used as a capture register or as a compare register is set by bit 0 (CRC00) of capture/compare control register 0.

When CR00 is used as a compare register, the value set in the CR00 is constantly compared with the 16-bit timer register (TM0) count value, and an interrupt request (INTTM00) is generated if they match. When TM0 is set to interval timer operation, this register is used to hold the interval time. When the PWM output operation is specified, it is used as a register that specifies a pulse width.

When CR00 is used as a capture register, it is possible to select the valid edge of the INTP0/TI00 pin or the INTP1/TI01 pin as the capture trigger. Setting of the INTP0/TI00 or INTP1/TI01 valid edge is performed by means of external interrupt mode register 0.

If CR00 is specified as a capture register and capture trigger is specified to be the valid edge of the INTP0/TI00 pin, the situation is as shown in the following table.

Table 8-5. INTP0/TI00 Pin Valid Edge and CR00 Capture Trigger Valid Edge

ES11	ES10	INTP0/TI00 Pin Valid Edge	CR00 Capture Trigger Valid Edge
0	0	Falling edge	Rising edge
0	1	Rising edge	Falling edge
1	0	Setting prohibited	
1	1	Both rising and falling edges	No capture operation

CR00 is set by a 16-bit memory manipulation instruction.

After $\overline{\text{RESET}}$ input, the value of CR00 is undefined.

Cautions 1. Set the data of PWM (14 bits) to the higher 4 bits of CR00. At this time, clear the lower 2 bits to 00.

2. Set a value other than 0000H to CR00. Therefore, when the 16-bit timer/event counter is used as an event counter, the 1-pulse count operation cannot be performed.

3. If the new value of CR00 is less than the value of the 16-bit timer register (TM0), TM0 continues counting. When an overflow occurs, it counts again from 0. Therefore, if the new value (M) of CR00 is less than the old value (N), the timer must be restarted after changing the value of CR00.

(2) Capture/compare register 01 (CR01)

CR01 is a 16-bit register which has the functions of both a capture register and a compare register. Whether it is used as a capture register or a compare register is set by bit 2 (CRC02) of capture/compare control register 0.

When CR01 is used as a compare register, the value set in the CR01 is constantly compared with the 16-bit timer register (TM0) count value, and an interrupt request (INTTM01) is generated if they match.

When CR01 is used as a capture register, it is possible to select the valid edge of the INTP0/TI00 pin as the capture trigger. Setting of the INTP0/TI00 valid edge is performed by means of external interrupt mode register 0 (INTM0).

CR01 is set with a 16-bit memory manipulation instruction.

After $\overline{\text{RESET}}$ input, the value of CR01 is undefined.

Caution If the valid edge of the TI00/P00 pin is input while CR01 is read, CR01 does not perform the capture operation, but retains data. However, the interrupt request flag (PIF0) is set when the valid edge is detected.

(3) 16-bit timer register (TM0)

TM0 is a 16-bit register which counts the count pulses.

TM0 is read by a 16-bit memory manipulation instruction. When TM0 is read, capture/compare register 01 (CR01) should first be set as a capture register.

$\overline{\text{RESET}}$ input clears TM0 to 0000H.

Caution Because reading of the value of TM0 is performed via CR01, the previously set value of CR01 is lost.

8.4 16-bit Timer/Event Counter Control Registers

The following seven types of registers are used to control the 16-bit timer/event counter.

- Timer clock select register 0 (TCL0)
- 16-bit timer mode control register (TMC0)
- Capture/compare control register 0 (CRC0)
- 16-bit timer output control register (TOC0)
- Port mode register 3 (PM3)
- External interrupt mode register 0 (INTM0)
- Sampling clock select register (SCS)

(1) Timer clock select register 0 (TCL0)

This register is used to set the count clock of the 16-bit timer register (TM0).
TCL0 is set with a 1-bit or 8-bit memory manipulation instruction.
$\overline{\text{RESET}}$ input clears TCL0 value to 00H.

Remark TCL0 has the function of setting the PCL output clock in addition to that of setting the count clock of the 16-bit timer register.

Figure 8-3. Timer Clock Select Register 0 Format

Symbol	⑦	6	5	4	3	2	1	0	Address	After Reset	R/W
TCL0	CLOE	TCL06	TCL05	TCL04	TCL03	TCL02	TCL01	TCL00	FF40H	00H	R/W

CLOE	PCL Output Control
0	Output disabled
1	Output enabled

TCL06	TCL05	TCL04	16-bit Timer Register Count Clock Selection	
			MCS = 1	MCS = 0
0	0	0	TI00 (Valid edge specifiable)	
0	0	1	Setting prohibited	f_X (5.0 MHz)
0	1	0	f_X (5.0 MHz)	$f_X/2$ (2.5 MHz)
0	1	1	$f_X/2$ (2.5 MHz)	$f_X/2^2$ (1.25 MHz)
1	0	0	$f_X/2^2$ (1.25 MHz)	$f_X/2^3$ (625 kHz)
1	1	1	Watch timer output (INTTM3)	
Other than above			Setting prohibited	

TCL03	TCL02	TCL01	TCL00	PCL Output Clock Selection	
				MCS = 1	MCS = 0
0	0	0	0	f_{XT} (32.768 kHz)	
0	1	0	1	f_X (5.0 MHz)	$f_X/2$ (2.5 MHz)
0	1	1	0	$f_X/2$ (2.5 MHz)	$f_X/2^2$ (1.25 MHz)
0	1	1	1	$f_X/2^2$ (1.25 MHz)	$f_X/2^3$ (625 kHz)
1	0	0	0	$f_X/2^3$ (625 kHz)	$f_X/2^4$ (313 kHz)
1	0	0	1	$f_X/2^4$ (313 kHz)	$f_X/2^5$ (156 kHz)
1	0	1	0	$f_X/2^5$ (156 kHz)	$f_X/2^6$ (78.1 kHz)
1	0	1	1	$f_X/2^6$ (78.1 kHz)	$f_X/2^7$ (39.1 kHz)
1	1	0	0	$f_X/2^7$ (39.1 kHz)	$f_X/2^8$ (19.5 kHz)
Other than above				Setting prohibited	

Cautions 1. Setting of the TI00/INTP0 pin valid edge is performed by external interrupt mode register 0, and selection of the sampling clock frequency is performed by the sampling clock select register.

2. When enabling PCL output, set TCL00 to TCL03, then set 1 in CLOE with a 1-bit memory manipulation instruction.

3. To read the count value when TI00 has been specified as the TM0 count clock, the value should be read from TM0, not from capture/compare register 01 (CR01).

4. When rewriting TCL0 to other data, stop the timer operation beforehand.

Remarks **1.** f_X: Main system clock oscillation frequency
2. f_{XT}: Subsystem clock oscillation frequency
3. TI00: 16-bit timer/event counter input pin
4. TM0: 16-bit timer register
5. MCS: Bit 0 of oscillation mode select register
6. Figures in parentheses apply to operation with f_X = 5.0 MHz of f_{XT} = 32.768 kHz.

(2) 16-bit timer mode control register (TMC0)

This register sets the 16-bit timer operating mode, the 16-bit timer register clear mode and output timing, and detects an overflow.

TMC0 is set with a 1-bit or 8-bit memory manipulation instruction.

$\overline{\text{RESET}}$ input clears TMC0 value to 00H.

Caution The 16-bit timer register starts operation at the moment a value other than 0, 0, 0 (operation stop mode) is set in TMC01 to TMC03, respectively. Set 0, 0, 0 in TMC01 to TMC03 to stop the operation.

Figure 8-4. 16-bit Timer Mode Control Register Format

Symbol	7	6	5	4	3	2	1	⓪	Address	After Reset	R/W
TMC0	0	0	0	0	TMC03	TMC02	TMC01	OVF0	FF48H	00H	R/W

TMC03	TMC02	TMC01	Operating Mode Clear Mode Selection	TO0 Output Timing Selection	Interrupt Generation
0	0	0	Operation stop (TM0 cleared to 0)	No change	Not generated
0	0	1	PWM mode (free running)	PWM pulse output	Generated on match between TM0 and CR00, and match between TM0 and CR01
0	1	0	Free running mode	Match between TM0 and CR00 or match between TM0 and CR01	
0	1	1		Match between TM0 and CR00, match between TM0 and CR01 or TI00 valid edge	
1	0	0	Clear & start on TI00 valid edge	Match between TM0 and CR00 or match between TM0 and CR01	
1	0	1		Match between TM0 and CR00, match between TM0 and CR01 or TI00 valid edge	
1	1	0	Clear & start on match between TM0 and CR00	Match between TM0 and CR00 or match between TM0 and CR01	
1	1	1		Match between TM0 and CR00, match between TM0 and CR01 or TI00 valid edge	

OVF0	16-bit Timer Register Overflow Detection
0	Overflow not detected
1	Overflow detected

Cautions 1. Switch the clear mode and the T00 output timing after stopping the timer operation (by setting TMC01 to TMC03 to 0, 0, 0).

2. Set the valid edge of the TI00/INTP0 pin with external interrupt mode register 0 and select the sampling clock frequency with a sampling clock select register.

3. When using the PWM mode, set the PWM mode and then set data to CR00.

4. If clear & start mode on match between TM0 and CR00 is selected, when the set value of CR00 is FFFFH and the TM0 value changes from FFFFH to 0000H, OVF0 flag is set to 1.

Remark TO0: 16-bit timer/event counter output pin
TI00: 16-bit timer/event counter input pin
TM0: 16-bit timer register
CR00: Capture/compare register 00
CR01: Capture/compare register 01

(3) Capture/compare control register 0 (CRC0)

This register controls the operation of the capture/compare registers 00 and 01 (CR00, CR01).
CRC0 is set with a 1-bit or 8-bit memory manipulation instruction.
$\overline{\text{RESET}}$ input sets CRC0 value to 04H.

Figure 8-5. Capture/Compare Control Register 0 Format

Symbol	7	6	5	4	3	2	1	0	Address	After Reset	R/W
CRC0	0	0	0	0	0	CRC02	CRC01	CRC00	FF4CH	04H	R/W

CRC02	CR01 Operating Mode Selection
0	Operates as compare register
1	Operates as capture register

CRC01	CR00 Capture Trigger Selection
0	Captures on valid edge of TI01
1	Captures on valid edge of TI00

CRC00	CR00 Operating Mode Selection
0	Operates as compare register
1	Operates as capture register

Cautions 1. Timer operation must be stopped before setting CRC0.

2. When clear & start mode on a match between TM0 and CR00 is selected with the 16-bit timer mode control register, CR00 should not be specified as a capture register.

(4) 16-bit timer output control register (TOC0)

This register controls the operation of the 16-bit timer/event counter output controller. It sets R-S type flip-flop (LV0) setting/resetting, the active level in PWM mode, inversion enabling/disabling in modes other than PWM mode, 16-bit timer/event counter timer output enabling/disabling, one-shot pulse output operation enabling/disabling, and output trigger for a one-shot pulse by software. TOC0 is set with a 1-bit or 8-bit memory manipulation instruction. $\overline{\text{RESET}}$ input clears TOC0 value to 00H.

Figure 8-6. 16-bit Timer Output Control Register Format

Symbol	7	⑥	⑤	4	③	②	1	⓪	Address	After Reset	R/W
TOC0	0	OSPT	OSPE	TOC04	LVS0	LVR0	TOC01	TOE0	FF4EH	00H	R/W

OSPT	Control of One-Shot Pulse Output Trigger by Software
0	One-shot pulse trigger not used
1	One-shot pulse trigger used

OSPE	One-Shot Pulse Output Operation Control
0	Continuous pulse output
1	One-shot pulse output

TOC04	Timer Output F/F Control by Match of CR01 and TM0
0	Inversion operation disabled
1	Inversion operation enabled

LVS0	LVR0	16-bit Timer/Event Counter Timer Output F/F Status Setting
0	0	No change
0	1	Timer output F/F reset (0)
1	0	Timer output F/F set (1)
1	1	Setting prohibited

TOC04	In PWM Mode	In Other Modes
	Active Level Selection	Timer Output F/F Control by Match of CR00 and TM0
0	Active high	Inversion operation disabled
1	Active low	Inversion operation enabled

TOE0	16-bit Timer/Event Counter Output Control
0	Output disabled (port mode)
1	Output enabled

Cautions 1. Timer operation must be stopped before setting TOC0 (except OSPT).

2. If LVS0 and LVR0 are read after data is set, they will be 0.

3. OSPT is cleared automatically after data setting, and will therefore be 0 if read.

(5) Port mode register 3 (PM3)

This register sets port 3 input/output in 1-bit units.

When using the P30/TO0 pin for timer output, set PM30 and output latch of P30 to 0.

PM3 is set with a 1-bit or 8-bit memory manipulation instruction.

$\overline{\text{RESET}}$ input sets PM3 value to FFH.

Figure 8-7. Port Mode Register 3 Format

Symbol	7	6	5	4	3	2	1	0	Address	After Reset	R/W
PM3	PM37	PM36	PM35	PM34	PM33	PM32	PM31	PM30	FF23H	FFH	R/W

PM3n	P3n Pin I/O Mode Selection (n = 0 to 7)
0	Output mode (output buffer ON)
1	Input mode (output buffer OFF)

(6) External interrupt mode register 0 (INTM0)

This register is used to set INTP0 to INTP2 valid edges.

INTM0 is set with an 8-bit memory manipulation instruction.

$\overline{\text{RESET}}$ input clears INTM0 value to 00H.

Figure 8-8. External Interrupt Mode Register 0 Format

Symbol	7	6	5	4	3	2	1	0	Address	After Reset	R/W
INTM0	ES31	ES30	ES21	ES20	ES11	ES10	0	0	FFECH	00H	R/W

ES31	ES30	INTP2 Valid Edge Selection
0	0	Falling edge
0	1	Rising edge
1	0	Setting prohibited
1	1	Both falling and rising edges

ES21	ES20	INTP1 Valid Edge Selection
0	0	Falling edge
0	1	Rising edge
1	0	Setting prohibited
1	1	Both falling and rising edges

ES11	ES10	INTP0 Valid Edge Selection
0	0	Falling edge
0	1	Rising edge
1	0	Setting prohibited
1	1	Both falling and rising edges

Caution Be sure to set bits 1 to 3 (TMC01 to TMC03) of the 16-bit timer mode control register to 0, 0, 0, and stop the timer operation before setting the valid edge of the INTP0/TI00 pin.

(7) Sampling clock select register (SCS)

This register sets clocks which undergo clock sampling of valid edges to be input to INTP0. When remote controlled reception is carried out using INTP0, digital noise is eliminated with sampling clock.

SCS is set with an 8-bit memory manipulation instruction.

$\overline{\text{RESET}}$ input clears SCS value to 00H.

Figure 8-9. Sampling Clock Select Register Format

Symbol	7	6	5	4	3	2	1	0	Address	After Reset	R/W
SCS	0	0	0	0	0	0	SCS1	SCS0	FF47H	00H	R/W

SCS1	SCS0	INTP0 Sampling Clock Selection	
		MCS = 1	MCS = 0
0	0	fxx/2^N	
0	1	fx/2^7 (39.1 kHz)	fx/2^8 (19.5 kHz)
1	0	fx/2^5 (156.3 kHz)	fx/2^6 (78.1 kHz)
1	1	fx/2^6 (78.1 kHz)	fx/2^7 (39.1 kHz)

Caution fxx/2^N is the clock supplied to the CPU, and fxx/2^5, fxx/2^6, and fxx/2^7 are clocks supplied to peripheral hardware. fxx/2^N is stopped in HALT mode.

Remarks

1. N: Value set in bits 0 to 2 (PCC0 to PCC2) of the processor clock control register (PCC) (N = 0 to 4)

2. fxx: Main system clock frequency (fx or fx/2)

3. fx: Main system clock oscillation frequency

4. MCS: Bit 0 of oscillation mode select register

5. Figures in parentheses apply to operation with fx = 5.0 MHz.

8.5 16-bit Timer/Event Counter Operations

8.5.1 Interval timer operations

Setting the 16-bit timer mode control register (TMC0) and capture/compare control register 0 (CRC0) as shown in Figure 8-10 allows operation as an interval timer. Interrupt requests are generated repeatedly using the count value set in 16-bit capture/compare register 00 (CR00) beforehand as the interval.

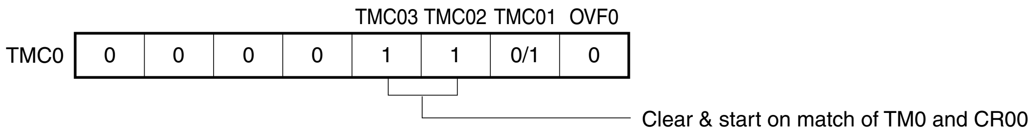
When the count value of the 16-bit timer register (TM0) matches the value set to CR00, counting continues with the TM0 value cleared to 0 and the interrupt request signal (INTTM00) is generated.

Count clock of the 16-bit timer/event counter can be selected with bits 4 to 6 (TCL04 to TCL06) of timer clock select register 0 (TCL0).

For the operation to be performed when the value of the compare register is changed during timer count operation, refer to **8.6 16-bit Timer/Event Counter Operating Precautions (3)**.

Figure 8-10. Control Register Settings for Interval Timer Operation

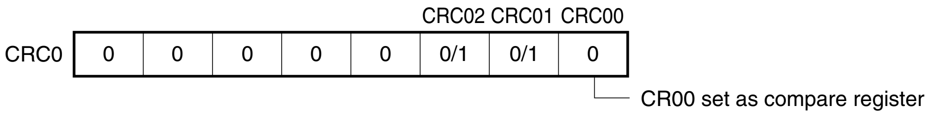
(a) 16-bit timer mode control register (TMC0)



					TMC03	TMC02	TMC01	OVF0
TMC0	0	0	0	0	1	1	0/1	0

Clear & start on match of TM0 and CR00

(b) Capture/compare control register 0 (CRC0)



						CRC02	CRC01	CRC00
CRC0	0	0	0	0	0	0/1	0/1	0

CR00 set as compare register

Remark 0/1: Setting 0 or 1 allows another function to be used simultaneously with the interval timer. See the description of the respective control registers for details.

Figure 8-11. Interval Timer Configuration Diagram

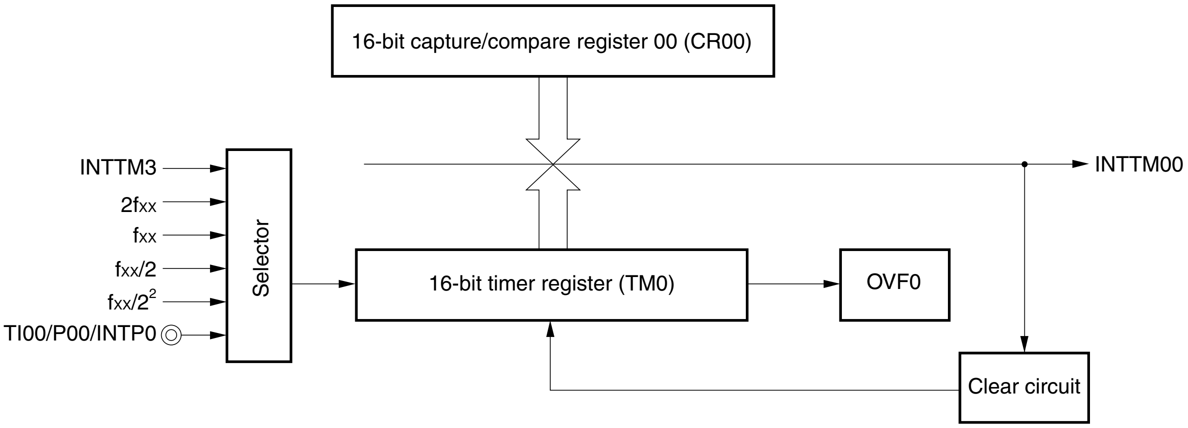
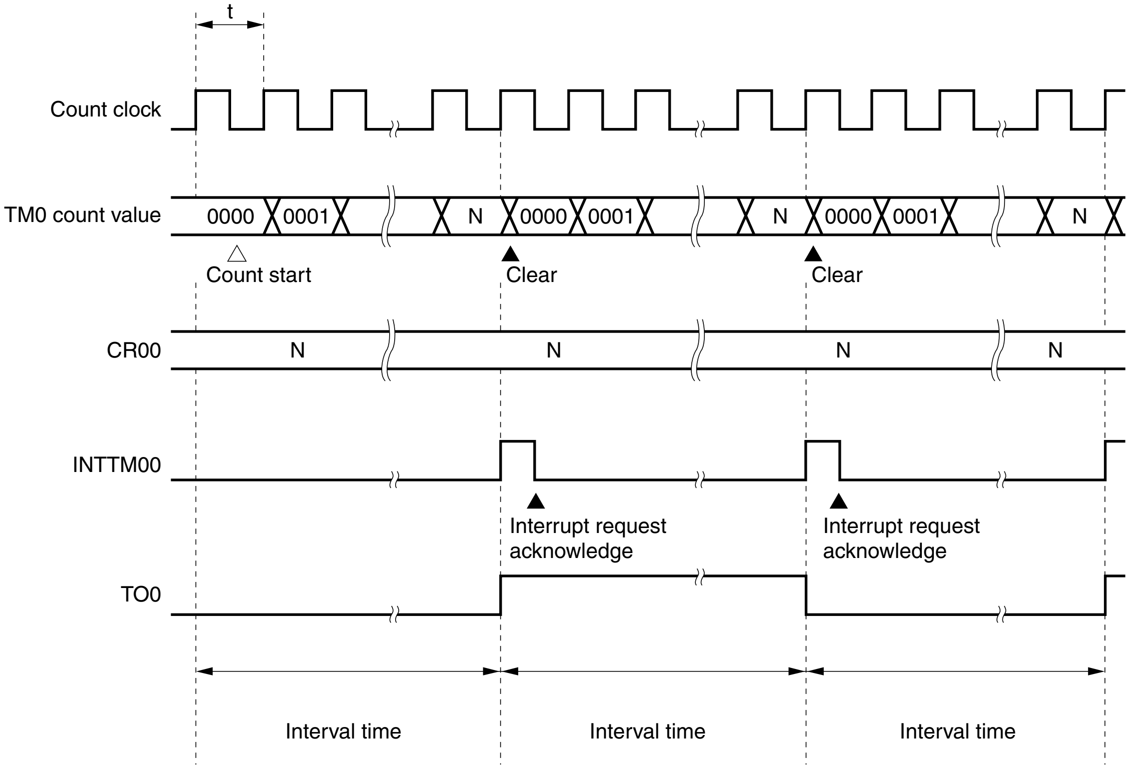


Figure 8-12. Interval Timer Operation Timings



Remark Interval time = (N + 1) × t: N = 0001H to FFFFH

Table 8-6. 16-bit Timer/Event Counter Interval Times

TCL06	TCL05	TCL04	Minimum Interval Time		Maximum Interval Time		Resolution	
			MCS = 1	MCS = 0	MCS = 1	MCS = 0	MCS = 1	MCS = 0
0	0	0	2 × TI00 input cycle		2^{16} × TI00 input cycle		TI00 input edge cycle	
0	0	1	Setting prohibited	2 × 1/fx (400 ns)	Setting prohibited	2^{16} × 1/fx (13.1 ms)	Setting prohibited	1/fx (200 ns)
0	1	0	2 × 1/fx (400 ns)	2^{2} × 1/fx (800 ns)	2^{16} × 1/fx (13.1 ms)	2^{17} × 1/fx (26.2 ms)	1/fx (200 ns)	2 × 1/fx (400 ns)
0	1	1	2^{2} × 1/fx (800 ns)	2^{3} × 1/fx (1.6 μs)	2^{17} × 1/fx (26.2 ms)	2^{18} × 1/fx (52.4 ms)	2 × 1/fx (400 ns)	2^{2} × 1/fx (800 ns)
1	0	0	2^{3} × 1/fx (1.6 μs)	2^{4} × 1/fx (3.2 μs)	2^{18} × 1/fx (52.4 ms)	2^{19} × 1/fx (104.9 ms)	2^{2} × 1/fx (800 ns)	2^{3} × 1/fx (1.6 μs)
1	1	1	2 × watch timer output cycle		2^{16} × watch timer output cycle		Watch timer output edge cycle	
Other than above			Setting prohibited					

Remarks **1.** fx: Main system clock oscillation frequency

2. MCS: Bit 0 of oscillation mode select register

3. Figures in parentheses apply to operation with fx = 5.0 MHz

8.5.2 PWM output operations

Setting the 16-bit timer mode control register (TMC0), capture/compare control register 0 (CRC0), and the 16-bit timer output control register (TOC0) as shown in Figure 8-13 allows operation as PWM output. Pulses with the duty rate determined by the value set in 16-bit capture/compare register 00 (CR00) beforehand are output from the TO0/P30 pin.

Set the active level width of the PWM pulse to the higher 14 bits of CR00. Select the active level with bit 1 (TOC01) of TOC0.

This PWM pulse has a 14-bit resolution. The pulse can be converted to an analog voltage by integrating it with an external low-pass filter (LPF). The PWM pulse is formed by a combination of the basic cycle determined by $2^{8}/\Phi$ and the sub-cycle determined by $2^{14}/\Phi$ so that the time constant of the external LPF can be shortened. Count clock Φ can be selected with bits 4 to 6 (TCL04 to TCL06) of timer clock select register 0 (TCL0).

PWM output enable/disable can be selected with bit 0 (TOE0) of TOC0.

Cautions 1. PWM operation mode should be selected before setting CR00.

2. Be sure to write 0 to bits 0 and 1 of CR00.

3. Do not select PWM operation mode for external clock input from the TI00/P00 pin.

Figure 8-13. Control Register Settings for PWM Output Operation

(a) 16-bit timer mode control register (TMC0)

					TMC03	TMC02	TMC01	OVF0
TMC0	0	0	0	0	0	0	1	0

TMC03, TMC02, TMC01: PWM mode

(b) Capture/compare control register 0 (CRC0)

						CRC02	CRC01	CRC00
CRC0	0	0	0	0	0	0/1	0/1	0

CRC00: CR00 set as compare register

(c) 16-bit timer output control register (TOC0)

		OSPT	OSPE	TOC04	LVS0	LVR0	TOC01	TOE0
TOC0	0	×	×	×	×	×	0/1	1

TOE0: TO0 output enabled

TOC01: Specifies active level

Remarks 1. 0/1: Setting 0 or 1 allows another function to be used simultaneously with PWM output. See the description of the respective control registers for details.

2. ×: don't care

By integrating 14-bit resolution PWM pulses with an external low-pass filter, they can be converted to an analog voltage and used for electronic tuning and D/A converter applications, etc.

The analog output voltage (V_{AN}) used for D/A conversion with the configuration shown in Figure 8-14 is as follows.

$$V_{AN} = V_{REF} \times \frac{\text{Capture/compare register 00 (CR00) value}}{2^{16}}$$

V_{REF}: External switching circuit reference voltage

Figure 8-14. Example of D/A Converter Configuration with PWM Output

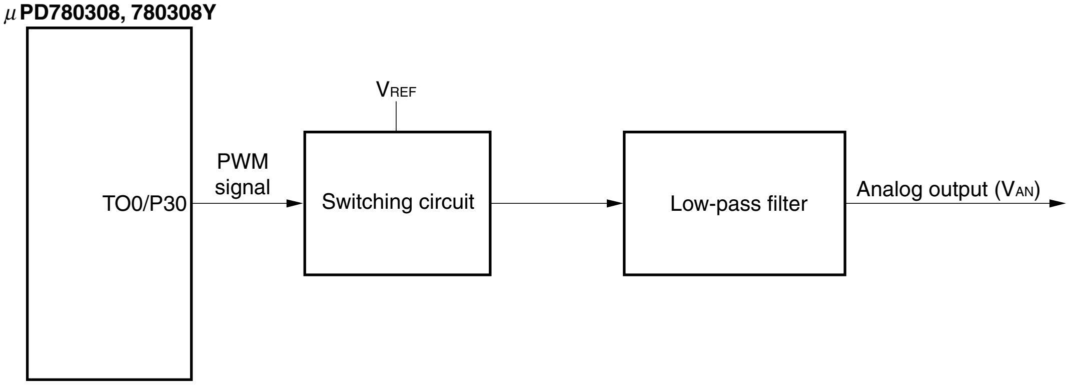
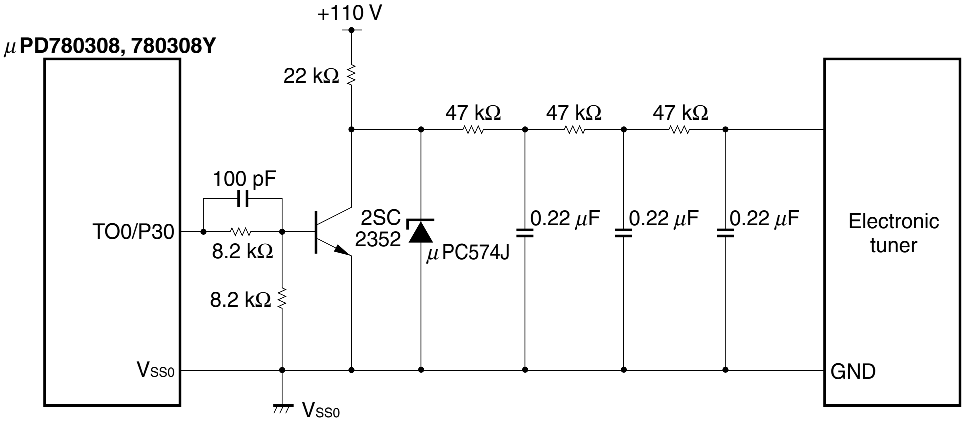


Figure 8-15 shows an example in which PWM output is converted to an analog voltage and used in a voltage synthesizer type TV tuner.

Figure 8-15. TV Tuner Application Circuit Example

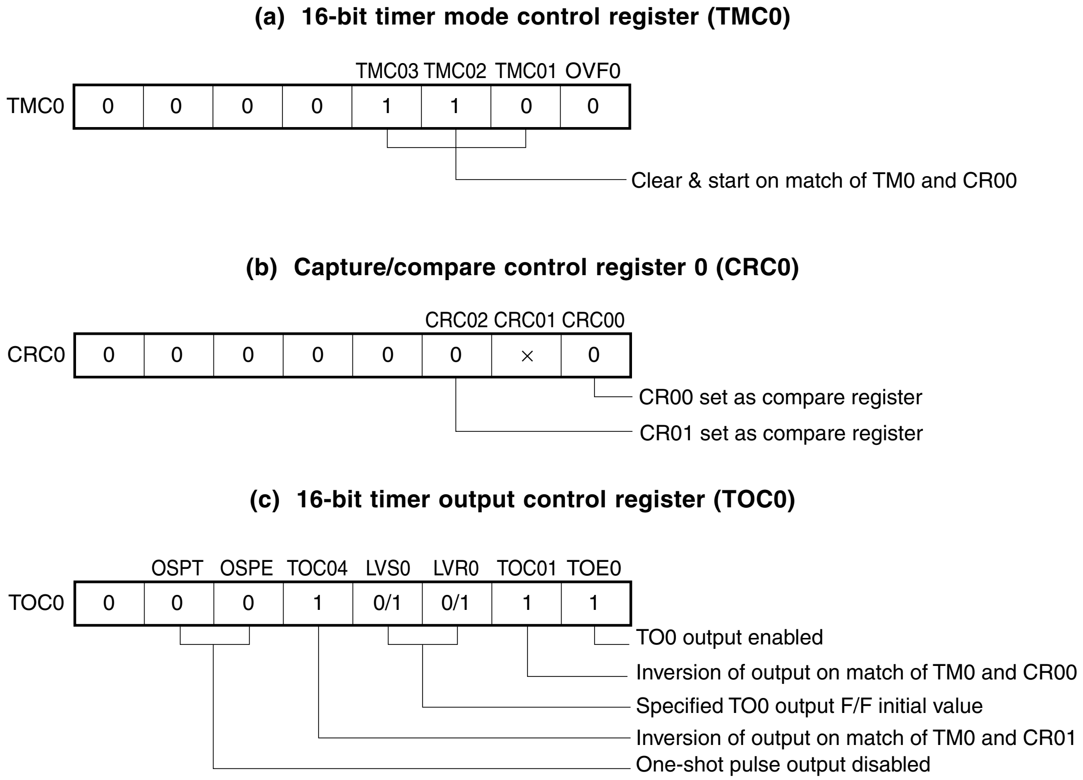


8.5.3 PPG output operations

Setting the 16-bit timer mode control register (TMC0) and capture/compare control register 0 (CRC0) as shown in Figure 8-16 allows operation as PPG (Programmable Pulse Generator) output.

In the PPG output operation, square waves are output from the TO0/P30 pin with the pulse width and the cycle that correspond to the count values set beforehand in 16-bit capture/compare register 01 (CR01) and in 16-bit capture/compare register 00 (CR00), respectively.

Figure 8-16. Control Register Settings for PPG Output Operation



(a) 16-bit timer mode control register (TMC0)

					TMC03	TMC02	TMC01	OVF0
TMC0	0	0	0	0	1	1	0	0

Clear & start on match of TM0 and CR00

(b) Capture/compare control register 0 (CRC0)

						CRC02	CRC01	CRC00
CRC0	0	0	0	0	0	0	×	0

CR00 set as compare register
CR01 set as compare register

(c) 16-bit timer output control register (TOC0)

		OSPT	OSPE	TOC04	LVS0	LVR0	TOC01	TOE0
TOC0	0	0	0	1	0/1	0/1	1	1

TO0 output enabled
Inversion of output on match of TM0 and CR00
Specified TO0 output F/F initial value
Inversion of output on match of TM0 and CR01
One-shot pulse output disabled

Caution Values in the following range should be set in CR00 and CR01:
0000H ≤ CR01 < CR00 ≤ FFFFH

Remark ×: don't care

8.5.4 Pulse width measurement operations

It is possible to measure the pulse width of the signals input to the TI00/P00 pin and TI01/P01 pin using the 16-bit timer register (TM0).

There are two measurement methods: measuring with TM0 used in free-running mode, and measuring by restarting the timer in synchronization with the edge of the signal input to the TI00/P00 pin.

(1) Pulse width measurement with free-running counter and one capture register

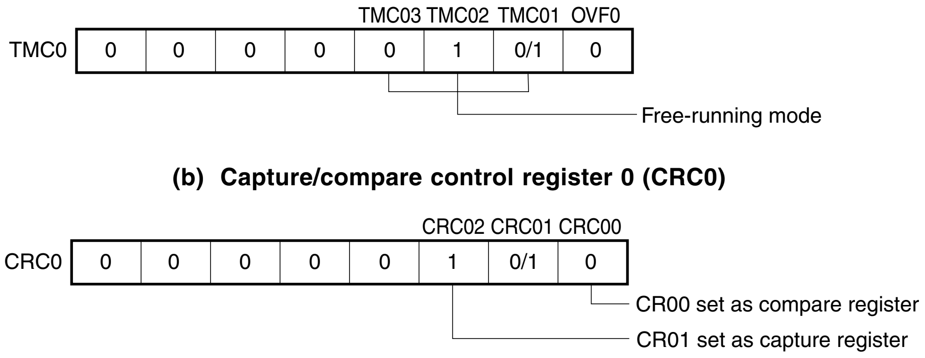
When the 16-bit timer register (TM0) is operated in free-running mode (see register settings in Figure 8-17), and the edge specified by external interrupt mode register 0 (INTM0) is input to the TI00/P00 pin, the value of TM0 is taken into 16-bit capture/compare register 01 (CR01) and an external interrupt request signal (INTP0) is set.

Any of three edge specifications can be selected—rising, falling, or both edges—by means of bits 2 and 3 (ES10 and ES11) of INTM0.

For valid edge detection, sampling is performed at the interval selected by means of the sampling clock select register (SCS), and a capture operation is only performed when a valid level is detected twice, thus eliminating noise with a short pulse width.

Figure 8-17. Control Register Settings for Pulse Width Measurement with Free-Running Counter and One Capture Register

(a) 16-bit timer mode control register (TMC0)



					TMC03	TMC02	TMC01	OVF0
TMC0	0	0	0	0	0	1	0/1	0

(b) Capture/compare control register 0 (CRC0)

						CRC02	CRC01	CRC00
CRC0	0	0	0	0	0	1	0/1	0

Remark 0/1: Setting 0 or 1 allows another function to be used simultaneously with pulse width measurement. See the description of the respective control registers for details.

Figure 8-18. Configuration Diagram for Pulse Width Measurement by Free-Running Counter

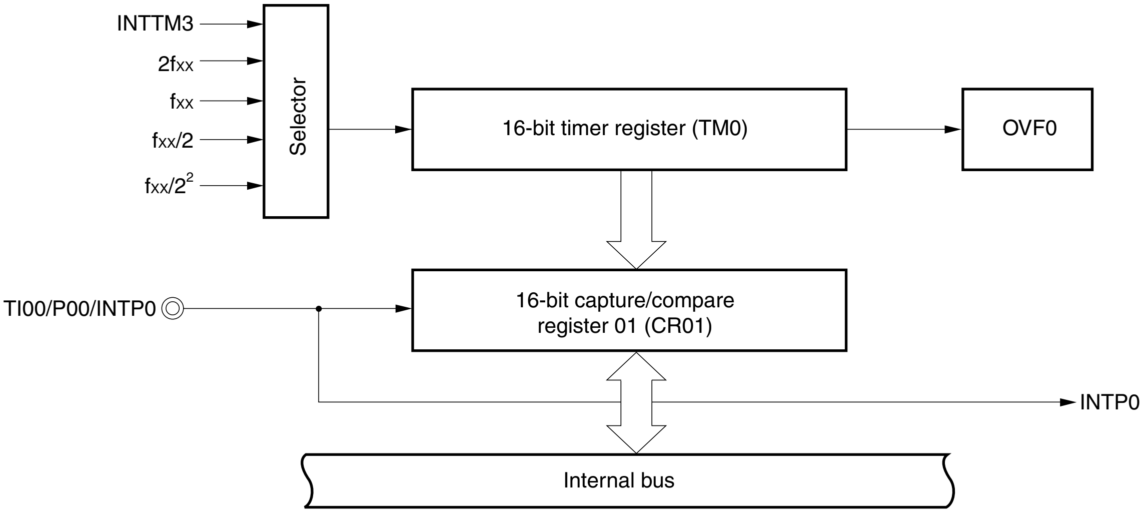
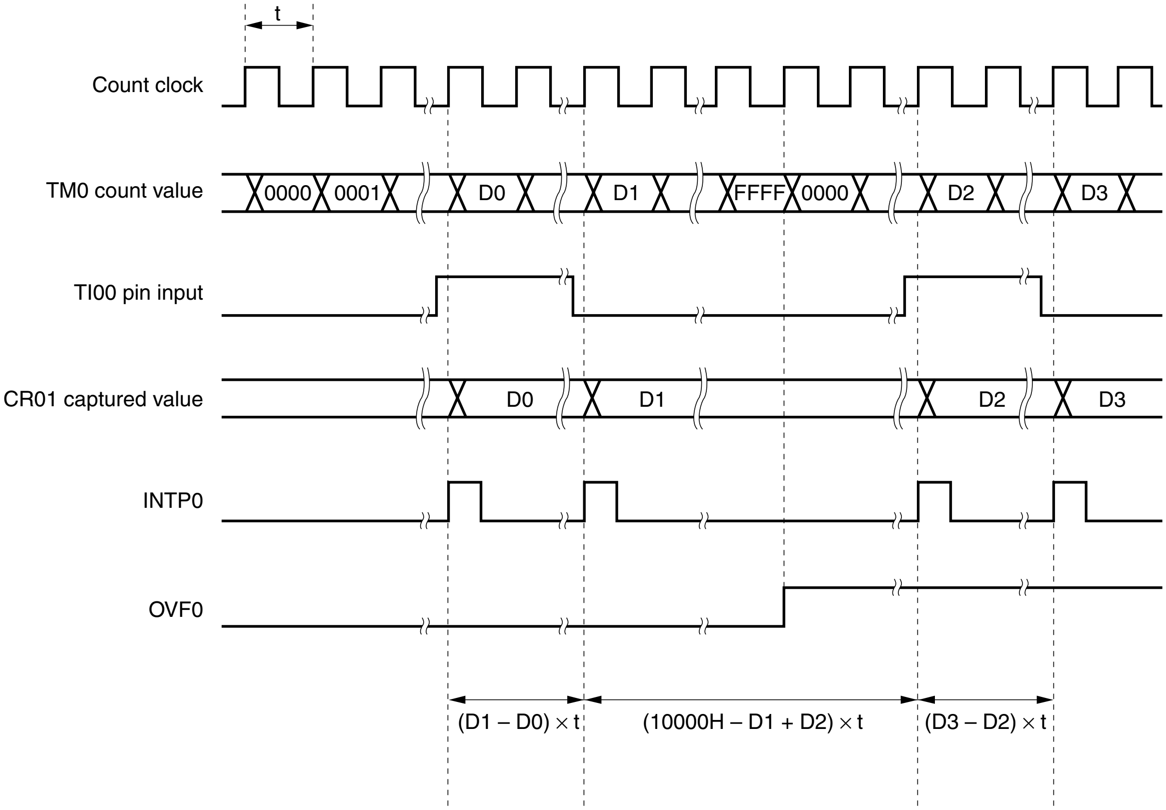

INTTM3
2fxx
fxx
fxx/2
fxx/2^2
Selector
16-bit timer register (TM0)
OVF0
TI00/P00/INTP0
16-bit capture/compare register 01 (CR01)
INTP0
Internal bus

Figure 8-19. Timing of Pulse Width Measurement Operation by Free-Running Counter and One Capture Register (with Both Edges Specified)



(2) Measurement of two pulse widths with free-running counter

When the 16-bit timer register (TM0) is operated in free-running mode (see register settings in Figure 8-20), it is possible to simultaneously measure the pulse widths of the two signals input to the TI00/P00 pin and the TI01/P01 pin.

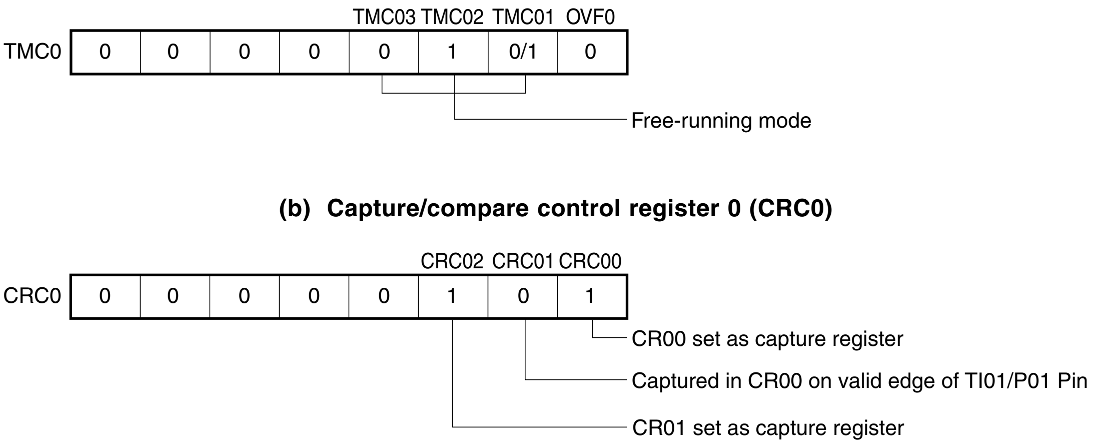
When the edge specified by bits 2 and 3 (ES10 and ES11) of external interrupt mode register 0 (INTM0) is input to the TI00/P00 pin, the value of TM0 is taken into 16-bit capture/compare register 01 (CR01) and an external interrupt request signal (INTP0) is set.

Also, when the edge specified by bits 4 and 5 (ES20 and ES21) of INTM0 is input to the TI01/P01 pin, the value of TM0 is taken into 16-bit capture/compare register 00 (CR00) and an external interrupt request signal (INTP1) is set.

Any of three edge specifications can be selected—rising, falling, or both edges—as the valid edges for the TI00/P00 pin and the TI01/P01 pin by means of bits 2 and 3 (ES10 and ES11) and bits 4 and 5 (ES20 and ES21) of INTM0, respectively.

For TI00/P00 pin valid edge detection, sampling is performed at the interval selected by means of the sampling clock select register (SCS), and a capture operation is only performed when a valid level is detected twice, thus eliminating noise with a short pulse width.

Figure 8-20. Control Register Settings for Two Pulse Width Measurements with Free-Running Counter



(a) 16-bit timer mode control register (TMC0)

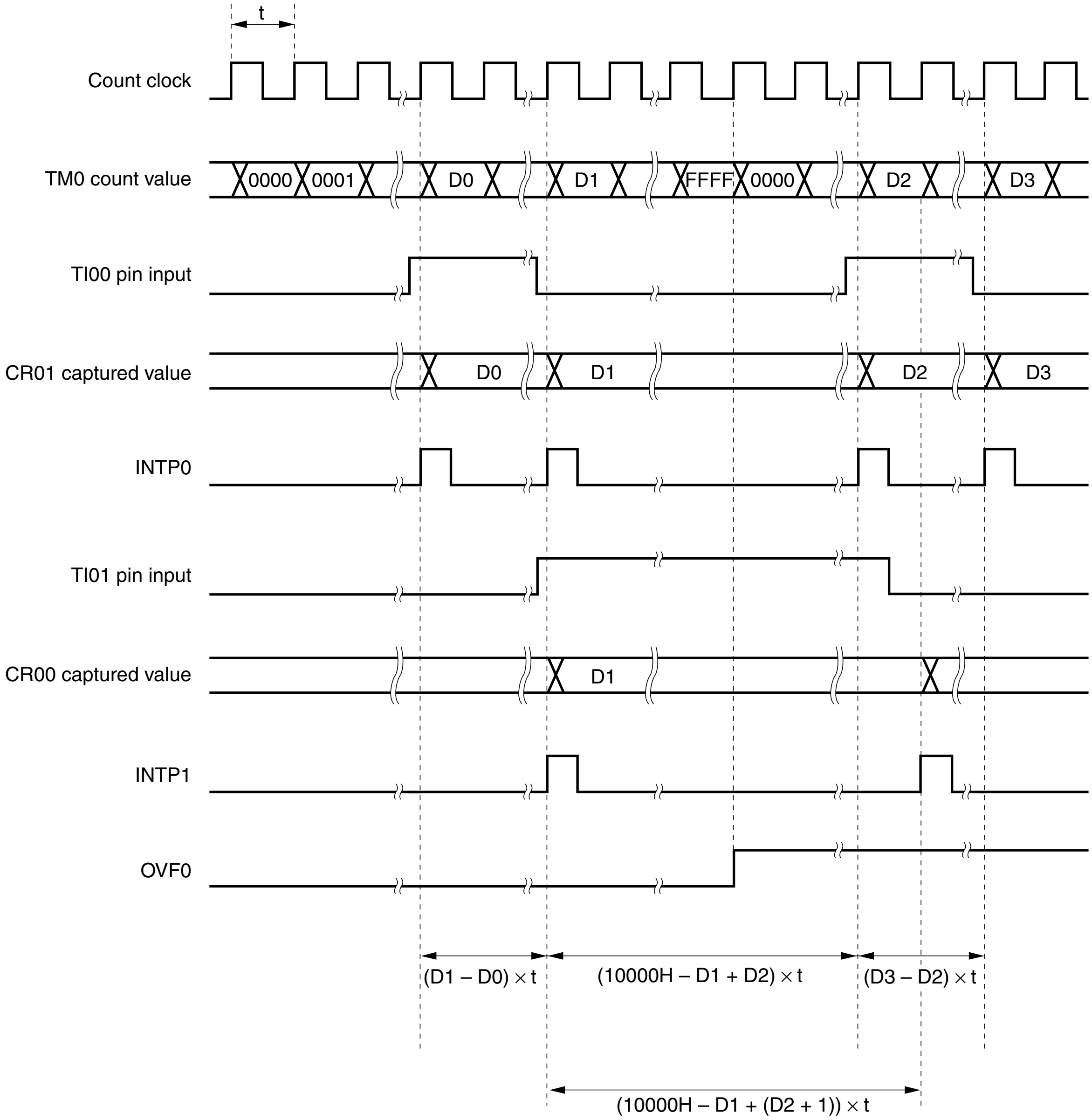
TMC03
TMC02
TMC01
OVF0
TMC0
0
0
0
0
0
1
0/1
0
Free-running mode

(b) Capture/compare control register 0 (CRC0)

CRC02
CRC01
CRC00
CRC0
0
0
0
0
0
1
0
1
CR00 set as capture register
Captured in CR00 on valid edge of TI01/P01 Pin
CR01 set as capture register

Remark 0/1: Setting 0 or 1 allows another function to be used simultaneously with pulse width measurement. See the description of the respective control registers for details.

Figure 8-21. Timing of Pulse Width Measurement Operation with Free-Running Counter (with Both Edges Specified)



(3) Pulse width measurement with free-running counter and two capture registers

When the 16-bit timer register (TM0) is operated in free-running mode (see register settings in Figure 8-22), it is possible to measure the pulse width of the signal input to the TI00/P00 pin.

When the edge specified by bits 2 and 3 (ES10 and ES11) of external interrupt mode register 0 (INTM0) is input to the TI00/P00 pin, the value of TM0 is taken into 16-bit capture/compare register 01 (CR01) and an external interrupt request signal (INTP0) is set.

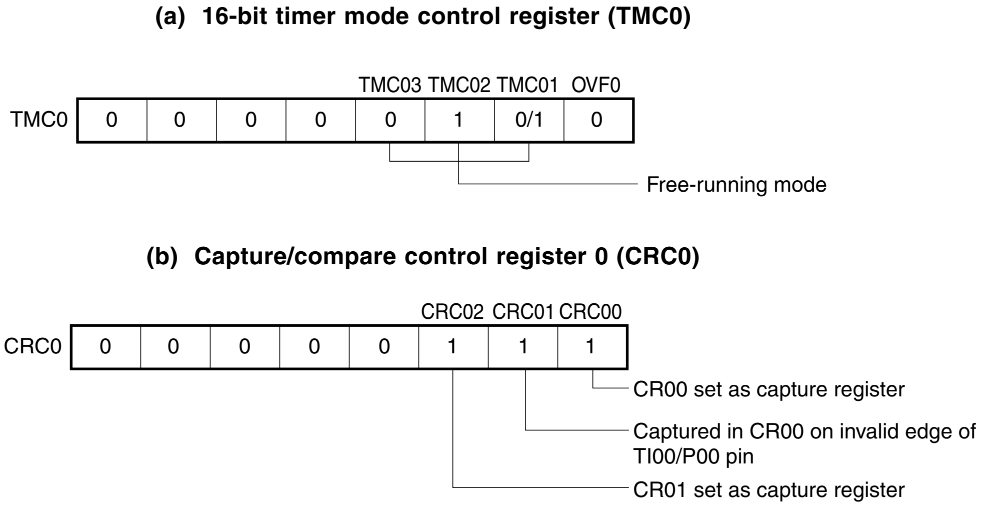
Also, on the inverse edge input of that of the capture operation into CR01, the value of TM0 is taken into 16-bit capture/compare register 00 (CR00).

Either of two edge specifications can be selected—rising or falling—as the valid edges for the TI00/P00 pin by means of bits 2 and 3 (ES10 and ES11) of INTM0.

For TI00/P00 pin valid edge detection, sampling is performed at the interval selected by means of the sampling clock select register (SCS), and a capture operation is only performed when a valid level is detected twice, thus eliminating noise with a short pulse width.

Caution If the valid edge of the TI00/P00 pin is specified to be both rising and falling edges, 16-bit capture/compare register 00 (CR00) cannot perform the capture operation.

Figure 8-22. Control Register Settings for Pulse Width Measurement with Free-Running Counter and Two Capture Registers



(a) 16-bit timer mode control register (TMC0)

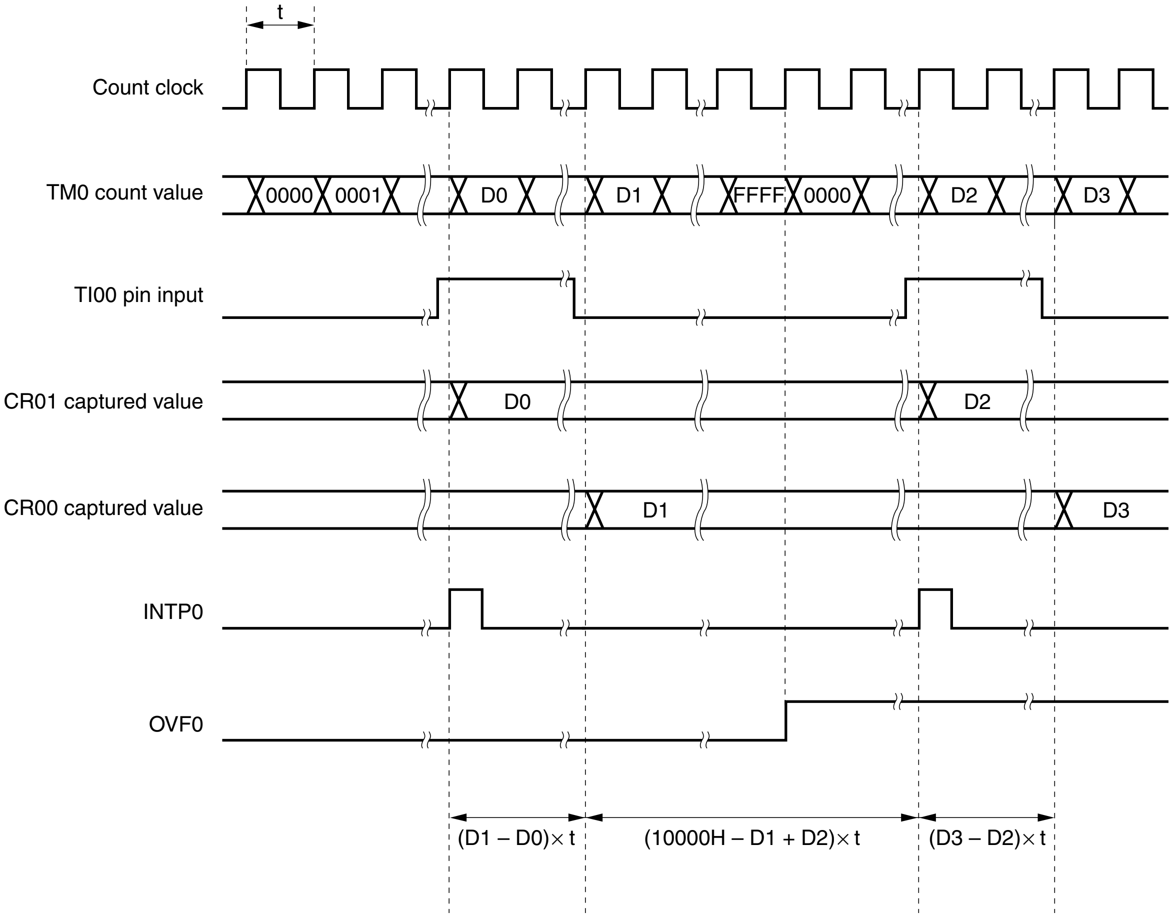
TMC03 TMC02 TMC01 OVF0
TMC0
0 0 0 0 0 1 0/1 0
Free-running mode

(b) Capture/compare control register 0 (CRC0)

CRC02 CRC01 CRC00
CRC0
0 0 0 0 0 1 1 1
CR00 set as capture register
Captured in CR00 on invalid edge of TI00/P00 pin
CR01 set as capture register

Remark 0/1: Setting 0 or 1 allows another function to be used simultaneously with pulse width measurement. See the description of the respective control registers for details.

Figure 8-23. Timing of Pulse Width Measurement Operation by Free-Running Counter and Two Capture Registers (with Rising Edge Specified)



(4) Pulse width measurement by means of restart

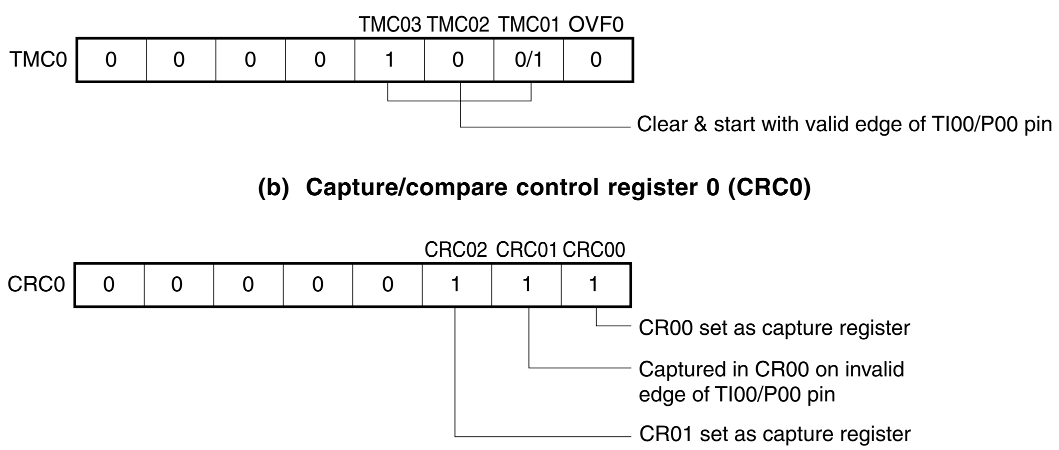
When input of a valid edge to the TI00/P00 pin is detected, the count value of the 16-bit timer register (TM0) is taken into 16-bit capture/compare register 01 (CR01), and then the pulse width of the signal input to the TI00/P00 pin is measured by clearing TM0 and restarting the count (see register settings in Figure 8-24).

The edge specification can be selected from two types, rising and falling edges by INTM0 bits 2 and 3 (ES10 and ES11).

In a valid edge detection, the sampling is performed by a cycle selected by the sampling clock select register (SCS), and a capture operation is only performed when a valid level is detected twice, thus eliminating noise with a short pulse width.

Caution If the valid edge of the TI00/P00 pin is specified to be both rising and falling edges, 16-bit capture/compare register 00 (CR00) cannot perform the capture operation.

Figure 8-24. Control Register Settings for Pulse Width Measurement by Means of Restart



(a) 16-bit timer mode control register (TMC0)

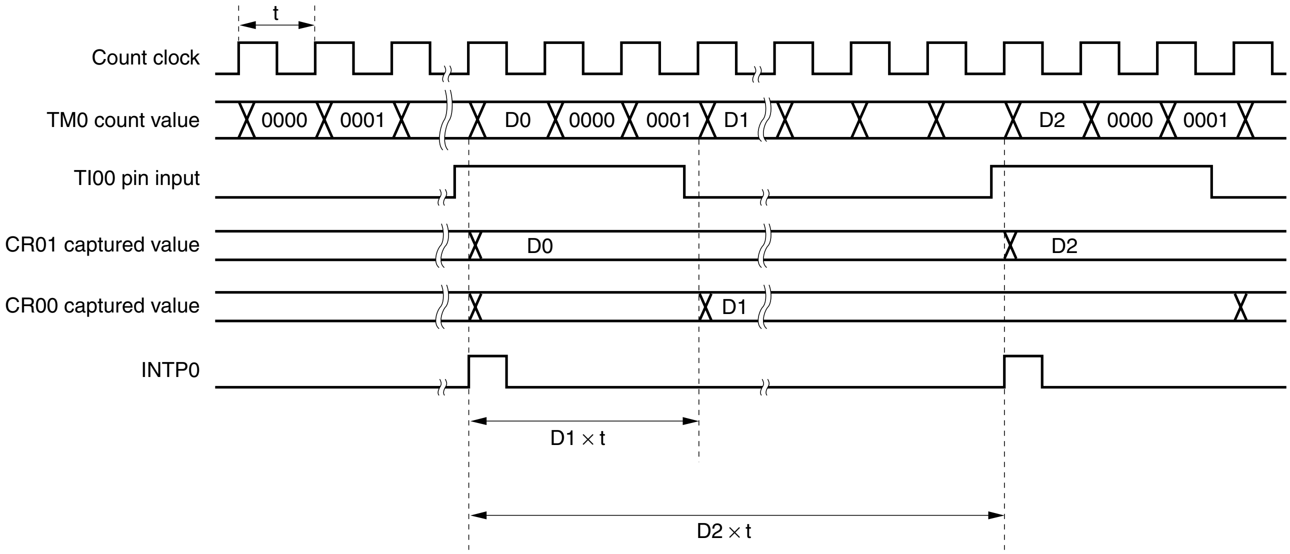
TMC03 TMC02 TMC01 OVF0
TMC0
0 0 0 0 1 0 0/1 0
Clear & start with valid edge of TI00/P00 pin

(b) Capture/compare control register 0 (CRC0)

CRC02 CRC01 CRC00
CRC0
0 0 0 0 0 1 1 1
CR00 set as capture register
Captured in CR00 on invalid edge of TI00/P00 pin
CR01 set as capture register

Remark 0/1: Setting 0 or 1 allows another function to be used simultaneously with pulse width measurement. See the description of the respective control registers for details.

Figure 8-25. Timing of Pulse Width Measurement Operation by Means of Restart (with Rising Edge Specified)



8.5.5 External event counter operation

The external event counter counts the number of external clock pulses to be input to the TI00/P00 pin with the 16-bit timer register (TM0).

TM0 is incremented each time the valid edge specified with external interrupt mode register 0 (INTM0) is input.

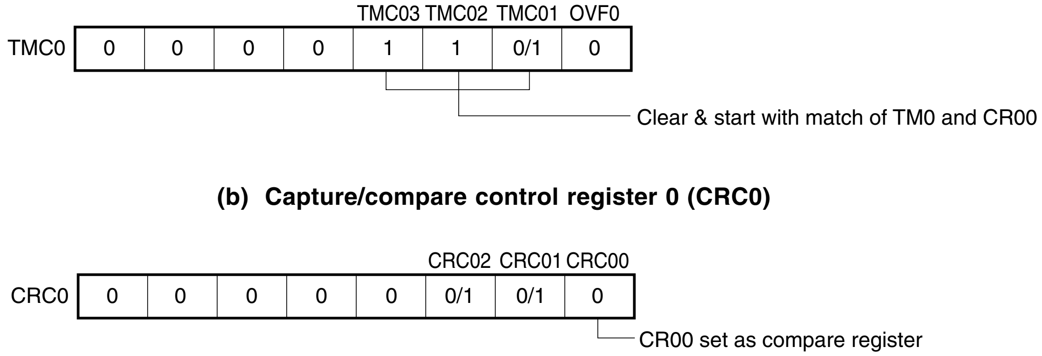
When the TM0 counted value matches the 16-bit capture/compare register 00 (CR00) value, TM0 is cleared to 0 and the interrupt request signal (INTTM00) is generated.

Set a value other than 0000H to CR00 (the 1-pulse count operation cannot be performed).

The rising edge, the falling edge or both edges can be selected with bits 2 and 3 (ES10 and ES11) of INTM0.

Because operation is carried out only after the valid edge is detected twice by sampling at the interval selected with the sampling clock select register (SCS), noise with short pulse widths can be eliminated.

Figure 8-26. Control Register Settings in External Event Counter Mode



(a) 16-bit timer mode control register (TMC0)

					TMC03	TMC02	TMC01	OVF0
TMC0	0	0	0	0	1	1	0/1	0

Clear & start with match of TM0 and CR00

(b) Capture/compare control register 0 (CRC0)

						CRC02	CRC01	CRC00
CRC0	0	0	0	0	0	0/1	0/1	0

CR00 set as compare register

Remark 0/1: Setting 0 or 1 allows another function to be used simultaneously with the external event counter. See the description of the respective control registers for details.

Figure 8-27. External Event Counter Configuration Diagram

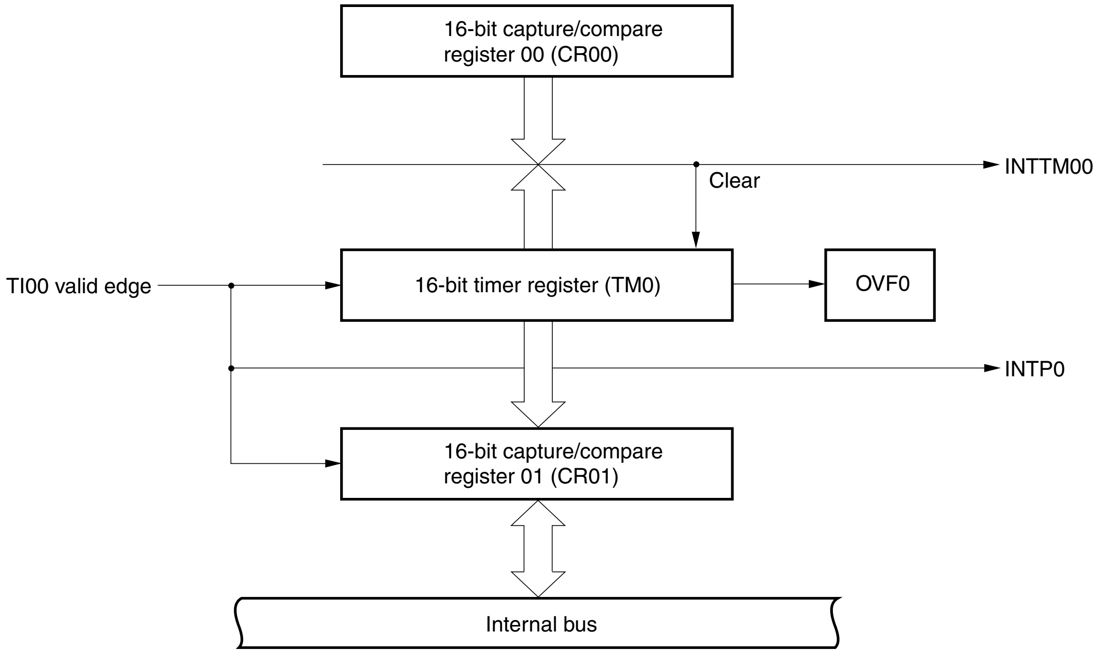
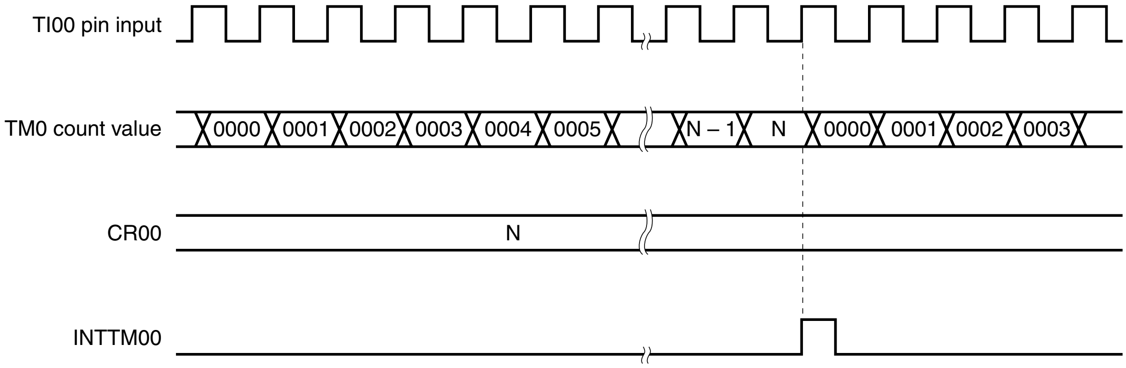

16-bit capture/compare register 00 (CR00)
INTTM00
Clear
TI00 valid edge
16-bit timer register (TM0)
OVF0
INTP0
16-bit capture/compare register 01 (CR01)
Internal bus

Figure 8-28. External Event Counter Operation Timings (with Rising Edge Specified)



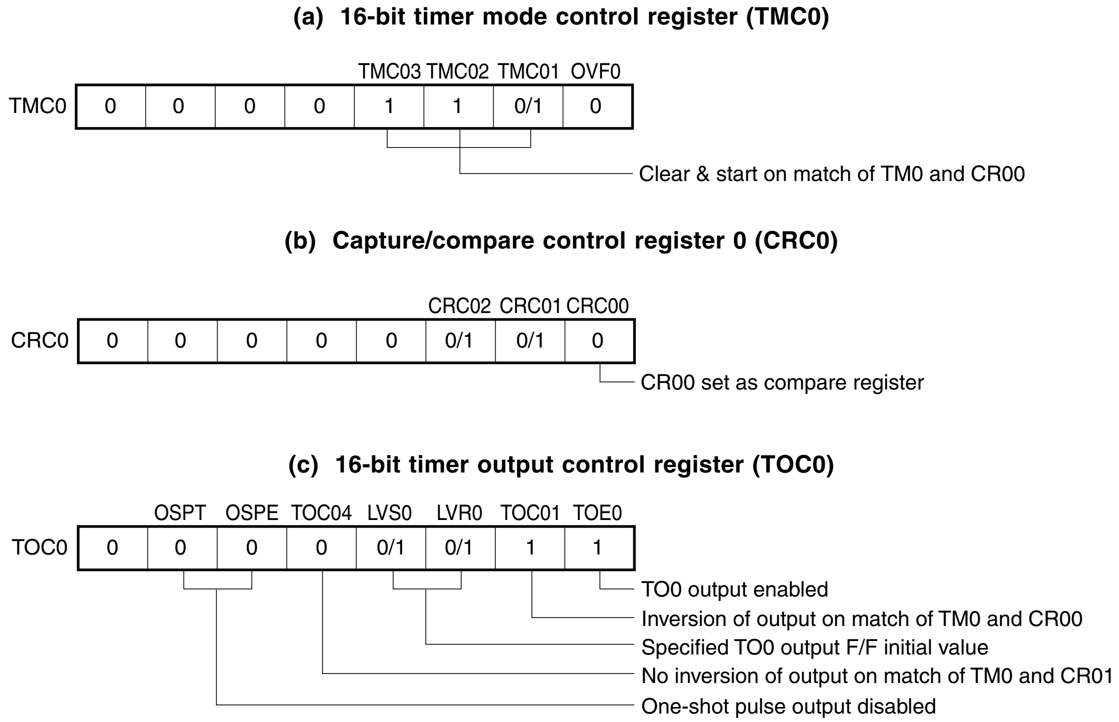
Caution When reading the external event counter count value, TM0 should be read.

8.5.6 Square-wave output operation

A square wave of any frequency is output at the interval specified by the count value set in advance to 16-bit capture/compare register 00 (CR00).

The TO0/P30 pin output status is reversed at intervals of the count value preset to CR00 by setting bit 0 (TOE0) and bit 1 (TOC01) of the 16-bit timer output control register (TOC0) to 1. This enables a square wave with any selected frequency to be output.

Figure 8-29. Control Register Settings in Square-Wave Output Mode



(a) 16-bit timer mode control register (TMC0)

					TMC03	TMC02	TMC01	OVF0
TMC0	0	0	0	0	1	1	0/1	0

Clear & start on match of TM0 and CR00

(b) Capture/compare control register 0 (CRC0)

						CRC02	CRC01	CRC00
CRC0	0	0	0	0	0	0/1	0/1	0

CR00 set as compare register

(c) 16-bit timer output control register (TOC0)

		OSPT	OSPE	TOC04	LVS0	LVR0	TOC01	TOE0
TOC0	0	0	0	0	0/1	0/1	1	1

TO0 output enabled
Inversion of output on match of TM0 and CR00
Specified TO0 output F/F initial value
No inversion of output on match of TM0 and CR01
One-shot pulse output disabled

Remark 0/1: Setting 0 or 1 allows another function to be used simultaneously with square-wave output. See the description of the respective control registers for details.

Figure 8-30. Square-Wave Output Operation Timing

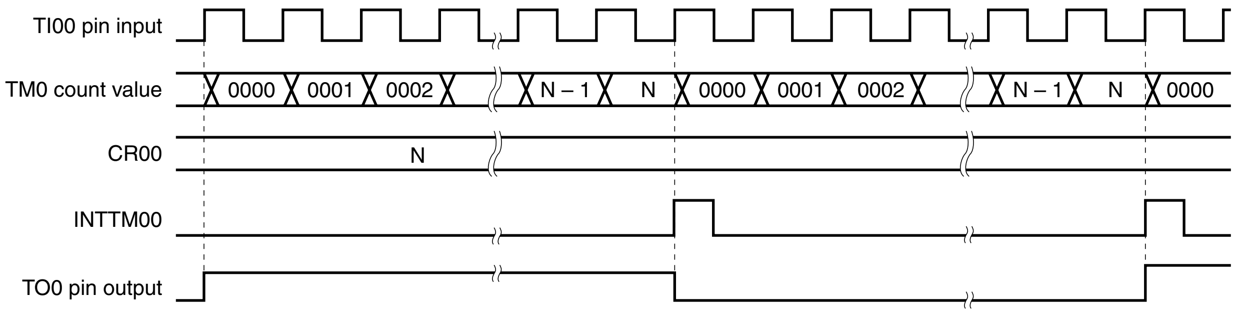


Table 8-7. 16-bit Timer/Event Count Square-Wave Output Ranges

Minimum Pulse Width		Maximum Pulse Width		Resolution	
MCS = 1	MCS = 0	MCS = 1	MCS = 0	MCS = 1	MCS = 0
2 × TI00 input cycle		2^{16} × TI00 input cycle		TI00 input edge cycle	
—	2 × 1/fx (400 ns)	—	2^{16} × 1/fx (13.1 ms)	—	1/fx (200 ns)
2 × 1/fx (400 ns)	2^{2} × 1/fx (800 ns)	2^{16} × 1/fx (13.1 ms)	2^{17} × 1/fx (26.2 ms)	1/fx (200 ns)	2 × 1/fx (400 ns)
2^{2} × 1/fx (800 ns)	2^{3} × 1/fx (1.6 μs)	2^{17} × 1/fx (26.2 ms)	2^{18} × 1/fx (52.4 ms)	2 × 1/fx (400 ns)	2^{2} × 1/fx (800 ns)
2^{3} × 1/fx (1.6 μs)	2^{4} × 1/fx (3.2 μs)	2^{18} × 1/fx (52.4 ms)	2^{19} × 1/fx (104.9 ms)	2^{2} × 1/fx (800 ns)	2^{3} × 1/fx (1.6 μs)
2 × watch timer output cycle		2^{16} × watch timer output cycle		Watch timer output edge cycle	

Remarks 1. fx: Main system clock oscillation frequency

2. MCS: Oscillation mode select register bit 0

3. Values in parentheses when operated at fx = 5.0 MHz

8.5.7 One-shot pulse output operation

It is possible to output one-shot pulses synchronized with a software trigger or an external trigger (TI00/P00 pin input).

(1) One-shot pulse output using software trigger

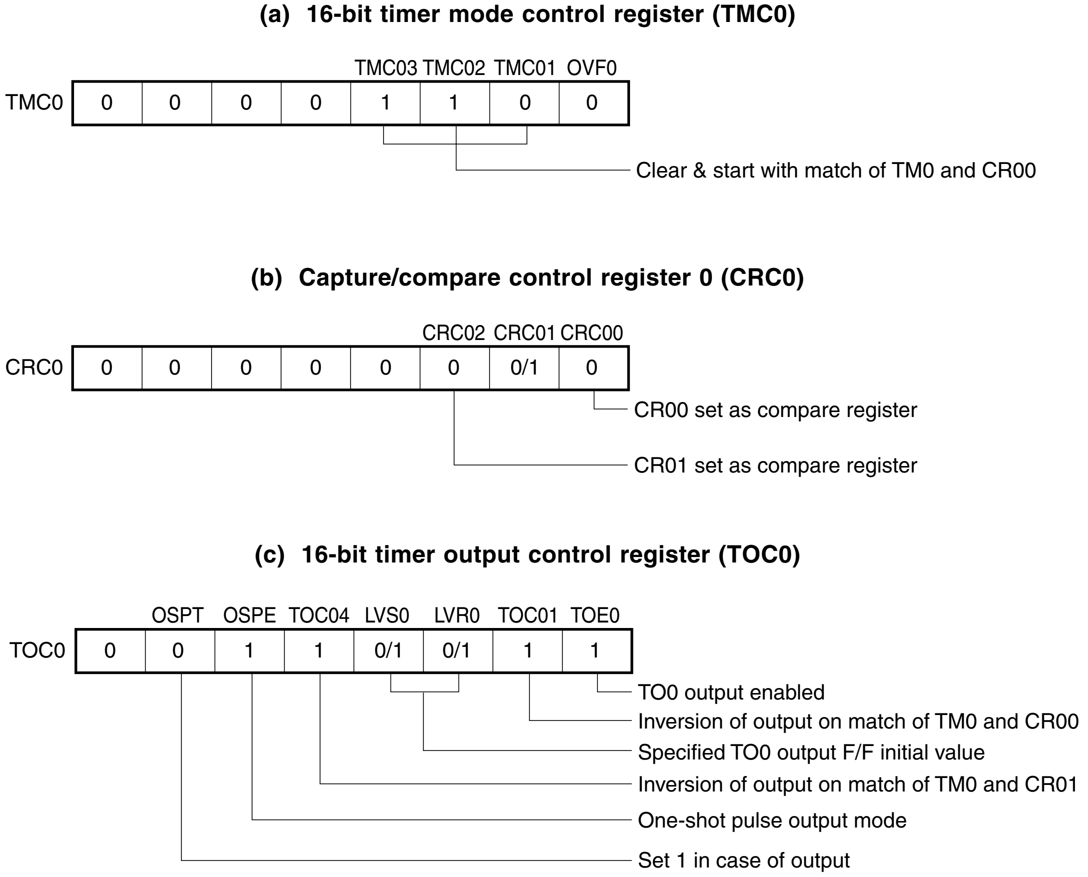
If the 16-bit timer mode control register (TMC0), capture/compare control register 0 (CRC0), and the 16-bit timer output control register (TOC0) are set as shown in Figure 8-31, and 1 is set in bit 6 (OSPT) of TOC0 by software, a one-shot pulse is output from the TO0/P30 pin.

By setting 1 in OSPT, the 16-bit timer/event counter is cleared and started, and output is activated by the count value set beforehand in 16-bit capture/compare register 01 (CR01). Thereafter, output is inactivated by the count value set beforehand in 16-bit capture/compare register 00 (CR00).

TM0 continues to operate after one-shot pulse is output. To stop TM0, 00H must be set to TMC0.

Caution When outputting one-shot pulse, do not set 1 in OSPT. When outputting one-shot pulse again, execute after the INTTM00, or interrupt match signal with CR00, is generated.

Figure 8-31. Control Register Settings for One-Shot Pulse Output Operation Using Software Trigger



(a) 16-bit timer mode control register (TMC0)

(b) Capture/compare control register 0 (CRC0)

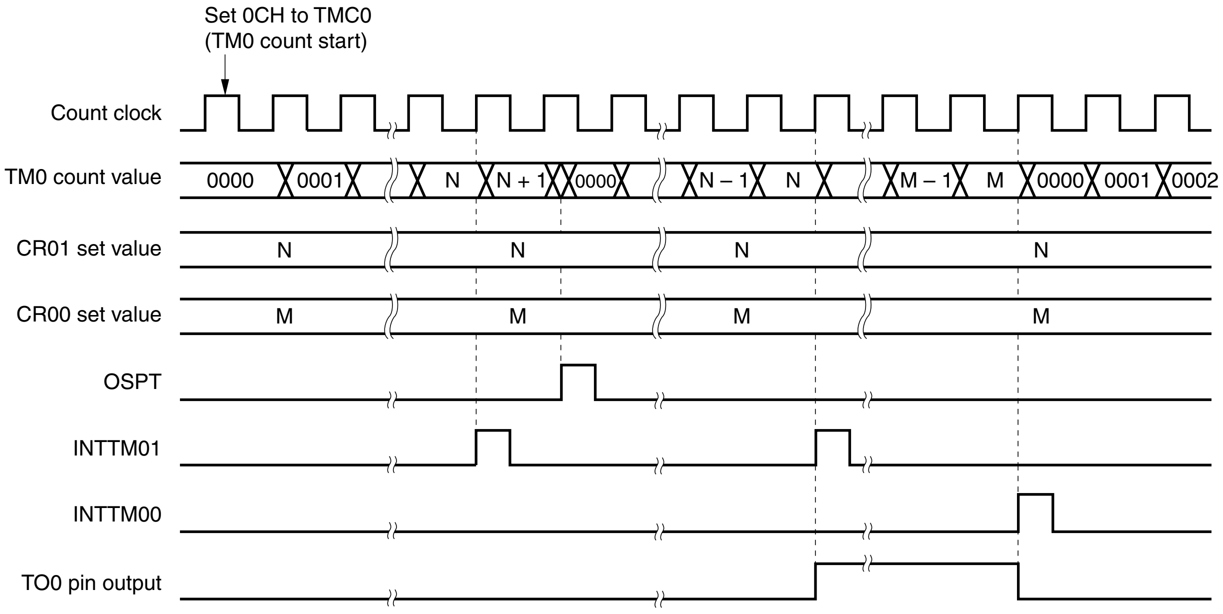
(c) 16-bit timer output control register (TOC0)

TMC03 TMC02 TMC01 OVF0
TMC0 0 0 0 0 1 1 0 0
Clear & start with match of TM0 and CR00
CRC02 CRC01 CRC00
CRC0 0 0 0 0 0 0 0/1 0
CR00 set as compare register
CR01 set as compare register
OSPT OSPE TOC04 LVS0 LVR0 TOC01 TOE0
TOC0 0 0 1 1 0/1 0/1 1 1
TO0 output enabled
Inversion of output on match of TM0 and CR00
Specified TO0 output F/F initial value
Inversion of output on match of TM0 and CR01
One-shot pulse output mode
Set 1 in case of output

Caution Values in the following range should be set in CR00 and CR01.
0000H ≤ CR01 < CR00 ≤ FFFFH

Remark 0/1: Setting 0 or 1 allows another function to be used simultaneously with one-shot pulse output. See the description of the respective control registers for details.

Figure 8-32. Timing of One-Shot Pulse Output Operation Using Software Trigger



Caution The 16-bit timer register starts operation at the moment a value other than 0, 0, 0 (operation stop mode) is set to TMC01 to TMC03, respectively.

(2) One-shot pulse output using external trigger

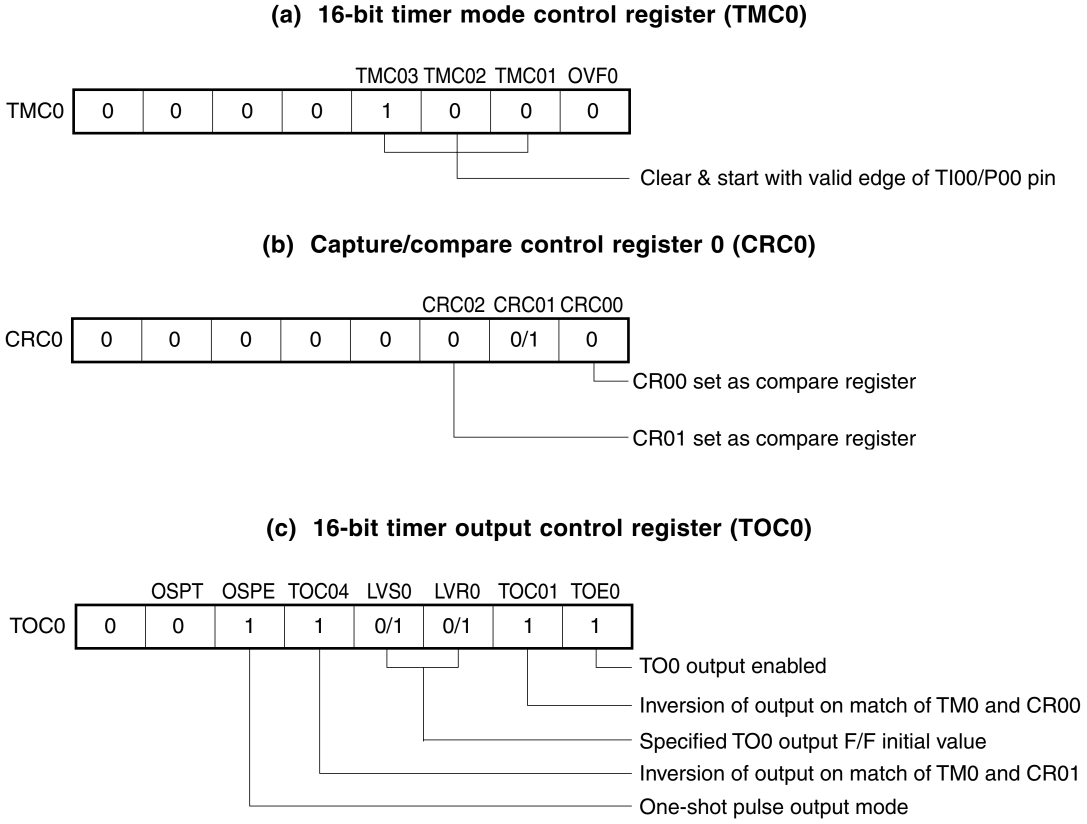
If the 16-bit timer mode control register (TMC0), capture/compare control register 0 (CRC0), and the 16-bit timer output control register (TOC0) are set as shown in Figure 8-33, a one-shot pulse is output from the TO0/P30 pin with a TI00/P00 valid edge as an external trigger.

Any of three edge specifications can be selected—rising, falling, or both edges — as the valid edges for the TI00/P00 pin by means of bits 2 and 3 (ES10 and ES11) of external interrupt mode register 0 (INTM0).

When a valid edge is input to the TI00/P00 pin, the 16-bit timer/event counter is cleared and started, and output is activated by the count values set beforehand in 16-bit capture/compare register 01 (CR01). Thereafter, output is inactivated by the count value set beforehand in 16-bit capture/compare register 00 (CR00).

Caution When outputting one-shot pulses, external trigger is ignored if generated again.

Figure 8-33. Control Register Settings for One-Shot Pulse Output Operation Using External Trigger



(a) 16-bit timer mode control register (TMC0)

					TMC03	TMC02	TMC01	OVF0
TMC0	0	0	0	0	1	0	0	0

Clear & start with valid edge of TI00/P00 pin

(b) Capture/compare control register 0 (CRC0)

						CRC02	CRC01	CRC00
CRC0	0	0	0	0	0	0	0/1	0

CR00 set as compare register
CR01 set as compare register

(c) 16-bit timer output control register (TOC0)

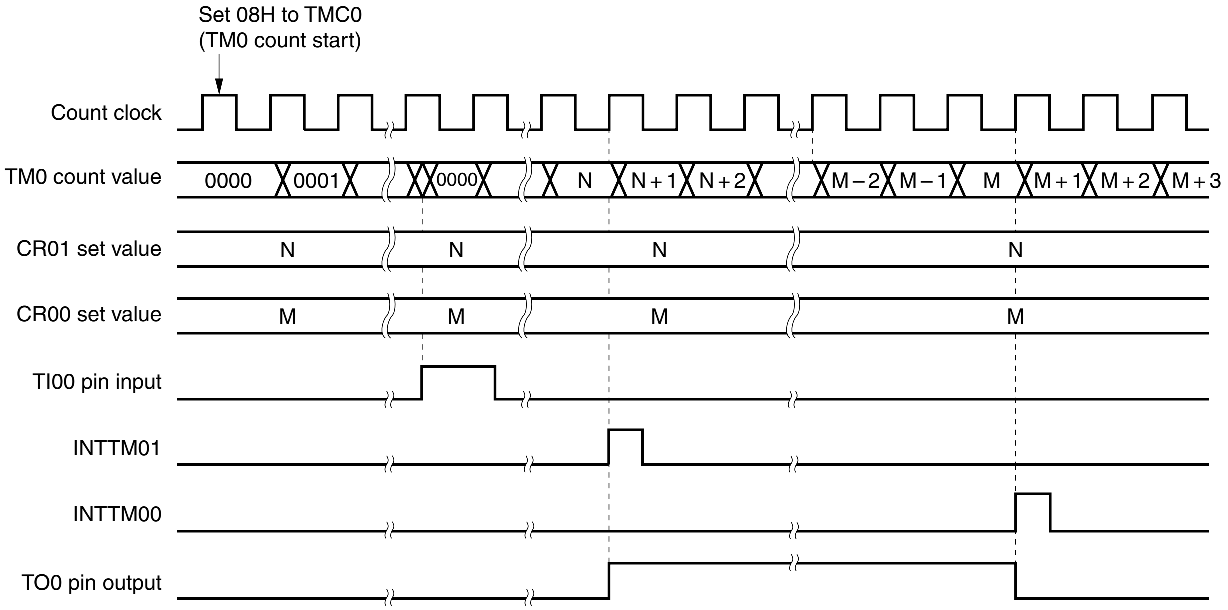
		OSPT	OSPE	TOC04	LVS0	LVR0	TOC01	TOE0
TOC0	0	0	1	1	0/1	0/1	1	1

TO0 output enabled
Inversion of output on match of TM0 and CR00
Specified TO0 output F/F initial value
Inversion of output on match of TM0 and CR01
One-shot pulse output mode

Caution Values in the following range should be set in CR00 and CR01.
0000H ≤ CR01 < CR00 ≤ FFFFH

Remark 0/1: Setting 0 or 1 allows another function to be used simultaneously with one-shot pulse output. See the description of the respective control registers for details.

Figure 8-34. Timing of One-Shot Pulse Output Operation Using External Trigger (with Rising Edge Specified)



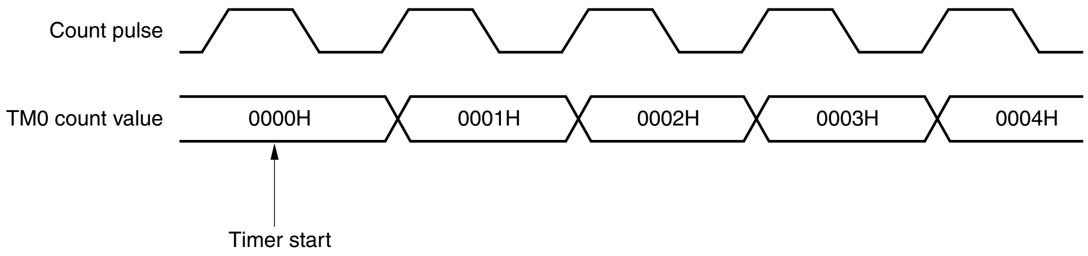
Caution The 16-bit timer register starts operation at the moment a value other than 0, 0, 0 (operation stop mode) is set to TMC01 to TMC03, respectively.

8.6 16-bit Timer/Event Counter Operating Precautions

(1) Timer start errors

An error with a maximum of one clock may occur concerning the time required for a match signal to be generated after timer start. This is because the 16-bit timer register (TM0) is started asynchronously with the count pulse.

Figure 8-35. 16-bit Timer Register Start Timing



(2) 16-bit compare register setting

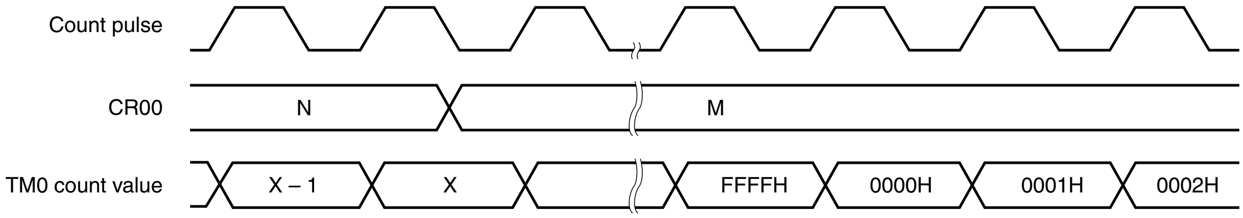
Set a value other than 0000H to 16-bit capture/compare register 00 (CR00).

Thus, when using the 16-bit capture/compare register as event counter, one-pulse count operation cannot be carried out.

(3) Operation after compare register change during timer count operation

If the value after 16-bit capture/compare register 00 (CR00) is changed is smaller than that of the 16-bit timer register (TM0), TM0 continues counting, overflows and then restarts counting from 0. Thus, if the value (M) after CR00 change is smaller than that (N) before change, it is necessary to restart the timer after changing CR00.

Figure 8-36. Timings After Change of Compare Register During Timer Count Operation

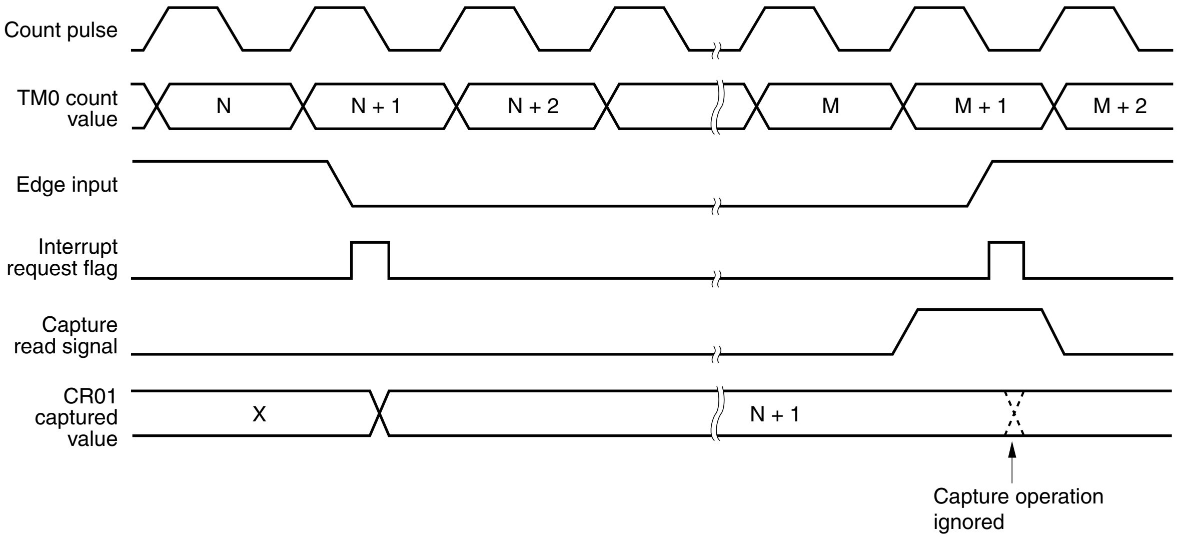


Remark N > X > M

(4) Capture register data retention timings

If the valid edge of the TI00/P00 pin is input during 16-bit capture/compare register 01 (CR01) read, CR01 holds data without carrying out capture operation. However, the interrupt request flag (PIF0) is set upon detection of the valid edge.

Figure 8-37. Capture Register Data Retention Timing



(5) Valid edge setting

Set the valid edge of the TI00/INTP0 pin after setting bits 1 to 3 (TMC01 to TMC03) of the 16-bit timer mode control register to 0, 0 and 0, respectively, and then stopping timer operation. Valid edge setting is carried out with bits 2 and 3 (ES10 and ES11) of external interrupt mode register 0.

(6) Re-trigger of one-shot pulse

(a) One-shot pulse output using software

When outputting one-shot pulse, do not set 1 in bit 6 (OSPT) of 16-bit timer output control register (TOC0). When outputting one-shot pulse again, execute it after the INTTM00, or interrupt match signal with 16-bit capture/compare register 00 (CR00), is generated.

(b) One-shot pulse output using external trigger

When outputting one-shot pulses, external trigger is ignored if generated again.

(7) Operation of OVF0 flag

OVF0 flag is set to 1 in the following case.

The clear & start mode on match between TM0 and CR00 is selected.

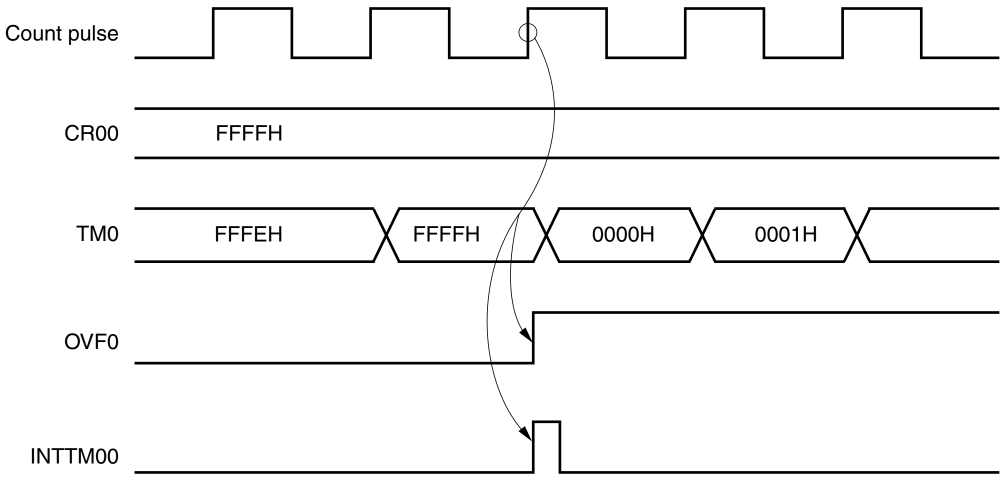
↓

CR00 is set to FFFFH.

↓

When TM0 is counted up from FFFFH to 0000H.

Figure 8-38. Operation Timing of OVF0 Flag



CHAPTER 9 8-BIT TIMER/EVENT COUNTER

9.1 8-bit Timer/Event Counter Functions

For the 8-bit timer/event counter, two modes are available. One is a mode for two-channel 8-bit timer/event counter to be used separately (the 8-bit timer/event counter mode) and the other is a mode for the 8-bit timer/event counter to be used as 16-bit timer/event counter (the 16-bit timer/event counter mode).

9.1.1 8-bit timer/event counter mode

8-bit timer/event counters 1 and 2 (TM1 and TM2) have the following functions.

- Interval timer
- External event counter
- Square-wave output

(1) 8-bit interval timer

Interrupt requests are generated at the preset time intervals.

Table 9-1. 8-bit Timer/Event Counter Interval Times

Minimum Interval Time		Maximum Interval Time		Resolution	
MCS = 1	MCS = 0	MCS = 1	MCS = 0	MCS = 1	MCS = 0
$2 \times 1/f_X$ (400 ns)	$2^2 \times 1/f_X$ (800 ns)	$2^9 \times 1/f_X$ (102.4 μs)	$2^{10} \times 1/f_X$ (204.8 μs)	$2 \times 1/f_X$ (400 ns)	$2^2 \times 1/f_X$ (800 ns)
$2^2 \times 1/f_X$ (800 ns)	$2^3 \times 1/f_X$ (1.6 μs)	$2^{10} \times 1/f_X$ (204.8 μs)	$2^{11} \times 1/f_X$ (409.6 μs)	$2^2 \times 1/f_X$ (800 ns)	$2^3 \times 1/f_X$ (1.6 μs)
$2^3 \times 1/f_X$ (1.6 μs)	$2^4 \times 1/f_X$ (3.2 μs)	$2^{11} \times 1/f_X$ (409.6 μs)	$2^{12} \times 1/f_X$ (819.2 μs)	$2^3 \times 1/f_X$ (1.6 μs)	$2^4 \times 1/f_X$ (3.2 μs)
$2^4 \times 1/f_X$ (3.2 μs)	$2^5 \times 1/f_X$ (6.4 μs)	$2^{12} \times 1/f_X$ (819.2 μs)	$2^{13} \times 1/f_X$ (1.64 ms)	$2^4 \times 1/f_X$ (3.2 μs)	$2^5 \times 1/f_X$ (6.4 μs)
$2^5 \times 1/f_X$ (6.4 μs)	$2^6 \times 1/f_X$ (12.8 μs)	$2^{13} \times 1/f_X$ (1.64 ms)	$2^{14} \times 1/f_X$ (3.28 ms)	$2^5 \times 1/f_X$ (6.4 μs)	$2^6 \times 1/f_X$ (12.8 μs)
$2^6 \times 1/f_X$ (12.8 μs)	$2^7 \times 1/f_X$ (25.6 μs)	$2^{14} \times 1/f_X$ (3.28 ms)	$2^{15} \times 1/f_X$ (6.55 ms)	$2^6 \times 1/f_X$ (12.8 μs)	$2^7 \times 1/f_X$ (25.6 μs)
$2^7 \times 1/f_X$ (25.6 μs)	$2^8 \times 1/f_X$ (51.2 μs)	$2^{15} \times 1/f_X$ (6.55 ms)	$2^{16} \times 1/f_X$ (13.1 ms)	$2^7 \times 1/f_X$ (25.6 μs)	$2^8 \times 1/f_X$ (51.2 μs)
$2^8 \times 1/f_X$ (51.2 μs)	$2^9 \times 1/f_X$ (102.4 μs)	$2^{16} \times 1/f_X$ (13.1 ms)	$2^{17} \times 1/f_X$ (26.2 ms)	$2^8 \times 1/f_X$ (51.2 μs)	$2^9 \times 1/f_X$ (102.4 μs)
$2^9 \times 1/f_X$ (102.4 μs)	$2^{10} \times 1/f_X$ (204.8 μs)	$2^{17} \times 1/f_X$ (26.2 ms)	$2^{18} \times 1/f_X$ (52.4 ms)	$2^9 \times 1/f_X$ (102.4 μs)	$2^{10} \times 1/f_X$ (204.8 μs)
$2^{11} \times 1/f_X$ (409.6 μs)	$2^{12} \times 1/f_X$ (819.2 μs)	$2^{19} \times 1/f_X$ (104.9 ms)	$2^{20} \times 1/f_X$ (209.7 ms)	$2^{11} \times 1/f_X$ (409.6 μs)	$2^{12} \times 1/f_X$ (819.2 μs)

Remarks 1. f_X: Main system clock oscillation frequency

2. MCS: Oscillation mode select register bit 0

3. Values in parentheses when operated at f_X = 5.0 MHz.

(2) External event counter

The number of pulses of an externally input signal can be measured.

(3) Square-wave output

A square wave with any selected frequency can be output.

Table 9-2. 8-bit Timer/Event Counter Square-Wave Output Ranges

Minimum Pulse Width		Maximum Pulse Width		Resolution	
MCS = 1	MCS = 0	MCS = 1	MCS = 0	MCS = 1	MCS = 0
2 × 1/fx (400 ns)	2^2 × 1/fx (800 ns)	2^9 × 1/fx (102.4 μs)	2^{10} × 1/fx (204.8 μs)	2 × 1/fx (400 ns)	2^2 × 1/fx (800 ns)
2^2 × 1/fx (800 ns)	2^3 × 1/fx (1.6 μs)	2^{10} × 1/fx (204.8 μs)	2^{11} × 1/fx (409.6 μs)	2^2 × 1/fx (800 ns)	2^3 × 1/fx (1.6 μs)
2^3 × 1/fx (1.6 μs)	2^4 × 1/fx (3.2 μs)	2^{11} × 1/fx (409.6 μs)	2^{12} × 1/fx (819.2 μs)	2^3 × 1/fx (1.6 μs)	2^4 × 1/fx (3.2 μs)
2^4 × 1/fx (3.2 μs)	2^5 × 1/fx (6.4 μs)	2^{12} × 1/fx (819.2 μs)	2^{13} × 1/fx (1.64 ms)	2^4 × 1/fx (3.2 μs)	2^5 × 1/fx (6.4 μs)
2^5 × 1/fx (6.4 μs)	2^6 × 1/fx (12.8 μs)	2^{13} × 1/fx (1.64 ms)	2^{14} × 1/fx (3.28 ms)	2^5 × 1/fx (6.4 μs)	2^6 × 1/fx (12.8 μs)
2^6 × 1/fx (12.8 μs)	2^7 × 1/fx (25.6 μs)	2^{14} × 1/fx (3.28 ms)	2^{15} × 1/fx (6.55 ms)	2^6 × 1/fx (12.8 μs)	2^7 × 1/fx (25.6 μs)
2^7 × 1/fx (25.6 μs)	2^8 × 1/fx (51.2 μs)	2^{15} × 1/fx (6.55 ms)	2^{16} × 1/fx (13.1 ms)	2^7 × 1/fx (25.6 μs)	2^8 × 1/fx (51.2 μs)
2^8 × 1/fx (51.2 μs)	2^9 × 1/fx (102.4 μs)	2^{16} × 1/fx (13.1 ms)	2^{17} × 1/fx (26.2 ms)	2^8 × 1/fx (51.2 μs)	2^9 × 1/fx (102.4 μs)
2^9 × 1/fx (102.4 μs)	2^{10} × 1/fx (204.8 μs)	2^{17} × 1/fx (26.2 ms)	2^{18} × 1/fx (52.4 ms)	2^9 × 1/fx (102.4 μs)	2^{10} × 1/fx (204.8 μs)
2^{11} × 1/fx (409.6 μs)	2^{12} × 1/fx (819.2 μs)	2^{19} × 1/fx (104.9 ms)	2^{20} × 1/fx (209.7 ms)	2^{11} × 1/fx (409.6 μs)	2^{12} × 1/fx (819.2 μs)

Remarks 1. fx: Main system clock oscillation frequency

2. MCS: Oscillation mode select register bit 0

3. Values in parentheses when operated at fx = 5.0 MHz.

9.1.2 16-bit timer/event counter mode

(1) 16-bit interval timer

Interrupt requests can be generated at the preset time intervals.

Table 9-3. Interval Times When 8-bit Timer/Event Counter Is Used as 16-bit Timer/Event Counter

Minimum Interval Time		Maximum Interval Time		Resolution	
MCS = 1	MCS = 0	MCS = 1	MCS = 0	MCS = 1	MCS = 0
2 × 1/fx (400 ns)	2^{2} × 1/fx (800 ns)	2^{17} × 1/fx (26.2 ms)	2^{18} × 1/fx (52.4 ms)	2 × 1/fx (400 ns)	2^{2} × 1/fx (800 ns)
2^{2} × 1/fx (800 ns)	2^{3} × 1/fx (1.6 μs)	2^{18} × 1/fx (52.4 ms)	2^{19} × 1/fx (104.9 ms)	2^{2} × 1/fx (800 ns)	2^{3} × 1/fx (1.6 μs)
2^{3} × 1/fx (1.6 μs)	2^{4} × 1/fx (3.2 μs)	2^{19} × 1/fx (104.9 ms)	2^{20} × 1/fx (209.7 ms)	2^{3} × 1/fx (1.6 μs)	2^{4} × 1/fx (3.2 μs)
2^{4} × 1/fx (3.2 μs)	2^{5} × 1/fx (6.4 μs)	2^{20} × 1/fx (209.7 ms)	2^{21} × 1/fx (419.4 ms)	2^{4} × 1/fx (3.2 μs)	2^{5} × 1/fx (6.4 μs)
2^{5} × 1/fx (6.4 μs)	2^{6} × 1/fx (12.8 μs)	2^{21} × 1/fx (419.4 ms)	2^{22} × 1/fx (838.9 ms)	2^{5} × 1/fx (6.4 μs)	2^{6} × 1/fx (12.8 μs)
2^{6} × 1/fx (12.8 μs)	2^{7} × 1/fx (25.6 μs)	2^{22} × 1/fx (838.9 ms)	2^{23} × 1/fx (1.7 s)	2^{6} × 1/fx (12.8 μs)	2^{7} × 1/fx (25.6 μs)
2^{7} × 1/fx (25.6 μs)	2^{8} × 1/fx (51.2 μs)	2^{23} × 1/fx (1.7 s)	2^{24} × 1/fx (3.4 s)	2^{7} × 1/fx (25.6 μs)	2^{8} × 1/fx (51.2 μs)
2^{8} × 1/fx (51.2 μs)	2^{9} × 1/fx (102.4 μs)	2^{24} × 1/fx (3.4 s)	2^{25} × 1/fx (6.7 s)	2^{8} × 1/fx (51.2 μs)	2^{9} × 1/fx (102.4 μs)
2^{9} × 1/fx (102.4 μs)	2^{10} × 1/fx (204.8 μs)	2^{25} × 1/fx (6.7 s)	2^{26} × 1/fx (13.4 s)	2^{9} × 1/fx (102.4 μs)	2^{10} × 1/fx (204.8 μs)
2^{11} × 1/fx (409.6 μs)	2^{12} × 1/fx (819.2 μs)	2^{27} × 1/fx (26.8 s)	2^{28} × 1/fx (53.7 s)	2^{11} × 1/fx (409.6 μs)	2^{12} × 1/fx (819.2 μs)

Remarks **1.** fx: Main system clock oscillation frequency

2. MCS: Oscillation mode select register bit 0

3. Values in parentheses when operated at fx = 5.0 MHz.

(2) External event counter

The number of pulses of an externally input signal can be measured.

(3) Square-wave output

A square wave with any selected frequency can be output.

Table 9-4. Square-Wave Output Ranges When 8-bit Timer/Event Counter Is Used as 16-bit Timer/Event Counter

Minimum Pulse Width		Maximum Pulse Width		Resolution	
MCS = 1	MCS = 0	MCS = 1	MCS = 0	MCS = 1	MCS = 0
$2 \times 1/f_X$ (400 ns)	$2^{2} \times 1/f_X$ (800 ns)	$2^{17} \times 1/f_X$ (26.2 ms)	$2^{18} \times 1/f_X$ (52.4 ms)	$2 \times 1/f_X$ (400 ns)	$2^{2} \times 1/f_X$ (800 ns)
$2^{2} \times 1/f_X$ (800 ns)	$2^{3} \times 1/f_X$ (1.6 μs)	$2^{18} \times 1/f_X$ (52.4 ms)	$2^{19} \times 1/f_X$ (104.9 ms)	$2^{2} \times 1/f_X$ (800 ns)	$2^{3} \times 1/f_X$ (1.6 μs)
$2^{3} \times 1/f_X$ (1.6 μs)	$2^{4} \times 1/f_X$ (3.2 μs)	$2^{19} \times 1/f_X$ (104.9 ms)	$2^{20} \times 1/f_X$ (209.7 ms)	$2^{3} \times 1/f_X$ (1.6 μs)	$2^{4} \times 1/f_X$ (3.2 μs)
$2^{4} \times 1/f_X$ (3.2 μs)	$2^{5} \times 1/f_X$ (6.4 μs)	$2^{20} \times 1/f_X$ (209.7 ms)	$2^{21} \times 1/f_X$ (419.4 ms)	$2^{4} \times 1/f_X$ (3.2 μs)	$2^{5} \times 1/f_X$ (6.4 μs)
$2^{5} \times 1/f_X$ (6.4 μs)	$2^{6} \times 1/f_X$ (12.8 μs)	$2^{21} \times 1/f_X$ (419.4 ms)	$2^{22} \times 1/f_X$ (838.9 ms)	$2^{5} \times 1/f_X$ (6.4 μs)	$2^{6} \times 1/f_X$ (12.8 μs)
$2^{6} \times 1/f_X$ (12.8 μs)	$2^{7} \times 1/f_X$ (25.6 μs)	$2^{22} \times 1/f_X$ (838.9 ms)	$2^{23} \times 1/f_X$ (1.7 s)	$2^{6} \times 1/f_X$ (12.8 μs)	$2^{7} \times 1/f_X$ (25.6 μs)
$2^{7} \times 1/f_X$ (25.6 μs)	$2^{8} \times 1/f_X$ (51.2 μs)	$2^{23} \times 1/f_X$ (1.7 s)	$2^{24} \times 1/f_X$ (3.4 s)	$2^{7} \times 1/f_X$ (25.6 μs)	$2^{8} \times 1/f_X$ (51.2 μs)
$2^{8} \times 1/f_X$ (51.2 μs)	$2^{9} \times 1/f_X$ (102.4 μs)	$2^{24} \times 1/f_X$ (3.4 s)	$2^{25} \times 1/f_X$ (6.7 s)	$2^{8} \times 1/f_X$ (51.2 μs)	$2^{9} \times 1/f_X$ (102.4 μs)
$2^{9} \times 1/f_X$ (102.4 μs)	$2^{10} \times 1/f_X$ (204.8 μs)	$2^{25} \times 1/f_X$ (6.7 s)	$2^{26} \times 1/f_X$ (13.4 s)	$2^{9} \times 1/f_X$ (102.4 μs)	$2^{10} \times 1/f_X$ (204.8 μs)
$2^{11} \times 1/f_X$ (409.6 μs)	$2^{12} \times 1/f_X$ (819.2 μs)	$2^{27} \times 1/f_X$ (26.8 s)	$2^{28} \times 1/f_X$ (53.7 s)	$2^{11} \times 1/f_X$ (409.6 μs)	$2^{12} \times 1/f_X$ (819.2 μs)

Remarks 1. f_X: Main system clock oscillation frequency

2. MCS: Oscillation mode select register bit 0

3. Values in parentheses when operated at f_X = 5.0 MHz.

9.2 8-bit Timer/Event Counter Configuration

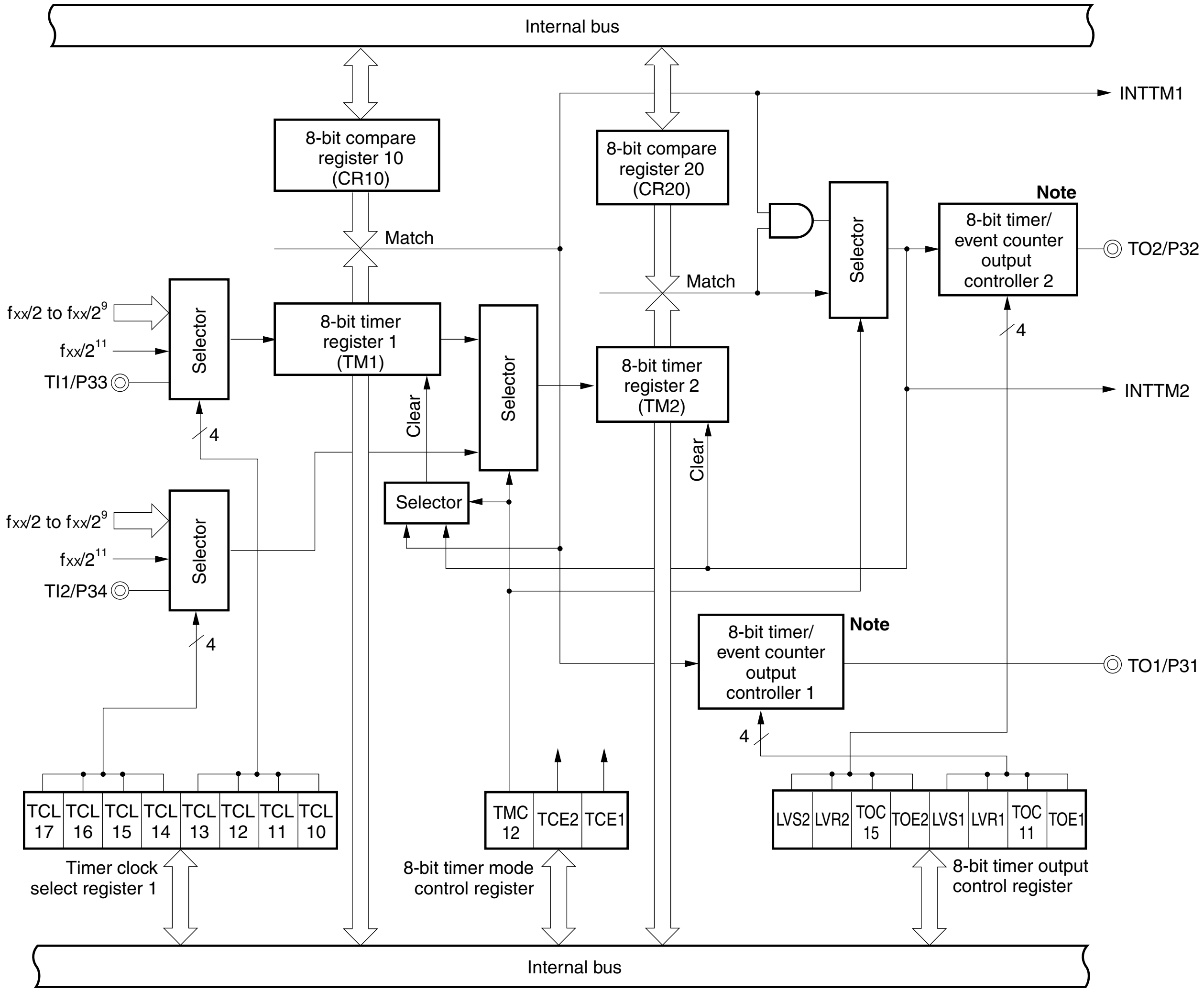
The 8-bit timer/event counter consists of the following hardware.

Table 9-5. 8-bit Timer/Event Counter Configuration

Item	Configuration
Timer register	8 bits × 2 (TM1, TM2)
Register	Compare register: 8 bits × 2 (CR10, CR20)
Timer output	2 (TO1, TO2)
Control register	Timer clock select register 1 (TCL1) 8-bit timer mode control register 1 (TMC1) 8-bit timer output control register (TOC1) Port mode register 3 (PM3)[Note]

Note For details, refer to **Figure 6-9 P30 to P37 Block Diagram**.

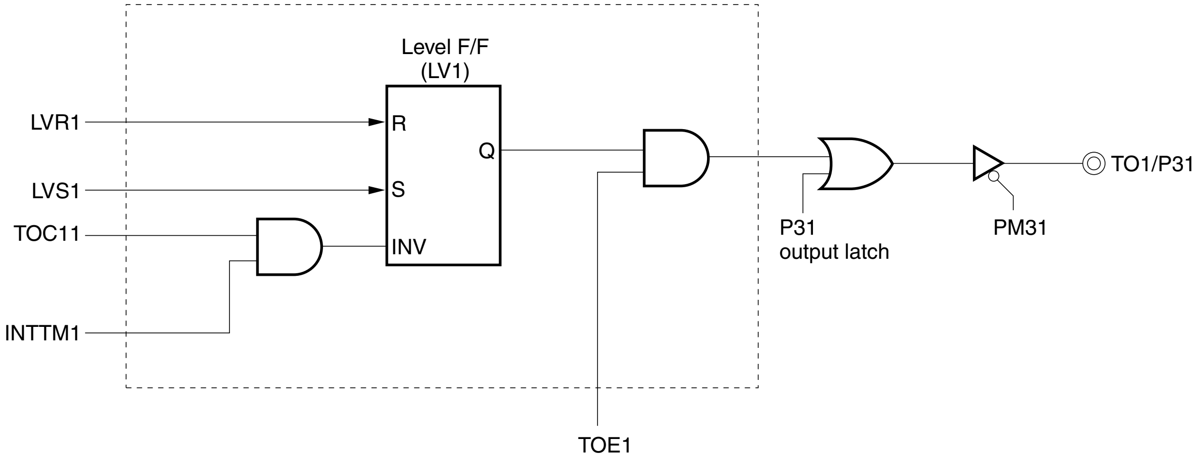
Figure 9-1. 8-bit Timer/Event Counter Block Diagram


Internal bus
INTTM1
8-bit compare register 10 (CR10)
8-bit compare register 20 (CR20)
Note
8-bit timer/ event counter output controller 2
TO2/P32
Match
Match
Selector
fxx/2 to fxx/2^9
fxx/2^11
TI1/P33
Selector
8-bit timer register 1 (TM1)
Selector
8-bit timer register 2 (TM2)
INTTM2
Clear
Clear
4
4
Selector
fxx/2 to fxx/2^9
fxx/2^11
TI2/P34
Selector
4
8-bit timer/ event counter output controller 1
Note
TO1/P31
4
TCL 17
TCL 16
TCL 15
TCL 14
TCL 13
TCL 12
TCL 11
TCL 10
TMC 12
TCE2
TCE1
LVS2
LVR2
TOC 15
TOE2
LVS1
LVR1
TOC 11
TOE1
Timer clock select register 1
8-bit timer mode control register
8-bit timer output control register
Internal bus

Note Refer to Figures 9-2 and 9-3 for details of 8-bit timer/event counter output controllers 1 and 2, respectively.

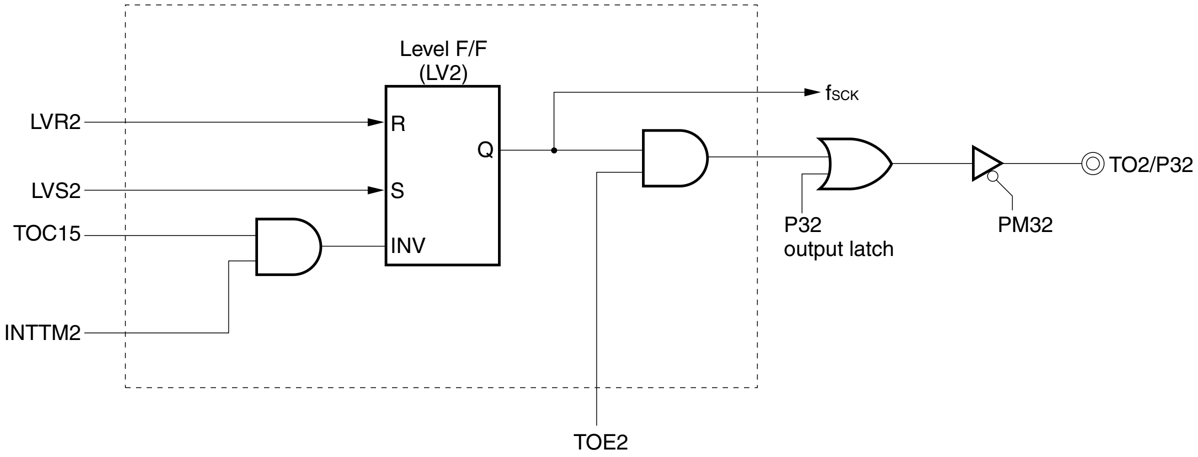
Remark fxx = fx/2 (MCS = 0), fxx = fx (MCS = 1)

Figure 9-2. Block Diagram of 8-bit Timer/Event Counter Output Controller 1



Remark The section in the broken line is an output controller.

Figure 9-3. Block Diagram of 8-bit Timer/Event Counter Output Controller 2



Remarks 1. The section in the broken line is an output controller.

2. f_{SCK}: Serial clock frequency

(1) Compare registers 10 and 20 (CR10, CR20)

These are 8-bit registers to compare the value set to CR10 to the 8-bit timer register 1 (TM1) count value, and the value set to CR20 to the 8-bit timer register 2 (TM2) count value, and, if they match, generate an interrupt request (INTTM1 and INTTM2, respectively).

When TM1 and TM2 are set to interval timer operation, these registers are used to hold the interval time. When the PWM output operation is specified, they are used as registers that specify a pulse width.

CR10 and CR20 are set with an 8-bit memory manipulation instruction. They cannot be set with a 16-bit memory manipulation instruction. When the compare register is used as an 8-bit timer/event counter, the 00H to FFH values can be set. When the compare register is used as a 16-bit timer/event counter, the 0000H to FFFFH values can be set.

$\overline{\text{RESET}}$ input makes CR10 and CR20 undefined.

Caution When using the compare register as 16-bit timer/event counter, be sure to set data after stopping timer operation.

(2) 8-bit timer registers 1, 2 (TM1, TM2)

These are 8-bit registers to count count pulses.

When TM1 and TM2 are used in the 8-bit timer × 2-channel mode, they are read with an 8-bit memory manipulation instruction. When TM1 and TM2 are used in 16-bit timer × 1-channel mode, the 16-bit timer (TMS) is read with a 16-bit memory manipulation instruction.

$\overline{\text{RESET}}$ input clears TM1 and TM2 to 00H.

9.3 8-bit Timer/Event Counter Control Registers

The following four types of registers are used to control the 8-bit timer/event counter.

- Timer clock select register 1 (TCL1)
- 8-bit timer mode control register 1 (TMC1)
- 8-bit timer output control register (TOC1)
- Port mode register 3 (PM3)

(1) Timer clock select register 1 (TCL1)

This register sets count clocks of 8-bit timer registers 1 and 2.
TCL1 is set with an 8-bit memory manipulation instruction.
$\overline{\text{RESET}}$ input clears TCL1 to 00H.

Figure 9-4. Timer Clock Select Register 1 Format

Symbol	7	6	5	4	3	2	1	0	Address	After Reset	R/W
TCL1	TCL17	TCL16	TCL15	TCL14	TCL13	TCL12	TCL11	TCL10	FF41H	00H	R/W

TCL17	TCL16	TCL15	TCL14	8-bit Timer Register 2 Count Clock Selection	
				MCS = 1	MCS = 0
0	0	0	0	TI2 falling edge	
0	0	0	1	TI2 rising edge	
0	1	1	0	fx/2 (2.5 MHz)	fx/2^{2} (1.25 MHz)
0	1	1	1	fx/2^{2} (1.25 MHz)	fx/2^{3} (625 kHz)
1	0	0	0	fx/2^{3} (625 kHz)	fx/2^{4} (313 kHz)
1	0	0	1	fx/2^{4} (313 kHz)	fx/2^{5} (156 kHz)
1	0	1	0	fx/2^{5} (156 kHz)	fx/2^{6} (78.1 kHz)
1	0	1	1	fx/2^{6} (78.1 kHz)	fx/2^{7} (39.1 kHz)
1	1	0	0	fx/2^{7} (39.1 kHz)	fx/2^{8} (19.5 kHz)
1	1	0	1	fx/2^{8} (19.5 kHz)	fx/2^{9} (9.8 kHz)
1	1	1	0	fx/2^{9} (9.8 kHz)	fx/2^{10} (4.9 kHz)
1	1	1	1	fx/2^{11} (2.4 kHz)	fx/2^{12} (1.2 kHz)
Other than above				Setting prohibited	

TCL13	TCL12	TCL11	TCL10	8-bit Timer Register 1 Count Clock Selection	
				MCS = 1	MCS = 0
0	0	0	0	TI1 falling edge	
0	0	0	1	TI1 rising edge	
0	1	1	0	fx/2 (2.5 MHz)	fx/2^{2} (1.25 MHz)
0	1	1	1	fx/2^{2} (1.25 MHz)	fx/2^{3} (625 kHz)
1	0	0	0	fx/2^{3} (625 kHz)	fx/2^{4} (313 kHz)
1	0	0	1	fx/2^{4} (313 kHz)	fx/2^{5} (156 kHz)
1	0	1	0	fx/2^{5} (156 kHz)	fx/2^{6} (78.1 kHz)
1	0	1	1	fx/2^{6} (78.1 kHz)	fx/2^{7} (39.1 kHz)
1	1	0	0	fx/2^{7} (39.1 kHz)	fx/2^{8} (19.5 kHz)
1	1	0	1	fx/2^{8} (19.5 kHz)	fx/2^{9} (9.8 kHz)
1	1	1	0	fx/2^{9} (9.8 kHz)	fx/2^{10} (4.9 kHz)
1	1	1	1	fx/2^{11} (2.4 kHz)	fx/2^{12} (1.2 kHz)
Other than above				Setting prohibited	

Caution When rewriting TCL1 to other data, stop the timer operation beforehand.

Remarks 1. fx: Main system clock oscillation frequency

2. TI1: 8-bit timer register 1 input pin

3. TI2: 8-bit timer register 2 input pin

4. MCS: Oscillation mode select register bit 0

5. Figures in parentheses apply to operation with fx = 5.0 MHz

(2) 8-bit timer mode control register (TMC1)

This register enables/stops operation of 8-bit timer registers 1 and 2 and sets the operating mode of 8-bit timer registers 1 and 2.

TMC1 is set with a 1-bit or 8-bit memory manipulation instruction.

$\overline{\text{RESET}}$ input clears TMC1 to 00H.

Figure 9-5. 8-bit Timer Mode Control Register Format

Symbol	7	6	5	4	3	2	①	⓪	Address	After Reset	R/W
TMC1	0	0	0	0	0	TMC12	TCE2	TCE1	FF49H	00H	R/W

TMC12	Operating Mode Selection
0	8-bit timer register × 2 channel mode (TM1, TM2)
1	16-bit timer register × 1 channel mode (TMS)

TCE2	8-bit Timer Register 2 Operation Control
0	Operation stop (TM2 clear to 0)
1	Operation enable

TCE1	8-bit Timer Register 1 Operation Control
0	Operation stop (TM1 clear to 0)
1	Operation enable

Cautions 1. Switch the operating mode after stopping timer operation.

2. When used as 16-bit timer register, TCE1 should be used for operation enable/stop.

(3) 8-bit timer output control register (TOC1)

This register controls operation of 8-bit timer/event counter output controllers 1 and 2.

It sets/resets the R-S flip-flops (LV1 and LV2) and enables/disables inversion and timer output of 8-bit timer registers 1 and 2.

TOC1 is set with a 1-bit or 8-bit memory manipulation instruction.

$\overline{\text{RESET}}$ input clears TOC1 to 00H.

Figure 9-6. 8-bit Timer Output Control Register Format

Symbol	⑦	⑥	5	④	③	②	1	⓪	Address	After Reset	R/W
TOC1	LVS2	LVR2	TOC15	TOE2	LVS1	LVR1	TOC11	TOE1	FF4FH	00H	R/W

LVS2	LVR2	8-bit Timer/Event Counter 2 Timer Output F/F Status Set
0	0	Unchanged
0	1	Timer output F/F reset (to 0)
1	0	Timer output F/F set (to 1)
1	1	Setting prohibited

TOC15	8-bit Timer/Event Counter 2 Timer Output F/F Control
0	Inverted operation disable
1	Inverted operation enable

TOE2	8-bit Timer/Event Counter 2 Output Control
0	Output disable (port mode)
1	Output enable

LVS1	LVR1	8-bit Timer/Event Counter 1 Timer Output F/F Status Set
0	0	Unchanged
0	1	Timer output F/F reset (to 0)
1	0	Timer output F/F set (to 1)
1	1	Setting prohibited

TOC11	8-bit Timer/Event Counter 1 Timer Output F/F Control
0	Inverted operation disable
1	Inverted operation enable

TOE1	8-bit Timer/Event Counter 1 Outptut Control
0	Output disable (port mode)
1	Output enable

Cautions 1. Be sure to set TOC1 after stopping timer operation.

2. LVS1, LVS2, LVR1 and LVR2 are 0 when read after data setting to them.

(4) Port mode register 3 (PM3)

This register sets port 3 input/output in 1-bit units.

When using the P31/TO1 and P32/TO2 pins for timer output, set PM31, PM32, and output latches of P31 and P32 to 0.

PM3 is set with a 1-bit or 8-bit memory manipulation instruction.

$\overline{\text{RESET}}$ input sets PM3 to FFH.

Figure 9-7. Port Mode Register 3 Format

Symbol	7	6	5	4	3	2	1	0	Address	After Reset	R/W
PM3	PM37	PM36	PM35	PM34	PM33	PM32	PM31	PM30	FF23H	FFH	R/W

PM3n	P3n Pin I/O Mode Selection (n = 0 to 7)
0	Output mode (output buffer ON)
1	Input mode (output buffer OFF)

9.4 8-bit Timer/Event Counter Operations

9.4.1 8-bit timer/event counter mode

(1) Interval timer operations

The 8-bit timer/event counter operates as an interval timer which generates interrupt requests repeatedly at intervals of the count value preset to 8-bit compare registers 10 and 20 (CR10 and CR20).

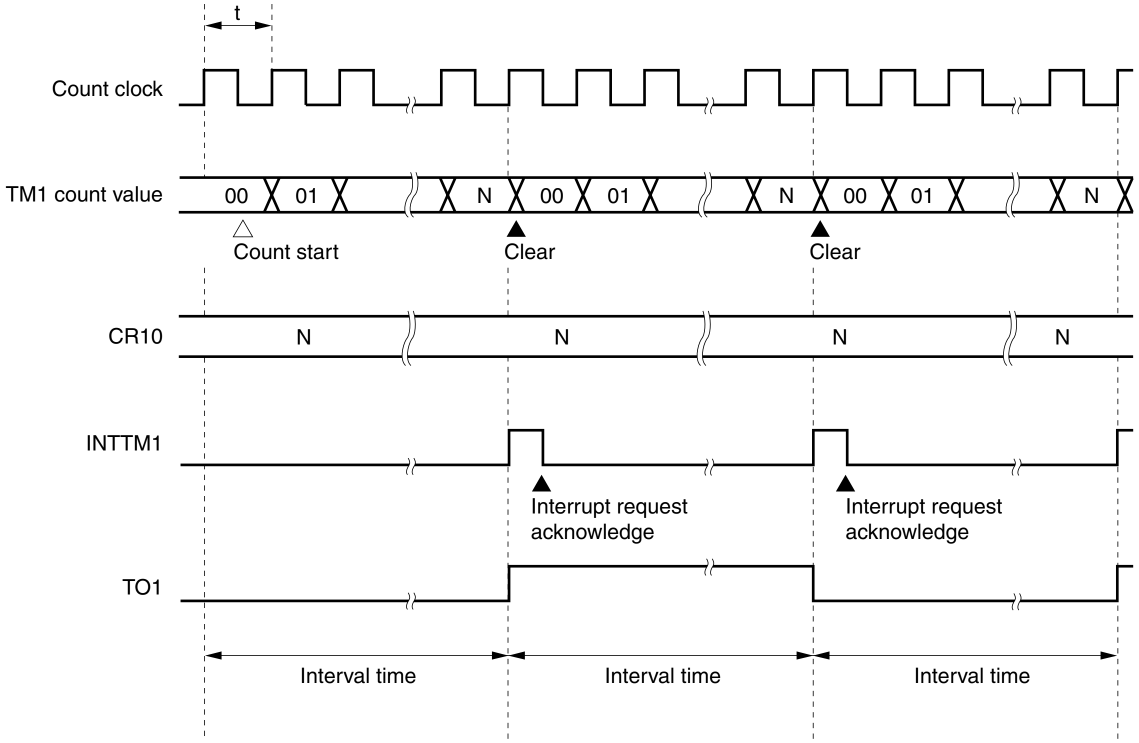
When the count values of 8-bit timer registers 1 and 2 (TM1 and TM2) match the values set to CR10 and CR20, counting continues with the TM1 and TM2 values cleared to 0 and the interrupt request signals (INTTM1 and INTTM2) are generated.

Count clock of TM1 can be selected with bits 0 to 3 (TCL10 to TCL13) of timer clock select register 1 (TCL1).

Count clock of TM2 can be selected with bits 4 to 7 (TCL14 to TCL17) of timer clock select register 1 (TCL1).

For the operation to be performed when the value of the compare register is changed during timer count operation, refer to **9.5 8-bit Timer/Event Counter Precautions (3)**.

Figure 9-8. Interval Timer Operation Timings



Remark Interval time = (N + 1) × t: N = 00H to FFH

Table 9-6. 8-bit Timer/Event Counter 1 Interval Time

TCL13	TCL12	TCL11	TCL10	Minimum Interval Time		Maximum Interval Time		Resolution	
				MCS = 1	MCS = 0	MCS = 1	MCS = 0	MCS = 1	MCS = 0
0	0	0	0	TI1 input cycle		$2^8 \times$ TI1 input cycle		TI1 input edge cycle	
0	0	0	1	TI1 input cycle		$2^8 \times$ TI1 input cycle		TI1 input edge cycle	
0	1	1	0	$2 \times 1/f_X$ (400 ns)	$2^2 \times 1/f_X$ (800 ns)	$2^9 \times 1/f_X$ (102.4 µs)	$2^{10} \times 1/f_X$ (204.8 µs)	$2 \times 1/f_X$ (400 ns)	$2^2 \times 1/f_X$ (800 ns)
0	1	1	1	$2^2 \times 1/f_X$ (800 ns)	$2^3 \times 1/f_X$ (1.6 µs)	$2^{10} \times 1/f_X$ (204.8 µs)	$2^{11} \times 1/f_X$ (409.6 µs)	$2^2 \times 1/f_X$ (800 ns)	$2^3 \times 1/f_X$ (1.6 µs)
1	0	0	0	$2^3 \times 1/f_X$ (1.6 µs)	$2^4 \times 1/f_X$ (3.2 µs)	$2^{11} \times 1/f_X$ (409.6 µs)	$2^{12} \times 1/f_X$ (819.2 µs)	$2^3 \times 1/f_X$ (1.6 µs)	$2^4 \times 1/f_X$ (3.2 µs)
1	0	0	1	$2^4 \times 1/f_X$ (3.2 µs)	$2^5 \times 1/f_X$ (6.4 µs)	$2^{12} \times 1/f_X$ (819.2 µs)	$2^{13} \times 1/f_X$ (1.64 ms)	$2^4 \times 1/f_X$ (3.2 µs)	$2^5 \times 1/f_X$ (6.4 µs)
1	0	1	0	$2^5 \times 1/f_X$ (6.4 µs)	$2^6 \times 1/f_X$ (12.8 µs)	$2^{13} \times 1/f_X$ (1.64 ms)	$2^{14} \times 1/f_X$ (3.28 ms)	$2^5 \times 1/f_X$ (6.4 µs)	$2^6 \times 1/f_X$ (12.8 µs)
1	0	1	1	$2^6 \times 1/f_X$ (12.8 µs)	$2^7 \times 1/f_X$ (25.6 µs)	$2^{14} \times 1/f_X$ (3.28 ms)	$2^{15} \times 1/f_X$ (6.55 ms)	$2^6 \times 1/f_X$ (12.8 µs)	$2^7 \times 1/f_X$ (25.6 µs)
1	1	0	0	$2^7 \times 1/f_X$ (25.6 µs)	$2^8 \times 1/f_X$ (51.2 µs)	$2^{15} \times 1/f_X$ (6.55 ms)	$2^{16} \times 1/f_X$ (13.1 ms)	$2^7 \times 1/f_X$ (25.6 µs)	$2^8 \times 1/f_X$ (51.2 µs)
1	1	0	1	$2^8 \times 1/f_X$ (51.2 µs)	$2^9 \times 1/f_X$ (102.4 µs)	$2^{16} \times 1/f_X$ (13.1 ms)	$2^{17} \times 1/f_X$ (26.2 ms)	$2^8 \times 1/f_X$ (51.2 µs)	$2^9 \times 1/f_X$ (102.4 µs)
1	1	1	0	$2^9 \times 1/f_X$ (102.4 µs)	$2^{10} \times 1/f_X$ (204.8 µs)	$2^{17} \times 1/f_X$ (26.2 ms)	$2^{18} \times 1/f_X$ (52.4 ms)	$2^9 \times 1/f_X$ (102.4 µs)	$2^{10} \times 1/f_X$ (204.8 µs)
1	1	1	1	$2^{11} \times 1/f_X$ (409.6 µs)	$2^{12} \times 1/f_X$ (819.2 µs)	$2^{19} \times 1/f_X$ (104.9 ms)	$2^{20} \times 1/f_X$ (209.7 ms)	$2^{11} \times 1/f_X$ (409.6 µs)	$2^{12} \times 1/f_X$ (819.2 µs)
Other than above				Setting prohibited					

Remarks 1. f_X: Main system clock oscillation frequency

2. TCL10 to TCL13: Bits 0 to 3 of timer clock select register 1 (TCL1)

3. MCS: Oscillation mode select register bit 0

4. Values in parentheses when operated at f_X = 5.0 MHz.

Table 9-7. 8-bit Timer/Event Counter 2 Interval Time

TCL17	TCL16	TCL15	TCL14	Minimum Interval Time		Maximum Interval Time		Resolution	
				MCS = 1	MCS = 0	MCS = 1	MCS = 0	MCS = 1	MCS = 0
0	0	0	0	TI2 input cycle		2^{8} × TI2 input cycle		TI2 input edge cycle	
0	0	0	1	TI2 input cycle		2^{8} × TI2 input cycle		TI2 input edge cycle	
0	1	1	0	$2 \times 1/f_X$ (400 ns)	$2^{2} \times 1/f_X$ (800 ns)	$2^{9} \times 1/f_X$ (102.4 μs)	$2^{10} \times 1/f_X$ (204.8 μs)	$2 \times 1/f_X$ (400 ns)	$2^{2} \times 1/f_X$ (800 ns)
0	1	1	1	$2^{2} \times 1/f_X$ (800 ns)	$2^{3} \times 1/f_X$ (1.6 μs)	$2^{10} \times 1/f_X$ (204.8 μs)	$2^{11} \times 1/f_X$ (409.6 μs)	$2^{2} \times 1/f_X$ (800 ns)	$2^{3} \times 1/f_X$ (1.6 μs)
1	0	0	0	$2^{3} \times 1/f_X$ (1.6 μs)	$2^{4} \times 1/f_X$ (3.2 μs)	$2^{11} \times 1/f_X$ (409.6 μs)	$2^{12} \times 1/f_X$ (819.2 μs)	$2^{3} \times 1/f_X$ (1.6 μs)	$2^{4} \times 1/f_X$ (3.2 μs)
1	0	0	1	$2^{4} \times 1/f_X$ (3.2 μs)	$2^{5} \times 1/f_X$ (6.4 μs)	$2^{12} \times 1/f_X$ (819.2 μs)	$2^{13} \times 1/f_X$ (1.64 ms)	$2^{4} \times 1/f_X$ (3.2 μs)	$2^{5} \times 1/f_X$ (6.4 μs)
1	0	1	0	$2^{5} \times 1/f_X$ (6.4 μs)	$2^{6} \times 1/f_X$ (12.8 μs)	$2^{13} \times 1/f_X$ (1.64 ms)	$2^{14} \times 1/f_X$ (3.28 ms)	$2^{5} \times 1/f_X$ (6.4 μs)	$2^{6} \times 1/f_X$ (12.8 μs)
1	0	1	1	$2^{6} \times 1/f_X$ (12.8 μs)	$2^{7} \times 1/f_X$ (25.6 μs)	$2^{14} \times 1/f_X$ (3.28 ms)	$2^{15} \times 1/f_X$ (6.55 ms)	$2^{6} \times 1/f_X$ (12.8 μs)	$2^{7} \times 1/f_X$ (25.6 μs)
1	1	0	0	$2^{7} \times 1/f_X$ (25.6 μs)	$2^{8} \times 1/f_X$ (51.2 μs)	$2^{15} \times 1/f_X$ (6.55 ms)	$2^{16} \times 1/f_X$ (13.1 ms)	$2^{7} \times 1/f_X$ (25.6 μs)	$2^{8} \times 1/f_X$ (51.2 μs)
1	1	0	1	$2^{8} \times 1/f_X$ (51.2 μs)	$2^{9} \times 1/f_X$ (102.4 μs)	$2^{16} \times 1/f_X$ (13.1 ms)	$2^{17} \times 1/f_X$ (26.2 ms)	$2^{8} \times 1/f_X$ (51.2 μs)	$2^{9} \times 1/f_X$ (102.4 μs)
1	1	1	0	$2^{9} \times 1/f_X$ (102.4 μs)	$2^{10} \times 1/f_X$ (204.8 μs)	$2^{17} \times 1/f_X$ (26.2 ms)	$2^{18} \times 1/f_X$ (52.4 ms)	$2^{9} \times 1/f_X$ (102.4 μs)	$2^{10} \times 1/f_X$ (204.8 μs)
1	1	1	1	$2^{11} \times 1/f_X$ (409.6 μs)	$2^{12} \times 1/f_X$ (819.2 μs)	$2^{19} \times 1/f_X$ (104.9 ms)	$2^{20} \times 1/f_X$ (209.7 ms)	$2^{11} \times 1/f_X$ (409.6 μs)	$2^{12} \times 1/f_X$ (819.2 μs)
Other than above				Setting prohibited					

Remarks

1. f_X: Main system clock oscillation frequency
2. TCL10 to TCL13: Bits 0 to 3 of timer clock select register 1 (TCL1)
3. MCS: Oscillation mode select register bit 0
4. Values in parentheses when operated at f_X = 5.0 MHz

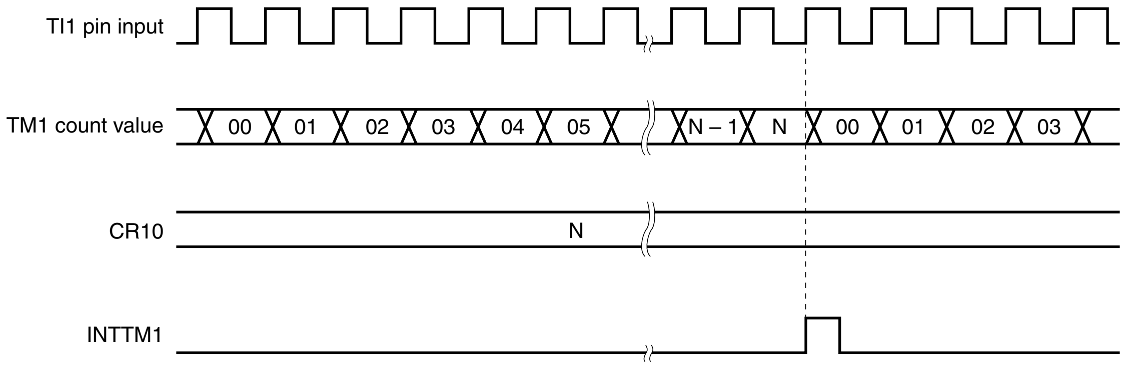
(2) External event counter operation

The external event counter counts the number of external clock pulses to be input to the TI1/P33 and TI2/P34 pins with 8-bit timer registers 1 and 2 (TM1 and TM2).

TM1 and TM2 are incremented each time the valid edge specified with timer clock select register 1 (TCL1) is input. Either the rising or falling edge can be selected.

When the TM1 and TM2 counted values match the values of 8-bit compare registers 10 and 20 (CR10 and CR20), TM1 and TM2 are cleared to 0 and the interrupt request signals (INTTM1 and INTTM2) are generated.

Figure 9-9. External Event Counter Operation Timings (with Rising Edge Specified)



Remark N = 00H to FFH

(3) Square-wave output operation

A square wave with any selected frequency is output at intervals of the value preset to 8-bit compare registers 10 and 20 (CR10 and CR20).

The TO1/P31 or TO2/P32 pin output status is reversed at intervals of the count value preset to CR10 or CR20 by setting bit 0 (TOE1) or bit 4 (TOE2) of the 8-bit timer output control register (TOC1) to 1. This enables a square wave with any selected frequency to be output.

Table 9-8. 8-bit Timer/Event Counter Square-Wave Output Ranges

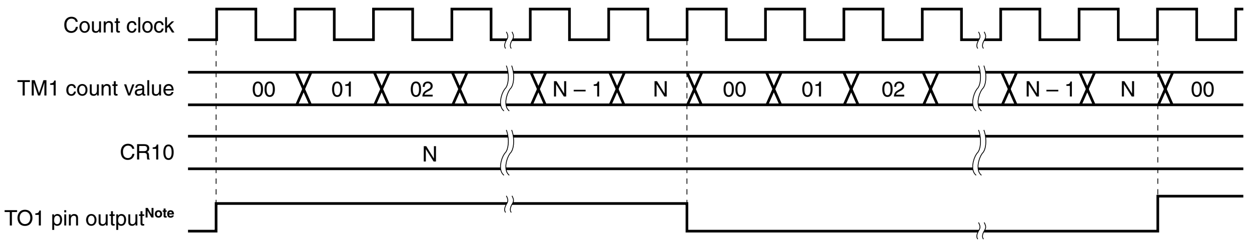
Minimum Pulse Width		Maximum Pulse Width		Resolution	
MCS = 1	MCS = 0	MCS = 1	MCS = 0	MCS = 1	MCS = 0
$2 \times 1/f_X$ (400 ns)	$2^{2} \times 1/f_X$ (800 ns)	$2^{9} \times 1/f_X$ (102.4 µs)	$2^{10} \times 1/f_X$ (204.8 µs)	$2 \times 1/f_X$ (400 ns)	$2^{2} \times 1/f_X$ (800 ns)
$2^{2} \times 1/f_X$ (800 ns)	$2^{3} \times 1/f_X$ (1.6 µs)	$2^{10} \times 1/f_X$ (204.8 µs)	$2^{11} \times 1/f_X$ (409.6 µs)	$2^{2} \times 1/f_X$ (800 ns)	$2^{3} \times 1/f_X$ (1.6 µs)
$2^{3} \times 1/f_X$ (1.6 µs)	$2^{4} \times 1/f_X$ (3.2 µs)	$2^{11} \times 1/f_X$ (409.6 µs)	$2^{12} \times 1/f_X$ (819.2 µs)	$2^{3} \times 1/f_X$ (1.6 µs)	$2^{4} \times 1/f_X$ (3.2 µs)
$2^{4} \times 1/f_X$ (3.2 µs)	$2^{5} \times 1/f_X$ (6.4 µs)	$2^{12} \times 1/f_X$ (819.2 µs)	$2^{13} \times 1/f_X$ (1.64 ms)	$2^{4} \times 1/f_X$ (3.2 µs)	$2^{5} \times 1/f_X$ (6.4 µs)
$2^{5} \times 1/f_X$ (6.4 µs)	$2^{6} \times 1/f_X$ (12.8 µs)	$2^{13} \times 1/f_X$ (1.64 ms)	$2^{14} \times 1/f_X$ (3.28 ms)	$2^{5} \times 1/f_X$ (6.4 µs)	$2^{6} \times 1/f_X$ (12.8 µs)
$2^{6} \times 1/f_X$ (12.8 µs)	$2^{7} \times 1/f_X$ (25.6 µs)	$2^{14} \times 1/f_X$ (3.28 ms)	$2^{15} \times 1/f_X$ (6.55 ms)	$2^{6} \times 1/f_X$ (12.8 µs)	$2^{7} \times 1/f_X$ (25.6 µs)
$2^{7} \times 1/f_X$ (25.6 µs)	$2^{8} \times 1/f_X$ (51.2 µs)	$2^{15} \times 1/f_X$ (6.55 ms)	$2^{16} \times 1/f_X$ (13.1 ms)	$2^{7} \times 1/f_X$ (25.6 µs)	$2^{8} \times 1/f_X$ (51.2 µs)
$2^{8} \times 1/f_X$ (51.2 µs)	$2^{9} \times 1/f_X$ (102.4 µs)	$2^{16} \times 1/f_X$ (13.1 ms)	$2^{17} \times 1/f_X$ (26.2 ms)	$2^{8} \times 1/f_X$ (51.2 µs)	$2^{9} \times 1/f_X$ (102.4 µs)
$2^{9} \times 1/f_X$ (102.4 µs)	$2^{10} \times 1/f_X$ (204.8 µs)	$2^{17} \times 1/f_X$ (26.2 ms)	$2^{18} \times 1/f_X$ (52.4 ms)	$2^{9} \times 1/f_X$ (102.4 µs)	$2^{10} \times 1/f_X$ (204.8 µs)
$2^{11} \times 1/f_X$ (409.6 µs)	$2^{12} \times 1/f_X$ (819.2 µs)	$2^{19} \times 1/f_X$ (104.9 ms)	$2^{20} \times 1/f_X$ (209.7 ms)	$2^{11} \times 1/f_X$ (409.6 µs)	$2^{12} \times 1/f_X$ (819.2 µs)

Remarks 1. f_X: Main system clock oscillation frequency

2. MCS: Oscillation mode select register bit 0

3. Values in parentheses when operated at f_X = 5.0 MHz.

Figure 9-10. Square Wave Output Operation Timing



Note The initial value of TO1 output can be set by using bits 2 and 3 (LVR1 and LVS1) of the 8-bit timer output control register (TOC1).

9.4.2 16-bit timer/event counter mode

When bit 2 (TMC12) of 8-bit timer mode control register 1 (TMC1) is set to 1, the 16-bit timer/event counter mode is set.

In this mode, the count clock is selected with bits 0 to 3 (TCL10 to TCL13) of timer clock select register 1 (TCL1). The overflow signal of 8-bit timer/event counter 1 (TM1) is used as the count clock to 8-bit timer/event counter 2 (TM2).

In this mode, the count operation enable/disable is selected with bit 0 (TCE1) of TMC1.

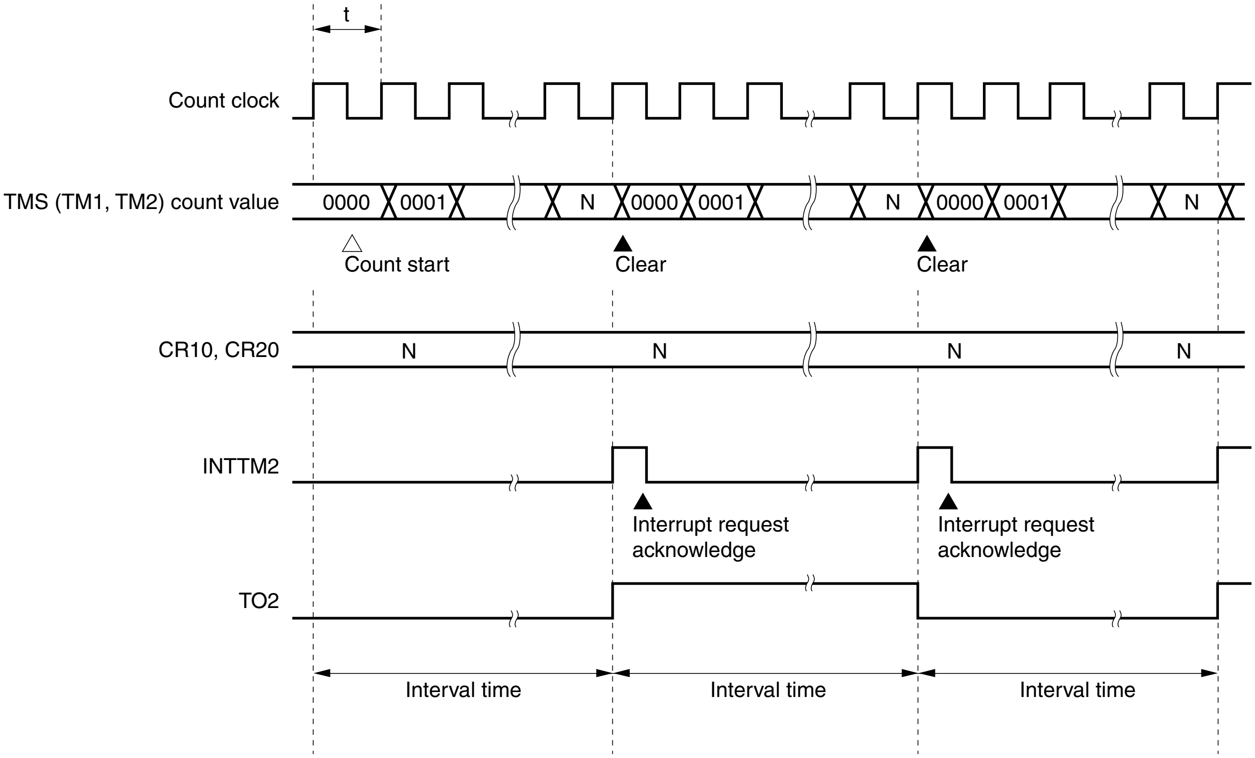
(1) Interval timer operation

The 8-bit timer/event counter can operate as an interval timer which generates interrupt requests repeatedly at intervals of the count value preset to 2-channel 8-bit compare registers (CR10 and CR20). To set a count value, set the value of the higher 8 bits to CR20, and the value of the lower 8 bits to CR10. For the count value that can be set (interval time), refer to Table 9-7.

When the 8-bit timer register 1 (TM1) and CR10 values match and the 8-bit timer register 2 (TM2) and CR20 values match, counting continues with the TM1 and TM2 values cleared to 0 and the interrupt request signal (INTTM2) is generated. For the operation timing of the interval timer, refer to Figure 9-11.

The count clock can be selected with bits 0 to 3 (TCL10 to TCL13) of timer clock select register 1 (TCL1). The overflow signal of TM1 is used as the count clock to TM2.

Figure 9-11. Interval Timer Operation Timing



Remark Interval time = (N + 1) × t: N = 0000H to FFFFH

Caution Even if the 16-bit timer/event counter mode is used, when the TM1 count value matches the CR10 value, interrupt request (INTTM1) is generated and the F/F of 8-bit timer/event counter output controller 1 is inverted. Thus, when using 8-bit timer/event counter as 16-bit interval timer, set the INTTM1 mask flag TMMK1 to 1 to disable INTTM1 acknowledgment.

When reading the 16-bit timer (TMS) count value, use the 16-bit memory manipulation instruction.

Table 9-9. Interval Times When 2-Channel 8-bit Timer/Event Counters (TM1 and TM2) Are Used as 16-bit Timer/Event Counter

TCL13	TCL12	TCL11	TCL10	Minimum Interval Time		Maximum Interval Time		Resolution	
				MCS = 1	MCS = 0	MCS = 1	MCS = 0	MCS = 1	MCS = 0
0	0	0	0	TI1 input cycle		$2^{8} \times$ TI1 input cycle		TI1 input edge cycle	
0	0	0	1	TI1 input cycle		$2^{8} \times$ TI1 input cycle		TI1 input edge cycle	
0	1	1	0	$2 \times 1/f_X$ (400 ns)	$2^{2} \times 1/f_X$ (800 ns)	$2^{17} \times 1/f_X$ (26.2 ms)	$2^{18} \times 1/f_X$ (52.4 ms)	$2 \times 1/f_X$ (400 ns)	$2^{2} \times 1/f_X$ (800 ns)
0	1	1	1	$2^{2} \times 1/f_X$ (800 ns)	$2^{3} \times 1/f_X$ (1.6 µs)	$2^{18} \times 1/f_X$ (52.4 ms)	$2^{19} \times 1/f_X$ (104.9 ms)	$2^{2} \times 1/f_X$ (800 ns)	$2^{3} \times 1/f_X$ (1.6 µs)
1	0	0	0	$2^{3} \times 1/f_X$ (1.6 µs)	$2^{4} \times 1/f_X$ (3.2 µs)	$2^{19} \times 1/f_X$ (104.9 ms)	$2^{20} \times 1/f_X$ (209.7 ms)	$2^{3} \times 1/f_X$ (1.6 µs)	$2^{4} \times 1/f_X$ (3.2 µs)
1	0	0	1	$2^{4} \times 1/f_X$ (3.2 µs)	$2^{5} \times 1/f_X$ (6.4 µs)	$2^{20} \times 1/f_X$ (209.7 ms)	$2^{21} \times 1/f_X$ (419.4 ms)	$2^{4} \times 1/f_X$ (3.2 µs)	$2^{5} \times 1/f_X$ (6.4 µs)
1	0	1	0	$2^{5} \times 1/f_X$ (6.4 µs)	$2^{6} \times 1/f_X$ (12.8 µs)	$2^{21} \times 1/f_X$ (419.4 ms)	$2^{22} \times 1/f_X$ (838.9 ms)	$2^{5} \times 1/f_X$ (6.4 µs)	$2^{6} \times 1/f_X$ (12.8 µs)
1	0	1	1	$2^{6} \times 1/f_X$ (12.8 µs)	$2^{7} \times 1/f_X$ (25.6 µs)	$2^{22} \times 1/f_X$ (838.9 ms)	$2^{23} \times 1/f_X$ (1.7 s)	$2^{6} \times 1/f_X$ (12.8 µs)	$2^{7} \times 1/f_X$ (25.6 µs)
1	1	0	0	$2^{7} \times 1/f_X$ (25.6 µs)	$2^{8} \times 1/f_X$ (51.2 µs)	$2^{23} \times 1/f_X$ (1.7 s)	$2^{24} \times 1/f_X$ (3.4 s)	$2^{7} \times 1/f_X$ (25.6 µs)	$2^{8} \times 1/f_X$ (51.2 µs)
1	1	0	1	$2^{8} \times 1/f_X$ (51.2 µs)	$2^{9} \times 1/f_X$ (102.4 µs)	$2^{24} \times 1/f_X$ (3.4 s)	$2^{25} \times 1/f_X$ (6.7 s)	$2^{8} \times 1/f_X$ (51.2 µs)	$2^{9} \times 1/f_X$ (102.4 µs)
1	1	1	0	$2^{9} \times 1/f_X$ (102.4 µs)	$2^{10} \times 1/f_X$ (204.8 µs)	$2^{25} \times 1/f_X$ (6.7 s)	$2^{26} \times 1/f_X$ (13.4 s)	$2^{9} \times 1/f_X$ (102.4 µs)	$2^{10} \times 1/f_X$ (204.8 µs)
1	1	1	1	$2^{11} \times 1/f_X$ (409.6 µs)	$2^{12} \times 1/f_X$ (819.2 µs)	$2^{27} \times 1/f_X$ (26.8 s)	$2^{28} \times 1/f_X$ (53.7 s)	$2^{11} \times 1/f_X$ (409.6 µs)	$2^{12} \times 1/f_X$ (819.2 µs)
Other than above				Setting prohibited					

Remarks 1. f_X: Main system clock oscillation frequency

2. TCL10 to TCL13: Bits 0 to 3 of timer clock select register 1 (TCL1)

3. MCS: Oscillation mode select register bit 0

4. Values in parentheses when operated at f_X = 5.0 MHz.

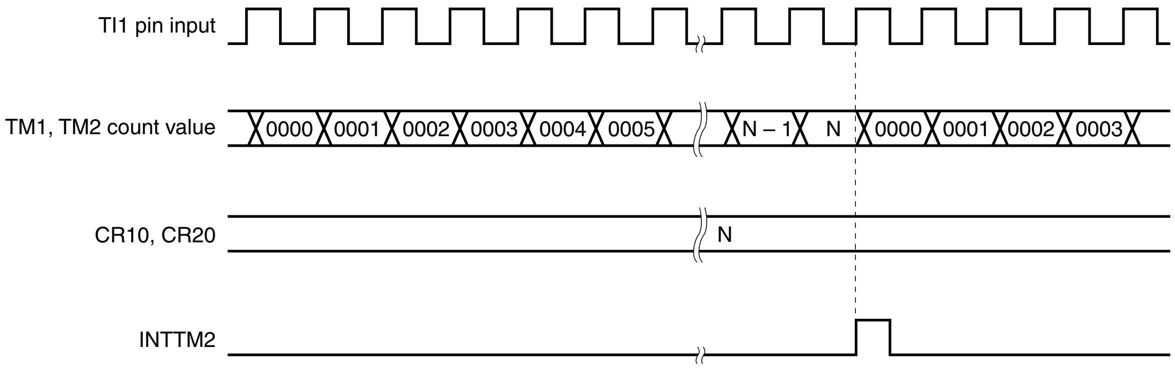
(2) External event counter operations

The external event counter counts the number of external clock pulses to be input to the TI1/P33 pin with 2-channel 8-bit timer registers 1 and 2 (TM1 and TM2).

TM1 is incremented each time the valid edge specified with timer clock select register 1 (TCL1) is input. When TM1 overflows as a result, TM2 is incremented with the overflow signal used as its count clock. Either the rising or falling edge can be selected.

When the TM1 and TM2 counted values match the values of 8-bit compare registers 10 and 20 (CR10 and CR20), TM1 and TM2 are cleared to 0 and the interrupt request signal (INTTM2) is generated.

Figure 9-12. External Event Counter Operation Timings (with Rising Edge Specified)



Caution Even if the 16-bit timer/event counter mode is used, when the TM1 count value matches the CR10 value, interrupt request (INTTM1) is generated and the F/F of 8-bit timer/event counter output controller 1 is inverted. Thus, when using 8-bit timer/event counter as 16-bit interval timer, set the INTTM1 mask flag TMMK1 to 1 to disable INTTM1 acknowledgment.

When reading the 16-bit timer (TMS) count value, use the 16-bit memory manipulation instruction.

(3) Square-wave output operation

A square wave with any selected frequency is output at intervals of the value preset to 8-bit compare registers 10 and 20 (CR10 and CR20).

The TO2/P32 pin output status is reversed at intervals of the count value preset to CR10 and CR20 by setting bit 4 (TOE2) of the 8-bit timer output control register (TOC1) to 1. This enables a square wave with any selected frequency to be output.

Table 9-10. Square-Wave Output Ranges When 2-Channel 8-bit Timer/Event Counters (TM1 and TM2) Are Used as 16-bit Timer/Event Counter

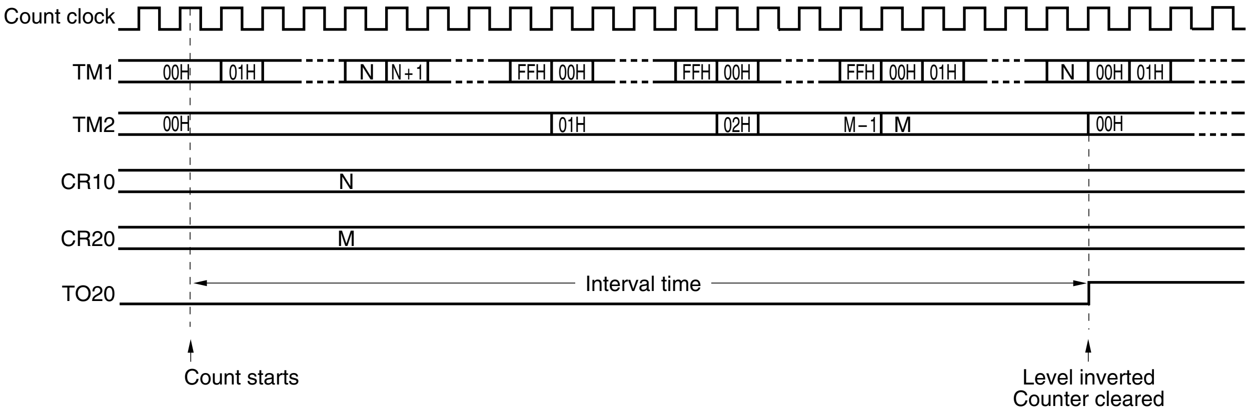
Minimum Pulse Width		Maximum Pulse Width		Resolution	
MCS = 1	MCS = 0	MCS = 1	MCS = 0	MCS = 1	MCS = 0
$2 \times 1/f_X$ (400 ns)	$2^{2} \times 1/f_X$ (800 ns)	$2^{17} \times 1/f_X$ (26.2 ms)	$2^{18} \times 1/f_X$ (52.4 ms)	$2 \times 1/f_X$ (400 ns)	$2^{2} \times 1/f_X$ (800 ns)
$2^{2} \times 1/f_X$ (800 ns)	$2^{3} \times 1/f_X$ (1.6 μs)	$2^{18} \times 1/f_X$ (52.4 ms)	$2^{19} \times 1/f_X$ (104.9 ms)	$2^{2} \times 1/f_X$ (800 ns)	$2^{3} \times 1/f_X$ (1.6 μs)
$2^{3} \times 1/f_X$ (1.6 μs)	$2^{4} \times 1/f_X$ (3.2 μs)	$2^{19} \times 1/f_X$ (104.9 ms)	$2^{20} \times 1/f_X$ (209.7 ms)	$2^{3} \times 1/f_X$ (1.6 μs)	$2^{4} \times 1/f_X$ (3.2 μs)
$2^{4} \times 1/f_X$ (3.2 μs)	$2^{5} \times 1/f_X$ (6.4 μs)	$2^{20} \times 1/f_X$ (209.7 ms)	$2^{21} \times 1/f_X$ (419.4 ms)	$2^{4} \times 1/f_X$ (3.2 μs)	$2^{5} \times 1/f_X$ (6.4 μs)
$2^{5} \times 1/f_X$ (6.4 μs)	$2^{6} \times 1/f_X$ (12.8 μs)	$2^{21} \times 1/f_X$ (419.4 ms)	$2^{22} \times 1/f_X$ (838.9 ms)	$2^{5} \times 1/f_X$ (6.4 μs)	$2^{6} \times 1/f_X$ (12.8 μs)
$2^{6} \times 1/f_X$ (12.8 μs)	$2^{7} \times 1/f_X$ (25.6 μs)	$2^{22} \times 1/f_X$ (838.9 ms)	$2^{23} \times 1/f_X$ (1.7 s)	$2^{6} \times 1/f_X$ (12.8 μs)	$2^{7} \times 1/f_X$ (25.6 μs)
$2^{7} \times 1/f_X$ (25.6 μs)	$2^{8} \times 1/f_X$ (51.2 μs)	$2^{23} \times 1/f_X$ (1.7 s)	$2^{24} \times 1/f_X$ (3.4 s)	$2^{7} \times 1/f_X$ (25.6 μs)	$2^{8} \times 1/f_X$ (51.2 μs)
$2^{8} \times 1/f_X$ (51.2 μs)	$2^{9} \times 1/f_X$ (102.4 μs)	$2^{24} \times 1/f_X$ (3.4 s)	$2^{25} \times 1/f_X$ (6.7 s)	$2^{8} \times 1/f_X$ (51.2 μs)	$2^{9} \times 1/f_X$ (102.4 μs)
$2^{9} \times 1/f_X$ (102.4 μs)	$2^{10} \times 1/f_X$ (204.8 μs)	$2^{25} \times 1/f_X$ (6.7 s)	$2^{26} \times 1/f_X$ (13.4 s)	$2^{9} \times 1/f_X$ (102.4 μs)	$2^{10} \times 1/f_X$ (204.8 μs)
$2^{11} \times 1/f_X$ (409.6 μs)	$2^{12} \times 1/f_X$ (819.2 μs)	$2^{27} \times 1/f_X$ (26.8 s)	$2^{28} \times 1/f_X$ (53.7 s)	$2^{11} \times 1/f_X$ (409.6 μs)	$2^{12} \times 1/f_X$ (819.2 μs)

Remarks 1. f_X: Main system clock oscillation frequency

2. MCS: Oscillation mode select register bit 0

3. Values in parentheses when operated at f_X = 5.0 MHz.

Figure 9-13. Square Wave Output Operation Timing

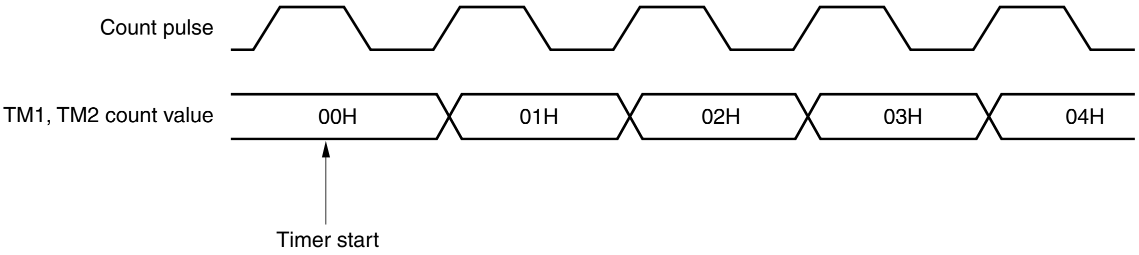


9.5 8-bit Timer/Event Counter Precautions

(1) Timer start errors

An error with a maximum of one clock may occur concerning the time required for a match signal to be generated after timer start. This is because 8-bit timer registers 1 and 2 (TM1 and TM2) are started asynchronously with the count pulse.

Figure 9-14. 8-bit Timer Register Start Timing



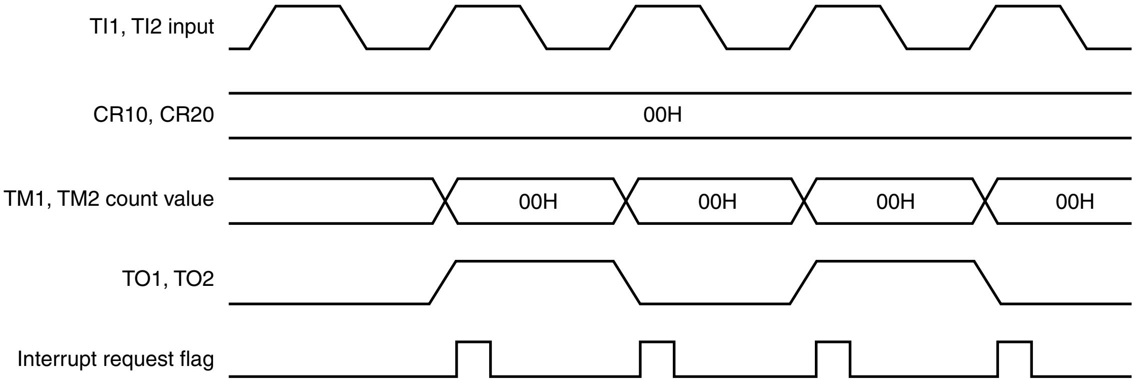
(2) 8-bit compare registers 10 and 20 setting

8-bit compare registers 10 and 20 (CR10 and CR20) can be set to 00H.

Thus, when these 8-bit compare registers are used as event counters, one-pulse count operation can be carried out.

When the 8-bit compare register is used as 16-bit timer/event counter, write data to CR10 and CR20 after setting bit 0 (TCE1) of 8-bit timer mode control register 1 to 0 and stopping timer operation.

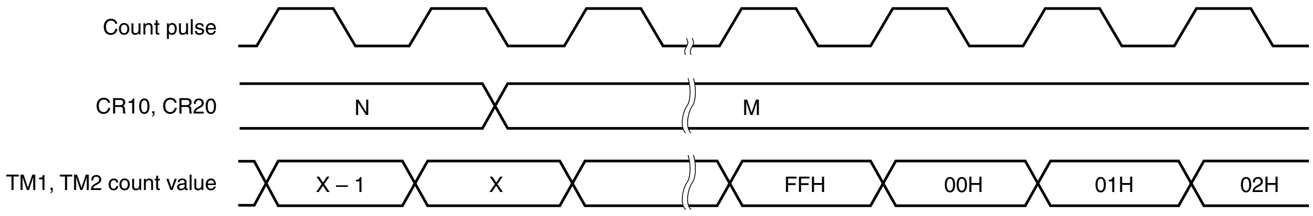
Figure 9-15. External Event Counter Operation Timing



(3) Operation after compare register change during timer count operation

If the values after 8-bit compare registers 10 and 20 (CR10 and CR20) are changed are smaller than those of 8-bit timer registers 1 and 2 (TM1 and TM2), TM1 and TM2 continue counting, overflow and then restart counting from 0. Thus, if the value (M) after CR10 and CR20 change is smaller than value (N) before the change, it is necessary to restart the timer after changing CR10 and CR20.

Figure 9-16. Timing After Compare Register Change During Timer Count Operation



Remark N > X > M

CHAPTER 10 WATCH TIMER

10.1 Watch Timer Functions

The watch timer has the following functions.

- Watch timer
- Interval timer

The watch timer and the interval timer can be used simultaneously.

(1) Watch timer

When the 32.768 kHz subsystem clock is used, a flag (WTIF) is set at 0.5 second or 0.25 second intervals. When the 4.19 MHz (standard: 4.194304 MHz) main system clock is used, a flag (WTIF) is set at 0.5 second or 0.25 second intervals.

Caution 0.5-second intervals cannot be generated with the 5.0 MHz main system clock. You should switch to the 32.768 kHz subsystem clock to generate 0.5-second intervals.

(2) Interval timer

Interrupt requests (INTTM3) are generated at the preset time interval.

Table 10-1. Interval Timer Interval Time

Interval Time	When Operated at f_{XX} = 5.0 MHz	When Operated at f_{XX} = 4.19 MHz	When Operated at f_{XT} = 32.768 kHz
$2^4 \times 1/f_W$	410 μs	488 μs	488 μs
$2^5 \times 1/f_W$	819 μs	977 μs	977 μs
$2^6 \times 1/f_W$	1.64 ms	1.95 ms	1.95 ms
$2^7 \times 1/f_W$	3.28 ms	3.91 ms	3.91 ms
$2^8 \times 1/f_W$	6.55 ms	7.81 ms	7.81 ms
$2^9 \times 1/f_W$	13.1 ms	15.6 ms	15.6 ms

Remark f_{XX}: Main system clock frequency (f_X or $f_X/2$)

f_X: Main system clock oscillation frequency

f_{XT}: Subsystem clock oscillation frequency

f_W: Watch timer clock frequency ($f_{XX}/2^7$ or f_{XT})

10.2 Watch Timer Configuration

The watch timer consists of the following hardware.

Table 10-2. Watch Timer Configuration

Item	Configuration
Counter	5 bits × 1
Control register	Timer clock select register 2 (TCL2) Watch timer mode control register (TMC2)

10.3 Watch Timer Control Registers

The following two types of registers are used to control the watch timer.

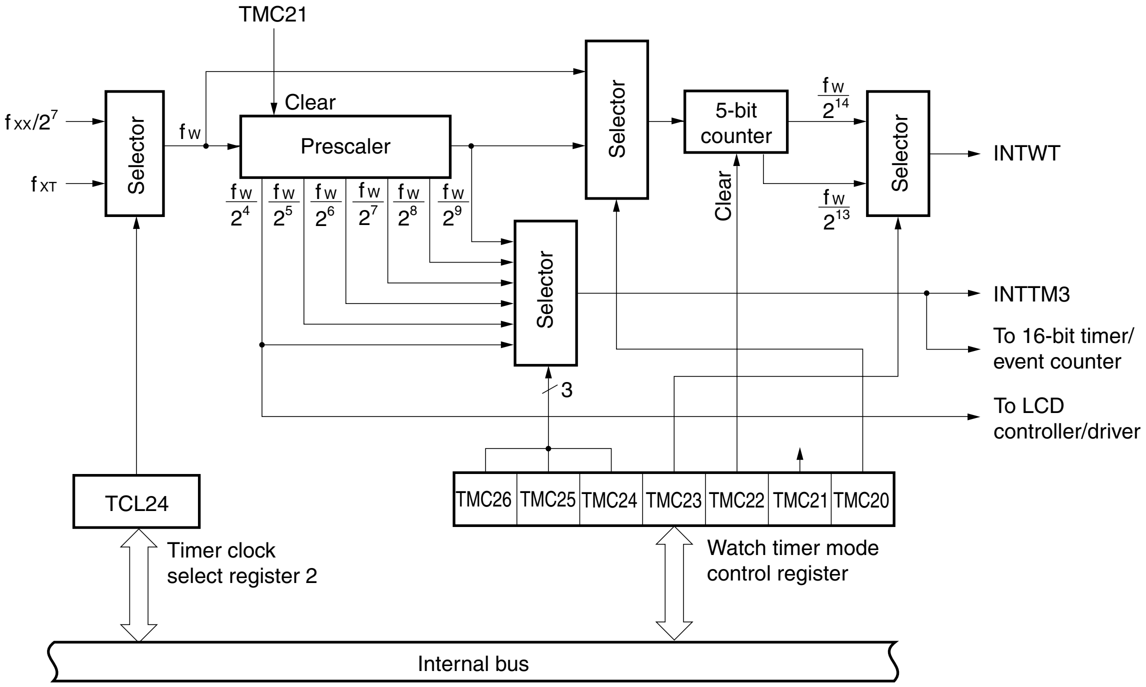
- Timer clock select register 2 (TCL2)
- Watch timer mode control register (TMC2)

(1) Timer clock select register 2 (TCL2)

This register sets the watch timer count clock.
TCL2 is set with an 8-bit memory manipulation instruction.
$\overline{\text{RESET}}$ input clears TCL2 to 00H.

Remark Besides setting the watch timer count clock, TCL2 sets the watchdog timer count clock and buzzer output frequency.

Figure 10-1. Watch Timer Block Diagram



Remark fxx = fx/2 (MCS = 0), fxx = fx (MCS = 1)

Figure 10-2. Timer Clock Select Register 2 Format

Symbol	7	6	5	4	3	2	1	0	Address	After Reset	R/W
TCL2	TCL27	TCL26	TCL25	TCL24	0	TCL22	TCL21	TCL20	FF42H	00H	R/W

TCL27	TCL26	TCL25	Buzzer Output Frequency Selection	
			MCS = 1	MCS = 0
0	×	×	Buzzer output disable	
1	0	0	$f_X/2^9$ (9.8 kHz)	$f_X/2^{10}$ (4.9 kHz)
1	0	1	$f_X/2^{10}$ (4.9 kHz)	$f_X/2^{11}$ (2.4 kHz)
1	1	0	$f_X/2^{11}$ (2.4 kHz)	$f_X/2^{12}$ (1.2 kHz)
1	1	1	Setting prohibited	

TCL24	Watch Timer Count Clock Selection	
	MCS = 1	MCS = 0
0	$f_X/2^7$ (39.1 kHz)	$f_X/2^8$ (19.5 kHz)
1	f_{XT} (32.768 kHz)	

TCL22	TCL21	TCL20	Watchdog Timer Count Clock Selection	
			MCS = 1	MCS = 0
0	0	0	$f_X/2^3$ (625 kHz)	$f_X/2^4$ (313 kHz)
0	0	1	$f_X/2^4$ (313 kHz)	$f_X/2^5$ (156 kHz)
0	1	0	$f_X/2^5$ (156 kHz)	$f_X/2^6$ (78.1 kHz)
0	1	1	$f_X/2^6$ (78.1 kHz)	$f_X/2^7$ (39.1 kHz)
1	0	0	$f_X/2^7$ (39.1 kHz)	$f_X/2^8$ (19.5 kHz)
1	0	1	$f_X/2^8$ (19.5 kHz)	$f_X/2^9$ (9.8 kHz)
1	1	0	$f_X/2^9$ (9.8 kHz)	$f_X/2^{10}$ (4.9 kHz)
1	1	1	$f_X/2^{11}$ (2.4 kHz)	$f_X/2^{12}$ (1.2 kHz)

Caution When rewriting TCL2 to other data, stop the timer operation beforehand.

Remarks 1. f_X: Main system clock oscillation frequency

2. f_{XT}: Subsystem clock oscillation frequency

3. ×: don't care

4. MCS: Oscillation mode select register bit 0

5. Figures in parentheses apply to operation with f_X = 5.0 MHz or f_{XT} = 32.768 kHz.

(2) Watch timer mode control register (TMC2)

This register sets the watch timer operating mode, watch flag set time and prescaler interval time and enables/disables prescaler and 5-bit counter operations.

TMC2 is set with a 1-bit or 8-bit memory manipulation instruction.

$\overline{\text{RESET}}$ input clears TMC2 to 00H.

Figure 10-3. Watch Timer Mode Control Register Format

Symbol	7	6	5	4	3	2	1	0	Address	After Reset	R/W
TMC2	0	TMC26	TMC25	TMC24	TMC23	TMC22	TMC21	TMC20	FF4AH	00H	R/W

TMC26	TMC25	TMC24	Prescaler Interval Time Selection		
			f_{XX} = 5.0 MHz Operation	f_{XX} = 4.19 MHz Operation	f_{XT} = 32.768 kHz Operation
0	0	0	$2^4/f_W$ (410 μs)	$2^4/f_W$ (488 μs)	$2^4/f_W$ (488 μs)
0	0	1	$2^5/f_W$ (819 μs)	$2^5/f_W$ (977 μs)	$2^5/f_W$ (977 μs)
0	1	0	$2^6/f_W$ (1.64 ms)	$2^6/f_W$ (1.95 ms)	$2^6/f_W$ (1.95 ms)
0	1	1	$2^7/f_W$ (3.28 ms)	$2^7/f_W$ (3.91 ms)	$2^7/f_W$ (3.91 ms)
1	0	0	$2^8/f_W$ (6.55 ms)	$2^8/f_W$ (7.81 ms)	$2^8/f_W$ (7.81 ms)
1	0	1	$2^9/f_W$ (13.1 ms)	$2^9/f_W$ (15.6 ms)	$2^9/f_W$ (15.6 ms)
Other than above			Setting prohibited		

TMC23	Watch Flag Set Time Selection		
	f_{XX} = 5.0 MHz Operation	f_{XX} = 4.19 MHz Operation	f_{XT} = 32.768 kHz Operation
0	$2^{14}/f_W$ (0.4 sec)	$2^{14}/f_W$ (0.5 sec)	$2^{14}/f_W$ (0.5 sec)
1	$2^{13}/f_W$ (0.2 sec)	$2^{13}/f_W$ (0.25 sec)	$2^{13}/f_W$ (0.25 sec)

TMC22	5-bit Counter Operation Control
0	Clear after operation stop
1	Operation enable

TMC21	Prescaler Operation Control
0	Clear after operation stop
1	Operation enable

TMC20	Watch Operating Mode Selection
0	Normal operating mode (flag set at $f_W/2^{14}$)
1	Fast feed operating mode (flag set at $f_W/2^5$)

Caution When the watch timer is used, the prescaler should not be cleared frequently.

Remark f_W: Watch timer clock frequency ($f_{XX}/2^7$ or f_{XT})

f_{XX}: Main system clock frequency (f_X or $f_X/2$)

f_X: Main system clock oscillation frequency

f_{XT}: Subsystem clock oscillation frequency

10.4 Watch Timer Operations

10.4.1 Watch timer operation

When the 32.768 kHz subsystem clock or 4.19 MHz main system clock is used, the timer operates as a watch timer with a 0.5-second or 0.25-second interval.

The watch timer sets the test input flag (WTIF) to 1 at the constant time interval. The standby state (STOP mode/ HALT mode) can be cleared by setting WTIF to 1.

When bit 2 (TIMC22) of the watch timer mode control register is set to 0, the 5-bit counter is cleared and the count operation stops.

For simultaneous operation of the interval timer, zero-second start can be achieved by setting TMC22 to 0 (maximum error: 26.2 ms when operated at f_{XX} = 5.0 MHz).

10.4.2 Interval timer operation

The watch timer operates as interval timer which generates interrupt requests repeatedly at an interval of the preset count value.

The interval time can be selected with bits 4 to 6 (TMC24 to TMC26) of the watch timer mode control register.

Table 10-3. Interval Timer Interval Time

TMC26	TMC25	TMC24	Interval Time	When Operated at f_{XX} = 5.0 MHz	When Operated at f_{XX} = 4.19 MHz	When Operated at f_{XT} = 32.768 kHz
0	0	0	$2^4 \times 1/f_W$	410 μs	488 μs	488 μs
0	0	1	$2^5 \times 1/f_W$	819 μs	977 μs	977 μs
0	1	0	$2^6 \times 1/f_W$	1.64 ms	1.95 ms	1.95 ms
0	1	1	$2^7 \times 1/f_W$	3.28 ms	3.91 ms	3.91 ms
1	0	0	$2^8 \times 1/f_W$	6.55 ms	7.81 ms	7.81 ms
1	0	1	$2^9 \times 1/f_W$	13.1 ms	15.6 ms	15.6 ms
Other than above			Setting prohibited			

Remark f_{XX}: Main system clock frequency (f_X or $f_X/2$)
f_X: Main system clock oscillation frequency
f_{XT}: Subsystem clock oscillation frequency
f_W: Watch timer clock frequency ($f_{XX}/2^7$ or f_{XT})

CHAPTER 11 WATCHDOG TIMER

11.1 Watchdog Timer Functions

The watchdog timer has the following functions.

- Watchdog timer
- Interval timer

Caution Select the watchdog timer mode or the interval timer mode with the watchdog timer mode register (WDTM).

(1) Watchdog timer mode

An inadvertent program loop is detected. Upon detection of the inadvertent program loop, a non-maskable interrupt request or $\overline{\text{RESET}}$ can be generated.

Table 11-1. Watchdog Timer Inadvertent Program Loop Detection Times

Loop Detection Time	MCS = 1	MCS = 0
2^{11} × 1/fxx	2^{11} × 1/fx (410 μs)	2^{12} × 1/fx (819 μs)
2^{12} × 1/fxx	2^{12} × 1/fx (819 μs)	2^{13} × 1/fx (1.64 ms)
2^{13} × 1/fxx	2^{13} × 1/fx (1.64 ms)	2^{14} × 1/fx (3.28 ms)
2^{14} × 1/fxx	2^{14} × 1/fx (3.28 ms)	2^{15} × 1/fx (6.55 ms)
2^{15} × 1/fxx	2^{15} × 1/fx (6.55 ms)	2^{16} × 1/fx (13.1 ms)
2^{16} × 1/fxx	2^{16} × 1/fx (13.1 ms)	2^{17} × 1/fx (26.2 ms)
2^{17} × 1/fxx	2^{17} × 1/fx (26.2 ms)	2^{18} × 1/fx (52.4 ms)
2^{19} × 1/fxx	2^{19} × 1/fx (104.9 ms)	2^{20} × 1/fx (209.7 ms)

Remarks 1. fxx: Main system clock frequency (fx or fx/2)

2. fx: Main system clock oscillation frequency

3. MCS: Oscillation mode select register bit 0

4. Figures in parentheses apply to operation with fx = 5.0 MHz.

(2) Interval timer mode

Interrupt requests are generated at the preset time intervals.

Table 11-2. Interval Times

Interval Time	MCS = 1	MCS = 0
2^{11} × 1/fxx	2^{11} × 1/fx (410 μs)	2^{12} × 1/fx (819 μs)
2^{12} × 1/fxx	2^{12} × 1/fx (819 μs)	2^{13} × 1/fx (1.64 ms)
2^{13} × 1/fxx	2^{13} × 1/fx (1.64 ms)	2^{14} × 1/fx (3.28 ms)
2^{14} × 1/fxx	2^{14} × 1/fx (3.28 ms)	2^{15} × 1/fx (6.55 ms)
2^{15} × 1/fxx	2^{15} × 1/fx (6.55 ms)	2^{16} × 1/fx (13.1 ms)
2^{16} × 1/fxx	2^{16} × 1/fx (13.1 ms)	2^{17} × 1/fx (26.2 ms)
2^{17} × 1/fxx	2^{17} × 1/fx (26.2 ms)	2^{18} × 1/fx (52.4 ms)
2^{19} × 1/fxx	2^{19} × 1/fx (104.9 ms)	2^{20} × 1/fx (209.7 ms)

Remarks 1. fxx: Main system clock frequency (fx or fx/2)

2. fx: Main system clock oscillation frequency

3. MCS: Oscillation mode select register bit 0

4. Figures in parentheses apply to operation with fx = 5.0 MHz.

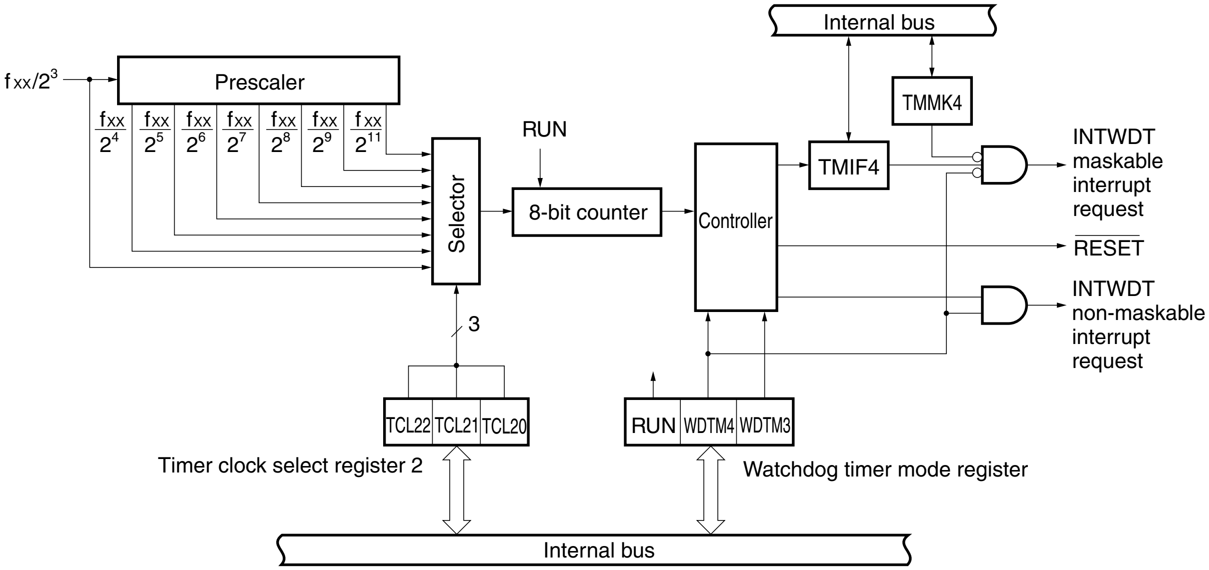
11.2 Watchdog Timer Configuration

The watchdog timer consists of the following hardware.

Table 11-3. Watchdog Timer Configuration

Item	Configuration
Control register	Timer clock select register 2 (TCL2) Watchdog timer mode register (WDTM)

Figure 11-1. Watchdog Timer Block Diagram



Remark fxx = fx/2 (MCS = 0), fxx = fx (MCS = 1)

11.3 Watchdog Timer Control Registers

The following two types of registers are used to control the watchdog timer.

- Timer clock select register 2 (TCL2)
- Watchdog timer mode register (WDTM)

(1) Timer clock select register 2 (TCL2)

This register sets the watchdog timer count clock.

TCL2 is set with an 8-bit memory manipulation instruction.

$\overline{\text{RESET}}$ input clears TCL2 to 00H.

Remark Besides setting the watchdog timer count clock, TCL2 sets the watch timer count clock and buzzer output frequency.

Figure 11-2. Timer Clock Select Register 2 Format

Symbol	7	6	5	4	3	2	1	0	Address	After Reset	R/W
TCL2	TCL27	TCL26	TCL25	TCL24	0	TCL22	TCL21	TCL20	FF42H	00H	R/W

TCL27	TCL26	TCL25	Buzzer Output Frequency Selection	
			MCS = 1	MCS = 0
0	×	×	Buzzer output disable	
1	0	0	$f_X/2^9$ (9.8 kHz)	$f_X/2^{10}$ (4.9 kHz)
1	0	1	$f_X/2^{10}$ (4.9 kHz)	$f_X/2^{11}$ (2.4 kHz)
1	1	0	$f_X/2^{11}$ (2.4 kHz)	$f_X/2^{12}$ (1.2 kHz)
1	1	1	Setting prohibited	

TCL24	Watch Timer Count Clock Selection	
	MCS = 1	MCS = 0
0	$f_X/2^7$ (39.1 kHz)	$f_X/2^8$ (19.5 kHz)
1	f_{XT} (32.768 kHz)	

TCL22	TCL21	TCL20	Watchdog Timer Count Clock Selection	
			MCS = 1	MCS = 0
0	0	0	$f_X/2^3$ (625 kHz)	$f_X/2^4$ (313 kHz)
0	0	1	$f_X/2^4$ (313 kHz)	$f_X/2^5$ (156 kHz)
0	1	0	$f_X/2^5$ (156 kHz)	$f_X/2^6$ (78.1 kHz)
0	1	1	$f_X/2^6$ (78.1 kHz)	$f_X/2^7$ (39.1 kHz)
1	0	0	$f_X/2^7$ (39.1 kHz)	$f_X/2^8$ (19.5 kHz)
1	0	1	$f_X/2^8$ (19.5 kHz)	$f_X/2^9$ (9.8 kHz)
1	1	0	$f_X/2^9$ (9.8 kHz)	$f_X/2^{10}$ (4.9 kHz)
1	1	1	$f_X/2^{11}$ (2.4 kHz)	$f_X/2^{12}$ (1.2 kHz)

Caution   When rewriting TCL2 to other data, stop the timer operation beforehand.

Remarks 1. f_X: Main system clock oscillation frequency

2. f_{XT}: Subsystem clock oscillation frequency

3. ×: don't care

4. MCS: Oscillation mode select register bit 0

5. Figures in parentheses apply to operation with f_X = 5.0 MHz or f_{XT} = 32.768 kHz.

(2) Watchdog timer mode register (WDTM)

This register sets the watchdog timer operating mode and enables/disables counting.

WDTM is set with a 1-bit or 8-bit memory manipulation instruction.

$\overline{\text{RESET}}$ input clears WDTM to 00H.

Figure 11-3. Watchdog Timer Mode Register Format

Symbol	⑦	6	5	4	3	2	1	0	Address	After Reset	R/W
WDTM	RUN	0	0	WDTM4	WDTM3	0	0	0	FFF9H	00H	R/W

RUN	Watchdog Timer Operation Mode Selection[Note 1]
0	Count stop
1	Counter is cleared and counting starts.

WDTM4	WDTM3	Watchdog Timer Operation Mode Selection[Note 2]
0	×	Interval timer mode (Maskable interrupt occurs upon generation of an overflow.)[Note 3]
1	0	Watchdog timer mode 1 (Non-maskable interrupt occurs upon generation of an overflow.)
1	0	Watchdog timer mode 2 (Reset operation is activated upon generation of an overflow.)

Notes 1. Once set to 1, WDTM3 and WDTM4 cannot be cleared to 0 by software.

2. Once set to 1, RUN cannot be cleared to 0 by software.
Thus, once counting starts, it can only be stopped by $\overline{\text{RESET}}$ input.

3. The watchdog timer starts operating as an interval timer as soon as RUN has been set to 1.

Cautions 1. When 1 is set in RUN so that the watchdog timer is cleared, the actual overflow time is up to 0.5% shorter than the time set by timer clock select register 2.

2. To use watchdog timer modes 1 and 2, confirm that the interrupt request flag (TMIF4) is 0 and then set the WDTM4 to 1.
If WDTM4 is set while TMIF4 is 1, the non-maskable interrupt request occurs regardless of the content of WDTM3.

Remark ×: don't care

11.4 Watchdog Timer Operations

11.4.1 Watchdog timer operation

When bit 4 (WDTM4) of the watchdog timer mode register (WDTM) is set to 1, the watchdog timer is operated to detect any inadvertent program loop.

The watchdog timer count clock (inadvertent program loop detection time interval) can be selected with bits 0 to 2 (TCL20 to TCL22) of timer clock select register 2 (TCL2).

Watchdog timer starts by setting bit 7 (RUN) of WDTM to 1. After the watchdog timer is started, set RUN to 1 within the set loop detection time interval. The watchdog timer can be cleared and counting is started by setting RUN to 1. If RUN is not set to 1 and the inadvertent program loop detection time is past, system reset or a non-maskable interrupt request is generated according to the WDTM bit 3 (WDTM3) value.

The watchdog timer continues operating in the HALT mode but it stops in the STOP mode. Thus, set RUN to 1 before the STOP mode is set, clear the watchdog timer and then execute the STOP instruction.

Cautions 1. The actual loop detection time may be shorter than the set time by a maximum of 0.5%.

2. When the subsystem clock is selected for the CPU clock, the watchdog timer count operation is stopped.

Table 11-4. Watchdog Timer Loop Detection Time

TCL22	TCL21	TCL20	Loop Detection Time	MCS = 1	MCS = 0
0	0	0	2^{11} × 1/fxx	2^{11} × 1/fx (410 μs)	2^{12} × 1/fx (819 μs)
0	0	1	2^{12} × 1/fxx	2^{12} × 1/fx (819 μs)	2^{13} × 1/fx (1.64 ms)
0	1	0	2^{13} × 1/fxx	2^{13} × 1/fx (1.64 ms)	2^{14} × 1/fx (3.28 ms)
0	1	1	2^{14} × 1/fxx	2^{14} × 1/fx (3.28 ms)	2^{15} × 1/fx (6.55 ms)
1	0	0	2^{15} × 1/fxx	2^{15} × 1/fx (6.55 ms)	2^{16} × 1/fx (13.1 ms)
1	0	1	2^{16} × 1/fxx	2^{16} × 1/fx (13.1 ms)	2^{17} × 1/fx (26.2 ms)
1	1	0	2^{17} × 1/fxx	2^{17} × 1/fx (26.2 ms)	2^{18} × 1/fx (52.4 ms)
1	1	1	2^{19} × 1/fxx	2^{19} × 1/fx (104.9 ms)	2^{20} × 1/fx (209.7 ms)

Remarks 1. fxx: Main system clock frequency (fx or fx/2)

2. fx: Main system clock oscillation frequency

3. MCS: Oscillation mode select register bit 0

4. Figures in parentheses apply to operation with fx = 5.0 MHz.

11.4.2 Interval timer operation

The watchdog timer operates as an interval timer which generates interrupt requests repeatedly at an interval of the preset count value when bit 4 (WDTM4) of the watchdog timer mode register (WDTM) is set to 0.

The count clock (interval timer) can be selected by bits 0 to 2 (TCL20 to TCL22) of timer clock select register 2 (TCL2). The watchdog timer starts operating as an interval timer when bit 7 (RUN) of WDTM is set to 1.

When the watchdog timer is operated as interval timer, the interrupt mask flag (TMMK4) and priority specify flag (TMPR4) are validated and the maskable interrupt request (INTWDT) can be generated. Among maskable interrupt requests, the INTWDT default has the highest priority.

The interval timer continues operating in the HALT mode but it stops in the STOP mode. Thus, set bit 7 of WDTM (RUN) to 1 before the STOP mode is set, clear the interval timer and then execute the STOP instruction.

Cautions 1. Once bit 4 (WDTM4) of WDTM is set to 1 (with the watchdog timer mode selected), the interval timer mode is not set unless $\overline{\text{RESET}}$ input is applied.

2. The interval time just after setting with WDTM may be shorter than the set time by a maximum of 0.5%.

3. When the subsystem clock is selected for the CPU clock, the watchdog timer count operation is stopped.

Table 11-5. Interval Timer Interval Time

TCL22	TCL21	TCL20	Interval Time	MCS = 1	MCS = 0
0	0	0	$2^{11} \times 1/f_{XX}$	$2^{11} \times 1/f_X$ (410 μs)	$2^{12} \times 1/f_X$ (819 μs)
0	0	1	$2^{12} \times 1/f_{XX}$	$2^{12} \times 1/f_X$ (819 μs)	$2^{13} \times 1/f_X$ (1.64 ms)
0	1	0	$2^{13} \times 1/f_{XX}$	$2^{13} \times 1/f_X$ (1.64 ms)	$2^{14} \times 1/f_X$ (3.28 ms)
0	1	1	$2^{14} \times 1/f_{XX}$	$2^{14} \times 1/f_X$ (3.28 ms)	$2^{15} \times 1/f_X$ (6.55 ms)
1	0	0	$2^{15} \times 1/f_{XX}$	$2^{15} \times 1/f_X$ (6.55 ms)	$2^{16} \times 1/f_X$ (13.1 ms)
1	0	1	$2^{16} \times 1/f_{XX}$	$2^{16} \times 1/f_X$ (13.1 ms)	$2^{17} \times 1/f_X$ (26.2 ms)
1	1	0	$2^{17} \times 1/f_{XX}$	$2^{17} \times 1/f_X$ (26.2 ms)	$2^{18} \times 1/f_X$ (52.4 ms)
1	1	1	$2^{19} \times 1/f_{XX}$	$2^{19} \times 1/f_X$ (104.9 ms)	$2^{20} \times 1/f_X$ (209.7 ms)

Remarks 1. f_{XX}: Main system clock frequency (f_X or $f_X/2$)

2. f_X: Main system clock oscillation frequency

3. MCS: Oscillation mode select register bit 0

4. Figures in parentheses apply to operation with f_X = 5.0 MHz.

CHAPTER 12 CLOCK OUTPUT CONTROLLER

12.1 Clock Output Controller Functions

The clock output controller is intended for carrier output during remote controlled transmission and clock output for supply to peripheral LSI. Clocks selected with timer clock select register 0 (TCL0) are output from the PCL/P35 pin.

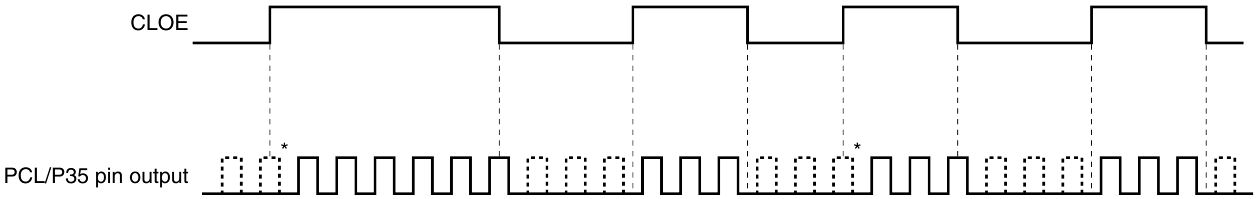
Follow the procedure below to output clock pulses.

(1) Select the clock pulse output frequency (with clock pulse output disabled) with bits 0 to 3 (TCL00 to TCL03) of TCL0.
(2) Set the P35 output latch to 0.
(3) Set bit 5 (PM35) of port mode register 3 to 0 (set to output mode).
(4) Set bit 7 (CLOE) of TCL0 to 1.

Caution Clock output cannot be used when setting P35 output latch to 1.

Remark When clock output enable/disable is switched, the clock output controller does not output pulses with small widths (see the portions marked with * in **Figure 12-1**).

Figure 12-1. Remote Controlled Output Application Example



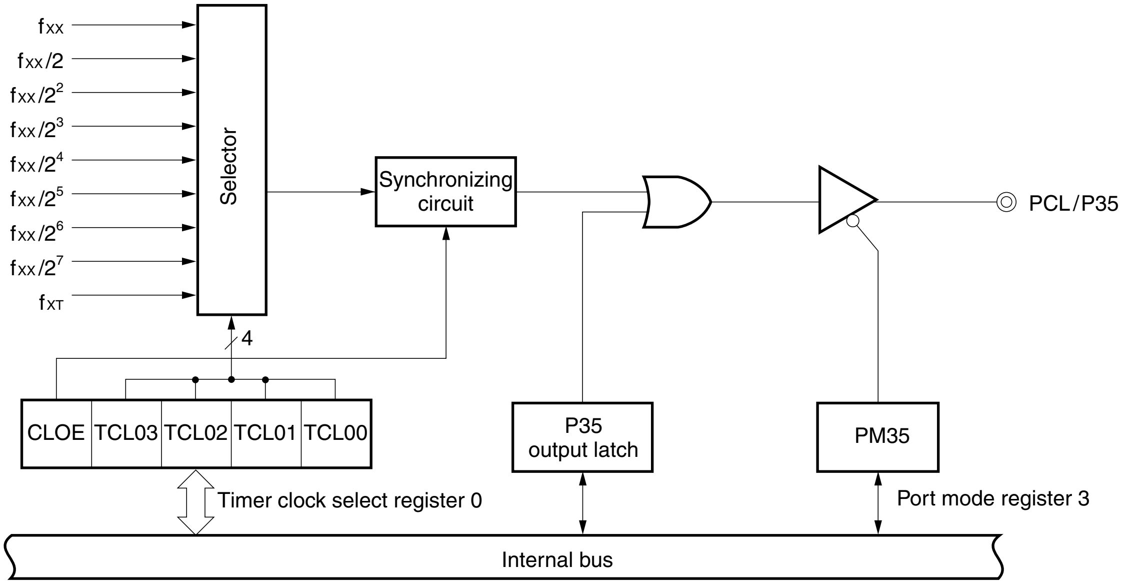
12.2 Clock Output Controller Configuration

The clock output controller consists of the following hardware.

Table 12-1. Clock Output Controller Configuration

Item	Configuration
Control register	Timer clock select register 0 (TCL0) Port mode register 3 (PM3)

Figure 12-2. Clock Output Controller Block Diagram



Remark fxx = fx/2 (MCS = 0), fxx = fx (MCS = 1)

12.3 Clock Output Function Control Registers

The following two types of registers are used to control the clock output function.

- Timer clock select register 0 (TCL0)
- Port mode register 3 (PM3)

(1) Timer clock select register 0 (TCL0)

This register sets PCL output clock.
TCL0 is set with a 1-bit or 8-bit memory manipulation instruction.
$\overline{\text{RESET}}$ input clears TCL0 to 00H.

Remark Besides setting PCL output clock, TCL0 sets the 16-bit timer register count clock.

Figure 12-3. Timer Clock Select Register 0 Format

Symbol	⑦	6	5	4	3	2	1	0	Address	After Reset	R/W
TCL0	CLOE	TCL06	TCL05	TCL04	TCL03	TCL02	TCL01	TCL00	FF40H	00H	R/W

CLOE	PCL Output Control
0	Output disabled
1	Output enabled

TCL06	TCL05	TCL04	16-bit Timer Register Count Clock Selection	
			MCS = 1	MCS = 0
0	0	0	TI00 (Valid edge specifiable)	
0	0	1	Setting prohibited	f_X (5.0 MHz)
0	1	0	f_X (5.0 MHz)	$f_X/2$ (2.5 MHz)
0	1	1	$f_X/2$ (2.5 MHz)	$f_X/2^2$ (1.25 MHz)
1	0	0	$f_X/2^2$ (1.25 MHz)	$f_X/2^3$ (625 kHz)
1	1	1	Watch timer output (INTTM3)	
Other than above			Setting prohibited	

TCL03	TCL02	TCL01	TCL00	PCL Output Clock Selection	
				MCS = 1	MCS = 0
0	0	0	0	f_{XT} (32.768 kHz)	
0	1	0	1	f_X (5.0 MHz)	$f_X/2$ (2.5 MHz)
0	1	1	0	$f_X/2$ (2.5 MHz)	$f_X/2^2$ (1.25 MHz)
0	1	1	1	$f_X/2^2$ (1.25 MHz)	$f_X/2^3$ (625 kHz)
1	0	0	0	$f_X/2^3$ (625 kHz)	$f_X/2^4$ (313 kHz)
1	0	0	1	$f_X/2^4$ (313 kHz)	$f_X/2^5$ (156 kHz)
1	0	1	0	$f_X/2^5$ (156 kHz)	$f_X/2^6$ (78.1 kHz)
1	0	1	1	$f_X/2^6$ (78.1 kHz)	$f_X/2^7$ (39.1 kHz)
1	1	0	0	$f_X/2^7$ (39.1 kHz)	$f_X/2^8$ (19.5 kHz)
Other than above				Setting prohibited	

Cautions

1. **Setting of the TI00/INTP0 pin valid edge is performed by external interrupt mode register 0, and selection of the sampling clock frequency is performed by the sampling clock select register.**
2. **When enabling PCL output, set TCL00 to TCL03, then set 1 in CLOE with a 1-bit memory manipulation instruction.**
3. **To read the count value when TI00 has been specified as the TM0 count clock, the value should be read from TM0, not from capture/compare register 01 (CR01).**
4. **When rewriting TCL0 to other data, stop the timer operation beforehand.**

Remarks **1.** f_X: Main system clock oscillation frequency

2. f_{XT}: Subsystem clock oscillation frequency

3. TI00: 16-bit timer/event counter input pin

4. TM0: 16-bit timer register

5. MCS: Oscillation mode select register bit 0

6. Figures in parentheses apply to operation with f_X = 5.0 MHz or f_{XT} = 32.768 kHz.

(2) Port mode register 3 (PM3)

This register sets port 3 input/output in 1-bit units.

When using the P35/PCL pin for clock output function, set PM35 and output latch of P35 to 0.

PM3 is set with a 1-bit or 8-bit memory manipulation instruction.

$\overline{\text{RESET}}$ input sets PM3 to FFH.

Figure 12-4. Port Mode Register 3 Format

Symbol	7	6	5	4	3	2	1	0	Address	After Reset	R/W
PM3	PM37	PM36	PM35	PM34	PM33	PM32	PM31	PM30	FF23H	FFH	R/W

PM3n	P3n Pin I/O Mode Selection (n = 0 to 7)
0	Output mode (output buffer ON)
1	Input mode (output buffer OFF)

CHAPTER 13 BUZZER OUTPUT CONTROLLER

13.1 Buzzer Output Controller Functions

The buzzer output controller outputs 1.2 kHz, 2.4 kHz, 4.9 kHz, or 9.8 kHz frequency square waves. The buzzer frequency selected with timer clock select register 2 (TCL2) is output from the BUZ/P36 pin.

Follow the procedure below to output the buzzer frequency.

(1) Select the buzzer output frequency with bits 5 to 7 (TCL25 to TCL27) of TCL2.
(2) Set the P36 output latch to 0.
(3) Set bit 6 (PM36) of port mode register 3 to 0 (set to output mode).

Caution Buzzer output cannot be used when setting P36 output latch to 1.

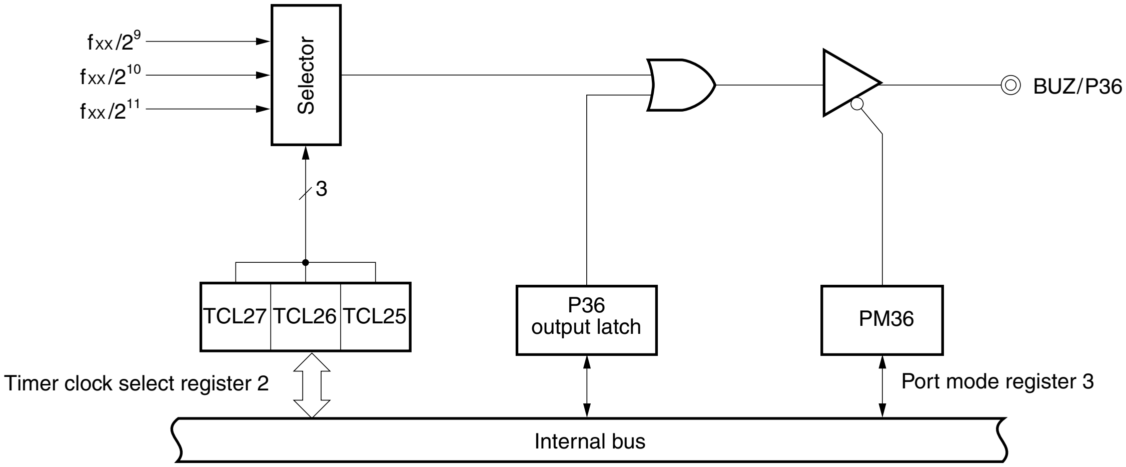
13.2 Buzzer Output Controller Configuration

The buzzer output controller consists of the following hardware.

Table 13-1. Buzzer Output Controller Configuration

Item	Configuration
Control register	Timer clock select register 2 (TCL2) Port mode register 3 (PM3)

Figure 13-1. Buzzer Output Controller Block Diagram



Remark fxx = fx/2 (MCS = 0), fxx = fx (MCS = 1)

13.3 Buzzer Output Function Control Registers

The following two types of registers are used to control the buzzer output function.

- Timer clock select register 2 (TCL2)
- Port mode register 3 (PM3)

(1) Timer clock select register 2 (TCL2)

This register sets the buzzer output frequency.
TCL2 is set with an 8-bit memory manipulation instruction.
$\overline{\text{RESET}}$ input clears TCL2 to 00H.

Remark Besides setting the buzzer output frequency, TCL2 sets the watch timer count clock and the watchdog timer count clock.

Figure 13-2. Timer Clock Select Register 2 Format

Symbol	7	6	5	4	3	2	1	0	Address	After Reset	R/W
TCL2	TCL27	TCL26	TCL25	TCL24	0	TCL22	TCL21	TCL20	FF42H	00H	R/W

TCL27	TCL26	TCL25	Buzzer Output Frequency Selection	
			MCS = 1	MCS = 0
0	×	×	Buzzer output disable	
1	0	0	$f_X/2^9$ (9.8 kHz)	$f_X/2^{10}$ (4.9 kHz)
1	0	1	$f_X/2^{10}$ (4.9 kHz)	$f_X/2^{11}$ (2.4 kHz)
1	1	0	$f_X/2^{11}$ (2.4 kHz)	$f_X/2^{12}$ (1.2 kHz)
1	1	1	Setting prohibited	

TCL24	Watch Timer Count Clock Selection	
	MCS = 1	MCS = 0
0	$f_X/2^7$ (39.1 kHz)	$f_X/2^8$ (19.5 kHz)
1	f_{XT} (32.768 kHz)	

TCL22	TCL21	TCL20	Watchdog Timer Count Clock Selection	
			MCS = 1	MCS = 0
0	0	0	$f_X/2^3$ (625 kHz)	$f_X/2^4$ (313 kHz)
0	0	1	$f_X/2^4$ (313 kHz)	$f_X/2^5$ (156 kHz)
0	1	0	$f_X/2^5$ (156 kHz)	$f_X/2^6$ (78.1 kHz)
0	1	1	$f_X/2^6$ (78.1 kHz)	$f_X/2^7$ (39.1 kHz)
1	0	0	$f_X/2^7$ (39.1 kHz)	$f_X/2^8$ (19.5 kHz)
1	0	1	$f_X/2^8$ (19.5 kHz)	$f_X/2^9$ (9.8 kHz)
1	1	0	$f_X/2^9$ (9.8 kHz)	$f_X/2^{10}$ (4.9 kHz)
1	1	1	$f_X/2^{11}$ (2.4 kHz)	$f_X/2^{12}$ (1.2 kHz)

Caution When rewriting TCL2 to other data, stop the timer operation beforehand.

Remarks 1. f_X: Main system clock oscillation frequency

2. f_{XT}: Subsystem clock oscillation frequency

3. ×: don't care

4. MCS: Oscillation mode select register bit 0

5. Figures in parentheses apply to operation with f_X = 5.0 MHz or f_{XT} = 32.768 kHz.

(2) Port mode register 3 (PM3)

This register sets port 3 input/output in 1-bit units.

When using the P36/BUZ pin for buzzer output function, set PM36 and output latch of P36 to 0.

PM3 is set with a 1-bit or 8-bit memory manipulation instruction.

$\overline{\text{RESET}}$ input sets PM3 to FFH.

Figure 13-3. Port Mode Register 3 Format

Symbol	7	6	5	4	3	2	1	0	Address	After Reset	R/W
PM3	PM37	PM36	PM35	PM34	PM33	PM32	PM31	PM30	FF23H	FFH	R/W

PM3n	P3n Pin I/O Mode Selection (n = 0 to 7)
0	Output mode (output buffer ON)
1	Input mode (output buffer OFF)

CHAPTER 14 A/D CONVERTER

14.1 A/D Converter Functions

The A/D converter converts an analog input into a digital value. It consists of 8 channels (ANI0 to ANI7) with an 8-bit resolution.

The conversion method is based on successive approximation and the conversion result is held in the 8-bit A/D conversion result register (ADCR).

The following two ways are available to start A/D conversion.

(1) Hardware start

Conversion is started by trigger input (INTP3).

(2) Software start

Conversion is started by setting the A/D converter mode register.

Select one channel of analog input from ANI0 to ANI7 to execute A/D conversion. In the case of hardware start, A/D conversion operation stops and an interrupt request (INTAD) is generated when the conversion operation ends. In the case of software start, the conversion operation is repeated. Each time the conversion operation ends, INTAD is generated.

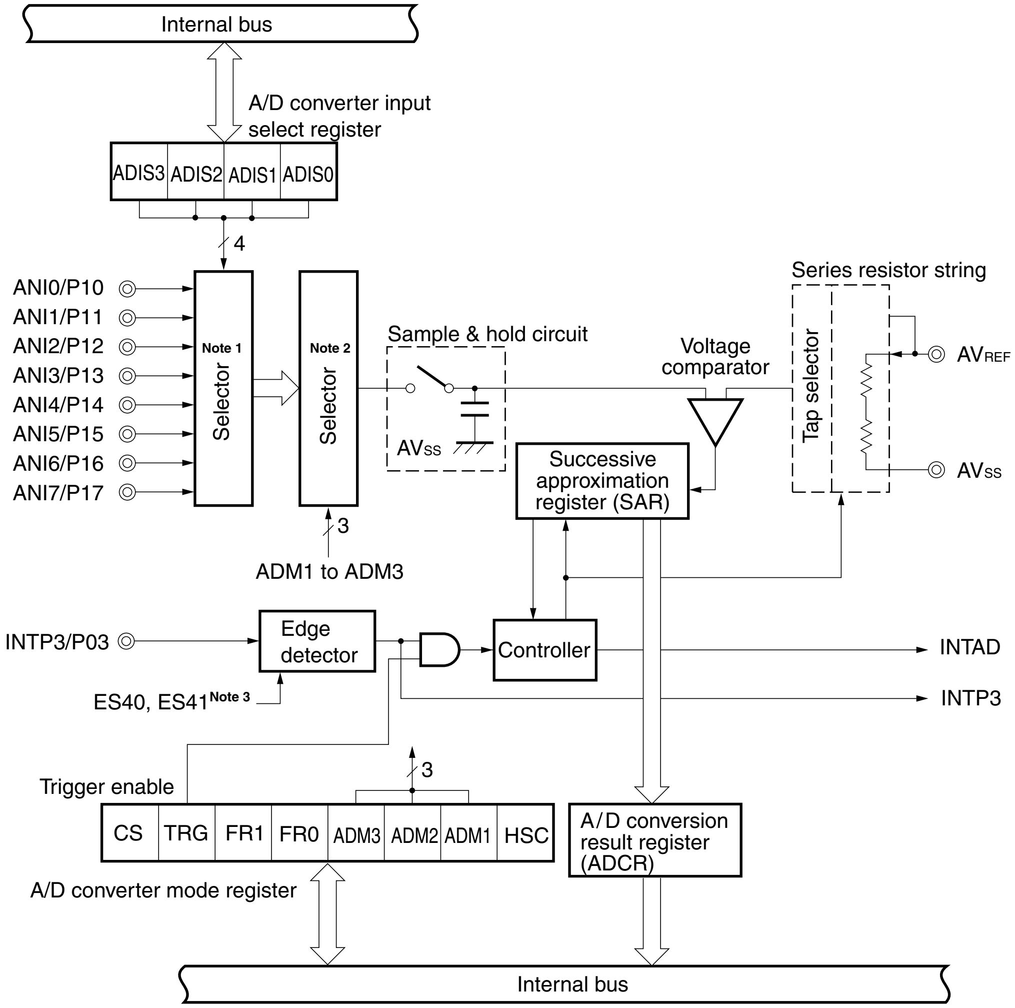
14.2 A/D Converter Configuration

The A/D converter consists of the following hardware.

Table 14-1. A/D Converter Configuration

Item	Configuration
Analog input	8 channels (ANI0 to ANI7)
Control register	A/D converter mode register (ADM) A/D converter input select register (ADIS) External interrupt mode register 1 (INTM1)
Register	Successive approximation register (SAR) A/D conversion result register (ADCR)

Figure 14-1. A/D Converter Block Diagram



Notes
1. Selector to select the number of channels to be used for analog input.
2. Selector to select the channel for A/D conversion.
3. Bits 0 and 1 of external interrupt mode register 1 (INTM1)

(1) Successive approximation register (SAR)

This register compares the analog input voltage value to the voltage tap (compare voltage) value applied from the series resistor string and holds the result from the most significant bit (MSB).

When up to the least significant bit (LSB) is set (termination of A/D conversion), the SAR contents are transferred to the A/D conversion result register.

(2) A/D conversion result register (ADCR)

This register holds the A/D conversion result. Each time A/D conversion terminates, the conversion result is loaded from the successive approximation register.

ADCR is read with an 8-bit memory manipulation instruction.

$\overline{\text{RESET}}$ input makes ADCR undefined.

(3) Sample & hold circuit

The sample & hold circuit samples each analog input signal sequentially applied from the input circuit and sends it to the voltage comparator. This circuit holds the sampled analog input voltage value during A/D conversion.

(4) Voltage comparator

The voltage comparator compares the analog input to the series resistor string output voltage.

(5) Series resistor string

The series resistor string is connected between AV_{REF} and AV_{SS} and generates a voltage to be compared with the analog input.

(6) ANI0 to ANI7 pins

These are 8-channel analog input pins to input analog signals to undergo A/D conversion to the A/D converter. Pins other than those selected as analog input by the A/D converter input select register (ADIS) can be used as I/O ports.

Caution Use ANI0 to ANI7 input voltages within the specified range. If a voltage higher than AV_{REF} or lower than AV_{SS} is applied (even if within the absolute maximum ratings), the converted value of the corresponding channel becomes indeterminate and may adversely affect the converted values of other channels.

(7) AV_{REF} pin

This pin inputs the A/D converter reference voltage.

It converts signals input to ANI0 to ANI7 into digital signals according to the voltage applied between AV_{REF} and AV_{SS}.

The current flowing in the series resistor string can be reduced by setting the voltage to be input to the AV_{REF} pin to AV_{SS} level in standby mode.

The AV_{REF} pin also functions to supply analog power to the A/D converter. When using the A/D converter, be sure to supply power to the AV_{REF} pin.

Caution When making the voltage applied to the AV_{REF} pin the same level as that of AV_{SS}, be sure to clear bit 7 (CS) of the A/D converter mode register (ADM) to 0.

(8) AV_{SS} pin

This is a GND potential pin of the A/D converter. Keep it at the same potential as the V_{SS0} pin when not using the A/D converter.

14.3 A/D Converter Control Registers

The following three types of registers are used to control the A/D converter.

- A/D converter mode register (ADM)
- A/D converter input select register (ADIS)
- External interrupt mode register 1 (INTM1)

(1) A/D converter mode register (ADM)

This register sets the analog input channel for A/D conversion, conversion time, conversion start/stop and external trigger.

ADM is set with a 1-bit or 8-bit memory manipulation instruction.

$\overline{\text{RESET}}$ input sets ADM to 01H.

Figure 14-2. A/D Converter Mode Register Format

Symbol	⑦	⑥	5	4	3	2	1	0	Address	After Reset	R/W
ADM	CS	TRG	FR1	FR0	ADM3	ADM2	ADM1	HSC	FF80H	01H	R/W

CS	A/D Conversion Operation Control
0	Operation stop
1	Operation start

TRG	External Trigger Selection
0	No external trigger (software starts)
1	Conversion started by external trigger (hardware starts)

FR1	FR0	HSC	A/D Conversion Time Selection[Note 1]			
			f_X = 5.0 MHz Operation		f_X = 4.19 MHz Operation	
			MCS = 1	MCS = 0	MCS = 1	MCS = 0
0	0	1	80/f_X (Setting prohibited[Note 2])	160/f_X (32.0 μs)	80/f_X (19.1 μs)	160/f_X (38.1 μs)
0	1	1	40/f_X (Setting prohibited[Note 2])	80/f_X (Setting prohibited[Note 2])	40/f_X (Setting prohibited[Note 2])	80/f_X (19.1 μs)
1	0	0	50/f_X (Setting prohibited[Note 2])	100/f_X (20.0 μs)	50/f_X (Setting prohibited[Note 2])	100/f_X (23.8 μs)
1	0	1	100/f_X (20.0 μs)	200/f_X (40.0 μs)	100/f_X (23.8 μs)	200/f_X (47.7 μs)
Other than above			Setting prohibited			

ADM3	ADM2	ADM1	Analog Input Channel Selection
0	0	0	ANI0
0	0	1	ANI1
0	1	0	ANI2
0	1	1	ANI3
1	0	0	ANI4
1	0	1	ANI5
1	1	0	ANI6
1	1	1	ANI7

Notes 1. Set so that the A/D conversion time is 19.1 μs or more.

2. Setting prohibited because A/D conversion time is less than 19.1 μs.

Cautions 1. The following sequence is recommended for power consumption reduction of A/D converter when the standby function is used: Clear bit 7 (CS) to 0 first to stop the A/D conversion operation, and then execute the HALT or STOP instruction.

2. When restarting the stopped A/D conversion operation, start the A/D conversion operation after clearing the interrupt request flag (ADIF) to 0.

Remarks 1. f_X: Main system clock oscillation frequency

2. MCS: Oscillation mode select register bit 0

(2) A/D converter input select register (ADIS)

This register determines whether the ANI0/P10 to ANI7/P17 pins should be used for analog input channels or ports. Pins other than those selected as analog input can be used as I/O ports.

ADIS is set with an 8-bit memory manipulation instruction.

$\overline{\text{RESET}}$ input clears ADIS to 00H.

Cautions 1. Set the analog input channel in the following order.

(1) Set the number of analog input channels with ADIS.

(2) Using A/D converter mode register (ADM), select one channel to undergo A/D conversion from among the channels set for analog input with ADIS.

2. No internal pull-up resistor can be used to the channels set for analog input with ADIS, irrespective of the value of bit 1 (PUO1) of pull-up resistor option register L.

Figure 14-3. A/D Converter Input Select Register Format

Symbol	7	6	5	4	3	2	1	0	Address	After Reset	R/W
ADIS	0	0	0	0	ADIS3	ADIS2	ADIS1	ADIS0	FF84H	00H	R/W

ADIS3	ADIS2	ADIS1	ADIS0	Number of Analog Input Channel Selection
0	0	0	0	No analog input channel (P10 to P17)
0	0	0	1	1 channel (ANI0, P11 to P17)
0	0	1	0	2 channel (ANI0, ANI1, P12 to P17)
0	0	1	1	3 channel (ANI0 to ANI2, P13 to P17)
0	1	0	0	4 channel (ANI0 to ANI3, P14 to P17)
0	1	0	1	5 channel (ANI0 to ANI4, P15 to P17)
0	1	1	0	6 channel (ANI0 to ANI5, P16, P17)
0	1	1	1	7 channel (ANI0 to ANI6, P17)
1	0	0	0	8 channel (ANI0 to ANI7)
Other than above				Setting prohibited

(3) External interrupt mode register 1 (INTM1)

This register sets the valid edge for INTP3 to INTP5.

INTM1 is set with an 8-bit memory manipulation instruction.

$\overline{\text{RESET}}$ input clears INTM1 to 00H.

Figure 14-4. External Interrupt Mode Register 1 Format

Symbol	7	6	5	4	3	2	1	0	Address	After Reset	R/W
INTM1	0	0	ES61	ES60	ES51	ES50	ES41	ES40	FFEDH	00H	R/W

ES61	ES60	INTP5 Valid Edge Selection
0	0	Falling edge
0	1	Rising edge
1	0	Setting prohibited
1	1	Both falling and rising edges

ES51	ES50	INTP4 Valid Edge Selection
0	0	Falling edge
0	1	Rising edge
1	0	Setting prohibited
1	1	Both falling and rising edges

ES41	ES40	INTP3 Valid Edge Selection
0	0	Falling edge
0	1	Rising edge
1	0	Setting prohibited
1	1	Both falling and rising edges

14.4 A/D Converter Operations

14.4.1 Basic operations of A/D converter

(1) Set the number of analog input channels with A/D converter input select register (ADIS).

(2) From among the analog input channels set with ADIS, select one channel for A/D conversion with A/D converter mode register (ADM).

(3) The sample & hold circuit samples the voltage input to the selected analog input channel.

(4) Sampling for the specified period of time sets the sample & hold circuit to the hold state so that the circuit holds the input analog voltage until termination of A/D conversion.

<R> (5) Set bit 7 (CS) of the A/D converter mode register (ADM). Bit 7 of the successive approximation register (SAR) is automatically set, and the series resistor string voltage tap is set to (1/2) AV_{REF} by the tap selector.

(6) The voltage difference between the series resistor string voltage tap and analog input is compared with a voltage comparator. If the analog input is greater than (1/2) AV_{REF}, the MSB of SAR remains set. If the input is smaller than (1/2) AV_{REF}, the MSB is reset.

(7) Next, bit 6 of SAR is automatically set and the operation proceeds to the next comparison. In this case, the series resistor string voltage tap is selected according to the preset value of bit 7 as described below.

- Bit 7 = 1 : (3/4) AV_{REF}
- Bit 7 = 0 : (1/4) AV_{REF}

The voltage tap and analog input voltage are compared and bit 6 of SAR is manipulated with the result as follows.

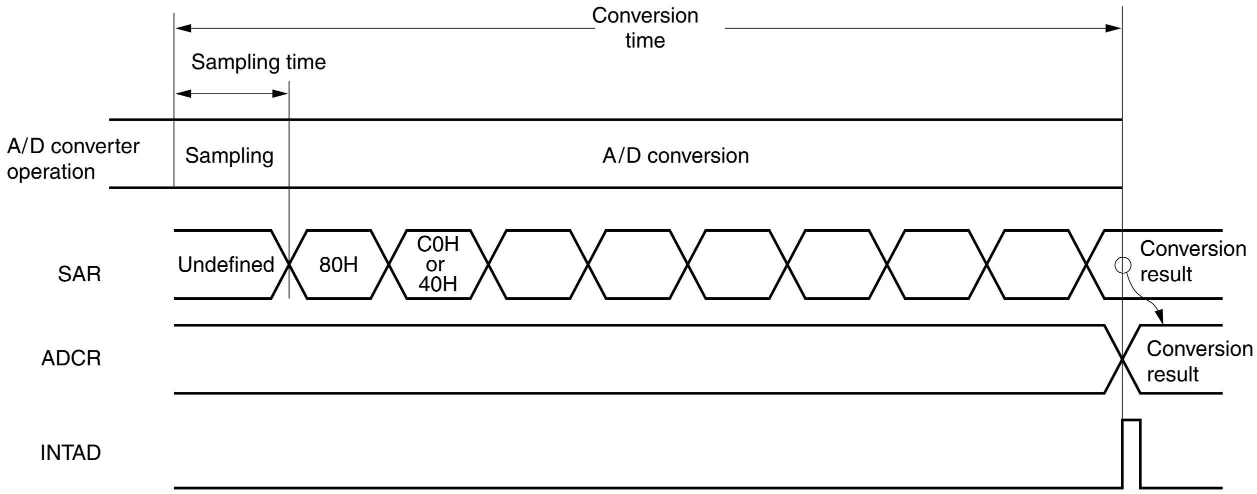
- Analog input voltage ≥ Voltage tap : Bit 6 = 1
- Analog input voltage ≤ Voltage tap : Bit 6 = 0

(8) Comparison of this sort continues up to bit 0 of SAR.

(9) Upon completion of the comparison of 8 bits, any effective digital resultant value remains in SAR and the resultant value is transferred to and latched in the A/D conversion result register (ADCR).

At the same time, the A/D conversion termination interrupt request (INTAD) can also be generated.

Figure 14-5. A/D Converter Basic Operation



A/D conversion operations are performed continuously until bit 7 (CS) of ADM is reset (to 0) by software.

If a write to the ADM register is performed during an A/D conversion operation, the conversion operation is initialized, and if the CS bit is set (to 1), conversion starts again from the beginning.

After $\overline{\text{RESET}}$ input, the value of ADCR is undefined.

14.4.2 Input voltage and conversion results

The relation between the analog input voltage input to the analog input pins (ANI0 to ANI7) and the A/D conversion result (the value stored in A/D conversion result register (ADCR)) is shown by the following expression.

$$\mathrm{ADCR} = \mathrm{INT}\ (\frac{V_{IN}}{AV_{REF}} \times 256 + 0.5)$$

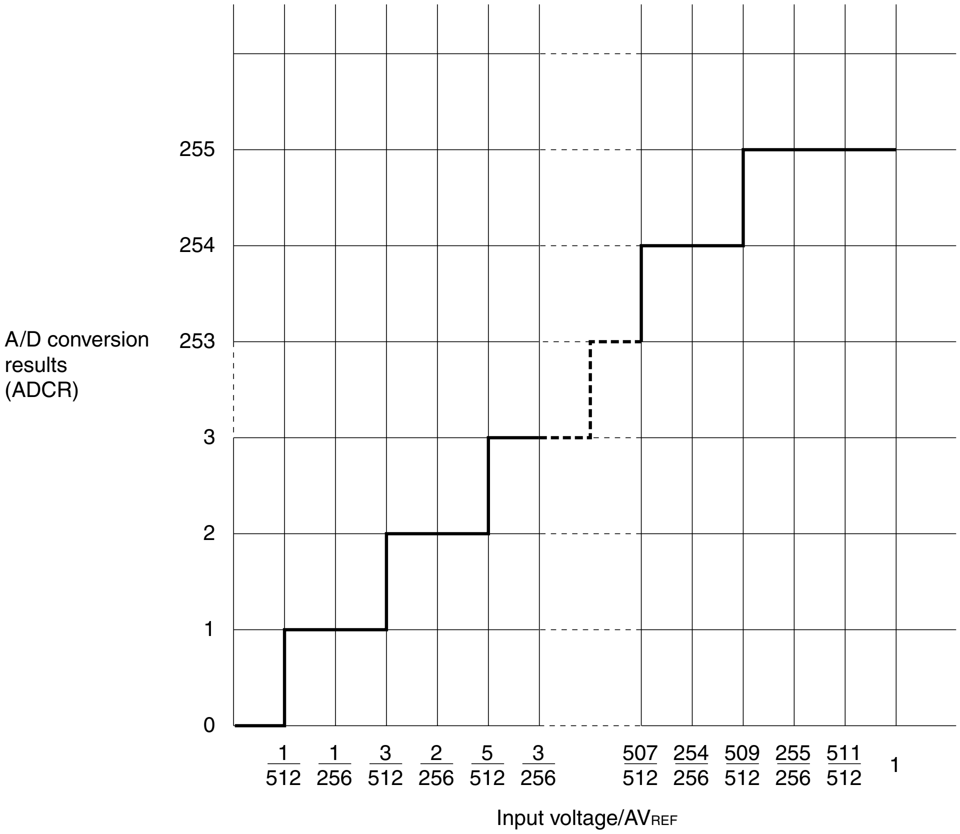
or

$$(\mathrm{ADCR} - 0.5) \times \frac{AV_{REF}}{256} \leq V_{IN} < (\mathrm{ADCR} + 0.5) \times \frac{AV_{REF}}{256}$$

INT( ): Function which returns integer parts of value in parentheses.
V_{IN}: Analog input voltage
AV_{REF}: AV_{REF} pin voltage
ADCR: A/D conversion result register (ADCR) value

Figure 14-6 shows the relation between the analog input voltage and the A/D conversion result.

Figure 14-6. Relationships Between Analog Input Voltage and A/D Conversion Result



14.4.3 A/D converter operating mode

One analog input channel is selected from among ANI0 to ANI7 with the A/D converter input select register (ADIS) and A/D converter mode register (ADM) and starts A/D conversion.

The following two ways are available to start A/D conversion.

- Hardware start: Conversion is started by trigger input (INTP3).
- Software start: Conversion is started by setting ADM.

The A/D conversion result is stored in the A/D conversion result register (ADCR) and the interrupt request signal (INTAD) is simultaneously generated.

(1) A/D conversion by hardware start

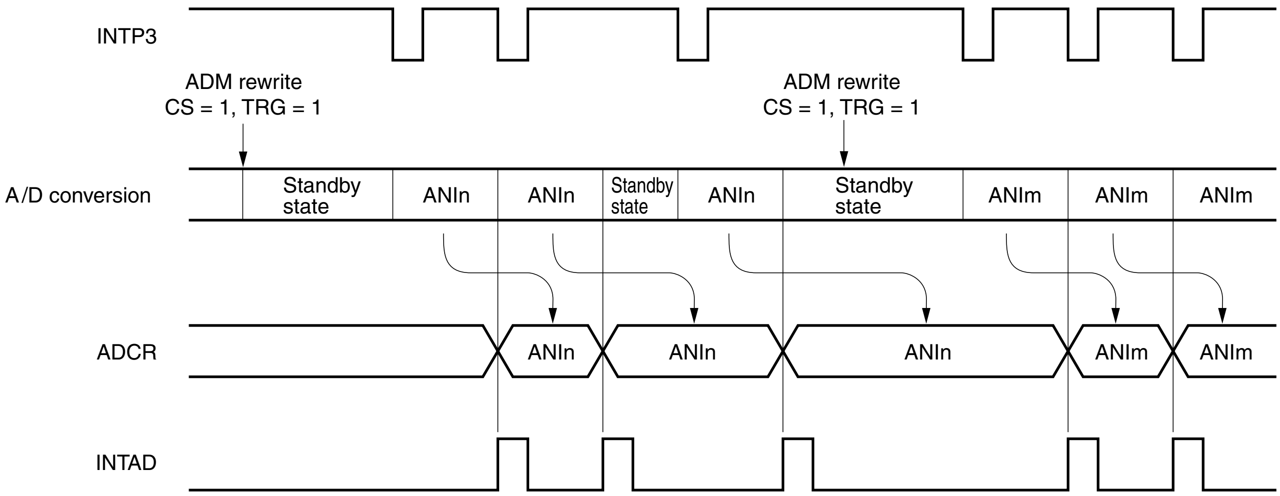
When bit 6 (TRG) and bit 7 (CS) of A/D converter mode register (ADM) are set to 1, the A/D conversion standby state is set. When the external trigger signal (INTP3) is input, the A/D conversion starts on the voltage applied to the analog input pins specified with bits 1 to 3 (ADM1 to ADM3) of ADM.

Upon termination of the A/D conversion, the conversion result is stored in the A/D conversion result register (ADCR) and the interrupt request signal (INTAD) is generated. After one A/D conversion operation is started and terminated, another operation is not started until a new external trigger signal is input.

If data with CS set to 1 is written to ADM again during A/D conversion, the converter suspends its A/D conversion operation and waits for a new external trigger signal to be input. When the external trigger input signal is reinput, A/D conversion is carried out from the beginning.

If data with CS set to 0 is written to ADM during A/D conversion, the A/D conversion operation stops immediately.

Figure 14-7. A/D Conversion by Hardware Start



Remarks 1. n = 0, 1, ... , 7

2. m = 0, 1, ... , 7

(2) A/D conversion by software start

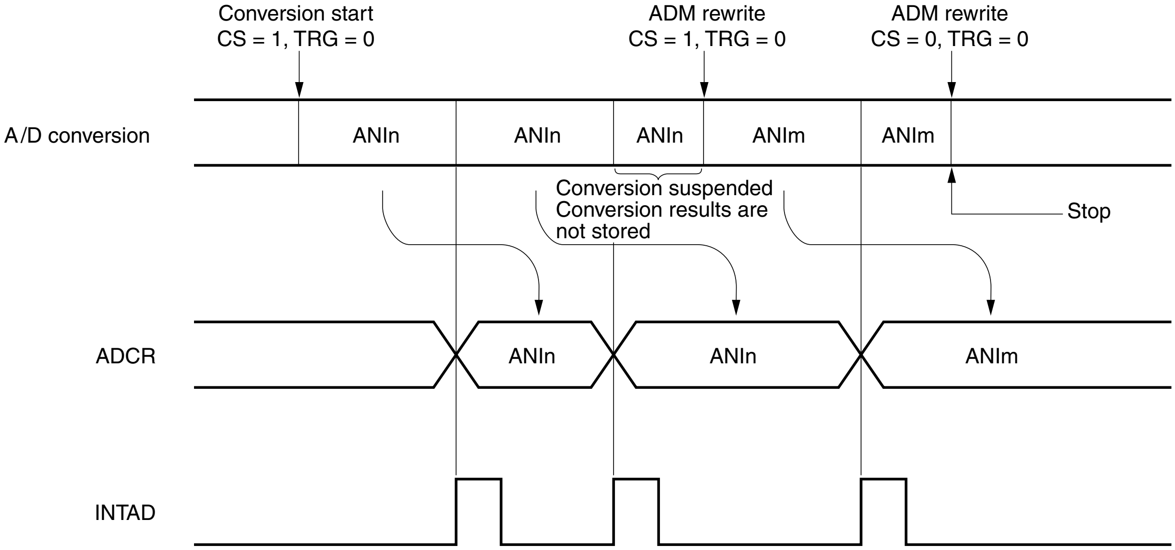
When bit 6 (TRG) and bit 7 (CS) of A/D converter mode register (ADM) are set to 0 and 1, respectively, the A/D conversion starts on the voltage applied to the analog input pins specified with bits 1 to 3 (ADM1 to ADM3) of ADM.

Upon termination of the A/D conversion, the conversion result is stored in the A/D conversion result register (ADCR) and the interrupt request signal (INTAD) is generated. After one A/D conversion operation is started and terminated, the next A/D conversion operation starts immediately. The A/D conversion operation continues repeatedly until new data is written to ADM.

If data with CS set to 1 is written to ADM again during A/D conversion, the converter suspends its A/D conversion operation and starts A/D conversion on the newly written data.

If data with CS set to 0 is written to ADM during A/D conversion, the A/D conversion operation stops immediately.

Figure 14-8. A/D Conversion by Software Start



Remarks 1. n = 0, 1, ... , 7

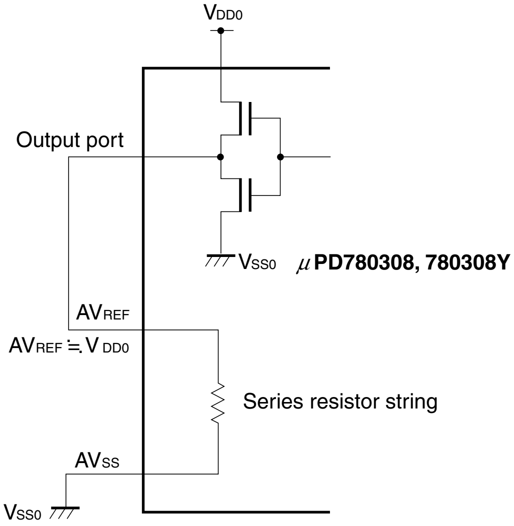
2. m = 0, 1, ... , 7

14.5 A/D Converter Cautions

(1) Current consumption in standby mode

The A/D converter operates on the main system clock. Therefore, its operation stops in STOP mode or in HALT mode with the subsystem clock. As a current still flows in the AV_{REF} pin at this time, this current must be cut in order to minimize the overall system power dissipation. In Figure 14-9, the power dissipation can be reduced by outputting a low-level signal to the output port in standby mode. However, there is no precision to the actual AV_{REF} voltage, and therefore the conversion values themselves lack precision and can only be used for relative comparison.

Figure 14-9. Example of Method of Reducing Current Consumption in Standby Mode



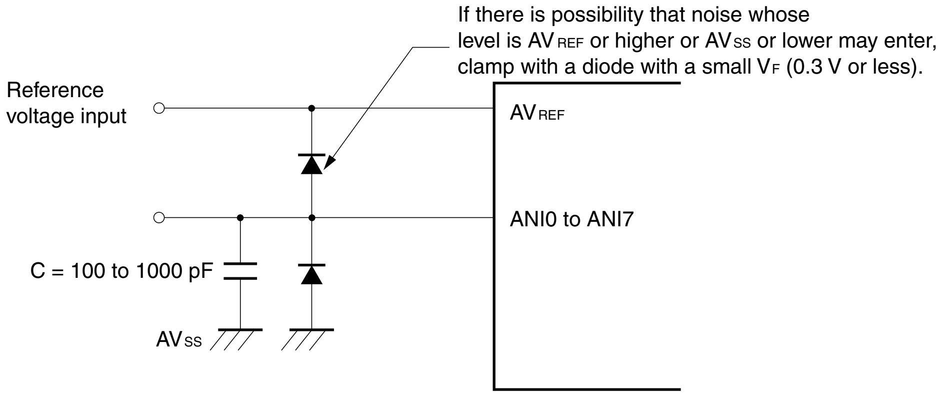
(2) Input range of ANI0 to ANI7

The input voltages of ANI0 to ANI7 should be within the specification range. In particular, if a voltage above AV_{REF} or below AV_{SS} is input (even if within the absolute maximum rating range), the conversion value for that channel will be indeterminate. The conversion values of the other channels may also be affected.

(3) Noise countermeasures

In order to maintain 8-bit resolution, attention must be paid to noise on pins AV_{REF} and ANI0 to ANI7. Since the effect increases in proportion to the output impedance of the analog input source, it is recommended that a capacitor be connected externally as shown in Figure 14-10 in order to reduce noise.

Figure 14-10. Analog Input Pin Disposition


If there is possibility that noise whose
level is AVREF or higher or AVSS or lower may enter,
clamp with a diode with a small VF (0.3 V or less).
Reference
voltage input
AVREF
ANI0 to ANI7
C = 100 to 1000 pF
AVSS

(4) Pins ANI0/P10 to ANI7/P17

The analog input pins ANI0 to ANI7 also function as I/O port (PORT1) pins. When A/D conversion is performed with any of pins ANI0 to ANI7 selected, be sure not to execute a PORT1 input instruction while conversion is in progress, as this may reduce the conversion resolution.

Also, if digital pulses are applied to a pin adjacent to the pin in the process of A/D conversion, the expected A/D conversion value may not be obtainable due to coupling noise. Therefore, avoid applying pulses to pins adjacent to the pin undergoing A/D conversion.

(5) AV_{REF} pin input impedance

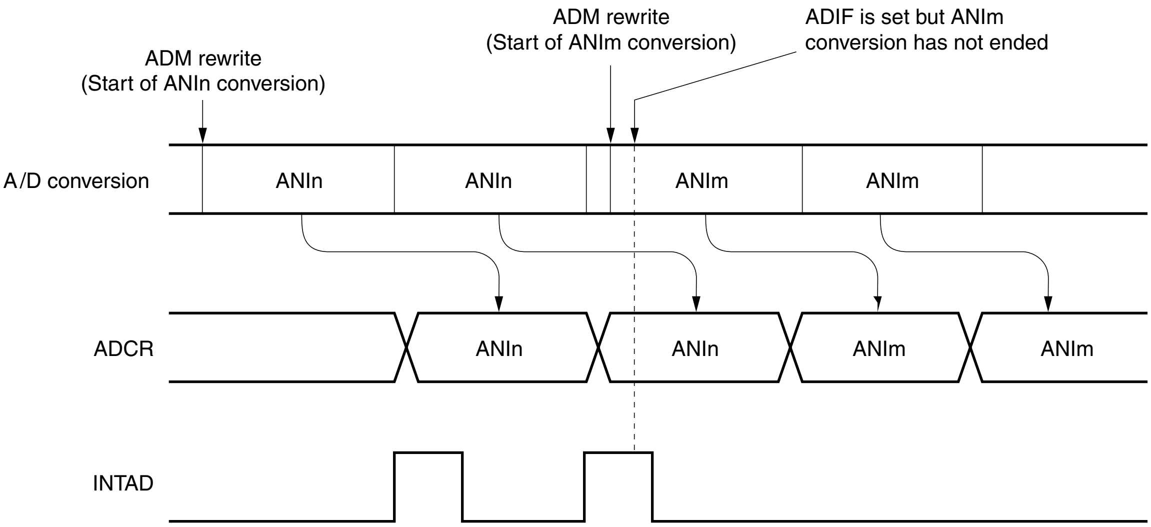
A series resistor string of approximately 10 kΩ is connected between the AV_{REF} pin and the AV_{SS} pin. Therefore, if the output impedance of the reference voltage source is high, this will result in parallel connection to the series resistor string between the AV_{REF} pin and the AV_{SS} pin, and there will be a large reference voltage error.

(6) Interrupt request flag (ADIF)

The interrupt request flag (ADIF) is not cleared even if the A/D converter mode register (ADM) is changed. Caution is therefore required since, if a change of analog input pin is performed during A/D conversion, the A/D conversion result and conversion end interrupt request flag for the pre-change analog input may be set just before the ADM rewrite, and when ADIF is read immediately after the ADM rewrite, ADIF may be set despite the fact that the A/D conversion for the post-change analog input has not ended.

When the A/D conversion is stopped and then resumed, clear the ADIF before it is resumed.

Figure 14-11. A/D Conversion End Interrupt Generation Timing



CHAPTER 15 SERIAL INTERFACE CHANNEL 0 (μPD780308 SUBSERIES)

The μPD780308 Subseries incorporates three channels of serial interfaces. Differences between channels 0, 2, and 3 are as follows (refer to **CHAPTER 17 SERIAL INTERFACE CHANNEL 2** for details of serial interface channel 2, and **CHAPTER 18 SERIAL INTERFACE CHANNEL 3** for details of serial interface channel 3, respectively).

Table 15-1. Differences Between Channels 0, 2, and 3

Serial Transfer Mode		Channel 0	Channel 2	Channel 3
3-wire serial I/O	Clock selection	$f_{xx}/2$, $f_{xx}/2^2$, $f_{xx}/2^3$, $f_{xx}/2^4$, $f_{xx}/2^5$, $f_{xx}/2^6$, $f_{xx}/2^7$, $f_{xx}/2^8$, external clock, TO2 output	External clock, baud rate generator output	$f_{xx}/2$, $f_{xx}/2^2$, $f_{xx}/2^3$, $f_{xx}/2^4$, $f_{xx}/2^5$, $f_{xx}/2^6$, $f_{xx}/2^7$, $f_{xx}/2^8$, external clock
	Transfer method	MSB/LSB switchable as the start bit	MSB/LSB switchable as the start bit	MSB/LSB switchable as the start bit
	Transfer end flag	Serial transfer end interrupt request flag (CSIIF0)	Serial transfer end interrupt request flag (SRIF)	Serial transfer end interrupt request flag (CSIIF3)
SBI (serial bus interface)		Use possible	None	None
2-wire serial I/O				
UART (Asynchronous serial interface)		None	Use possible	None

15.1 Serial Interface Channel 0 Functions

Serial interface channel 0 employs the following four modes.

- Operation stop mode
- 3-wire serial I/O mode
- SBI (serial bus interface) mode
- 2-wire serial I/O mode

Caution Do not change the operating mode (3-wire serial I/O, 2-wire serial I/O, or SBI) while serial interface channel 0 is enabled. To change the operating mode, stop the serial operation once.

(1) Operation stop mode

This mode is used when serial transfer is not carried out. Power consumption can be reduced.

(2) 3-wire serial I/O mode (MSB-/LSB-first selectable)

This mode is used for 8-bit data transfer using three lines, one each for serial clock ($\overline{\text{SCK0}}$), serial output (SO0) and serial input (SI0). This mode enables simultaneous transmission/reception and therefore reduces the data transfer processing time.

The start bit of transferred 8-bit data is switchable between MSB and LSB, so that devices can be connected regardless of their start bit recognition.

This mode should be used when connecting with peripheral I/O devices or display controllers which incorporate a conventional synchronous clocked serial interface as is the case with the 75X/XL, 78K, and 17K Series.

(3) SBI (serial bus interface) mode (MSB-first)

This mode is used for 8-bit data transfer with two or more devices using two lines of serial clock ($\overline{\text{SCK0}}$) and serial data bus (SB0 or SB1).

The SBI mode conforms to the NEC serial bus format, and transfers or receives three types of data: “addresses”, “commands”, and “data”.

- Address: Data to select the target device for serial communication
- Command: Data to give an instruction to the target device
- Data: Data actually transferred

Actually, the master device outputs an “address” to the serial bus to select one of the slave devices with which the master device is to communicate. After that, “commands” and “data” are transferred or received between the master and slave devices. The receiver can automatically identify the received data as an “address”, “command”, or “data” by hardware.

This function enables the I/O ports to be used effectively and the application program serial interface control portions to be simplified.

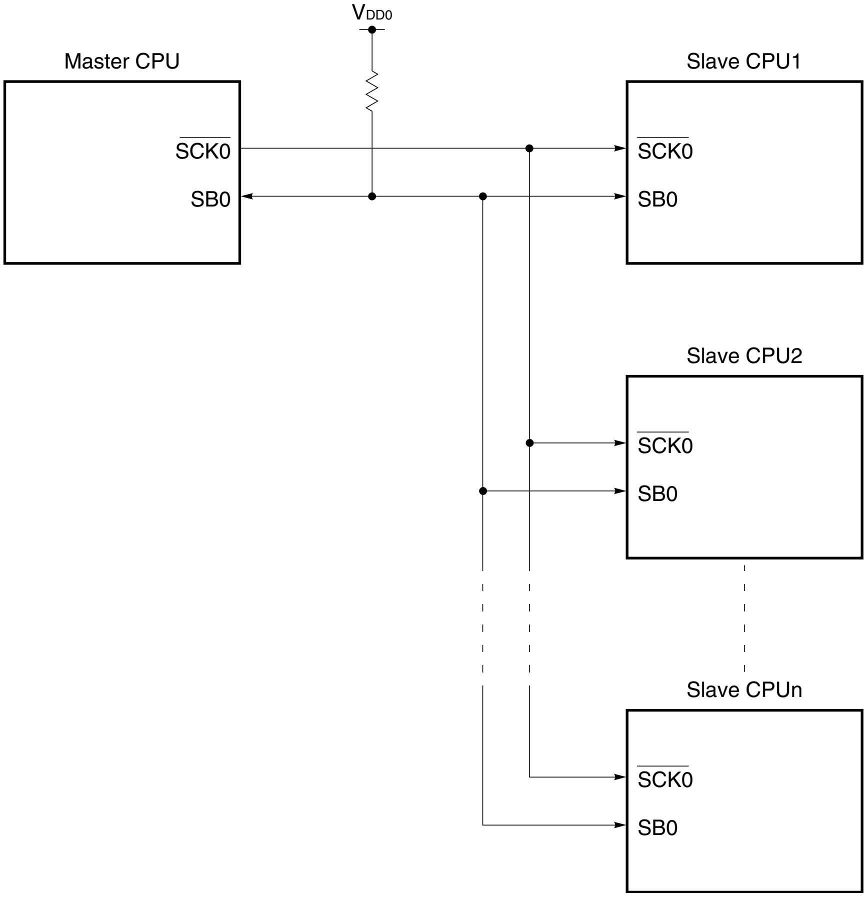
In this mode, the wake-up function for handshake and the output function of acknowledge and busy signals can also be used.

(4) 2-wire serial I/O mode (MSB-first)

This mode is used for 8-bit data transfer using two lines of serial clock ($\overline{\text{SCK0}}$) and serial data bus (SB0 or SB1).

This mode enables to cope with any one of the possible data transfer formats by controlling the $\overline{\text{SCK0}}$ level and the SB0 or SB1 output level. Thus, the handshake line previously necessary for connection of two or more devices can be removed, resulting in the increased number of available I/O ports.

Figure 15-1. Serial Bus Interface (SBI) System Configuration Example



15.2 Serial Interface Channel 0 Configuration

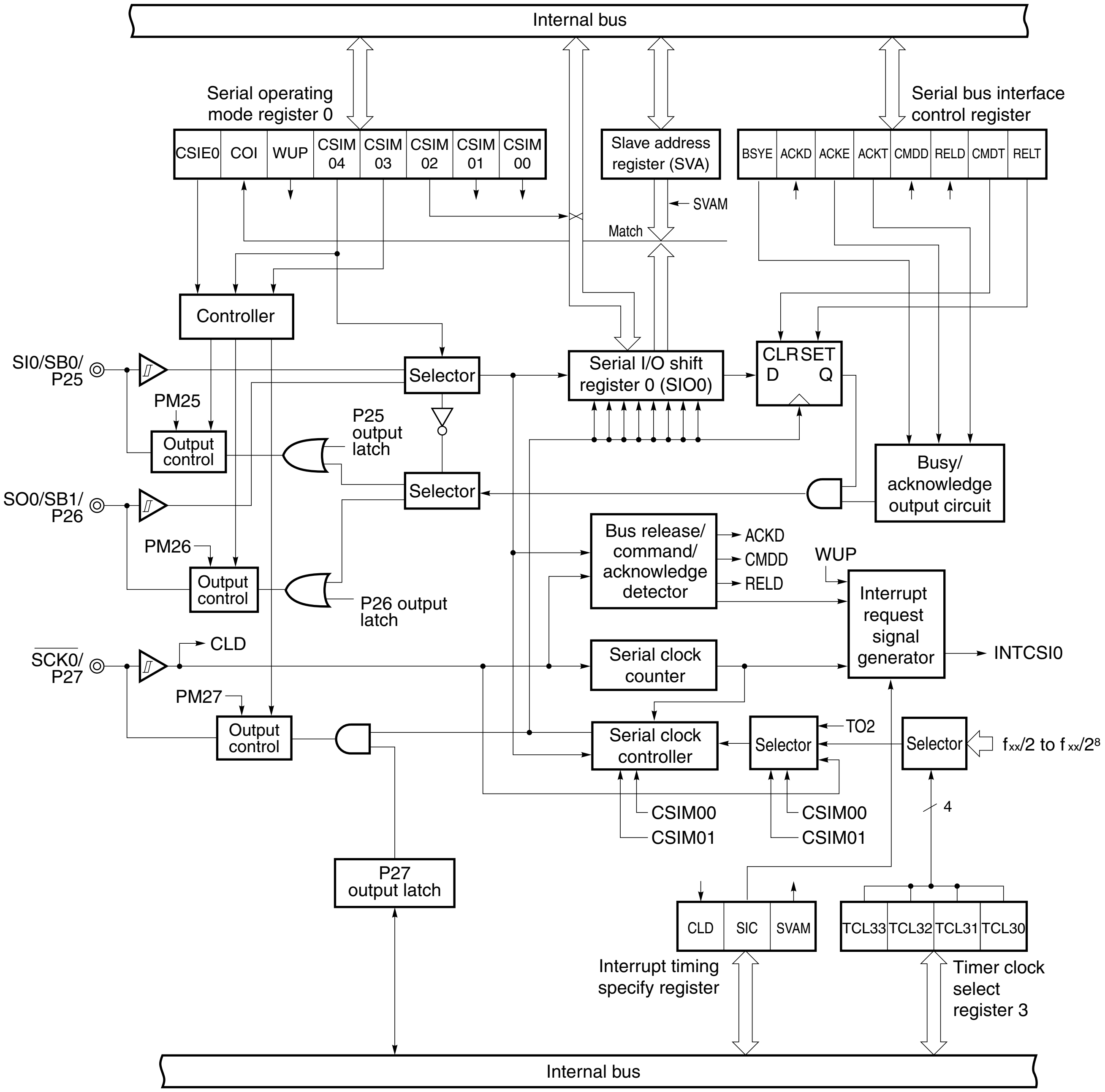
Serial interface channel 0 consists of the following hardware.

Table 15-2. Serial Interface Channel 0 Configuration

Item	Configuration
Register	Serial I/O shift register 0 (SIO0) Slave address register (SVA)
Control register	Timer clock select register 3 (TCL3) Serial operating mode register 0 (CSIM0) Serial bus interface control register (SBIC) Interrupt timing specify register (SINT) Port mode register 2 (PM2)**Note**

Note Refer to **Figure 6-5 P25, P26 Block Diagram (μPD780308 Subseries)** and **Figure 6-6 P27 Block Diagram (μPD780308 Subseries)**.

Figure 15-2. Serial Interface Channel 0 Block Diagram



Remarks 1. Output control performs selection between CMOS output and N-ch open-drain output.

2. fxx = fx/2 (MCS = 0), fxx = fx (MCS = 1)

(1) Serial I/O shift register 0 (SIO0)

This is an 8-bit register to carry out parallel/serial conversion and to carry out serial transmission/reception (shift operation) in synchronization with the serial clock.

SIO0 is set with an 8-bit memory manipulation instruction.

When bit 7 (CSIE0) of serial operating mode register 0 (CSIM0) is 1, writing data to SIO0 starts serial operation.

In transmission, data written to SIO0 is output to the serial output (SO0) or serial data bus (SB0/SB1). In reception, data is read from the serial input (SI0) or SB0/SB1 to SIO0.

Note that, if a bus is driven in the SBI mode or 2-wire serial I/O mode, the bus pin must serve for both input and output. Thus, in the case of a device for reception, write FFH to SIO0 in advance (except when address reception is carried out by setting bit 5 (WUP) of CSIM0 to 1).

In the SBI mode, the busy state can be cleared by writing data to SIO0. In this case, bit 7 (BSYE) of the serial bus interface control register (SBIC) is not cleared to 0.

$\overline{\text{RESET}}$ input makes SIO0 undefined.

(2) Slave address register (SVA)

This is an 8-bit register to set the slave address value for connection of a slave device to the serial bus.

SVA is set with an 8-bit memory manipulation instruction. This register is not used in the 3-wire serial I/O mode.

The master device outputs a slave address for selection of a particular slave device to the connected slave device. These two data (the slave address output from the master device and the SVA value) are compared with an address comparator. If they match, the slave device has been selected. In that case, bit 6 (COI) of serial operating mode register 0 (CSIM0) becomes 1.

By setting bit 4 (SVAM) of the interrupt timing specify register (SINT) to 1, the address can be compared using the data of the LSB-masked higher 7 bits.

If no matching is detected in address reception, bit 2 (RELD) of the serial bus interface control register (SBIC) is cleared to 0.

In the SBI mode, the wake-up function can be used by setting bit 5 (WUP) of CSIM0 to 1.

In this case, an interrupt request signal (INTCSI0) is generated only when the slave address output by the master matches the value of SVA. This interrupt request indicates that the master has requested for communication. If bit 5 (SIC) of the interrupt timing specify register (SINT) is set to 1, the wake-up function cannot be used even if WUP is set to 1 (the interrupt request signal is generated when bus release is detected). When using the wake-up function, clear SIC to 0.

When the device is used as the master or slave in the SBI or 2-wire serial I/O mode, detect an error by using SVA.

$\overline{\text{RESET}}$ input makes SVA undefined.

(3) SO0 latch

This latch holds SI0/SB0/P25 and SO0/SB1/P26 pin levels. It can be directly controlled by software. In the SBI mode, this latch is set upon termination of the 8th serial clock.

(4) Serial clock counter

This counter counts the serial clocks to be output and input during transmission/reception and to check whether 8-bit data has been transmitted/received.

(5) Serial clock controller

This circuit controls serial clock supply to serial I/O shift register 0 (SIO0). When the internal system clock is used, the circuit also controls clock output to the $\overline{\text{SCK0}}$/P27 pin.

(6) Interrupt request signal generator

This circuit controls interrupt request signal generation. It generates the interrupt request signal in the following cases.

- In the 3-wire serial I/O mode and 2-wire serial I/O mode
  This circuit generates an interrupt request signal every eight serial clocks.

- In the SBI mode
  When WUP[Note] is 0..... Generates an interrupt request signal every eight serial clocks.
  When WUP[Note] is 1..... Generates an interrupt request signal when the serial I/O shift register 0 (SIO0) value matches the slave address register (SVA) value after address reception.

Note WUP is wake-up function specify bit. It is bit 5 of serial operating mode register 0 (CSIM0). Clear bit 5 (SIC) of the interrupt timing specify register to 0 when using the wake-up function (WUP = 1).

(7) Busy/acknowledge output circuit and bus release/command/acknowledge detector

These two circuits output and detect various control signals in the SBI mode.
These do not operate in the 3-wire serial I/O mode and 2-wire serial I/O mode.

15.3 Serial Interface Channel 0 Control Registers

The following four types of registers are used to control serial interface channel 0.

- Timer clock select register 3 (TCL3)
- Serial operating mode register 0 (CSIM0)
- Serial bus interface control register (SBIC)
- Interrupt timing specify register (SINT)

(1) Timer clock select register 3 (TCL3)

This register sets the serial clock of serial interface channel 0.
TCL3 is set with an 8-bit memory manipulation instruction.
$\overline{\text{RESET}}$ input sets TCL3 to 88H.

Figure 15-3. Timer Clock Select Register 3 Format

Symbol	7	6	5	4	3	2	1	0	Address	After Reset	R/W
TCL3	1	0	0	0	TCL33	TCL32	TCL31	TCL30	FF43H	88H	R/W

TCL33	TCL32	TCL31	TCL30	Serial Interface Channel 0 Serial Clock Selection	
				MCS = 1	MCS = 0
0	1	1	0	Setting prohibited	$f_X/2^2$ (1.25 MHz)
0	1	1	1	$f_X/2^2$ (1.25 MHz)	$f_X/2^3$ (625 kHz)
1	0	0	0	$f_X/2^3$ (625 kHz)	$f_X/2^4$ (313 kHz)
1	0	0	1	$f_X/2^4$ (313 kHz)	$f_X/2^5$ (156 kHz)
1	0	1	0	$f_X/2^5$ (156 kHz)	$f_X/2^6$ (78.1 kHz)
1	0	1	1	$f_X/2^6$ (78.1 kHz)	$f_X/2^7$ (39.1 kHz)
1	1	0	0	$f_X/2^7$ (39.1 kHz)	$f_X/2^8$ (19.5 kHz)
1	1	0	1	$f_X/2^8$ (19.5 kHz)	$f_X/2^9$ (9.8 kHz)
Other than above				Setting prohibited	

Cautions 1. Set bit 4 to bit 6 to 0, and bit 7 to 1.

2. When rewriting TCL3 to other data, stop the serial transfer operation beforehand.

Remarks 1. f_X: Main system clock oscillation frequency

2. MCS: Oscillation mode select register bit 0

3. Figures in parentheses apply to operation with f_X = 5.0 MHz.

(2) Serial operating mode register 0 (CSIM0)

This register sets serial interface channel 0 serial clock, operating mode, operation enable/stop, wake-up function and displays the address comparator match signal.

CSIM0 is set with a 1-bit or 8-bit memory manipulation instruction.

$\overline{\text{RESET}}$ input clears CSIM0 to 00H.

Caution Do not change the operating mode (3-wire serial I/O, 2-wire serial I/O, or SBI) while serial interface channel 0 is enabled. To change the operating mode, stop the serial operation once.

Figure 15-4. Serial Operating Mode Register 0 Format (1/2)

Symbol	⑦	⑥	⑤	4	3	2	1	0	Address	After Reset	R/W
CSIM0	CSIE0	COI	WUP	CSIM04	CSIM03	CSIM02	CSIM01	CSIM00	FF60H	00H	R/W[Note 1]

R/W

CSIE0	Serial Interface Channel 0 Operation Control
0	Operation stopped
1	Operation enabled

R

COI	Slave Address Comparison Result Flag[Note 2]
0	Slave address register not equal to serial I/O shift register 0 data
1	Slave address register equal to serial I/O shift register 0 data

R/W

WUP	Wake-up Function Control[Note 3]
0	Interrupt request signal generation with each serial transfer in any mode
1	Interrupt request signal generation when the address received after bus release (when CMDD = RELD = 1) matches the slave address register data in SBI mode

Notes

1. Bit 6 (COI) is a read-only bit.
2. When CSIE0 = 0, COI becomes 0.
3. Clear bit 5 (SIC) of the interrupt timing specify register (SINT) to 0 when using the wake-up function (WUP = 1).

Figure 15-4. Serial Operating Mode Register 0 Format (2/2)

R/W

CSIM 04	CSIM 03	CSIM 02	PM25	P25	PM26	P26	PM27	P27	Operation Mode	Start Bit	SIO/SB0/P25 Pin Function	SO0/SB1/P26 Pin Function	$\overline{\text{SCK0}}$/P27 Pin Function
0	×	0	Note 1 1	Note 1 ×	0	0	0	1	3-wire serial I/O mode	MSB	SI0[Note 1] (Input)	SO0 (CMOS output)	$\overline{\text{SCK0}}$ (CMOS I/O)
		1								LSB			
1	0	0	Note 2 ×	Note 2 ×	0	0	0	1	SBI mode	MSB	P25 (CMOS I/O)	SB1 (N-ch open-drain I/O)	$\overline{\text{SCK0}}$ (CMOS I/O)
		1	0	0	Note 2 ×	Note 2 ×	0	1			SB0 (N-ch open-drain I/O)	P26 (CMOS I/O)	
1	1	0	Note 2 ×	Note 2 ×	0	0	0	1	2-wire serial I/O mode	MSB	P25 (CMOS I/O)	SB1 (N-ch open-drain I/O)	$\overline{\text{SCK0}}$ (N-ch open-drain I/O)
		1	0	0	Note 2 ×	Note 2 ×	0	1			SB0 (N-ch open-drain I/O)	P26 (CMOS I/O)	

R/W

CSIM01	CSIM00	Serial Interface Channel 0 Clock Selection
0	×	Input clock to $\overline{\text{SCK0}}$ pin from off-chip
1	0	8-bit timer register 2 (TM2) output
1	1	Clock specified with bits 0 to 3 of timer clock select register 3 (TCL3)

Notes 1. Can be used as P25 (CMOS I/O) when used only for transmission.

2. Can be used freely as port function.

Remark ×: don’t care

PM××: Port mode register

P××: Port output latch

(3) Serial bus interface control register (SBIC)

This register sets serial interface operation and displays statuses.

SBIC is set with a 1-bit or 8-bit memory manipulation instruction.

$\overline{\text{RESET}}$ input clears SBIC to 00H.

Figure 15-5. Serial Bus Interface Control Register Format (1/2)

Symbol	⑦	⑥	⑤	④	③	②	①	⓪	Address	After Reset	R/W
SBIC	BSYE	ACKD	ACKE	ACKT	CMDD	RELD	CMDT	RELT	FF61H	00H	R/W[Note 1]

R/W

BSYE[Note 2]	Synchronizing Busy Signal Output Control
0	Disables busy signal which is output in synchronization with the falling edge of $\overline{\text{SCK0}}$ clock just after execution of the instruction to be cleared to 0.
1	Outputs busy signal at the falling edge of $\overline{\text{SCK0}}$ clock following the acknowledge signal.

R

ACKD	Acknowledge Detection	
	Clear Conditions (ACKD = 0)	Set Conditions (ACKD = 1)
	• Falling edge of the $\overline{\text{SCK0}}$ immediately after the busy mode is released while executing the transfer start instruction • When CSIE0 = 0 • When $\overline{\text{RESET}}$ input is applied	• When acknowledge signal ($\overline{\text{ACK}}$) is detected at the rising edge of $\overline{\text{SCK0}}$ clock after completion of transfer

R/W

ACKE	Acknowledge Signal Output Control	
0	Acknowledge signal automatic output disable (output with ACKT enable)	
1	Before completion of transfer	Acknowledge signal is output in synchronization with the 9th clock falling edge of $\overline{\text{SCK0}}$ (automatically output when ACKE = 1).
	After completion of transfer	Acknowledge signal is output in synchronization with the falling edge of $\overline{\text{SCK0}}$ just after execution of the instruction to be set to 1 (automatically output when ACKE = 1). However, not automatically cleared to 0 after acknowledge signal output.

Notes 1. Bits 2, 3, and 6 (RELD, CMDD, and ACKD) are read-only bits.

2. The busy mode can be cancelled by start of serial interface transfer. However, the BSYE flag is not cleared to 0.

Remark CSIE0: Bit 7 of serial operating mode register 0 (CSIM0)

Figure 15-5. Serial Bus Interface Control Register Format (2/2)

R/W	ACKT	Acknowledge signal is output in synchronization with the falling edge clock of $\overline{\text{SCK0}}$ just after execution of the instruction to be set to 1, and after acknowledge signal output, is automatically cleared to 0. Used as ACKE = 0. Also cleared to 0 upon start of serial interface transfer or when CSIE0 = 0.

R	CMDD	Command Detection
	Clear Conditions (CMDD = 0)	Set Conditions (CMDD = 1)
	• When transfer start instruction is executed • When bus release signal (REL) is detected • When CSIE0 = 0 • When $\overline{\text{RESET}}$ input is applied	• When command signal (CMD) is detected

R	RELD	Bus Release Detection
	Clear Conditions (RELD = 0)	Set Conditions (RELD =1)
	• When transfer start instruction is executed • If SIO0 and SVA values do not match in address reception • When CSIE0 = 0 • When $\overline{\text{RESET}}$ input is applied	• When bus release signal (REL) is detected

R/W	CMDT	Used for command signal output. When CMDT = 1, SO latch is cleared to (0). After SO latch clearance, automatically cleared to 0. Also cleared to 0 when CSIE0 = 0.

R/W	RELT	Used for bus release signal output. When RELT = 1, SO latch is set to 1. After SO latch setting, automatically cleared to 0. Also cleared to 0 when CSIE0 = 0.

Remarks 1. Bits 0, 1, and 4 (RELD, CMDT, and ACKT) are 0 when they are read after data has been set.

2. CSIE0: Bit 7 of serial operating mode register 0 (CSIM0)

(4) Interrupt timing specify register (SINT)

This register sets the bus release interrupt and address mask functions and displays the $\overline{\text{SCK0}}$ pin level status.

SINT is set with a 1-bit or 8-bit memory manipulation instruction.

$\overline{\text{RESET}}$ input clears SINT to 00H.

Figure 15-6. Interrupt Timing Specify Register Format

Symbol	7	⑥	⑤	④	3	2	1	0	Address	After Reset	R/W
SINT	0	CLD	SIC	SVAM	0	0	0	0	FF63H	00H	R/W[Note 1]

R	CLD	$\overline{\text{SCK0}}$ Pin Level[Note 2]
	0	Low level
	1	High level

R/W	SIC	INTCSI0 Interrupt Cause Selection[Note 3]
	0	CSIIF0 is set upon termination of serial interface channel 0 transfer
	1	CSIIF0 is set upon bus release detection or termination of serial interface channel 0 transfer

R/W	SVAM	SVA Bit to Be Used as Slave Address
	0	Bits 0 to 7
	1	Bits 1 to 7

Notes **1.** Bit 6 (CLD) is a read-only bit.

2. When CSIE0 = 0, CLD becomes 0.

3. When using the wake-up function in the SBI mode, set SIC to 0.

Caution **Be sure to set bit 0 to bit 3 to 0.**

Remark SVA: Slave address register

CSIIF0: Interrupt request flag corresponding to INTCSI0

CSIE0: Bit 7 of serial operating mode register 0 (CSIM0)

15.4 Serial Interface Channel 0 Operations

The following four operating modes are available for serial interface channel 0.

- Operation stop mode
- 3-wire serial I/O mode
- SBI mode
- 2-wire serial I/O mode

15.4.1 Operation stop mode

Serial transfer is not carried out in the operation stop mode. Thus, power consumption can be reduced. Serial I/O shift register 0 (SIO0) does not carry out shift operation either and thus it can be used as an ordinary 8-bit register.

In the operation stop mode, the P25/SI0/SB0, P26/SO0/SB1 and P27/$\overline{\text{SCK0}}$ pins can be used as ordinary I/O ports.

(1) Register setting

The operation stop mode is set with serial operating mode register 0 (CSIM0).

CSIM0 is set with a 1-bit or 8-bit memory manipulation instruction.

$\overline{\text{RESET}}$ input clears CSIM0 to 00H.

Symbol	⑦	⑥	⑤	4	3	2	1	0	Address	After Reset	R/W
CSIM0	CSIE0	COI	WUP	CSIM04	CSIM03	CSIM02	CSIM01	CSIM00	FF60H	00H	R/W

R/W	CSIE0	Serial Interface Channel 0 Operation Control
	0	Operation stopped
	1	Operation enabled

15.4.2 3-wire serial I/O mode operation

The 3-wire serial I/O mode is valid for connection of peripheral I/O units and display controllers which incorporate a conventional synchronous clocked serial interface as is the case with the 75X/XL, 78K, and 17K Series.

Communication is carried out with three lines of serial clock ($\overline{\text{SCK0}}$), serial output (SO0), and serial input (SI0).

(1) Register setting

The 3-wire serial I/O mode is set with serial operating mode register 0 (CSIM0) and the serial bus interface control register (SBIC).

(a) Serial operating mode register 0 (CSIM0)

CSIM0 is set with a 1-bit or 8-bit memory manipulation instruction.

$\overline{\text{RESET}}$ input clears CSIM0 to 00H.

Symbol	⑦	⑥	⑤	4	3	2	1	0	Address	After Reset	R/W
CSIM0	CSIE0	COI	WUP	CSIM04	CSIM03	CSIM02	CSIM01	CSIM00	FF60H	00H	R/W$^{\text{Note 1}}$

R/W

CSIE0	Serial Interface Channel 0 Operation Control
0	Operation stopped
1	Operation enabled

R/W

WUP	Wake-up Function Control$^{\text{Note 2}}$
0	Interrupt request signal generation with each serial transfer in any mode
1	Interrupt request signal generation when the address received after bus release (when CMDD = RELD = 1) matches the slave address register data in SBI mode

R/W

CSIM 04	CSIM 03	CSIM 02	PM25	P25	PM26	P26	PM27	P27	Operation Mode	Start Bit	SIO/SB0/P25 Pin Function	SO0/SB1/P26 Pin Function	$\overline{\text{SCK0}}$/P27 Pin Function
0	×	0	Note 3 1	Note 3 ×	0	0	0	1	3-wire serial I/O mode	MSB	SI0$^{\text{Note 3}}$ (Input)	SO0 (CMOS output)	$\overline{\text{SCK0}}$ (CMOS I/O)
		1								LSB			
1	0	SBI mode (See **15.4.3 SBI mode operation**)											
1	1	2-wire serial I/O mode (See **15.4.4 2-wire serial I/O mode operation**)											

R/W

CSIM01	CSIM00	Serial Interface Channel 0 Clock Selection
0	×	Input clock to $\overline{\text{SCK0}}$ pin from off-chip
1	0	8-bit timer register 2 (TM2) output
1	1	Clock specified with bits 0 to 3 of timer clock select register 3 (TCL3)

Notes 1. Bit 6 (COI) is a read-only bit.

2. Be sure to set WUP to 0 when the 3-wire serial I/O mode is selected.

3. Can be used as P25 (CMOS I/O) when used only for transmission.

Remark	
×:	don’t care
PM××:	Port mode register
P××:	Port output latch

(b) Serial bus interface control register (SBIC)

SBIC is set with a 1-bit or 8-bit memory manipulation instruction.

$\overline{\text{RESET}}$ input clears SBIC to 00H.

Symbol	⑦	⑥	⑤	④	③	②	①	⓪	Address	After Reset	R/W
SBIC	BSYE	ACKD	ACKE	ACKT	CMDD	RELD	CMDT	RELT	FF61H	00H	R/W

R/W	CMDT	When CMDT = 1, SO latch is cleared to 0. After SO latch clearance, automatically cleared to 0. Also cleared to 0 when CSIE0 = 0.

R/W	RELT	When RELT = 1, SO latch is set to 1. After SO latch setting, automatically cleared to 0. Also cleared to 0 when CSIE0 = 0.

CSIE0: Bit 7 of serial operating mode register 0 (CSIM0)

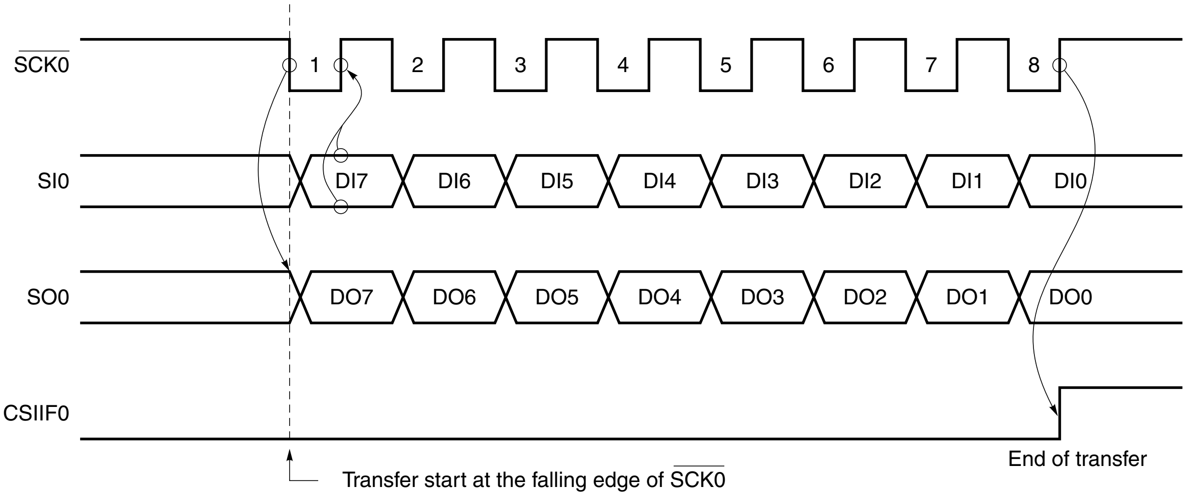
(2) Communication operation

The 3-wire serial I/O mode is used for data transmission/reception in 8-bit units. Each bit of data is transmitted or received in synchronization with the serial clock.

Shift operation of serial I/O shift register 0 (SIO0) is carried out at the falling edge of the serial clock ($\overline{\text{SCK0}}$). The transmitted data is held in the SO0 latch and is output from the SO0 pin. The received data input to the SI0 pin is latched in SIO0 at the rising edge of $\overline{\text{SCK0}}$.

Upon termination of 8-bit transfer, SIO0 operation stops automatically and the interrupt request flag (CSIIF0) is set.

Figure 15-7. 3-Wire Serial I/O Mode Timings



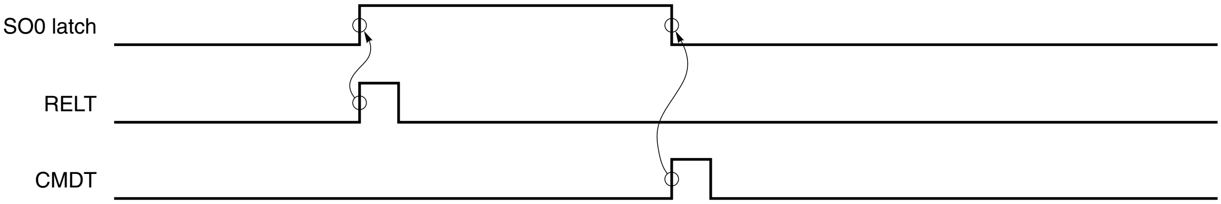
The SO0 pin is a CMOS output pin and outputs current SO0 latch statuses. Thus, the SO0 pin output status can be manipulated by setting bit 0 (RELT) and bit 1 (CMDT) of the serial bus interface control register (SBIC). However, do not carry out this manipulation during serial transfer.

Control the $\overline{\text{SCK0}}$ pin output level in the output mode (internal system clock mode) by manipulating the P27 output latch (refer to **15.4.5 $\overline{\text{SCK0}}$/P27 pin output manipulation**).

(3) Other signals

Figure 15-8 shows RELT and CMDT operations.

Figure 15-8. RELT and CMDT Operations



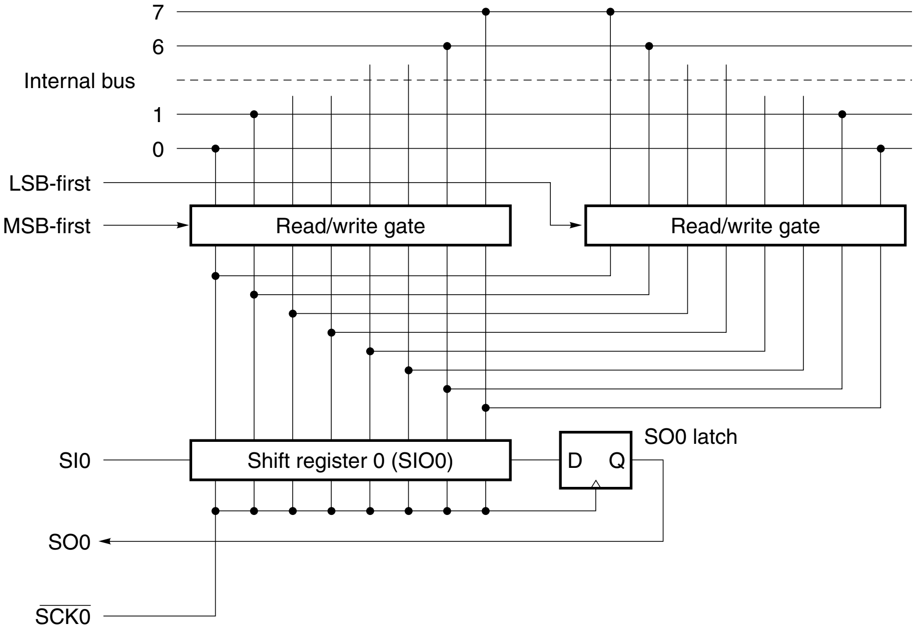
(4) MSB/LSB switching as the start bit

The 3-wire serial I/O mode enables to select transfer to start from MSB or LSB.

Figure 15-9 shows the configuration of serial I/O shift register 0 (SIO0) and internal bus. As shown in the figure, MSB/LSB can be read/written in reverse form.

MSB/LSB switching as the start bit can be specified with bit 2 (CSIM02) of serial operating mode register 0 (CSIM0).

Figure 15-9. Circuit of Switching in Transfer Bit Order



Start bit switching is realized by switching the bit order for data write to SIO0. The SIO0 shift order remains unchanged.

Thus, switching between MSB-first and LSB-first must be performed before writing data to the shift register.

(5) Transfer start

Serial transfer is started by setting transfer data to serial I/O shift register 0 (SIO0) when the following two conditions are satisfied.

- Serial interface channel 0 operation control bit (CSIE0) = 1
- Internal serial clock is stopped or $\overline{\text{SCK0}}$ is at high level after 8-bit serial transfer.

Caution If CSIE0 is set to "1" after data write to SIO0, transfer does not start.

Upon termination of 8-bit transfer, serial transfer automatically stops and the interrupt request flag (CSIIF0) is set.

15.4.3 SBI mode operation

SBI (Serial Bus Interface) is a high-speed serial interface in compliance with the NEC serial bus format.

SBI uses a single master device and employs the clocked serial I/O format with the addition of a bus configuration function. This function enables devices to communicate using only two lines. Thus, when making up a serial bus with two or more microcontrollers and peripheral ICs, the number of ports to be used and the number of wires on the board can be decreased.

The master device outputs three kinds of data to slave devices on the serial data bus: “addresses” to select a device to be communicated with, “commands” to instruct the selected device, and “data” which is actually required.

The slave device can identify the received data into “address”, “command”, or “data”, by hardware. An application program that controls serial interface channel 0 can be simplified by using this function.

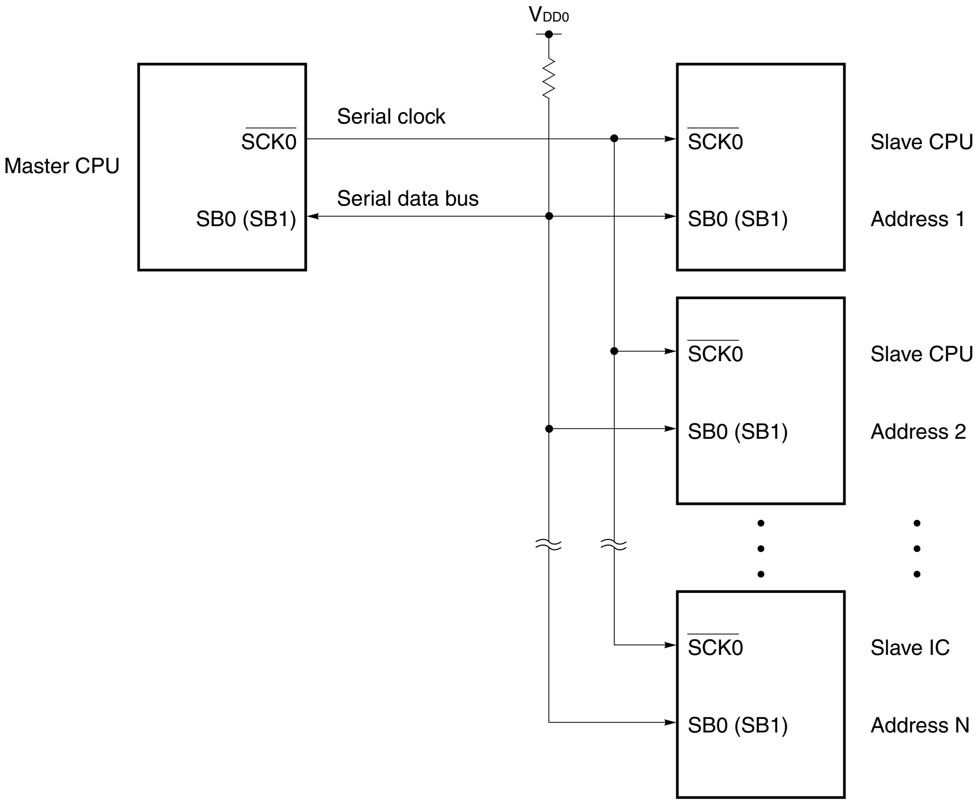
The SBI function is incorporated into various devices including 75X/XL Series and 78K Series.

Figure 15-10 shows a serial bus configuration example when a CPU having a serial interface compliant with SBI and peripheral ICs are used.

In SBI, the SB0 (SB1) serial data bus is an open-drain output pin and therefore the serial data bus line behaves in the same way as the wired-OR configuration. In addition, a pull-up resistor must be connected to the serial data bus line.

When the SBI mode is used, refer to **(11) SBI mode precautions (d)** described later.

Figure 15-10. Example of Serial Bus Configuration with SBI



Caution When exchanging the master CPU/slave CPU, a pull-up resistor is necessary for the serial clock line ($\overline{SCK0}$) as well because serial clock line ($\overline{SCK0}$) input/output switching is carried out asynchronously between the master and slave CPUs.

(1) SBI functions

In the conventional serial I/O format, when a serial bus is configured by connecting two or more devices, many ports and wiring are necessary, to provide chip select signal to identify command and data, and to judge the busy state, because only the data transfer function is available. If these operations are to be controlled by software, the software must be heavily loaded.

In SBI, a serial bus can be configured with two signal lines of serial clock $\overline{\text{SCK0}}$ and serial data bus SB0 (SB1). Thus, use of SBI leads to reduction in the number of microcontroller ports and that of wirings and routings on the board.

The SBI functions are described below.

(a) Address/command/data identify function

Serial data is distinguished into addresses, commands, and data.

(b) Chip select function by address transmission

The master executes slave chip selection by address transmission.

(c) Wake-up function

The slave can easily judge address reception (chip select judgment) with the wake-up function (which can be set/reset by software).

When the wake-up function is set, the interrupt request signal (INTCSI0) is generated upon reception of a match address.

Thus, when communication is executed with two or more devices, the CPU except the selected slave devices can operate regardless of under way serial communications.

(d) Acknowledge signal ($\overline{\text{ACK}}$) control function

The acknowledge signal to check serial data reception is controlled.

(e) Busy signal ($\overline{\text{BUSY}}$) control function

The busy signal to report the slave busy state is controlled.

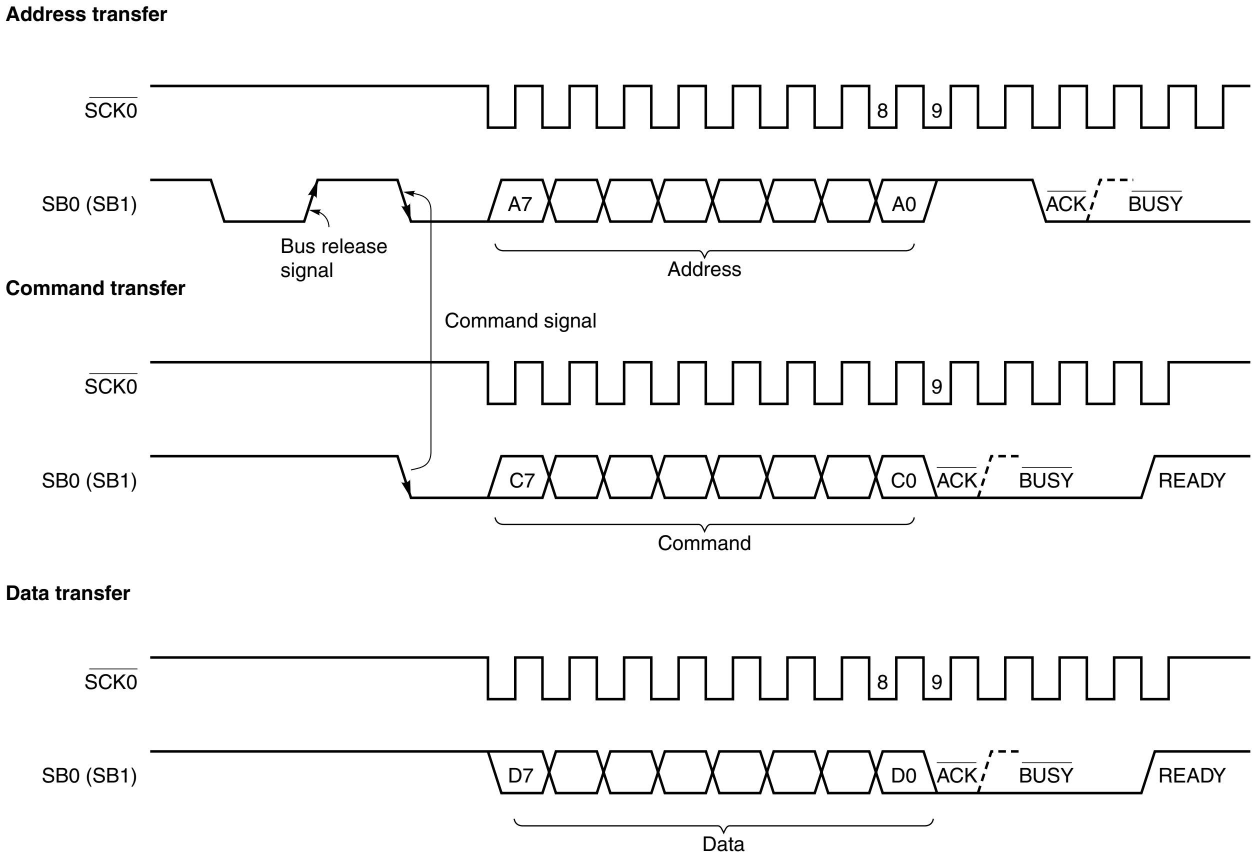
(2) SBI definition

The SBI serial data format and the signals to be used are defined as follows.

Serial data to be transferred with SBI consists of three kinds of data: “address”, “command”, and “data”.

Figure 15-11 shows the address, command, and data transfer timings.

Figure 15-11. SBI Transfer Timings



Remark The dotted line indicates the READY status.

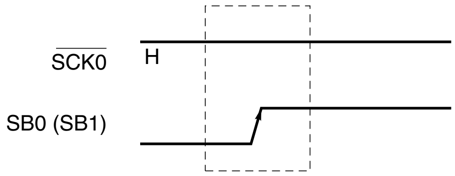
The bus release signal and the command signal are output by the master device. $\overline{\text{BUSY}}$ is output by the slave signal. $\overline{\text{ACK}}$ can be output by either the master or slave device (normally, the 8-bit data receiver outputs). Serial clocks continue to be output by the master device from 8-bit data transfer start to $\overline{\text{BUSY}}$ reset.

(a) Bus release signal (REL)

The bus release signal is a signal with the SB0 (SB1) line which has changed from the low level to the high level when the $\overline{\text{SCK0}}$ line is at the high level (without serial clock output).

This signal is output by the master device.

Figure 15-12. Bus Release Signal

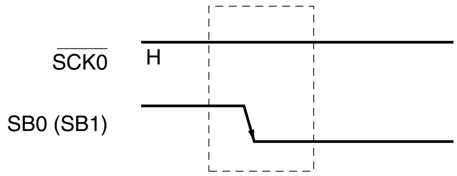


The bus release signal indicates that the master device is going to transmit an address to the slave device. The slave device incorporates hardware to detect the bus release signal.

(b) Command signal (CMD)

The command signal is a signal with the SB0 (SB1) line which has changed from the high level to the low level when the $\overline{\text{SCK0}}$ line is at the high level (without serial clock output). This signal is output by the master device.

Figure 15-13. Command Signal



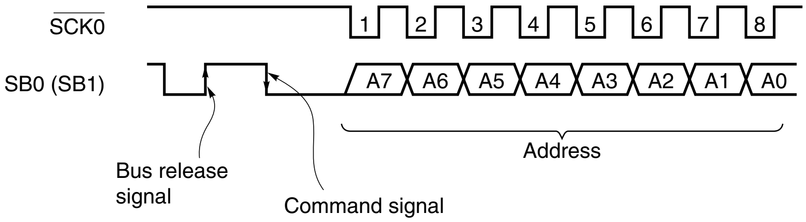
The command signal indicates that the master is to transmit a command to the slave (however, the command signal following the bus release signal indicates that an address is transmitted).

The slave device incorporates hardware to detect the command signal.

(c) Address

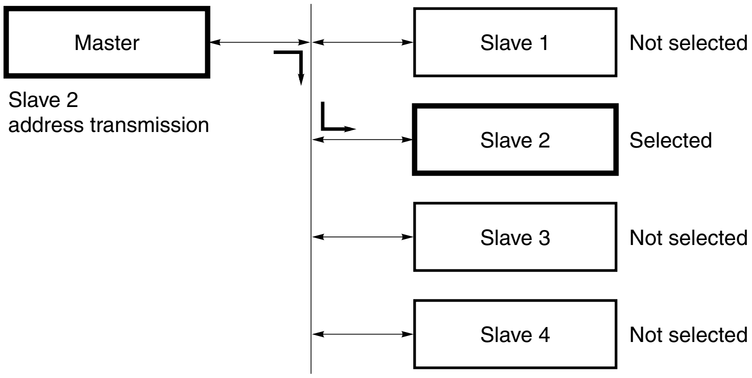
An address is 8-bit data which the master device outputs to the slave device connected to the bus line in order to select a particular slave device.

Figure 15-14. Addresses



8-bit data following bus release and command signals is defined as an "address". In the slave device, this condition is detected by hardware and whether or not 8-bit data matches the own specification number (slave address) is checked by hardware. If the 8-bit data matches the slave address, the slave device has been selected. After that, communication with the master device continues until a release instruction is received from the master device.

Figure 15-15. Slave Selection with Address



(d) Command and data

The master device transmits commands to, and transmits/receives data to/from the slave device selected by address transmission.

Figure 15-16. Commands

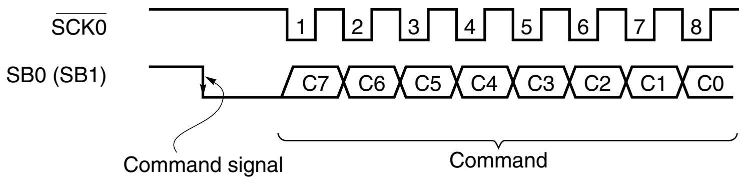
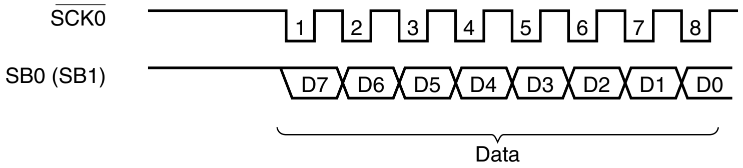


Figure 15-17. Data

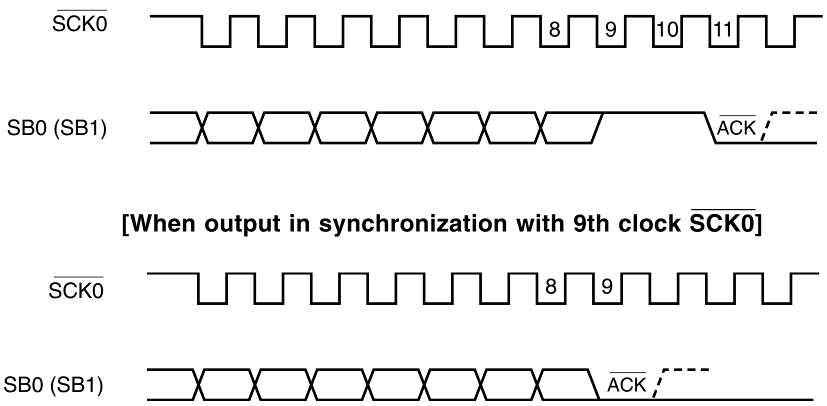


8-bit data following a command signal is defined as "command" data. 8-bit data without command signal is defined as "data". Command and data operation procedures are allowed to determine by user according to communications specifications.

(e) Acknowledge signal ($\overline{\mathrm{ACK}}$)

The acknowledge signal is used to check serial data reception between transmitter and receiver.

Figure 15-18. Acknowledge Signal



[When output in synchronization with 11th clock $\overline{\mathrm{SCK0}}$]

[When output in synchronization with 9th clock $\overline{\mathrm{SCK0}}$]

$\overline{\mathrm{SCK0}}$
8
9
10
11
SB0 (SB1)
$\overline{\mathrm{ACK}}$
$\overline{\mathrm{SCK0}}$
8
9
SB0 (SB1)
$\overline{\mathrm{ACK}}$

Remark The dotted line indicates the READY status.

The acknowledge signal is one-shot pulse to be generated at the falling edge of $\overline{\mathrm{SCK0}}$ after 8-bit data transfer. It can be positioned anywhere and can be synchronized with any clock $\overline{\mathrm{SCK0}}$.

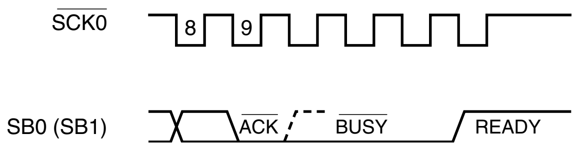
After 8-bit data transmission, the transmitter checks whether the receiver has returned the acknowledge signal. If the acknowledge signal is not returned for the preset period of time after data transmission, it can be judged that data reception has not been carried out correctly.

(f) Busy signal ($\overline{\text{BUSY}}$) and ready signal (READY)

The $\overline{\text{BUSY}}$ signal is intended to report to the master device that the slave device is preparing for data transmission/reception.

The READY signal is intended to report to the master device that the slave device is ready for data transmission/reception.

Figure 15-19. $\overline{\text{BUSY}}$ and READY Signals



Remark The dotted line indicates the READY status.

In SBI, the slave device notifies the master device of the busy state by setting SB0 (SB1) line to the low level.

The $\overline{\text{BUSY}}$ signal output follows the acknowledge signal output from the master or slave device. It is set/ reset at the falling edge of $\overline{\text{SCK0}}$. When the $\overline{\text{BUSY}}$ signal is reset, the master device automatically terminates the output of $\overline{\text{SCK0}}$ serial clock.

When the $\overline{\text{BUSY}}$ signal is reset and the READY signal is set, the master device can start the next transfer.

(3) Register setting

The SBI mode is set with serial operating mode register 0 (CSIM0), the serial bus interface control register (SBIC), and the interrupt timing specify register (SINT).

(a) Serial operating mode register 0 (CSIM0)

CSIM0 is set with a 1-bit or 8-bit memory manipulation instruction.

$\overline{\text{RESET}}$ input clears CSIM0 to 00H.

Symbol	⑦	⑥	⑤	4	3	2	1	0	Address	After Reset	R/W
CSIM0	CSIE0	COI	WUP	CSIM04	CSIM03	CSIM02	CSIM01	CSIM00	FF60H	00H	R/W[Note 1]

R/W	CSIE0	Serial Interface Channel 0 Operation Control
	0	Operation stopped
	1	Operation enabled

R	COI	Slave Address Comparison Result Flag[Note 2]
	0	Slave address register not equal to serial I/O shift register 0 data
	1	Slave address register equal to serial I/O shift register 0 data

R/W	WUP	Wake-up Function Control[Note 3]
	0	Interrupt request signal generation with each serial transfer in any mode
	1	Interrupt request signal generation when the address received after bus release (when CMDD = RELD = 1) matches the slave address register data in SBI mode

R/W	CSIM 04	CSIM 03	CSIM 02	PM25	P25	PM26	P26	PM27	P27	Operation Mode	Start Bit	SI0/SB0/P25 Pin Function	SO0/SB1/P26 Pin Function	$\overline{\text{SCK0}}$/P27 Pin Function
	0	×	3-wire serial I/O mode (See **15.4.2 3-wire serial I/O mode operation**)											
	1	0	0	Note 4 ×	Note 4 ×	0	0	0	1	SBI mode	MSB	P25 (CMOS I/O)	SB1 (N-ch open-drain I/O)	$\overline{\text{SCK0}}$ (CMOS I/O)
			1	0	0	Note 4 ×	Note 4 ×	0	1			SB0 (N-ch open-drain I/O)	P26 (CMOS I/O)	
	1	1	2-wire serial I/O mode (See **15.4.4 2-wire serial I/O mode operation**)											

R/W	CSIM01	CSIM00	Serial Interface Channel 0 Clock Selection
	0	×	Input clock to $\overline{\text{SCK0}}$ pin from off-chip
	1	0	8-bit timer register 2 (TM2) output
	1	1	Clock specified with bits 0 to 3 of timer clock select register 3 (TCL3)

Notes 1. Bit 6 (COI) is a read-only bit.

2. COI is 0 when CSIE0 = 0.

3. Set bit 5 (SIC) of the interrupt timing specify register (SINT) to 1 when using the wake-up function (WUP = 1).

4. These pins can be used freely as port pins.

Remark ×: don't care

PM××: Port mode register

P××: Port output latch

(b) Serial bus interface control register (SBIC)

SBIC is set with a 1-bit or 8-bit memory manipulation instruction.

$\overline{\text{RESET}}$ input clears SBIC to 00H.

Symbol	⑦	⑥	⑤	④	③	②	①	⓪	Address	After Reset	R/W
SBIC	BSYE	ACKD	ACKE	ACKT	CMDD	RELD	CMDT	RELT	FF61H	00H	R/W[Note 1]

R/W

BSYE[Note2]	Synchronizing Busy Signal Output Control
0	Disables busy signal which is output in synchronization with the falling edge of $\overline{\text{SCK0}}$ clock just after execution of the instruction to be cleared (to 0).
1	Outputs busy signal at the falling edge of $\overline{\text{SCK0}}$ clock following the acknowledge signal.

R

ACKD	Acknowledge Detection
Clear Conditions (ACKD = 0)	Set Conditions (ACKD = 1)
• $\overline{\text{SCK0}}$ fall immediately after the busy mode is released during the transfer start instruction execution. • When CSIE0 = 0 • When $\overline{\text{RESET}}$ input is applied	• When acknowledge signal ($\overline{\text{ACK}}$) is detected at the rising edge of $\overline{\text{SCK0}}$ clock after completion of transfer

R/W

ACKE	Acknowledge Signal Output Control	
0	Acknowledge signal automatic output disable (output with ACKT enable)	
1	Before completion of transfer	Acknowledge signal is output in synchronization with the 9th clock falling edge of $\overline{\text{SCK0}}$ (automatically output when ACKE = 1).
	After completion of transfer	Acknowledge signal is output in synchronization with falling edge clock of $\overline{\text{SCK0}}$ just after execution of the instruction to be set to 1 (automatically output when ACKE = 1). However, not automatically cleared to 0 after acknowledge signal output.

(continued)

Notes 1. Bits 2, 3, and 6 (RELD, CMDD, and ACKD) are read-only bits.

2. The busy mode can be cleared by start of serial interface transfer. However, the BSYE flag is not cleared to 0.

Remarks 1. Bits 0, 1, and 4 (RELT, CMDT, and ACKT) are 0 when read after data setting.

2. CSIE0: Bit 7 of serial operating mode register 0 (CSIM0)

R/W	ACKT	Acknowledge signal is output in synchronization with the falling edge clock of $\overline{\text{SCK0}}$ just after execution of the instruction to be set (to 1) and, after acknowledge signal output, is automatically cleared (to 0). Used as ACKE = 0. Also cleared (to 0) upon start of serial interface transfer or when CSIE0 = 0.

R

CMDD	Command Detection
Clear Conditions (CMDD = 0)	Set Conditions (CMDD = 1)
• When transfer start instruction is executed • When bus release signal (REL) is detected • When CSIE0 = 0 • When $\overline{\text{RESET}}$ input is applied	• When command signal (CMD) is detected

R

RELD	Bus Release Detection
Clear Conditions (RELD = 0)	Set Conditions (RELD = 1)
• When transfer start instruction is executed • If SIO0 and SVA values do not match in address reception • When CSIE0 = 0 • When $\overline{\text{RESET}}$ input is applied	• When bus release signal (REL) is detected

R/W	CMDT	Used for command signal output. When CMDT = 1, SO latch is cleared (to 0). After SO latch clearance, automatically cleared (to 0). Also cleared to 0 when CSIE0 = 0.

R/W	RELT	Used for bus release signal output. When RELT = 1, SO latch is set (to 1). After SO latch setting, automatically cleared (to 0). Also cleared to 0 when CSIE0 = 0.

Remarks 1. Bits 0, 1, and 4 (RELT, CMDT, and ACKT) are 0 when read after data setting.

2. CSIE0: Bit 7 of serial operating mode register 0 (CSIM0)

(c) Interrupt timing specify register (SINT)

SINT is set with a 1-bit or 8-bit memory manipulation instruction.
$\overline{\text{RESET}}$ input clears SINT to 00H.

Symbol	7	⑥	⑤	④	3	2	1	0	Address	After Reset	R/W
SINT	0	CLD	SIC	SVAM	0	0	0	0	FF63H	00H	R/W[Note 1]

R	CLD	$\overline{\text{SCK0}}$ Pin Level[Note 2]
	0	Low level
	1	High level

R/W	SIC	INTCSI0 Interrupt Cause Selection[Note 3]
	0	CSIIF0 is set upon termination of serial interface channel 0 transfer
	1	CSIIF0 is set upon bus release detection or termination of serial interface channel 0 transfer

R/W	SVAM	SVA Bit to Be Used as Slave Address
	0	Bits 0 to 7
	1	Bits 1 to 7

Notes 1. Bit 6 (CLD) is a read-only bit.

2. When CSIE0 = 0, CLD becomes 0.

3. When using the wake-up function in the SBI mode, set SIC to 0.

Caution Be sure to set bit 0 to bit 3 to 0.

Remark SVA: Slave address register
CSIIF0: Interrupt request flag corresponding to INTCSI0
CSIE0: Bit 7 of serial operating mode register 0 (CSIM0)

(4) Various signals

Figures 15-20 to 15-25 show various signals and serial bus interface control register (SBIC) flag operations in SBI. Table 15-3 lists various signals in SBI.

Figure 15-20. RELT, CMDT, RELD, and CMDD Operations (Master)

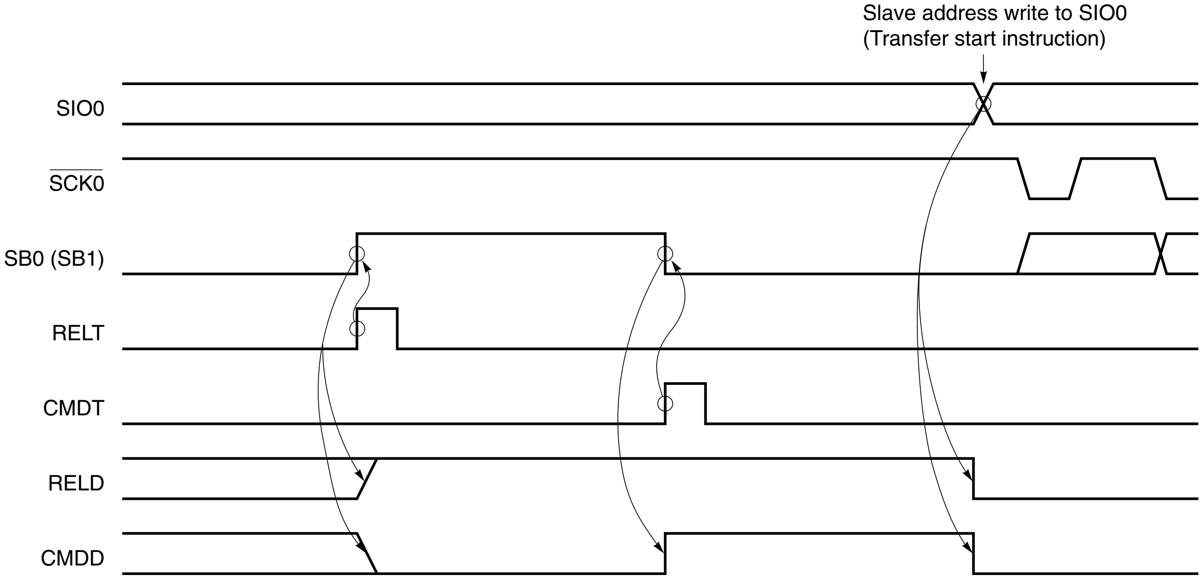


Figure 15-21. RELD and CMDD Operations (Slave)

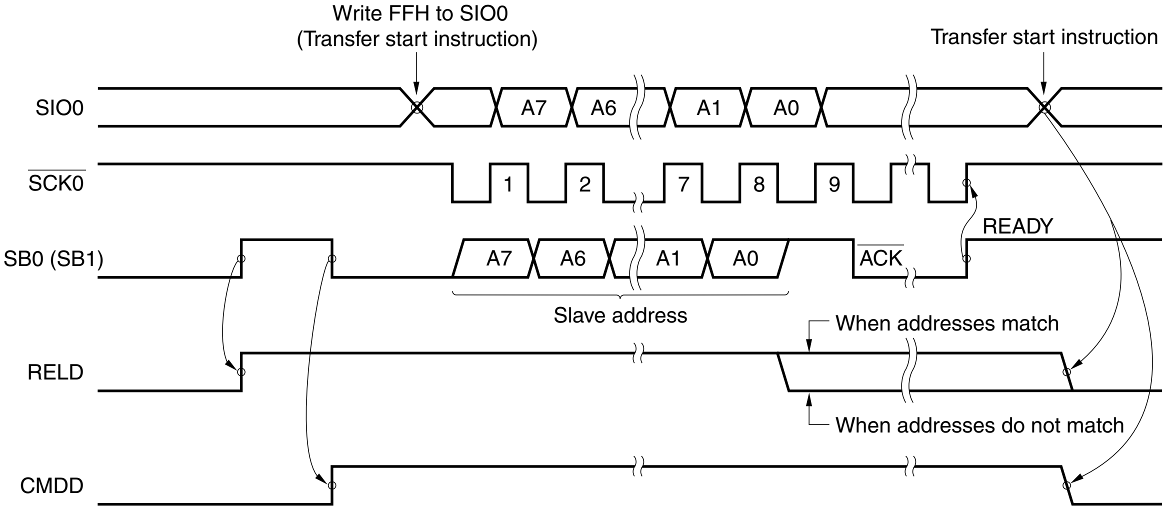
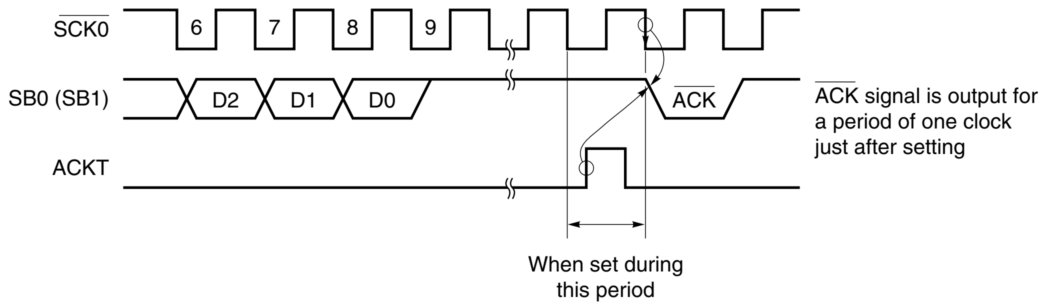
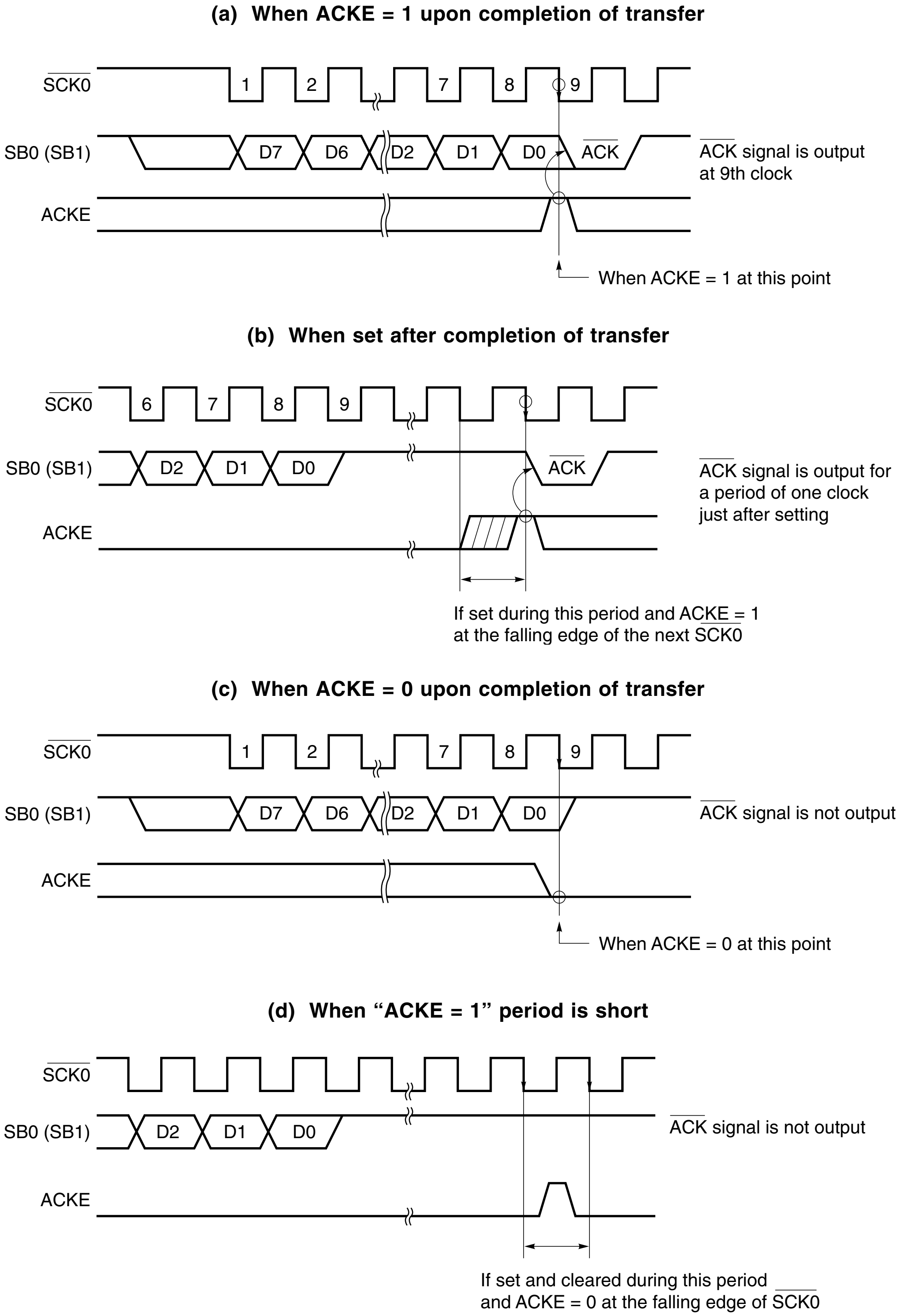


Figure 15-22. ACKT Operation


SCK0
6
7
8
9
SB0 (SB1)
D2
D1
D0
ACK
ACKT
ACK signal is output for a period of one clock just after setting
When set during this period

Caution   Do not set ACKT before termination of transfer.

Figure 15-23. ACKE Operations



(a) When ACKE = 1 upon completion of transfer

ACK signal is output at 9th clock

When ACKE = 1 at this point

(b) When set after completion of transfer

ACK signal is output for a period of one clock just after setting

If set during this period and ACKE = 1 at the falling edge of the next SCK0

(c) When ACKE = 0 upon completion of transfer

ACK signal is not output

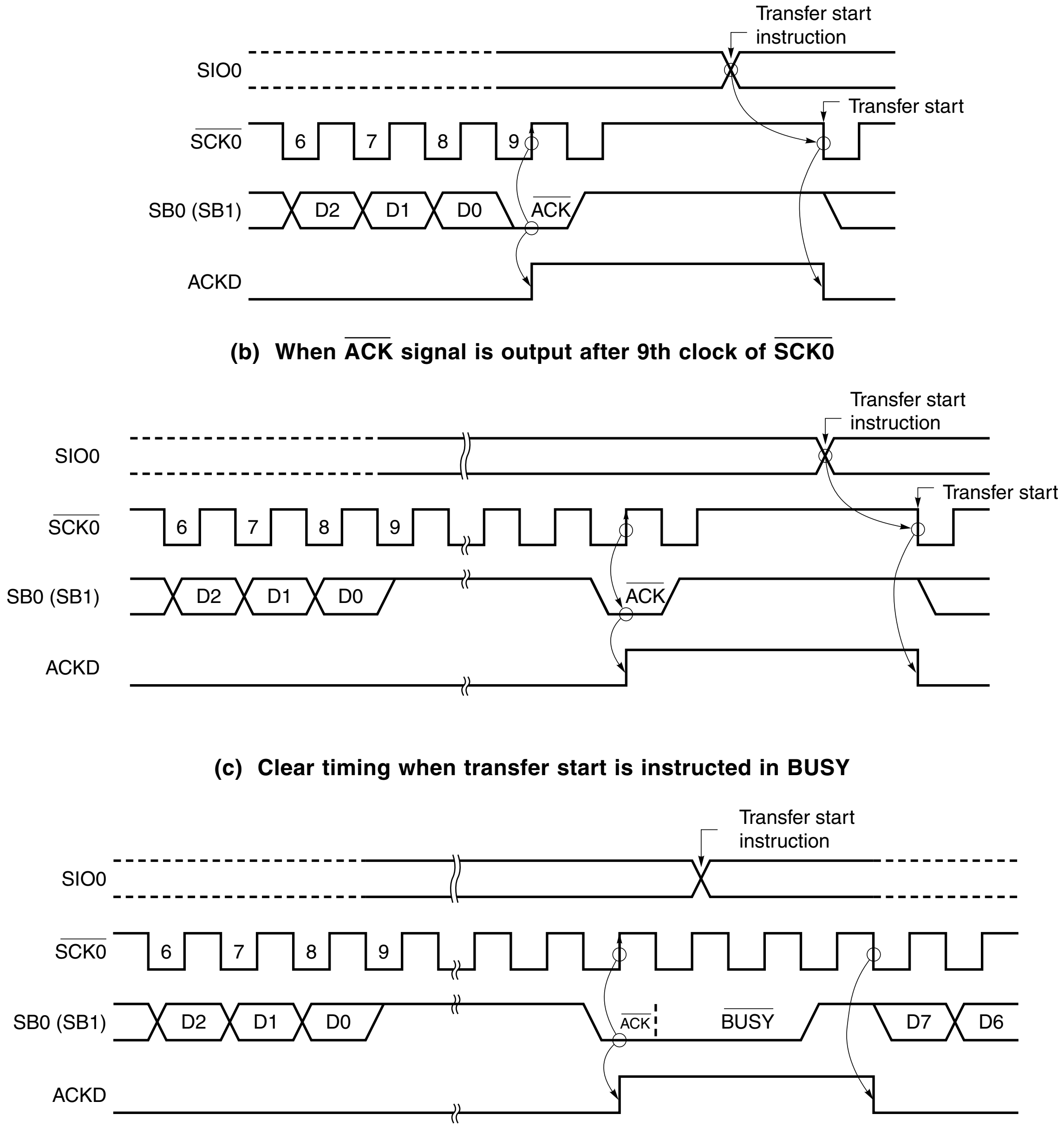
When ACKE = 0 at this point

(d) When "ACKE = 1" period is short

ACK signal is not output

If set and cleared during this period and ACKE = 0 at the falling edge of SCK0

Figure 15-24. ACKD Operations

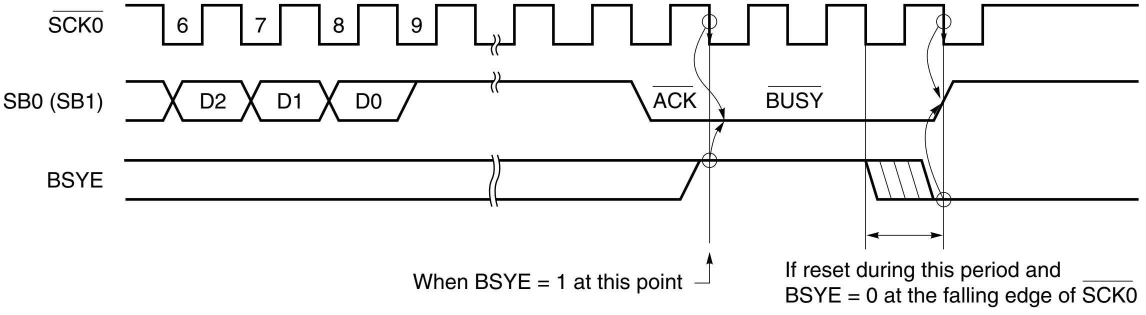


(a) When $\overline{\text{ACK}}$ signal is output at 9th clock of $\overline{\text{SCK0}}$

(b) When $\overline{\text{ACK}}$ signal is output after 9th clock of $\overline{\text{SCK0}}$

(c) Clear timing when transfer start is instructed in BUSY

Figure 15-25. BSYE Operation



SCK0
6
7
8
9
SB0 (SB1)
D2
D1
D0
ACK
BUSY
BSYE
When BSYE = 1 at this point
If reset during this period and
BSYE = 0 at the falling edge of SCK0

Table 15-3. Various Signals in SBI Mode (1/2)

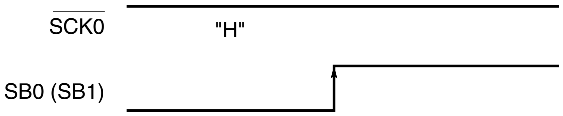
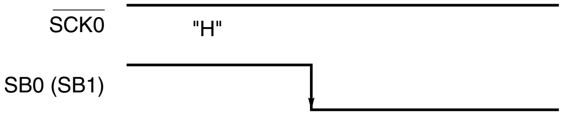
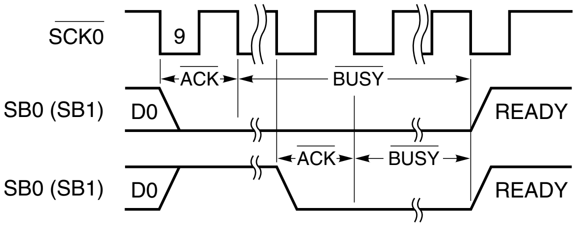
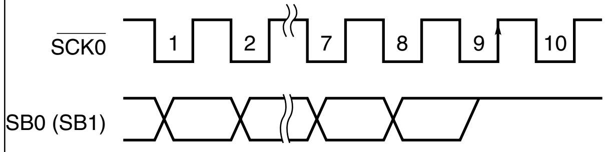
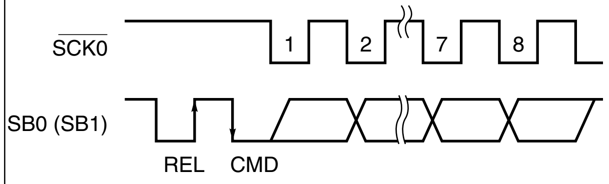
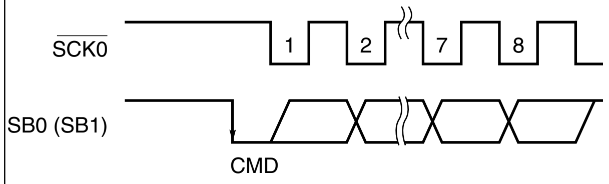
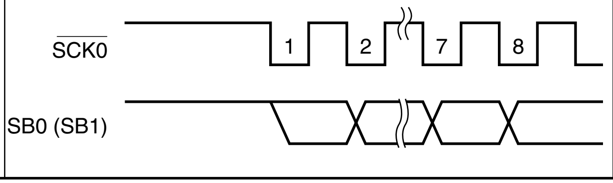
Signal Name	Output Device	Definition	Timing Chart	Output Condition	Effects on Flag	Meaning of Signal
Bus release signal (REL)	Master	SB0 (SB1) rising edge when $\overline{\text{SCK0}}$ = 1		• RELT set	• RELD set • CMDD clear	CMD signal is output to indicate that transmit data is an address.
Command signal (CMD)	Master	SB0 (SB1) falling edge when $\overline{\text{SCK0}}$ = 1		• CMDT set	• CMDD set	i) Transmit data is an address after REL signal output. ii) REL signal is not output and transmit data is a command.
Acknowledge signal ($\overline{\text{ACK}}$)	Master/ slave	Low-level signal to be output to SB0 (SB1) during one-clock period of $\overline{\text{SCK0}}$ after completion of serial reception	[Synchronous BUSY output]  $\overline{\text{SCK0}}$ 9 $\overline{\text{ACK}}$ BUSY SB0 (SB1) D0 READY SB0 (SB1) D0 $\overline{\text{ACK}}$ BUSY READY	①ACKE = 1 ②ACKT set	• ACKD set	Completion of reception
Busy signal ($\overline{\text{BUSY}}$)	Slave	[Synchronous BUSY signal] Low-level signal to be output to SB0 (SB1) following acknowledge signal		• BSYE = 1	—	Serial receive disable because of processing
Ready signal (READY)	Slave	High-level signal to be output to SB0 (SB1) before serial transfer start and after completion of serial transfer		①BSYE = 0 ②Execution of instruction for data write to SIO0 (transfer start instruction)	—	Serial receive enable

Table 15-3. Various Signals in SBI Mode (2/2)

Signal Name	Output Device	Definition	Timing Chart	Output Condition	Effects on Flag	Meaning of Signal
Serial clock ($\overline{\text{SCK0}}$)	Master	Synchronous clock to output address/command/ data, $\overline{\text{ACK}}$ signal, synchronous $\overline{\text{BUSY}}$ signal, etc. Address/command/data are transferred with the first eight synchronous clocks.		When CSIE0 = 1, execution of instruction for data write to SIO0 (serial transfer start instruction)[Note 2]	CSIIF0 set (rising edge of 9th clock of $\overline{\text{SCK0}}$)[Note 1]	Timing of signal output to serial data bus
Address (A7 to A0)	Master	8-bit data to be transferred in synchronization with $\overline{\text{SCK0}}$ after output of REL and CMD signals				Address value of slave device on the serial bus
Commands (C7 to C0)	Master	8-bit data to be transferred in synchronization with $\overline{\text{SCK0}}$ after output of only CMD signal without REL signal output				Instructions and messages to the slave device
Data (D7 to D0)	Master/ slave	8-bit data to be transferred in synchronization with $\overline{\text{SCK0}}$ without output of REL and CMD signals				Numeric values to be processed with slave or master device

Notes 1. When WUP = 0, CSIIF0 is set at the rising edge of the 9th clock of $\overline{\text{SCK0}}$.
When WUP = 1, an address is received. Only when the address matches the slave address register (SVA) value, CSIIF0 is set.

2. In $\overline{\text{BUSY}}$ state, transfer starts after the READY state is set.

(5) Pin configuration

The serial clock pin $\overline{\text{SCK0}}$ and serial data bus pin SB0 (SB1) have the following configurations.

(a) $\overline{\text{SCK0}}$ .... Serial clock I/O pin

<1> Master ... CMOS and push-pull output

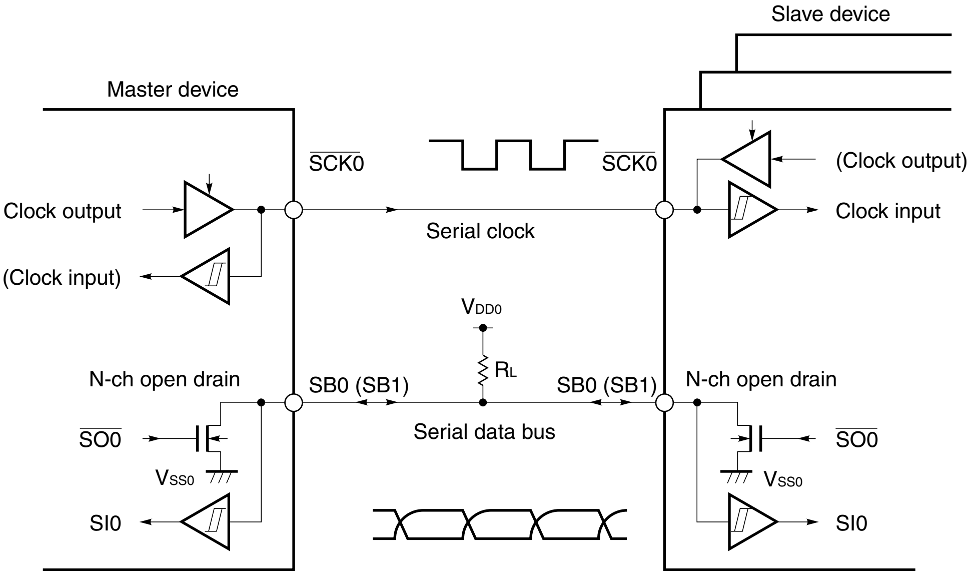
<2> Slave...... Schmitt input

(b) SB0 (SB1) .... Serial data I/O dual-function pin

Both master and slave devices have an N-ch open-drain output and a Schmitt input.

Because the serial data bus line has an N-ch open-drain output, an external pull-up resistor is necessary.

Figure 15-26. Pin Configuration



Caution Because the N-ch open-drain output pin must go into a high-impedance state at time of data reception, write FFH to serial I/O shift register 0 (SIO0) in advance. The N-ch open-drain can go into a high-impedance state at any time of transfer. However, when the wake-up function specify bit (WUP) = 1, the N-ch transistor always goes into a high-impedance state. Thus, it is not necessary to write FFH to SIO0 before reception.

(6) Address match detection method

In the SBI mode, the master transmits a slave address to select a specific slave device.

Address match is automatically detected by hardware. If the slave address transmitted by the master matches the address set to the slave address register (SVA) when the wake-up function specify bit (WUP) = 1, CSIIF0 is set.

If bit 5 (SIC) of the interrupt timing specify register (SINT) is set to 1, the wake-up function cannot be used even if WUP is set to 1 (an interrupt request signal is generated when bus release is detected). To use the wake-up function, therefore, clear SIC to 0.

Cautions 1. Slave selection/non-selection is detected by matching of the slave address received after bus release (RELD = 1).

For this match detection, match interrupt (INTCSI0) of the address to be generated with WUP = 1 is normally used. Thus, execute selection/non-selection detection by slave address when WUP = 1.

2. When detecting selection/non-selection without the use of interrupt with WUP = 0, do so by means of transmission/reception of the command preset by program instead of using the address match detection method.

(7) Error detection

In the SBI mode, the serial bus SB0 (SB1) status being transmitted is fetched into the destination device, that is, serial I/O shift register 0 (SIO0). Thus, transmit errors can be detected in the following way.

(a) Method of comparing SIO0 data before transmission to that after transmission

In this case, if two data differ from each other, a transmit error is judged to have occurred.

(b) Method of using the slave address register (SVA)

Transmit data is set to both SIO0 and SVA and is transmitted. After termination of transmission, COI bit (match signal coming from the address comparator) of serial operating mode register 0 (CSIM0) is tested. If “1”, normal transmission is judged to have been carried out. If “0”, a transmit error is judged to have occurred.

(8) Communication operation

In the SBI mode, the master device selects normally one slave device as communication target from among two or more devices by outputting an “address” to the serial bus.

After the communication target device has been determined, commands and data are transmitted/received and serial communication is realized between the master and slave devices.

Figures 15-27 to 15-30 show data communication timing charts.

Shift operation of the shift register is carried out at the falling edge of serial clock ($\overline{\text{SCK0}}$). Transmit data is latched into the SO0 latch and is output with MSB set as the first bit from the SB0/P25 or SB1/P26 pin. Receive data input to the SB0 (or SB1) pin at the rising edge of $\overline{\text{SCK0}}$ is latched into the shift register.

Figure 15-27. Address Transmission from Master Device to Slave Device (WUP = 1)

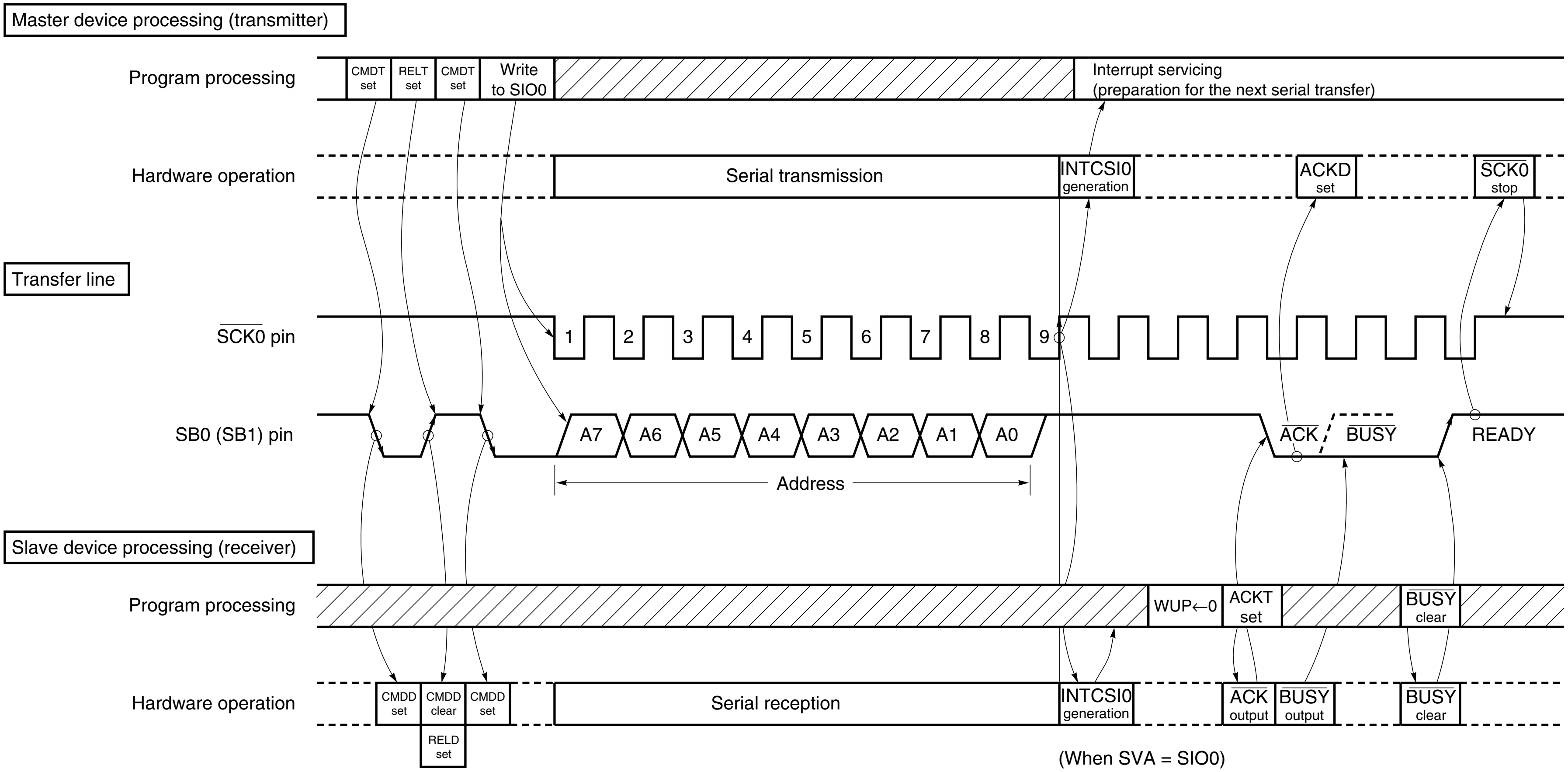


Figure 15-28. Command Transmission from Master Device to Slave Device

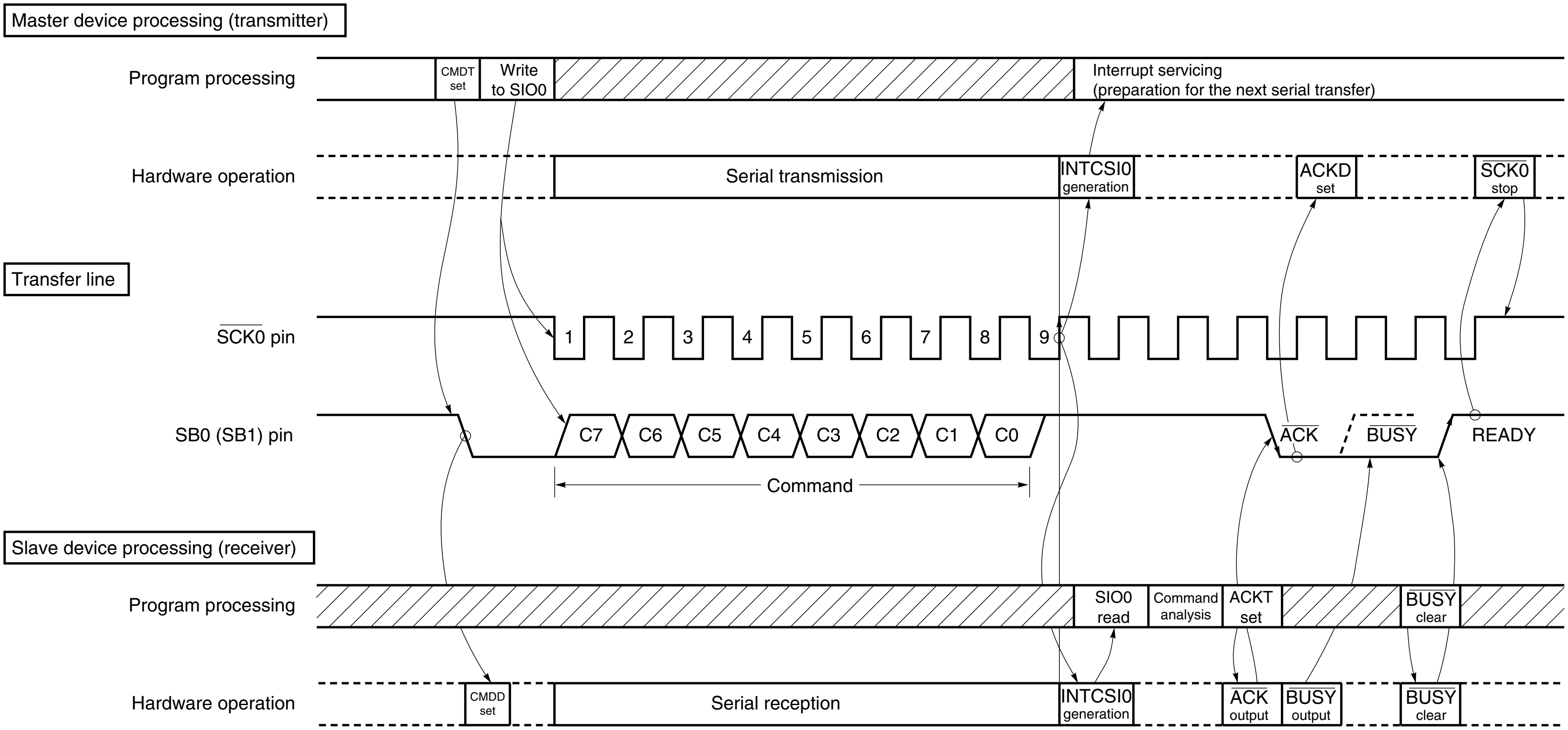


Figure 15-29. Data Transmission from Master Device to Slave Device

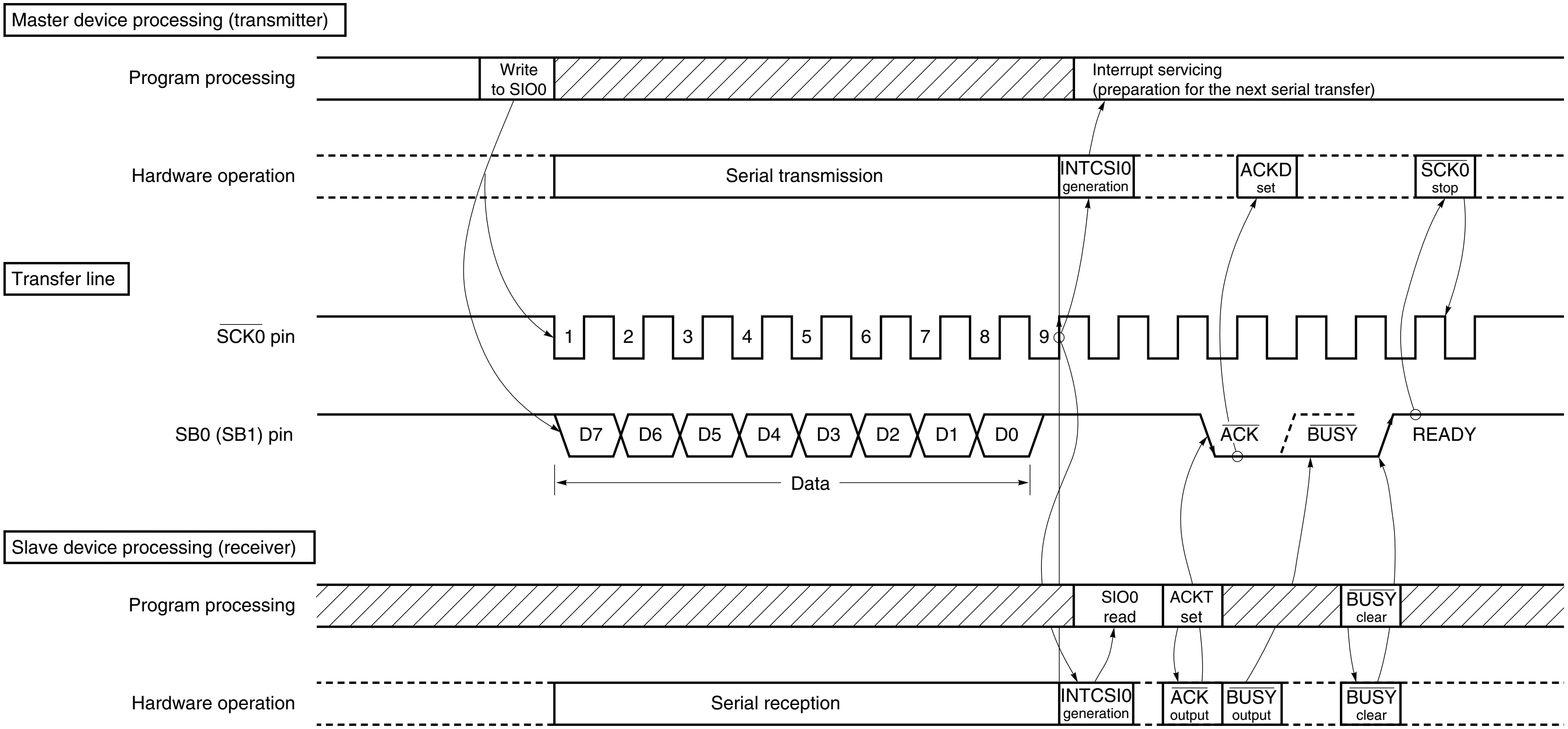
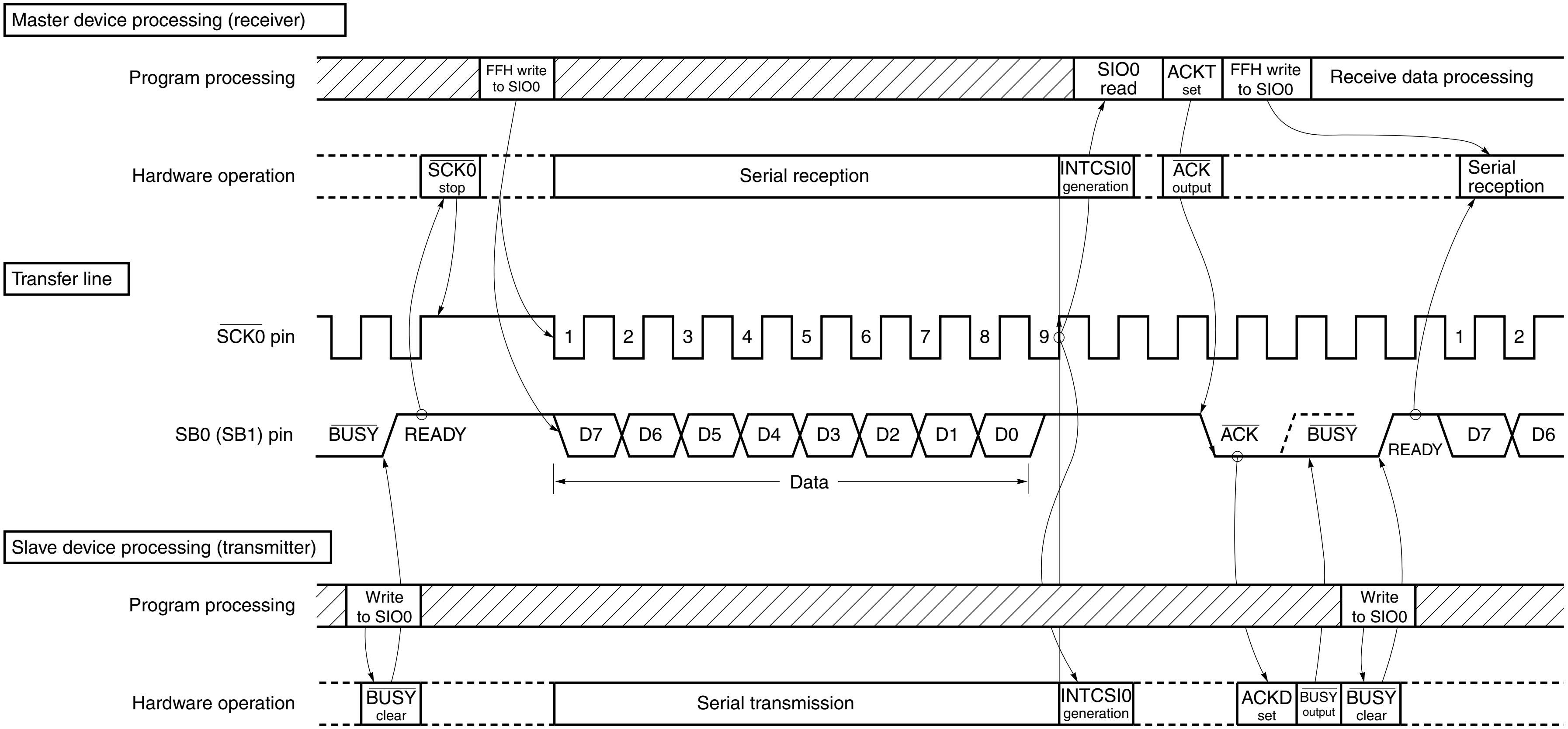

Master device processing (transmitter)
Program processing
Write to SIO0
Interrupt servicing (preparation for the next serial transfer)
Hardware operation
Serial transmission
INTCSI0 generation
ACKD set
$\overline{\text{SCK0}}$ stop
Transfer line
$\overline{\text{SCK0}}$ pin
1
2
3
4
5
6
7
8
9
SB0 (SB1) pin
D7
D6
D5
D4
D3
D2
D1
D0
$\overline{\text{ACK}}$
$\overline{\text{BUSY}}$
READY
Data
Slave device processing (receiver)
Program processing
SIO0 read
ACKT set
$\overline{\text{BUSY}}$ clear
Hardware operation
Serial reception
INTCSI0 generation
$\overline{\text{ACK}}$ output
$\overline{\text{BUSY}}$ output
$\overline{\text{BUSY}}$ clear

Figure 15-30. Data Transmission from Slave Device to Master Device



(9) Transfer start

Serial transfer is started by setting transfer data to serial I/O shift register 0 (SIO0) when the following two conditions are satisfied.

- Serial interface channel 0 operation control bit (CSIE0) = 1
- Internal serial clock is stopped or $\overline{\text{SCK0}}$ is at high level after 8-bit serial transfer.

Cautions 1. If CSIE0 is set to "1" after data write to SIO0, transfer does not start.

2. Because the N-ch transistor output pin must go into a high-impedance state for data reception, write FFH to SIO0 in advance.
However, when the wake-up function specify bit (WUP) = 1, the N-ch transistor always goes into a high-impedance state. Thus, it is not necessary to write FFH to SIO0 before reception.

3. If data is written to SIO0 when the slave is busy, the data is not lost.
When the busy state is cleared and SB0 (or SB1) input is set to the high level (READY) state, transfer starts.

Upon termination of 8-bit transfer, serial transfer automatically stops and the interrupt request flag (CSIIF0) is set.

For pins (SB0 or SB1) which are to be used for data input/output, be sure to carry out the following settings before serial transfer of the 1st byte after $\overline{\text{RESET}}$ input.

<1> Set the P25 and P26 output latches to 1.

<2> Set bit 0 (RELT) of the serial bus interface control register (SBIC) to 1.

<3> Reset the P25 and P26 output latches from 1 to 0.

(10) Identifying busy status of slave

When device is in the master mode, follow the procedure below to judge whether slave device is in the busy state or not.

<1> Detect acknowledge signal ($\overline{\mathrm{ACK}}$) or interrupt request signal generation.
<2> Set the port mode register PM25 (or PM26) of the SB0/P25 (or SB1/P26) pin into the input mode.
<3> Read out the pin state (when the pin level is high, the READY state is set).

After the detection of the READY state, set the port mode register to 0 and return to the output mode.

(11) SBI mode precautions

(a) Slave selection/non-selection is detected by match detection of the slave address received after bus release (RELD = 1).
For this match detection, match interrupt (INTCSI0) of the address to be generated with WUP = 1 is normally used. Thus, execute selection/non-selection detection by slave address when WUP = 1.

(b) When detecting selection/non-selection without the use of interrupt with WUP = 0, do so by means of transmission/reception of the command preset by program instead of using the address match detection method.

(c) If WUP is set to 1 during $\overline{\mathrm{BUSY}}$ signal output, $\overline{\mathrm{BUSY}}$ is not cleared. In SBI, the $\overline{\mathrm{BUSY}}$ signal continues to be output after $\overline{\mathrm{BUSY}}$ clear instruction generation to the falling edge of the next serial clock ($\overline{\mathrm{SCK0}}$). Before setting WUP to 1, be sure to clear $\overline{\mathrm{BUSY}}$ and then check that the SB0 (SB1) has become high-level.

(d) For pins which are to be used for data input/output, be sure to carry out the following settings before serial transfer of the 1st byte after $\overline{\mathrm{RESET}}$ input.

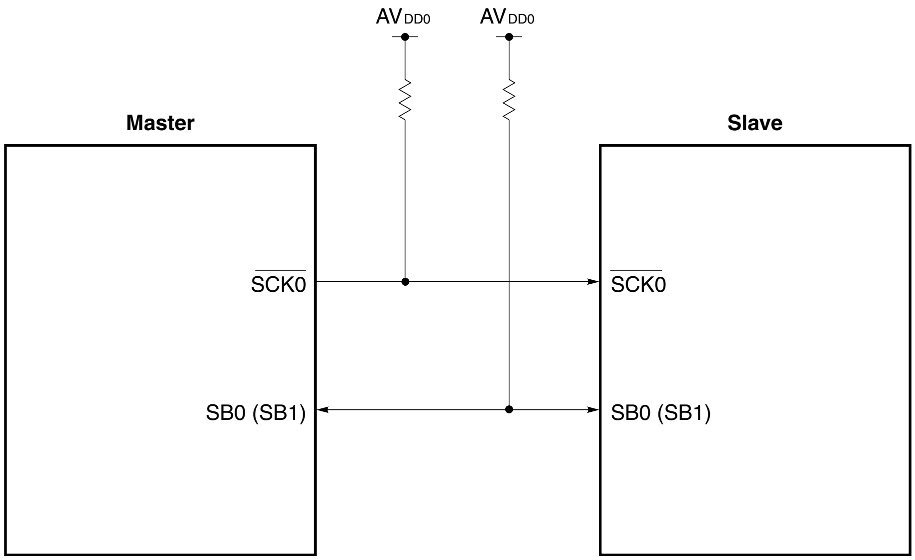
<1> Set the P25 and P26 output latches to 1.
<2> Set bit 0 (RELT) of the serial bus interface control register to 1.
<3> Reset the P25 and P26 output latches from 1 to 0.

15.4.4 2-wire serial I/O mode operation

The 2-wire serial I/O mode can cope with any communication format by program.

Communication is basically carried out with two lines of serial clock ($\overline{\text{SCK0}}$) and serial data input/output (SB0 or SB1).

Figure 15-31. Serial Bus Configuration Example Using 2-Wire Serial I/O Mode



(1) Register setting

The 2-wire serial I/O mode is set with serial operating mode register 0 (CSIM0), the serial bus interface control register (SBIC), and the interrupt timing specify register (SINT).

(a) Serial operating mode register 0 (CSIM0)

CSIM0 is set with a 1-bit or 8-bit memory manipulation instruction.

$\overline{\text{RESET}}$ input clears CSIM0 to 00H.

Symbol	⑦	⑥	⑤	4	3	2	1	0	Address	After Reset	R/W
CSIM0	CSIE0	COI	WUP	CSIM04	CSIM03	CSIM02	CSIM01	CSIM00	FF60H	00H	R/W[Note 1]

R/W

CSIE0	Serial Interface Channel 0 Operation Control
0	Operation stopped
1	Operation enabled

R

COI	Slave Address Comparison Result Flag[Note 2]
0	Slave address register not equal to serial I/O shift register 0 data
1	Slave address register equal to serial I/O shift register 0 data

R/W

WUP	Wake-up Function Control[Note 3]
0	Interrupt request signal generation with each serial transfer in any mode
1	Interrupt request signal generation when the address received after bus release (when CMDD = RELD = 1) matches the slave address register data in SBI mode

R/W

CSIM 04	CSIM 03	CSIM 02	PM25	P25	PM26	P26	PM27	P27	Operation Mode	Start Bit	SIO/SB0/P25 Pin Function	SO0/SB1/P26 Pin Function	$\overline{\text{SCK0}}$/P27 Pin Function
0	×	3-wire serial I/O mode (See **15.4.2 3-wire serial I/O mode operation**)											
1	0	SBI mode (See **15.4.3 SBI mode operation**)											
1	1	0	Note 4 ×	Note 4 ×	0	0	0	1	2-wire serial I/O mode	MSB	P25 (CMOS I/O)	SB1 (N-ch open-drain I/O)	$\overline{\text{SCK0}}$ (N-ch open-drain I/O)
		1	0	0	Note 4 ×	Note 4 ×	0	1			SB0 (N-ch open-drain I/O)	P26 (CMOS I/O)	

R/W

CSIM01	CSIM00	Serial Interface Channel 0 Clock Selection
0	×	Input clock to $\overline{\text{SCK0}}$ pin from off-chip
1	0	8-bit timer register 2 (TM2) output
1	1	Clock specified with bits 0 to 3 of timer clock select register 3 (TCL3)

Notes 1. Bit 6 (COI) is a read-only bit.

2. When CSIE0 = 0, COI becomes 0.

3. Be sure to set WUP to 0 when the 2-wire serial I/O mode is selected.

4. Can be used freely as port function.

Remark ×: don't care

PM××: Port mode register

P××: Port output latch

(b) Serial bus interface control register (SBIC)

SBIC is set with a 1-bit or 8-bit memory manipulation instruction.

$\overline{\text{RESET}}$ input clears SBIC to 00H.

Symbol	⑦	⑥	⑤	④	③	②	①	⓪	Address	After Reset	R/W
SBIC	BSYE	ACKD	ACKE	ACKT	CMDD	RELD	CMDT	RELT	FF61H	00H	R/W

R/W	CMDT	When CMDT = 1, SO latch is cleared to 0. After SO latch clearance, automatically cleared to 0. Also cleared to 0 when CSIE0 = 0.

R/W	RELT	When RELT = 1, SO latch is set to 1. After SO latch setting, automatically cleared to 0. Also cleared to 0 when CSIE0 = 0.

(c) Interrupt timing specify register (SINT)

SINT is set with a 1-bit or 8-bit memory manipulation instruction.

$\overline{\text{RESET}}$ input clears SINT to 00H.

Symbol	7	⑥	⑤	④	3	2	1	0	Address	After Reset	R/W
SINT	0	CLD	SIC	SVAM	0	0	0	0	FF63H	00H	R/W[Note 1]

R

CLD	$\overline{\text{SCK0}}$ Pin Level[Note 2]
0	Low level
1	High level

R/W

SIC	INTCSI0 Interrupt Cause Selection
0	CSIIF0 is set upon termination of serial interface channel 0 transfer
1	CSIIF0 is set upon bus release detection or termination of serial interface channel 0 transfer

Notes 1. Bit 6 (CLD) is a read-only bit.

2. When CSIE0 = 0, CLD becomes 0.

Caution Be sure to set bit 0 to bit 3 to 0.

Remark CSIIF0: Interrupt request flag corresponding to INTCSI0

CSIE0: Bit 7 of serial operating mode register 0 (CSIM0)

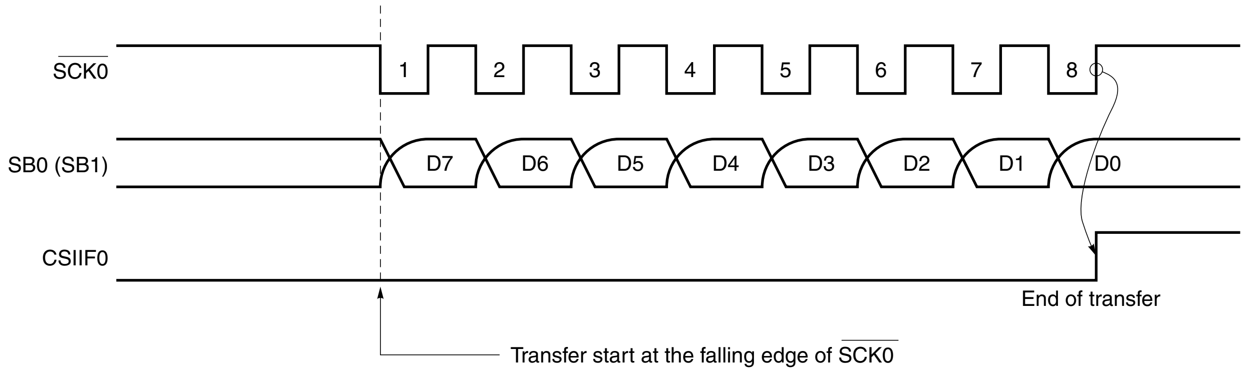
(2) Communication operation

The 2-wire serial I/O mode is used for data transmission/reception in 8-bit units. Each bit of data is transmitted or received in synchronization with the serial clock.

Shift operation of serial I/O shift register 0 (SIO0) is carried out in synchronization with the falling edge of the serial clock ($\overline{\text{SCK0}}$). The transmit data is held in the SO0 latch and is output from the SB0/P25 (or SB1/P26) pin on an MSB-first basis. The receive data input from the SB0 (or SB1) pin is latched into the shift register at the rising edge of $\overline{\text{SCK0}}$.

Upon termination of 8-bit transfer, the shift register operation stops automatically and the interrupt request flag (CSIIF0) is set.

Figure 15-32. 2-Wire Serial I/O Mode Timings



End of transfer

Transfer start at the falling edge of $\overline{\text{SCK0}}$

The SB0 (SB1) pin specified for the serial data bus is an N-ch open-drain I/O and thus it must be externally connected to a pull-up resistor. Because the N-ch transistor output pin must go into a high-impedance state for data reception, write FFH to SIO0 in advance.

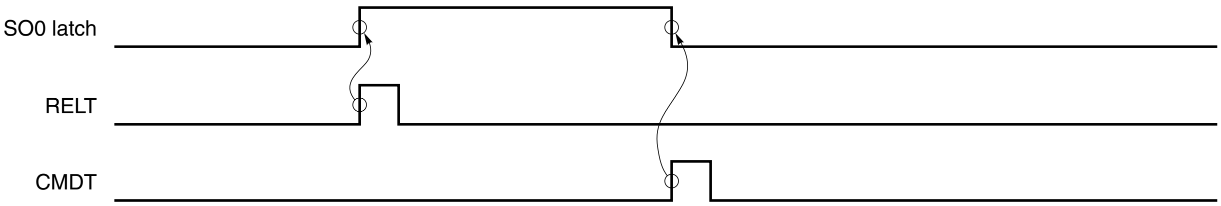
The SB0 (or SB1) pin generates the SO0 latch status and thus the SB0 (or SB1) pin output status can be manipulated by setting bit 0 (RELT) and bit 1 (CMDT) of the serial bus interface control register (SBIC). However, do not carry out this manipulation during serial transfer.

Control the $\overline{\text{SCK0}}$ pin output level in the output mode (internal system clock mode) by manipulating the P27 output latch (refer to **15.4.5 $\overline{\text{SCK0}}$/P27 pin output manipulation**).

(3) Other signals

Figure 15-33 shows RELT and CMDT operations.

Figure 15-33. RELT and CMDT Operations



(4) Transfer start

Serial transfer is started by setting transfer data to serial I/O shift register 0 (SIO0) when the following two conditions are satisfied.

- Serial interface channel 0 operation control bit (CSIE0) = 1
- Internal serial clock is stopped or $\overline{\text{SCK0}}$ is at high level after 8-bit serial transfer.

Cautions 1. If CSIE0 is set to "1" after data write to SIO0, transfer does not start.

2. Because the N-ch transistor output pin must go into a high-impedance state for data reception, write FFH to SIO0 in advance.

Upon termination of 8-bit transfer, serial transfer automatically stops and the interrupt request flag (CSIIF0) is set.

(5) Error detection

In the 2-wire serial I/O mode, the serial bus SB0 (SB1) status being transmitted is fetched into the destination device, that is, serial I/O shift register 0 (SIO0). Thus, transmit error can be detected in the following way.

(a) Method of comparing SIO0 data before transmission to that after transmission

In this case, if two data differ from each other, a transmit error is judged to have occurred.

(b) Method of using the slave address register (SVA)

Transmit data is set to both SIO0 and SVA and is transmitted. After termination of transmission, COI bit (match signal coming from the address comparator) of serial operating mode register 0 (CSIM0) is tested. If "1", normal transmission is judged to have been carried out. If "0", a transmit error is judged to have occurred.

15.4.5 $\overline{\text{SCK0}}$/P27 pin output manipulation

Because the $\overline{\text{SCK0}}$/P27 pin incorporates an output latch, static output is also possible by software in addition to normal serial clock output.

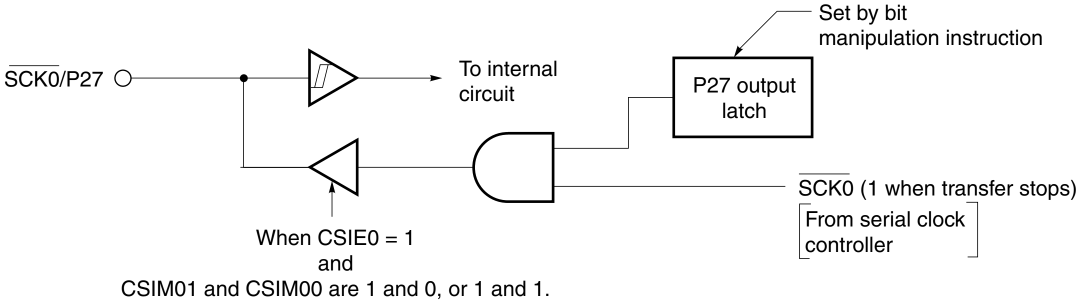
P27 output latch manipulation enables any value of $\overline{\text{SCK0}}$ to be set by software (SI0/SB0 and SO0/SB1 pin to be controlled with the RELT and CMDT bits of the serial bus interface control register (SBIC)).

$\overline{\text{SCK0}}$/P27 pin output manipulating procedure is described below.

<1> Set serial operating mode register 0 (CSIM0) ($\overline{\text{SCK0}}$ pin is set in the output mode and serial operation is enabled). While serial transfer is suspended, $\overline{\text{SCK0}}$ is set to 1.

<2> Manipulate the content of the P27 output latch by executing the bit manipulation instruction.

Figure 15-34. $\overline{\text{SCK0}}$/P27 Pin Configuration


$\overline{\text{SCK0}}$/P27
To internal circuit
Set by bit manipulation instruction
P27 output latch
$\overline{\text{SCK0}}$ (1 when transfer stops)
From serial clock controller
When CSIE0 = 1
and
CSIM01 and CSIM00 are 1 and 0, or 1 and 1.

CHAPTER 16 SERIAL INTERFACE CHANNEL 0 (μPD780308Y SUBSERIES)

The μPD780308Y Subseries incorporates three channels of serial interfaces. Differences between channels 0, 2, and 3 are as follows (refer to **CHAPTER 17 SERIAL INTERFACE CHANNEL 2** for details of serial interface channel 2, and **CHAPTER 18 SERIAL INTERFACE CHANNEL 3** for details of serial interface channel 3, respectively).

Table 16-1. Differences Between Channels 0, 2, and 3

Serial Transfer Mode		Channel 0	Channel 2	Channel 3
3-wire serial I/O	Clock selection	$f_{xx}/2$, $f_{xx}/2^2$, $f_{xx}/2^3$, $f_{xx}/2^4$, $f_{xx}/2^5$, $f_{xx}/2^6$, $f_{xx}/2^7$, $f_{xx}/2^8$, external clock, TO2 output	External clock, baud rate generator output	$f_{xx}/2$, $f_{xx}/2^2$, $f_{xx}/2^3$, $f_{xx}/2^4$, $f_{xx}/2^5$, $f_{xx}/2^6$, $f_{xx}/2^7$, $f_{xx}/2^8$, external clock
	Transfer method	MSB/LSB switchable as the start bit	MSB/LSB switchable as the start bit	MSB/LSB switchable as the start bit
	Transfer end flag	Serial transfer end interrupt request flag (CSIIF0)	Serial transfer end interrupt request flag (SRIF)	Serial transfer end interrupt request flag (CSIIF3)
I^2C bus (Inter IC Bus)		Use possible	None	None
2-wire serial I/O				
UART (Asynchronous serial interface)		None	Use possible	None

16.1 Serial Interface Channel 0 Functions

Serial interface channel 0 employs the following four modes.

- Operation stop mode
- 3-wire serial I/O mode
- 2-wire serial I/O mode
- I^{2}C (Inter IC) bus mode

Caution Do not change the operating mode (3-wire serial I/O, 2-wire serial I/O, or I^{2}C bus) while serial interface channel 0 is enabled. To change the operating mode, stop the serial operation once.

(1) Operation stop mode

This mode is used when serial transfer is not carried out. Power consumption can be reduced.

(2) 3-wire serial I/O mode (MSB-/LSB-first selectable)

This mode is used for 8-bit data transfer using three lines, one each for serial clock ($\overline{\text{SCK0}}$), serial output (SO0) and serial input (SI0). This mode enables simultaneous transmission/reception and therefore reduces the data transfer processing time.

The start bit of transferred 8-bit data is switchable between MSB and LSB, so that devices can be connected regardless of their start bit recognition.

This mode should be used when connecting with peripheral I/O devices or display controllers which incorporate a conventional synchronous clocked serial interface as is the case with the 75X/XL, 78K, and 17K Series.

(3) 2-wire serial I/O mode (MSB-first)

This mode is used for 8-bit data transfer using two lines of serial clock ($\overline{\text{SCK0}}$) and serial data bus (SB0 or SB1).

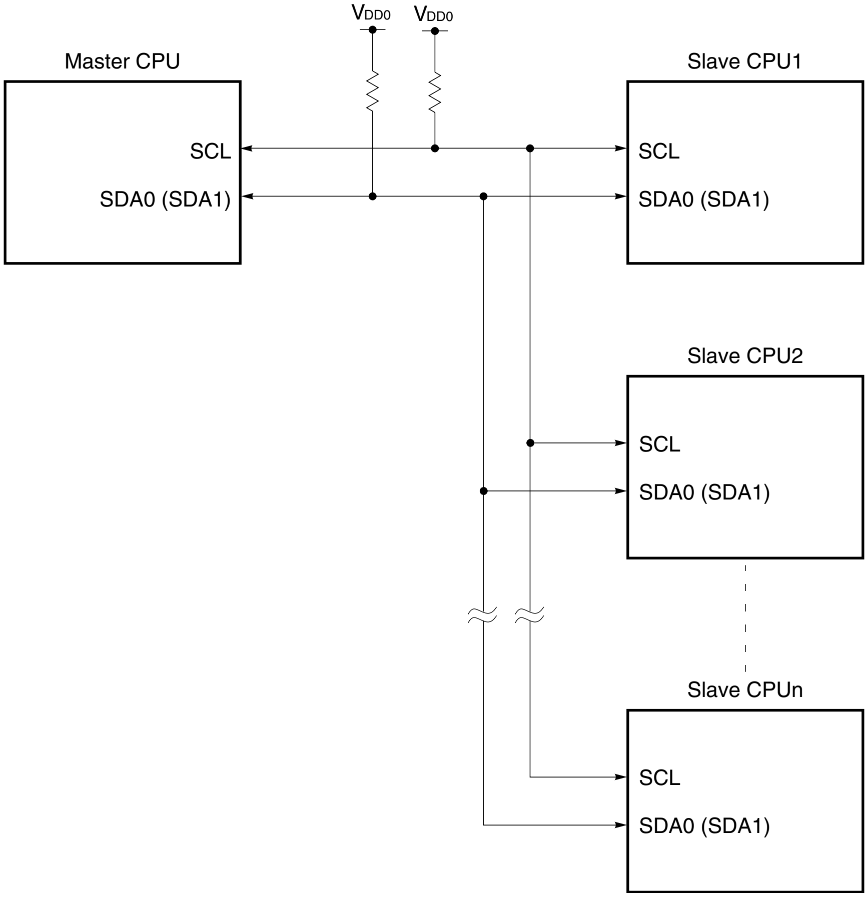
This mode enables to cope with any one of the possible data transfer formats by controlling the $\overline{\text{SCK0}}$ level and the SB0 or SB1 output level. Thus, the handshake line previously necessary for connection of two or more devices can be removed, resulting in the increased number of available I/O ports.

(4) I^{2}C (Inter IC) bus mode (MSB-first)

This mode is used for 8-bit data transfer with two or more devices using two lines of serial clock (SCL) and serial data bus (SDA0 or SDA1).

This mode is in compliance with the I^{2}C bus format. In this mode, the transmitter outputs three kinds of data onto the serial data bus: “start condition”, “data”, and “stop condition”, to be actually sent or received. The receiver automatically distinguishes the received data into “start condition”, “data”, or “stop condition”, by hardware.

Figure 16-1. Serial Bus Configuration Example Using I^{2}C Bus



16.2 Serial Interface Channel 0 Configuration

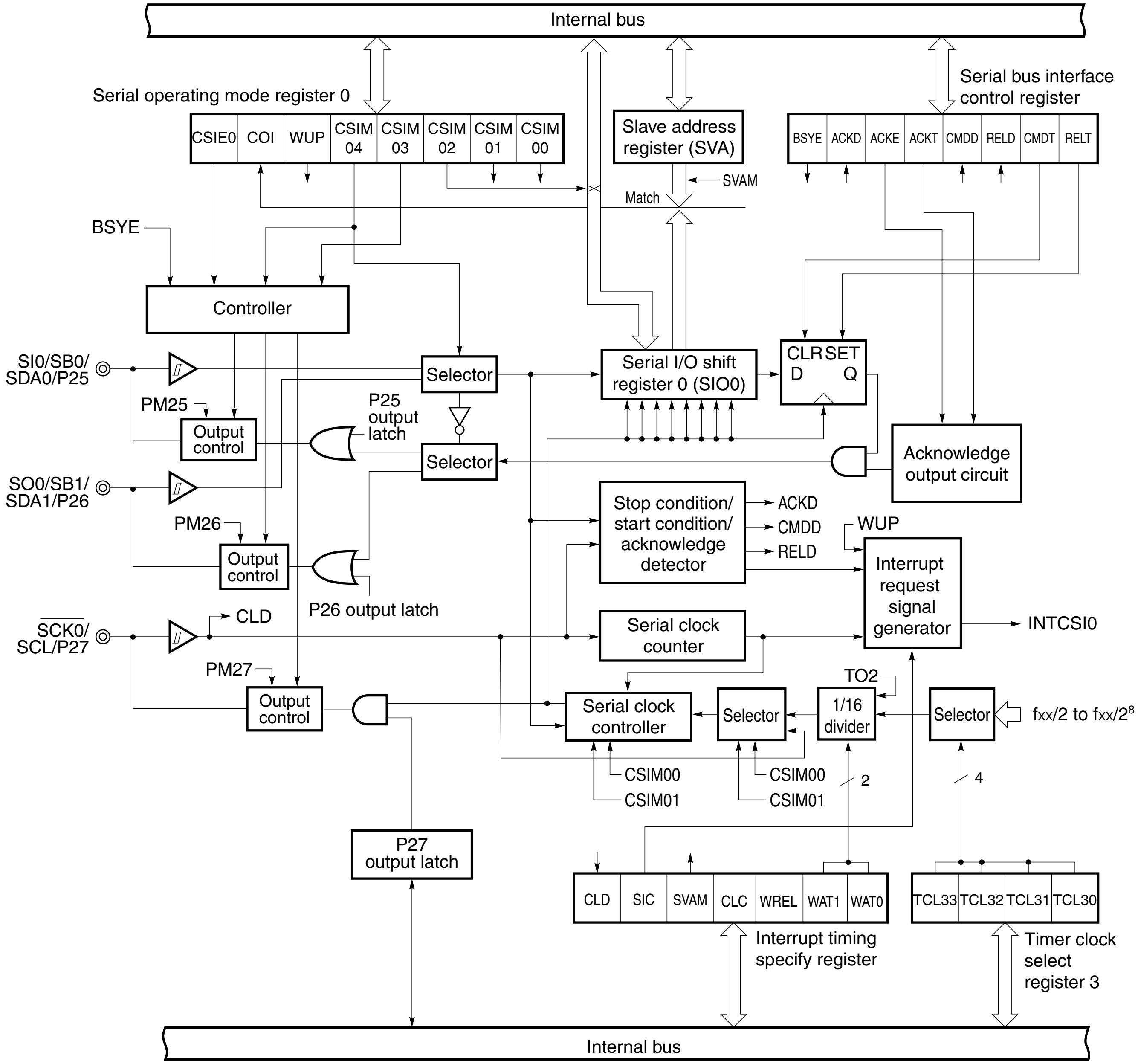
Serial interface channel 0 consists of the following hardware.

Table 16-2. Serial Interface Channel 0 Configuration

Item	Configuration
Register	Serial I/O shift register 0 (SIO0) Slave address register (SVA)
Control register	Timer clock select register 3 (TCL3) Serial operating mode register 0 (CSIM0) Serial bus interface control register (SBIC) Interrupt timing specify register (SINT) Port mode register 2 (PM2)[Note]

Note Refer to **Figure 6-7 P25, P26 Block Diagram (μPD780308Y Subseries)** and **Figure 6-8 P27 Block Diagram (μPD780308Y Subseries)**.

Figure 16-2. Serial Interface Channel 0 Block Diagram



Remarks 1. Output control performs selection between CMOS output and N-ch open-drain output.

2. fxx = fx/2 (MCS = 0), fxx = fx (MCS = 1)

(1) Serial I/O shift register 0 (SIO0)

This is an 8-bit register to carry out parallel-serial conversion and to carry out serial transmission/reception (shift operation) in synchronization with the serial clock.

SIO0 is set with an 8-bit memory manipulation instruction.

When bit 7 (CSIE0) of serial operating mode register 0 (CSIM0) is 1, writing data to SIO0 starts serial operation. In transmission, data written to SIO0 is output to the serial output (SO0) or serial data bus (SB0/SB1). In reception, data is read from the serial input (SI0) or SB0/SB1 to SIO0.

Note that, if a bus is driven in the I^{2}C bus mode or 2-wire serial I/O mode, the bus pin must serve for both input and output. Therefore, the transmission N-ch transistor of the device which will start reception of data must be turned off beforehand. Consequently, write FFH to SIO0 in advance.
In the I^{2}C bus mode, set SIO0 to FFH with bit 7 (BSYE) of the serial bus interface control register (SBIC) set to 1.
$\overline{\text{RESET}}$ input makes SIO0 undefined.

Caution Do not execute an instruction that writes SIO0 while WUP (bit 5 of serial operating mode register 0 (CSIM0)) is 1 in the I^{2}C bus mode. Even if this instruction is not executed, data can be received when the wake-up function is used (WUP = 1). For the details of the wake-up function, refer to 16.4.4 (1) (c) Wake-up function.

(2) Slave address register (SVA)

This is an 8-bit register to set the slave address value for connection of a slave device to the serial bus.
SVA is set with an 8-bit memory manipulation instruction. This register is not used in the 3-wire serial I/O mode.
The master device outputs a slave address for selection of a particular slave device to the connected slave device. These two data (the slave address output from the master device and the SVA value) are compared with an address comparator. If they match, the slave device has been selected. In that case, bit 6 (COI) of serial operating mode register 0 (CSIM0) becomes 1.
By setting bit 4 (SVAM) of the interrupt timing specify register (SINT) to 1, the address can be compared using the data of the LSB-masked higher 7 bits.
If no matching is detected in address reception, bit 2 (RELD) of the serial bus interface control register (SBIC) is cleared to 0.
In the I^{2}C bus mode, the wake-up function can be used by setting bit 5 (WUP) of CSIM0 to 1. In this case, an interrupt request signal (INTCSI0) is generated when the slave address output by the master matches the value of SVA (the interrupt request signal is also generated when the stop condition is detected). This interrupt request indicates that the master has requested for communication. Note that SIC must be set to 1 when the wake-up function is used.
When the device is used as the master or slave in the 2-wire serial I/O or I^{2}C bus mode, detect an error by using the slave address register (SVA).
$\overline{\text{RESET}}$ input makes SVA undefined.

(3) SO0 latch

This latch holds SI0/SB0/SDA0/P25 and SO0/SB1/SDA1/P26 pin levels. It can be directly controlled by software.

(4) Serial clock counter

This counter counts the serial clocks to be output and input during transmission/reception and to check whether 8-bit data has been transmitted/received.

(5) Serial clock controller

This circuit controls serial clock supply to serial I/O shift register 0 (SIO0). When the internal system clock is used, the circuit also controls clock output to the $\overline{\text{SCK0}}$/SCL/P27 pin.

(6) Interrupt request signal generator

This circuit controls interrupt request signal generation. It generates interrupt request signals according to the settings of interrupt timing specify register (SINT) bits 0 and 1 (WAT0, WAT1) and serial operating mode register 0 (CSIM0) bit 5 (WUP), as shown in Table 16-3.

(7) Acknowledge output circuit and stop condition/start condition/acknowledge detector

These two circuits output and detect various control signals in the I^2C bus mode.

These do not operate in the 3-wire serial I/O mode and 2-wire serial I/O mode.

Table 16-3. Serial Interface Channel 0 Interrupt Request Signal Generation

Serial Transfer Mode	BSYE	WUP	WAT1	WAT0	ACKE	Description
3-wire or 2-wire serial I/O mode	0	0	0	0	0	An interrupt request signal is generated each time 8 serial clocks are counted.
	Other than above					Setting prohibited
I^2C bus mode (transmit)	0	0	1	0	0	An interrupt request signal is generated each time 8 serial clocks are counted (8-clock wait). Normally, during transmission the settings WAT21, WAT0 = 1, 0, are not used. They are used only when wanting to coordinate receive time and processing systematically using software. ACK information is generated by the receiving side, thus ACKE should be set to 0 (disable).
			1	1	0	An interrupt request signal is generated each time 9 serial clocks are counted (9-clock wait). ACK information is generated by the receiving side, thus ACKE should be set to 0 (disable).
	Other than above					Setting prohibited
I^2C bus mode (receive)	1	0	1	0	0	An interrupt request signal is generated each time 8 serial clocks are counted (8-clock wait). ACK information is output by manipulating ACKT by software after an interrupt request is generated.
			1	1	0/1	An interrupt request signal is generated each time 9 serial clocks are counted (9-clock wait). To automatically generate ACK information, preset ACKE to 1 before transfer start. However, in the case of the master, set ACKE to 0 (disable) before receiving the last data.
	1	1	1	1	1	Generates an interrupt request signal when the values of serial I/O shift register 0 (SIO0) and slave address register (SVA) match, or when the stop condition is detected, after an address has been received. To automatically generate ACK information, preset ACKE to 1 (enable) before transfer start.
	Other than above					Setting prohibited

BSYE: Bit 7 of serial bus interface control register (SBIC)

ACKE: Bit 5 of serial bus interface control register (SBIC)

16.3 Serial Interface Channel 0 Control Registers

The following four types of registers are used to control serial interface channel 0.

- Timer clock select register 3 (TCL3)
- Serial operating mode register 0 (CSIM0)
- Serial bus interface control register (SBIC)
- Interrupt timing specify register (SINT)

(1) Timer clock select register 3 (TCL3)

This register sets the serial clock of serial interface channel 0.
TCL3 is set with an 8-bit memory manipulation instruction.
$\overline{\text{RESET}}$ input sets TCL3 to 88H.

Figure 16-3. Timer Clock Select Register 3 Format

Symbol	7	6	5	4	3	2	1	0	Address	After Reset	R/W
TCL3	1	0	0	0	TCL33	TCL32	TCL31	TCL30	FF43H	88H	R/W

TCL33	TCL32	TCL31	TCL30	Serial Interface Channel 0 Serial Clock Selection			
				Serial Clock in I²C Bus Mode		Serial Clock in 2-Wire or 3-Wire Serial I/O Mode	
				MCS = 1	MCS = 0	MCS = 1	MCS = 0
0	1	1	0	Setting prohibited	$f_X/2^6$ (78.1 kHz)	Setting prohibited	$f_X/2^2$ (1.25 MHz)
0	1	1	1	$f_X/2^6$ (78.1 kHz)	$f_X/2^7$ (39.1 kHz)	$f_X/2^2$ (1.25 MHz)	$f_X/2^3$ (625 kHz)
1	0	0	0	$f_X/2^7$ (39.1 kHz)	$f_X/2^8$ (19.5 kHz)	$f_X/2^3$ (625 kHz)	$f_X/2^4$ (313 kHz)
1	0	0	1	$f_X/2^8$ (19.5 kHz)	$f_X/2^9$ (9.77 kHz)	$f_X/2^4$ (313 kHz)	$f_X/2^5$ (156 kHz)
1	0	1	0	$f_X/2^9$ (9.77 kHz)	$f_X/2^{10}$ (4.88 kHz)	$f_X/2^5$ (156 kHz)	$f_X/2^6$ (78.1 kHz)
1	0	1	1	$f_X/2^{10}$ (4.88 kHz)	$f_X/2^{11}$ (2.44 kHz)	$f_X/2^6$ (78.1 kHz)	$f_X/2^7$ (39.1 kHz)
1	1	0	0	$f_X/2^{11}$ (2.44 kHz)	$f_X/2^{12}$ (1.22 kHz)	$f_X/2^7$ (39.1 kHz)	$f_X/2^8$ (19.5 kHz)
1	1	0	1	$f_X/2^{12}$ (1.22 kHz)	$f_X/2^{13}$ (0.61 kHz)	$f_X/2^8$ (19.5 kHz)	$f_X/2^9$ (9.8 kHz)
Other than above				Setting prohibited			

Cautions 1. Set bit 4 to bit 6 to 0, and bit 7 to 1.

2. When rewriting TCL3 to other data, stop the serial transfer operation beforehand.

Remarks 1. f_X: Main system clock oscillation frequency

2. MCS: Oscillation mode select register bit 0

3. Figures in parentheses apply to operation with f_X = 5.0 MHz.

(2) Serial operating mode register 0 (CSIM0)

This register sets serial interface channel 0 serial clock, operating mode, operation enable/stop, wake-up function and displays the address comparator match signal.

CSIM0 is set with a 1-bit or 8-bit memory manipulation instruction.

$\overline{\text{RESET}}$ input clears CSIM0 to 00H.

Caution Do not change the operating mode (3-wire serial I/O, 2-wire serial I/O, or I^{2}C bus) while serial interface channel 0 is enabled. To change the operating mode, stop the serial operation once.

Figure 16-4. Serial Operating Mode Register 0 Format (1/2)

Symbol	⑦	⑥	⑤	4	3	2	1	0	Address	After Reset	R/W
CSIM0	CSIE0	COI	WUP	CSIM04	CSIM03	CSIM02	CSIM01	CSIM00	FF60H	00H	R/W[Note 1]

R/W	CSIE0	Serial Interface Channel 0 Operation Control
	0	Operation stopped
	1	Operation enabled

R	COI	Slave Address Comparison Result Flag[Note 2]
	0	Slave address register not equal to serial I/O shift register 0 data
	1	Slave address register equal to serial I/O shift register 0 data

R/W	WUP	Wake-up Function Control[Note 3]
	0	Interrupt request signal generation with each serial transfer in any mode
	1	Interrupt request signal generation when the address received after detecting start condition (when CMDD = 1) matches the slave address register data in I^{2}C bus mode

Notes 1. Bit 6 (COI) is a read-only bit.

2. When CSIE0 = 0, COI becomes 0.

3. Set bit 5 (SIC) of the interrupt timing specify register (SINT) to 1 when using the wake-up function (WUP = 1). Do not execute an instruction that writes to serial I/O shift register 0 (SIO0) while WUP = 1.

Remark

×:	don't care
PM××:	Port mode register
P××:	Port output latch

Figure 16-4. Serial Operating Mode Register 0 Format (2/2)

R/W

CSIM 04	CSIM 03	CSIM 02	PM25	P25	PM26	P26	PM27	P27	Operation Mode	Start Bit	SI0/SB0/SDA0/ P25 Pin Function	SO0/SB1/SDA1/ P26 Pin Function	$\overline{\text{SCK0}}$/SCL/P27 Pin Function
0	×	0	1[Note 2]	×[Note 2]	0	0	0	1	3-wire serial I/O mode	MSB	SI0[Note 1] (Input)	SO0 (CMOS output)	$\overline{\text{SCK0}}$ (CMOS I/O)
		1								LSB			
1	1	0	×[Note 3]	×[Note 3]	0	0	0	1	2-wire serial I/O mode or I^{2}C bus mode	MSB	P25 (CMOS I/O)	SB1/SDA1 (N-ch open-drain I/O)	$\overline{\text{SCK0}}$/SCL (N-ch open-drain I/O)
		1	0	0	×[Note 3]	×[Note 3]	0	1			SB0/SDA0 (N-ch open-drain I/O)	P26 (CMOS I/O)	

R/W

CSIM01	CSIM00	Serial Interface Channel 0 Clock Selection
0	×	Input clock to $\overline{\text{SCK0}}$/SCL pin from off-chip
1	0	8-bit timer register 2 (TM2) output[Note 4]
1	1	Clock specified with bits 0 to 3 of timer clock select register 3 (TCL3)

Notes 1. Set bit 5 (SIC) of the interrupt timing specify register (SINT) to 1 when using the wake-up function (WUP = 1). Do not execute an instruction that writes to serial I/O shift register 0 (SIO0) while WUP = 1.

2. This pin can be used as P25 (CMOS I/O) only to transmit data.

3. These pins can be used freely as port pins.

4. In the I^{2}C bus mode, the clock frequency becomes 1/16 of that output from TO2.

(3) Serial bus interface control register (SBIC)

This register sets serial bus interface operation and displays statuses.

SBIC is set with a 1-bit or 8-bit memory manipulation instruction.

$\overline{\text{RESET}}$ input clears SBIC to 00H.

Figure 16-5. Serial Bus Interface Control Register Format (1/2)

Symbol	⑦	⑥	⑤	④	③	②	①	⓪	Address	After Reset	R/W
SBIC	BSYE	ACKD	ACKE	ACKT	CMDD	RELD	CMDT	RELT	FF61H	00H	R/W[Note 1]

R/W	[Note 2] BSYE	Control of N-ch Open-Drain Output for Transmission in I^{2}C Bus Mode[Note 3]
	0	Output enabled (transmission)
	1	Output disabled (reception)

R	ACKD	Acknowledge Detection
	Clear Conditions (ACKD = 0)	Set Conditions (ACKD = 1)
	• When transfer start instruction is executed • When CSIE0 = 0 • When $\overline{\text{RESET}}$ input is applied	• When acknowledge signal is detected at the rising edge of SCL clock after completion of transfer

R/W	ACKE	Acknowledge Signal Automatic Output Control[Note 4]
	0	Disables acknowledge signal automatic output. (However, output with ACKT is enabled) Used for reception when 8-clock wait mode is selected or for transmission.[Note 5]
	1	Enables acknowledge signal automatic output. Outputs acknowledge signal in synchronization with the falling edge of the 9th SCL clock cycle (automatically output when ACKE = 1). However, not automatically cleared to 0 after acknowledge signal output. Used in reception with 9-clock wait mode selected.

R/W	ACKT	Keeps SDA0 (SDA1) low from set instruction (ACKT = 1) execution to the next falling edge of SCL. Used to generate the $\overline{\text{ACK}}$ signal by software when 8-clock wait mode is selected. Also cleared to 0 upon start of serial interface transfer or when CSIE0 = 0.

R	CMDD	Start Condition Detection
	Clear Conditions (CMDD = 0)	Set Conditions (CMDD = 1)
	• When transfer start instruction is executed • When stop condition signal is detected • When CSIE0 = 0 • When $\overline{\text{RESET}}$ input is applied	• When start condition signal is detected

Notes 1. Bits 2, 3, and 6 (RELD, CMDD, and ACKD) are read-only bits.

2. The busy mode can be cancelled by start of serial interface transfer or reception of address signal. However, the BSYE flag is not cleared to 0.

3. When using the wake-up function, be sure to set BSYE to 1.

4. Setting should be performed before transfer.

5. If 8-clock wait mode is selected, the acknowledge signal at reception time must be output using ACKT.

Remarks 1. Bits 0, 1, and 4 (RELT, CMDT, and ACKT) are 0 when read after data setting.

2. CSIE0: Bit 7 of serial operating mode register 0 (CSIM0)

Figure 16-5. Serial Bus Interface Control Register Format (2/2)

R

RELD	Stop Condition Detection
Clear Conditions (RELD = 0)	Set Conditions (RELD =1)
• When transfer start instruction is executed • If SIO0 and SVA values do not match in address reception • When CSIE0 = 0 • When $\overline{\text{RESET}}$ input is applied	• When stop condition signal is detected

R/W		
	CMDT	Used for start condition signal output. When CMDT = 1, SO latch is cleared (to 0). After SO latch clearance, automatically cleared to 0. Also cleared to 0 when CSIE0 = 0.

R/W		
	RELT	Used for stop condition signal output. When RELT = 1, SO latch is set to 1. After SO latch setting, automatically cleared to 0. Also cleared to 0 when CSIE0 = 0.

CSIE0: Bit 7 of serial operating mode register 0 (CSIM0)

(4) Interrupt timing specify register (SINT)

This register controls interrupt, wait, and clock level, sets the address mask functions, and displays the $\overline{\text{SCK0}}$/SCL pin level status.

SINT is set with a 1-bit or 8-bit memory manipulation instruction.

$\overline{\text{RESET}}$ input clears SINT to 00H.

Figure 16-6. Interrupt Timing Specify Register Format (1/2)

Symbol	7	⑥	⑤	④	③	②	1	0	Address	After Reset	R/W
SINT	0	CLD	SIC	SVAM	CLC	WREL	WAT1	WAT0	FF63H	00H	R/W[Note 1]

R	CLD	$\overline{\text{SCK0}}$/SCL Pin Level[Note 2]
	0	Low level
	1	High level

R/W	SIC	INTCSI0 Interrupt Cause Selection[Note 3]
	0	CSIIF0 is set to 1 upon termination of serial interface channel 0 transfer
	1	CSIIF0 is set to 1 upon stop condition detection or termination of serial interface channel 0 transfer

R/W	SVAM	SVA Bit to Be Used as Slave Address
	0	Bits 0 to 7
	1	Bits 1 to 7

R/W	CLC	Clock Level Control[Note 4]
	0	Used in I^{2}C bus mode. Make output level of SCL pin low unless serial transfer is being performed.
	1	Used in I^{2}C bus mode. Make SCL pin enter high-impedance state unless serial transfer is being performed (except for clock line which is kept high). Used to enable master device to generate start condition and stop condition signals.

Notes **1.** Bit 6 (CLD) is a read-only bit.

2. When CSIE0 = 0, CLD becomes 0.

3. When using wake-up function in the I^{2}C bus mode, set SIC to 1.

4. When not using the I^{2}C bus mode, set CLC to 0.

Remark		
	SVA:	Slave address register
	CSIIF0:	Interrupt request flag corresponding to INTCSI0
	CSIE0:	Bit 7 of serial operating mode register 0 (CSIM0)

Figure 16-6. Interrupt Timing Specify Register Format (2/2)

R/W

WREL	Wait Sate Cancellation Control
0	Wait state has been cancelled.
1	Cancels wait state. Automatically cleared to 0 when the state is cancelled. (Used to cancel wait state by means of WAT0 and WAT1.)

R/W

WAT1	WAT0	Wait and Interrupt Control
0	0	Generates interrupt service request at rising edge of 8th $\overline{\text{SCK0}}$ clock cycle (keeping clock output in high impedance).
0	1	Setting prohibited
1	0	Used in I^{2}C bus mode (8-clock wait). Generates interrupt service request at rising edge of 8th SCL clock cycle. (In the case of master device, makes SCL output low to enter wait state after 8 clock pulses are output. In the case of slave device, makes SCL output low to request wait state after 8 clock pulses are input.)
1	1	Used in I^{2}C bus mode (9-clock wait). Generates interrupt service request at rising edge of 9th SCL clock cycle. (In the case of master device, makes SCL output low to enter wait state after 9 clock pulses are output. In the case of slave device, makes SCL output low to request wait state after 9 clock pulses are input.)

16.4 Serial Interface Channel 0 Operations

The following four operating modes are available for serial interface channel 0.

- Operation stop mode
- 3-wire serial I/O mode
- 2-wire serial I/O mode
- I^{2}C (Inter IC) bus mode

16.4.1 Operation stop mode

Serial transfer is not carried out in the operation stop mode. Thus, power consumption can be reduced. Serial I/O shift register 0 (SIO0) does not carry out shift operation either and thus it can be used as ordinary 8-bit register.

In the operation stop mode, the P25/SI0/SB0/SDA0, P26/SO0/SB1/SDA1 and P27/$\overline{\text{SCK0}}$/SCL pins can be used as ordinary I/O ports.

(1) Register setting

The operation stop mode is set with serial operating mode register 0 (CSIM0).

CSIM0 is set with a 1-bit or 8-bit memory manipulation instruction.

$\overline{\text{RESET}}$ input clears CSIM0 to 00H.

Symbol	⑦	⑥	⑤	4	3	2	1	0	Address	After Reset	R/W
CSIM0	CSIE0	COI	WUP	CSIM04	CSIM03	CSIM02	CSIM01	CSIM00	FF60H	00H	R/W

R/W	CSIE0	Serial Interface Channel 0 Operation Control
	0	Operation stopped
	1	Operation enabled

16.4.2 3-wire serial I/O mode operation

The 3-wire serial I/O mode is valid for connection of peripheral I/O units and display controllers which incorporate a conventional synchronous clocked serial interface as is the case with the 75X/XL, 78K, and 17K Series.

Communication is carried out with three lines of serial clock ($\overline{\text{SCK0}}$), serial output (SO0), and serial input (SI0).

(1) Register setting

The 3-wire serial I/O mode is set with serial operating mode register 0 (CSIM0) and the serial bus interface control register (SBIC).

(a) Serial operating mode register 0 (CSIM0)

CSIM0 is set with a 1-bit or 8-bit memory manipulation instruction.

$\overline{\text{RESET}}$ input clears CSIM0 to 00H.

Symbol	⑦	⑥	⑤	4	3	2	1	0	Address	After Reset	R/W
CSIM0	CSIE0	COI	WUP	CSIM04	CSIM03	CSIM02	CSIM01	CSIM00	FF60H	00H	R/W[Note 1]

R/W

CSIE0	Serial Interface Channel 0 Operation Control
0	Operation stopped
1	Operation enabled

R/W

WUP	Wake-up Function Control[Note 2]
0	Interrupt request signal generation with each serial transfer in any mode
1	Interrupt request signal generation when the address received after detecting start condition (when CMDD = 1) matches the slave address register data in I²C bus mode

R/W

CSIM 04	CSIM 03	CSIM 02	PM25	P25	PM26	P26	PM27	P27	Operation Mode	Start Bit	SIO/SB0/SDA0 /P25 Pin Function	SO0/SB1/SDA1 /P26 Pin Function	$\overline{\text{SCK0}}$/SCL/P27 Pin Function
0	×	0	1[Note 3]	×[Note 3]	0	0	0	1	3-wire serial I/O mode	MSB	SI0[Note 3] (Input)	SO0 (CMOS output)	$\overline{\text{SCK0}}$ (CMOS I/O)
		1								LSB			
1	1	2-wire serial I/O mode (See **16.4.3 2-wire serial I/O mode operation**) or I²C bus mode (See **16.4.4 I²C bus mode operation**)											

R/W

CSIM01	CSIM00	Serial Interface Channel 0 Clock Selection
0	×	Input clock to $\overline{\text{SCK0}}$ pin from off-chip
1	0	8-bit timer register 2 (TM2) output
1	1	Clock specified with bits 0 to 3 of timer clock select register 3 (TCL3)

Notes 1. Bit 6 (COI) is a read-only bit.

2. Be sure to set WUP to 0 when the 3-wire serial I/O mode is selected.

3. Can be used as P25 (CMOS I/O) when used only for transmission.

Remark ×: don't care

PM××: Port mode register

P××: Port output latch

(b) Serial bus interface control register (SBIC)

SBIC is set with a 1-bit or 8-bit memory manipulation instruction.

$\overline{\text{RESET}}$ input clears SBIC to 00H.

Symbol	⑦	⑥	⑤	④	③	②	①	⓪	Address	After Reset	R/W
SBIC	BSYE	ACKD	ACKE	ACKT	CMDD	RELD	CMDT	RELT	FF61H	00H	R/W

R/W	CMDT	When CMDT = 1, SO latch is cleared to 0. After SO latch clearance, automatically cleared to 0. Also cleared to 0 when CSIE0 = 0.

R/W	RELT	When RELT = 1, SO latch is set to 1. After SO latch setting, automatically cleared to 0. Also cleared to 0 when CSIE0 = 0.

CSIE0: Bit 7 of serial operating mode register 0 (CSIM0)

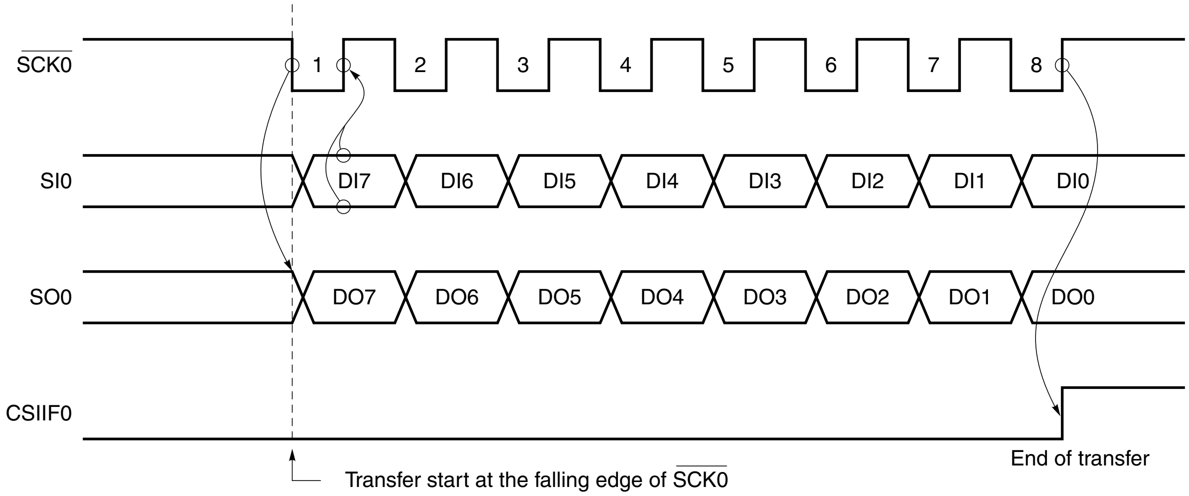
(2) Communication operation

The 3-wire serial I/O mode is used for data transmission/reception in 8-bit units. Each bit of data is transmitted or received in synchronization with the serial clock.

Shift operation of serial I/O shift register 0 (SIO0) is carried out at the falling edge of the serial clock ($\overline{\text{SCK0}}$). The transmitted data is held in the SO0 latch and is output from the SO0 pin. The received data input to the SI0 pin is latched in SIO0 at the rising edge of $\overline{\text{SCK0}}$.

Upon termination of 8-bit transfer, SIO0 operation stops automatically and the interrupt request flag (CSIIF0) is set.

Figure 16-7. 3-Wire Serial I/O Mode Timings



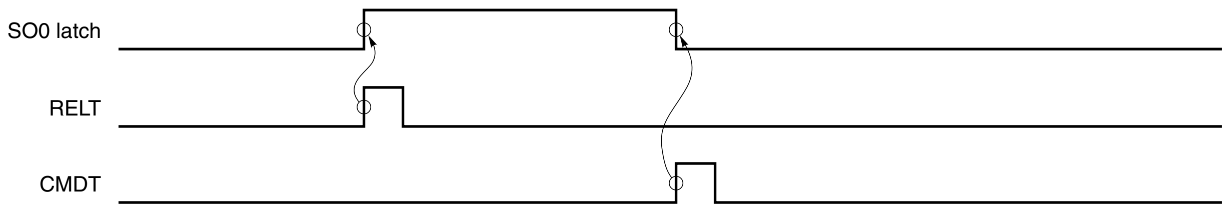
The SO0 pin is a CMOS output pin and outputs current SO0 latch statuses. Thus, the SO0 pin output status can be manipulated by setting bit 0 (RELT) and bit 1 (CMDT) of the serial bus interface control register (SBIC). However, do not carry out this manipulation during serial transfer.

Control the $\overline{\text{SCK0}}$ pin output level in the output mode (internal system clock mode) by manipulating the P27 output latch (refer to **16.4.7 $\overline{\text{SCK0}}$/SCL/P27 pin output manipulation**).

(3) Other signals

Figure 16-8 shows RELT and CMDT operations.

Figure 16-8. RELT and CMDT Operations



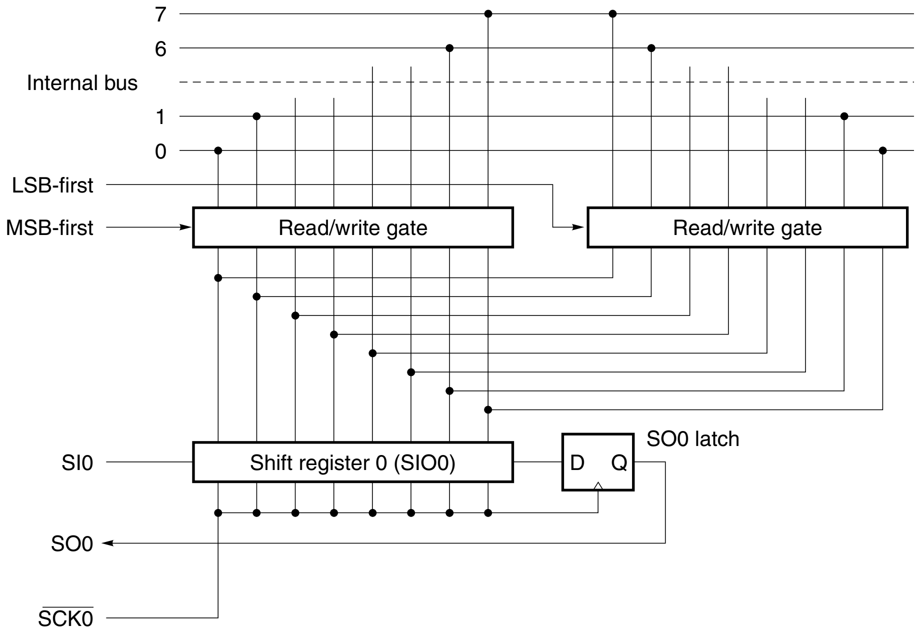
(4) MSB/LSB switching as the start bit

The 3-wire serial I/O mode enables to select transfer to start from MSB or LSB.

Figure 16-9 shows the configuration of serial I/O shift register 0 (SIO0) and internal bus. As shown in the figure, MSB/LSB can be read/written in reverse form.

MSB/LSB switching as the start bit can be specified with bit 2 (CSIM02) of serial operating mode register 0 (CSIM0).

Figure 16-9. Circuit of Switching in Transfer Bit Order



Start bit switching is realized by switching the bit order for data write to SIO0. The SIO0 shift order remains unchanged.

Thus, switching between MSB-first and LSB-first must be performed before writing data to the shift register.

(5) Transfer start

Serial transfer is started by setting transfer data to serial I/O shift register 0 (SIO0) when the following two conditions are satisfied.

- Serial interface channel 0 operation control bit (CSIE0) = 1
- Internal serial clock is stopped or $\overline{\text{SCK0}}$ is at high level after 8-bit serial transfer.

Caution If CSIE0 is set to "1" after data write to SIO0, transfer does not start.

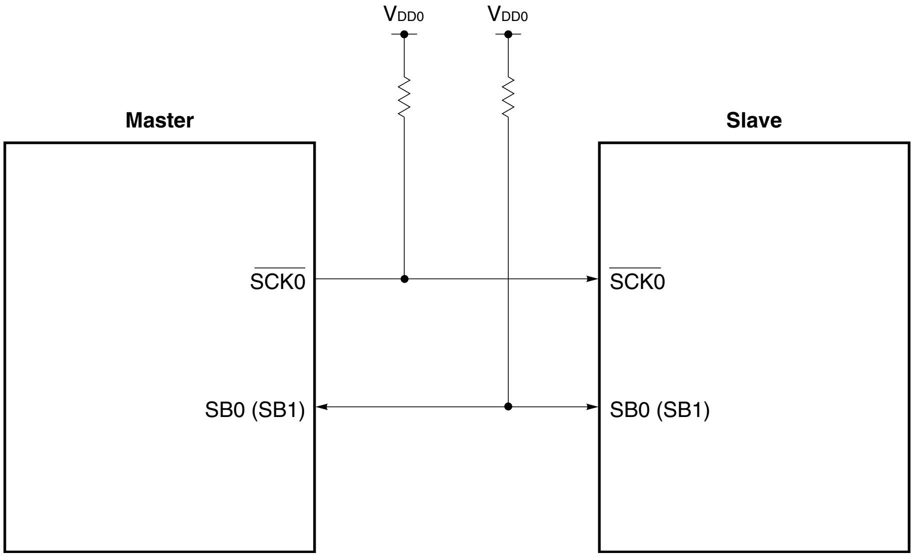
Upon termination of 8-bit transfer, serial transfer automatically stops and the interrupt request flag (CSIIF0) is set.

16.4.3 2-wire serial I/O mode operation

The 2-wire serial I/O mode can cope with any communication format by program.

Communication is basically carried out with two lines of serial clock ($\overline{\text{SCK0}}$) and serial data input/output (SB0 or SB1).

Figure 16-10. Serial Bus Configuration Example Using 2-Wire Serial I/O Mode



(1) Register setting

The 2-wire serial I/O mode is set with serial operating mode register 0 (CSIM0), the serial bus interface control register (SBIC), and the interrupt timing specify register (SINT).

(a) Serial operating mode register 0 (CSIM0)

CSIM0 is set with a 1-bit or 8-bit memory manipulation instruction.

$\overline{\text{RESET}}$ input clears CSIM0 to 00H.

Symbol	⑦	⑥	⑤	4	3	2	1	0	Address	After Reset	R/W
CSIM0	CSIE0	COI	WUP	CSIM04	CSIM03	CSIM02	CSIM01	CSIM00	FF60H	00H	R/W^Note 1^

R/W	CSIE0	Serial Interface Channel 0 Operation Control
	0	Operation stopped
	1	Operation enabled

R	COI	Slave Address Comparison Result Flag^Note 2^
	0	Slave address register not equal to serial I/O shift register 0 data
	1	Slave address register equal to serial I/O shift register 0 data

R/W	WUP	Wake-up Function Control^Note 3^
	0	Interrupt request signal generation with each serial transfer in any mode
	1	Interrupt request signal generation when the address received after detecting start condition (when CMDD = 1) matches the slave address register data in I^{2}C bus mode

R/W	CSIM 04	CSIM 03	CSIM 02	PM25	P25	PM26	P26	PM27	P27	Operation Mode	Start Bit	SIO/SB0/SDA0 /P25 Pin Function	SO0/SB1/SDA1 /P26 Pin Function	$\overline{\text{SCK0}}$/SCL/P27 Pin Function
	0	×	3-wire serial I/O mode (See **16.4.2 3-wire serial I/O mode operation**)											
	1	1	0	Note 4 ×	Note 4 ×	0	0	0	1	2-wire serial I/O mode or I^{2}C bus mode	MSB	P25 (CMOS I/O)	SB1/SDA1 (N-ch open-drain I/O)	$\overline{\text{SCK0}}$/SCL (N-ch open-drain I/O)
			1	0	0	Note 4 ×	Note 4 ×	0	1			SB0/SDA0 (N-ch open-drain I/O)	P26 (CMOS I/O)	

R/W	CSIM01	CSIM00	Serial Interface Channel 0 Clock Selection
	0	×	Input clock to $\overline{\text{SCK0}}$ pin from off-chip
	1	0	8-bit timer register 2 (TM2) output
	1	1	Clock specified with bits 0 to 3 of timer clock select register 3 (TCL3)

Notes 1. Bit 6 (COI) is a read-only bit.

2. When CSIE0 = 0, COI becomes 0.

3. Be sure to set WUP to 0 when the 2-wire serial I/O mode is selected.

4. Can be used freely as port function.

Remark ×: don't care

PM××: Port mode register

P××: Port output latch

(b) Serial bus interface control register (SBIC)

SBIC is set with a 1-bit or 8-bit memory manipulation instruction.

$\overline{\text{RESET}}$ input clears SBIC to 00H.

Symbol	⑦	⑥	⑤	④	③	②	①	⓪	Address	After Reset	R/W
SBIC	BSYE	ACKD	ACKE	ACKT	CMDD	RELD	CMDT	RELT	FF61H	00H	R/W

R/W		
	CMDT	When CMDT = 1, SO latch is cleared to 0. After SO latch clearance, automatically cleared to 0. Also cleared to 0 when CSIE0 = 0.

R/W		
	RELT	When RELT = 1, SO latch is set to 1. After SO latch setting, automatically cleared to 0. Also cleared to 0 when CSIE0 = 0.

(c) Interrupt timing specify register (SINT)

SINT is set with a 1-bit or 8-bit memory manipulation instruction.

$\overline{\text{RESET}}$ input clears SINT to 00H.

Symbol	7	⑥	⑤	④	③	②	1	0	Address	After Reset	R/W
SINT	0	CLD	SIC	SVAM	CLC	WREL	WAT1	WAT0	FF63H	00H	R/W[Note 1]

R	CLD	$\overline{\text{SCK0}}$ Pin Level[Note 2]
	0	Low level
	1	High level

R/W	SIC	INTCSI0 Interrupt Cause Selection
	0	CSIIF0 is set to 1 upon termination of serial interface channel 0 transfer
	1	CSIIF0 is set to 1 upon stop condition detection or termination of serial interface channel 0 transfer

Notes 1. Bit 6 (CLD) is a read-only bit.

2. When CSIE0 = 0, CLD becomes 0.

Caution Be sure to set bit 0 to bit 3 to 0 when 2-wire serial I/O mode is used.

Remark CSIIF0: Interrupt request flag corresponding to INTCSI0

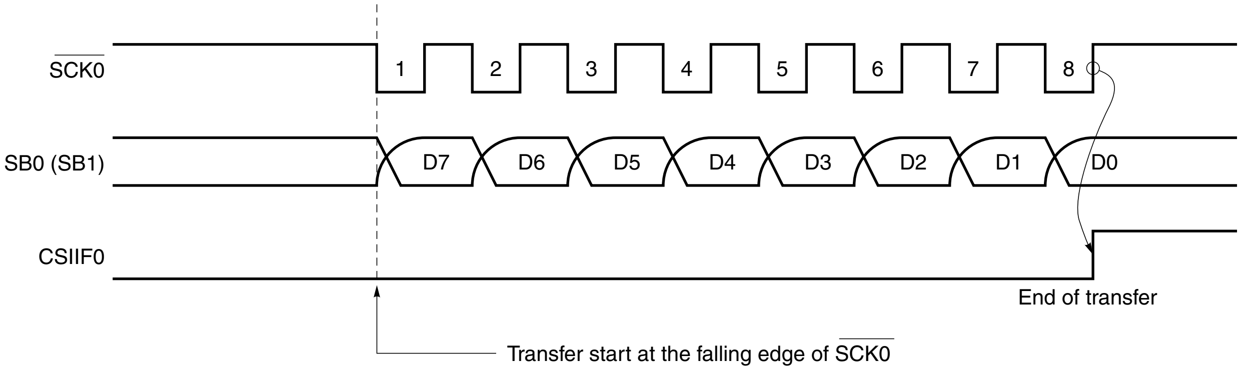
(2) Communication operation

The 2-wire serial I/O mode is used for data transmission/reception in 8-bit units. Each bit of data is transmitted or received in synchronization with the serial clock.

Shift operation of serial I/O shift register 0 (SIO0) is carried out in synchronization with the falling edge of the serial clock ($\overline{\text{SCK0}}$). The transmit data is held in the SO0 latch and is output from the SB0/SDA0/P25 (or SB1/SDA1/P26) pin on an MSB-first basis. The receive data input from the SB0 (or SB1) pin is latched into the shift register at the rising edge of $\overline{\text{SCK0}}$.

Upon termination of 8-bit transfer, the shift register operation stops automatically and the interrupt request flag (CSIIF0) is set.

Figure 16-11. 2-Wire Serial I/O Mode Timings



The SB0 (or SB1) pin specified for the serial data bus is an N-ch open-drain I/O and thus it must be externally connected to a pull-up resistor. Because the N-ch transistor output pin must go into a high-impedance state for data reception, write FFH to SIO0 in advance.

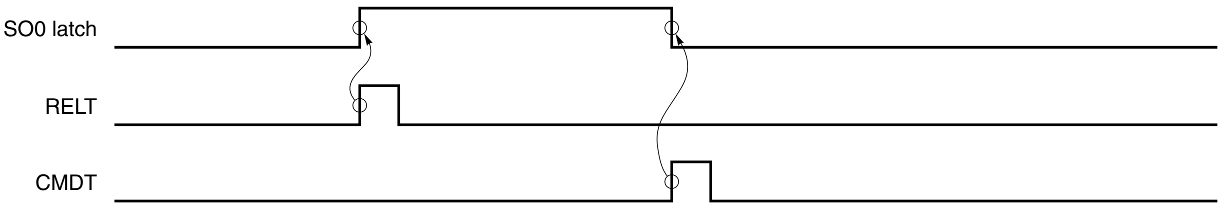
The SB0 (or SB1) pin generates the SO0 latch status and thus the SB0 (or SB1) pin output status can be manipulated by setting the RELT and CMDT bits. However, do not carry out this manipulation during serial transfer.

Control the $\overline{\text{SCK0}}$ pin output level in the output mode (internal system clock mode) by manipulating the P27 output latch (refer to **16.4.7 $\overline{\text{SCK0}}$/SCL/P27 pin output manipulation**).

(3) Other signals

Figure 16-12 shows RELT and CMDT operations.

Figure 16-12. RELT and CMDT Operations



(4) Transfer start

Serial transfer is started by setting transfer data to serial I/O shift register 0 (SIO0) when the following two conditions are satisfied.

- Serial interface channel 0 operation control bit (CSIE0) = 1
- Internal serial clock is stopped or $\overline{\text{SCK0}}$ is at high level after 8-bit serial transfer

Cautions 1. If CSIE0 is set to "1" after data write to SIO0, transfer does not start.

2. Because the N-ch transistor output pin must go into a high-impedance state for data reception, write FFH to SIO0 in advance.

Upon termination of 8-bit transfer, serial transfer automatically stops and the interrupt request flag (CSIIF0) is set.

(5) Error detection

In the 2-wire serial I/O mode, the serial bus SB0 (SB1) status being transmitted is fetched into the destination device, that is, serial I/O shift register 0 (SIO0). Thus, transmit error can be detected in the following way.

(a) Method of comparing SIO0 data before transmission to that after transmission

In this case, if two data differ from each other, a transmit error is judged to have occurred.

(b) Method of using the slave address register (SVA)

Transmit data is set to both SIO0 and SVA and is transmitted. After termination of transmission, COI bit (match signal coming from the address comparator) of serial operating mode register 0 (CSIM0) is tested. If "1", normal transmission is judged to have been carried out. If "0", a transmit error is judged to have occurred.

16.4.4 I^{2}C bus mode operation

The I^{2}C bus mode is provided for when communication operations are performed between a single master device and multiple slave devices. This mode configures a serial bus that includes only a single master device, and is based on the clocked serial I/O format with the addition of bus configuration functions, which allows the master device to communicate with a number of (slave) devices using only two lines: serial clock (SCL) line and serial data bus (SDA0 or SDA1) line. Consequently, when the user plans to configure a serial bus which includes multiple microcontrollers and peripheral devices, using this configuration results in reduction of the required number of port pins and on-board wires.

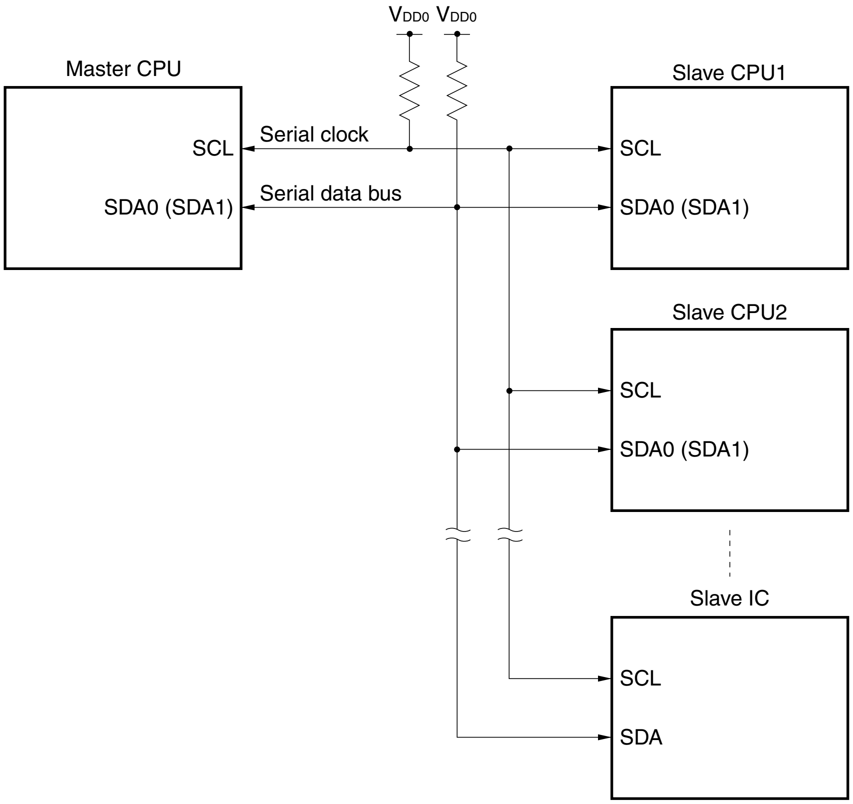
In the I^{2}C bus specification, the master sends start condition, data, and stop condition signals to slave devices through the serial data bus, while slave devices automatically detect and distinguish the type of signals due to the signal detection function incorporated as hardware. This simplifies I^{2}C bus control sections in the application program.

An example of a serial bus configuration is shown in Figure 16-13. This system below is composed of CPUs and peripheral ICs having serial interface hardware that complies with the I^{2}C bus specification.

Note that pull-up resistors are required to connect to both serial clock line and serial data bus line, because open-drain buffers are used for the serial clock pin (SCL) and the serial data bus pin (SDA0 or SDA1) on the I^{2}C bus.

The signals used in the I^{2}C bus mode are described in Table 16-4.

Figure 16-13. Example of Serial Bus Configuration Using I^{2}C Bus



(1) I^{2}C bus mode functions

In the I^{2}C bus mode, the following functions are available.

(a) Automatic identification of serial data

Slave devices automatically detect and identifies start condition, data, and stop condition signals sent in series through the serial data bus.

(b) Chip selection by specifying device addresses

The master device can select a specific slave device connected to the I^{2}C bus and communicate with it by sending in advance the address data corresponding to the destination device.

(c) Wake-up function

When address data is sent from the master device, slave devices compare it with the value registered in their internal slave address registers. If the values in one of the slave devices match, the slave device internally generates an interrupt request signal to terminate the current processing and communicates with the master device (The interrupt request is also generated when the stop condition is detected). Therefore, CPUs other than the selected slave device on the I^{2}C bus can perform independent operations during the serial communication.

(d) Acknowledge signal ($\overline{\text{ACK}}$) control function

The master device and a slave device send and receive acknowledge signals to confirm that the serial communication has been executed normally.

(e) Wait signal ($\overline{\text{WAIT}}$) control function

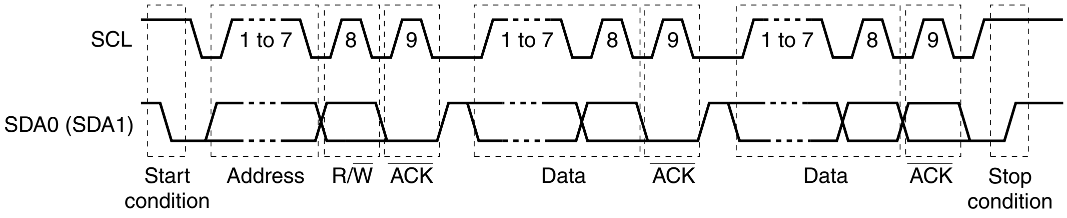
When a slave device is preparing for data transmission or reception and requires more waiting time, the slave device outputs a wait signal on the bus to inform the master device of the wait status.

(2) I^{2}C bus definition

This section describes the format of serial data communications and functions of the signals used in the I^{2}C bus mode.

First, the transfer timings of the start condition, data, and stop condition signals, which are output onto the signal data bus of the I^{2}C bus, are shown in Figure 16-14.

Figure 16-14. I^{2}C Bus Serial Data Transfer Timing



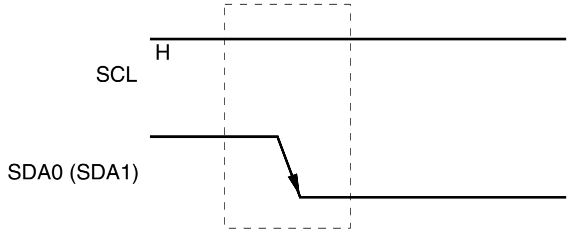
The start condition, slave address, and stop condition signals are output by the master. The acknowledge signal ($\overline{\text{ACK}}$) is output by either the master or the slave device (normally by the device which has received the 8-bit data that was sent). A serial clock (SCL) is continuously supplied from the master device.

(a) Start condition

When the SDA0 (SDA1) pin level is changed from high to low while the SCL pin is high, this transition is recognized as the start condition signal. This start condition signal, which is created using the SCL and SDA0 (or SDA1) pins, is output from the master device to slave devices to initiate a serial transfer. See **16.4.5 Cautions on use of I²C bus mode** for details of the start condition output.

The start condition signal is detected by hardware incorporated in slave devices.

Figure 16-15. Start Condition



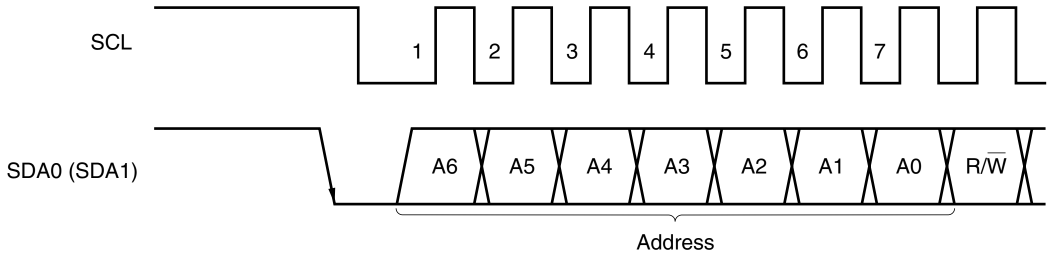
(b) Address

The 7 bits following the start condition signal are defined as an address.

The 7-bit address data is output by the master device to specify a specific slave from among those connected to the bus line. Each slave device on the bus line must therefore have a different address.

Therefore, after a slave device detects the start condition, it compares the 7-bit address data received and the data of the slave address register (SVA). After the comparison, only the slave device in which the data are a match becomes the communication partner, and subsequently performs communication with the master device until the master device sends a start condition or stop condition signal.

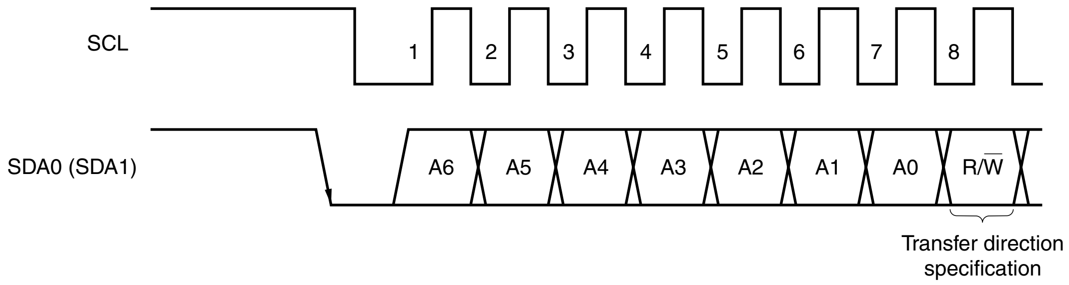
Figure 16-16. Address



(c) Transfer direction specification

The 1 bit that follows the 7-bit address data will be sent from the master device, and it is defined as the transfer direction specification bit. If this bit is 0, it is the master device which will send data to the slave. If it is 1, it is the slave device which will send data to the master.

Figure 16-17. Transfer Direction Specification

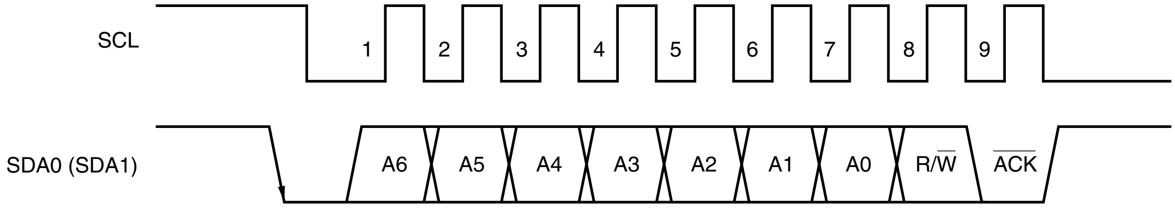


(d) Acknowledge signal ($\overline{\text{ACK}}$)

The acknowledge signal indicates that the transferred serial data has definitely been received. This signal is used between the sending side and receiving side devices for confirmation of correct data transfer. In principle, the receiving side device returns an acknowledge signal to the sending device each time it receives 8-bit data. The only exception is when the receiving side is the master device and the 8-bit data is the last transfer data; the master device outputs no acknowledge signal in this case.

The sending side that has transferred 8-bit data waits for the acknowledge signal which will be sent from the receiving side. If the sending side device receives the acknowledge signal, which means a successful data transfer, it proceeds to the next processing. If this signal is not sent back from the slave device, this means that the data sent has not been received by the slave device, and therefore the master device outputs a stop condition signal to terminate subsequent transmissions.

Figure 16-18. Acknowledge Signal



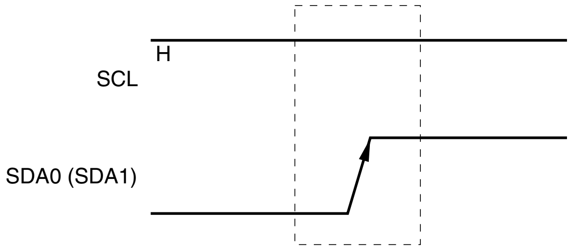
(e) Stop condition

If the SDA0 (SDA1) pin level changes from low to high while the SCL pin is high, this transition is defined as a stop condition signal.

The stop condition signal is output from the master to the slave device to terminate a serial transfer.

The stop condition signal is detected by hardware incorporated in the slave device.

Figure 16-19. Stop Condition

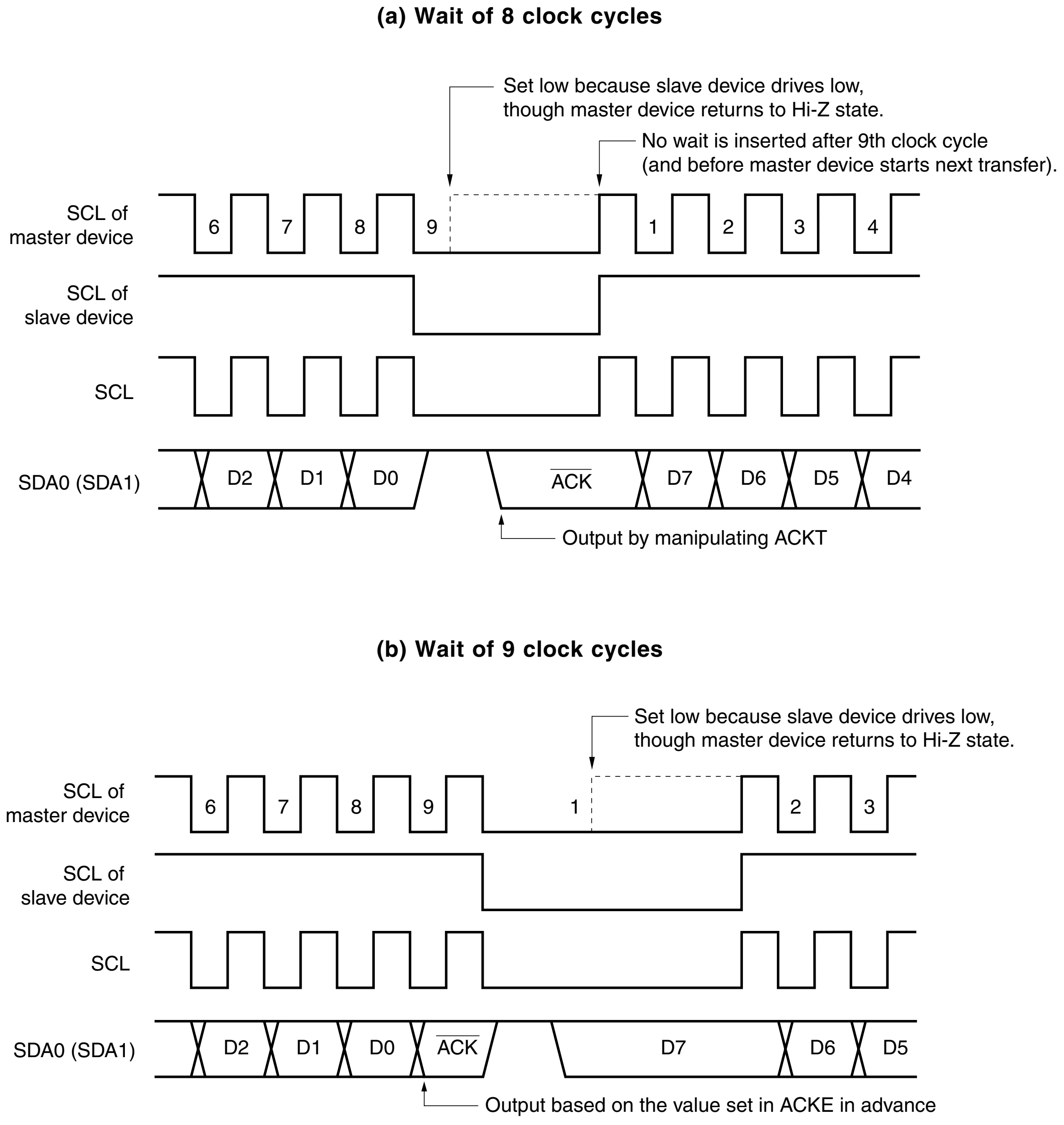


(f) Wait signal ($\overline{\text{WAIT}}$)

The wait signal is output by a slave device to inform the master device that the slave device is in wait state due to preparing for transmitting or receiving data.

During the wait state, the slave device continues to output the wait signal by keeping the SCL pin low to delay subsequent transfers. When the wait state is released, the master device can start the next transfer. For the releasing operation of slave devices, see **16.4.5 Cautions on use of I^{2}C bus mode**.

Figure 16-20. Wait Signal


(a) Wait of 8 clock cycles

(b) Wait of 9 clock cycles

Set low because slave device drives low,
though master device returns to Hi-Z state.
No wait is inserted after 9th clock cycle
(and before master device starts next transfer).
SCL of master device
6
7
8
9
1
2
3
4
SCL of slave device
SCL
SDA0 (SDA1)
D2
D1
D0
ACK
D7
D6
D5
D4
Output by manipulating ACKT
Set low because slave device drives low,
though master device returns to Hi-Z state.
SCL of master device
6
7
8
9
1
2
3
SCL of slave device
SCL
SDA0 (SDA1)
D2
D1
D0
ACK
D7
D6
D5
Output based on the value set in ACKE in advance

(3) Register setting

The I^{2}C bus mode setting is performed by serial operating mode register 0 (CSIM0), the serial bus interface control register (SBIC), and the interrupt timing specify register (SINT).

(a) Serial operating mode register 0 (CSIM0)

CSIM0 is set with a 1-bit or 8-bit memory manipulation instruction.

$\overline{\text{RESET}}$ input clears CSIM0 to 00H.

Symbol	⑦	⑥	⑤	4	3	2	1	0	Address	After Reset	R/W
CSIM0	CSIE0	COI	WUP	CSIM04	CSIM03	CSIM02	CSIM01	CSIM00	FF60H	00H	R/W[Note 1]

R/W

CSIE0	Serial Interface Channel 0 Operation Control
0	Operation stopped
1	Operation enabled

R

COI	Slave Address Comparison Result Flag[Note 2]
0	Slave address register not equal to serial I/O shift register 0 data
1	Slave address register equal to serial I/O shift register 0 data

R/W

WUP	Wake-up Function Control[Note 3]
0	Interrupt request signal generation with each serial transfer in any mode
1	Interrupt request signal generation when the address received after start condition detection (when CMDD = 1) matches the slave address register data in I^{2}C bus mode

R/W

CSIM 04	CSIM 03	CSIM 02	PM25	P25	PM26	P26	PM27	P27	Operation Mode	Start Bit	SI0/SB0/SDA0/ P25 Pin Function	SO0/SB1/SDA1/ P26 Pin Function	$\overline{\text{SCK0}}$/SCL/P27 Pin Function
0	×	3-wire serial I/O mode (See **16.4.2 3-wire serial I/O mode operation**)											
1	1	0	×[Note 4]	×[Note 4]	0	0	0	1	2-wire serial I/O or I^{2}C bus mode	MSB	P25 (CMOS I/O)	SB1/SDA1 (N-ch open-drain I/O)	$\overline{\text{SCK0}}$/SCL (N-ch open-drain I/O)
		1	0	0	×[Note 4]	×[Note 4]	0	1			SB0/SDA0 (N-ch open-drain I/O)	P26 (CMOS I/O)	

R/W

CSIM01	CSIM00	Serial Interface Channel 0 Clock Selection
0	×	Input clock to SCL pin from off-chip
1	0	8-bit timer register 2 (TM2) output[Note 5]
1	1	Clock specified with bits 0 to 3 of timer clock select register 3 (TCL3)

Notes 1. Bit 6 (COI) is a read-only bit.

2. When CSIE0 = 0, COI becomes 0.

3. Set bit 5 (SIC) of the interrupt timing specify register (SINT) to 1 when using the wake-up function (WUP = 1). Do not execute an instruction that writes to serial I/O shift register 0 (SIO0) while WUP = 1.

4. These pins can be used freely as port pins.

5. In the I^{2}C bus mode, the clock frequency becomes 1/16 of that output from TO2.

Remark ×: don't care

PM××: Port mode register

P××: Port output latch

(b) Serial bus interface control register (SBIC)

SBIC is set with a 1-bit or 8-bit memory manipulation instruction.
$\overline{\text{RESET}}$ input clears SBIC to 00H.

Symbol	⑦	⑥	⑤	④	③	②	①	⓪	Address	After Reset	R/W
SBIC	BSYE	ACKD	ACKE	ACKT	CMDD	RELD	CMDT	RELT	FF61H	00H	R/W[Note 1]

R/W

Note 2 BSYE	Control of N-ch Open-Drain Output for Transmission in I^{2}C Bus Mode[Note 3]
0	Output enabled (transmission)
1	Output disabled (reception)

R

<table>
<tr><th>ACKD</th><th>Acknowledge Detection</th></tr>
<tr><td>Clear Conditions (ACKD = 0)</td><td>Set Conditions (ACKD = 1)</td></tr>
<tr><td>• When transfer start instruction is executed
• When $\overline{\text{CSIE0}}$ = 0
• When $\overline{\text{RESET}}$ input is applied</td><td>• When acknowledge signal is detected at the rising edge of SCL clock after completion of transfer</td></tr>
</table>

R/W

ACKE	Acknowledge Signal Automatic Output Control[Note 4]
0	Disabled (with ACKT enabled). Used when receiving data in the 8-clock wait mode or when transmitting data[Note 5]
1	Enabled. After completion of transfer, acknowledge signal is output in synchronization with the 9th falling edge of SCL clock (automatically output when ACKE = 1). However, not automatically cleared to 0 after acknowledge signal output. Used for reception when the 9-clock wait mode is selected.

R/W

ACKT	SDA0 (SDA1) is set to low after the set instruction execution (ACKT = 1) before the next SCL falling edge. Used for generating an $\overline{\text{ACK}}$ signal by software if the 8-clock wait mode is selected. Cleared to 0 if CSIE0 = 0 when a transfer by the serial interface is started.

R

<table>
<tr><th>CMDD</th><th>Start Condition Detection</th></tr>
<tr><td>Clear Conditions (CMDD = 0)</td><td>Set Conditions (CMDD = 1)</td></tr>
<tr><td>• When transfer start instruction is executed
• When stop condition is detected
• When CSIE0 = 0
• When $\overline{\text{RESET}}$ input is applied</td><td>• When start condition is detected</td></tr>
</table>

(continued)

Notes 1. Bits 2, 3, and 6 (RELD, CMDD, and ACKD) are read-only bits.

2. The busy mode can be released by the start of a serial interface transfer or reception of an address signal. However, the BSYE flag is not cleared to 0.

3. When using the wake-up function, be sure to set BSYE to 1.

4. This setting must be performed prior to transfer start.

5. In the 8-clock wait mode, use ACKT for output of the acknowledge signal after normal data reception.

Remark CSIE0: Bit 7 of serial operating mode register 0 (CSIM0)

R

RELD	Stop Condition Detection
Clear Conditions (RELD = 0)	Set Conditions (RELD = 1)
• When transfer start instruction is executed • If SIO0 and SVA values do not match in address reception • When CSIE0 = 0 • When $\overline{\text{RESET}}$ input is applied	• When stop condition is detected

R/W

CMDT	Use for start condition output. When CMDT = 1, SO latch is cleared to 0. After clearing SO latch, automatically cleared to 0. Also cleared to 0 when CSIE0 = 0.

R/W

RELT	Used for stop condition output. When RELT = 1, SO latch is set to 1. After SO latch setting, automatically cleared to 0. Also cleared to 0 when CSIE0 = 0.

(c) Interrupt timing specify register (SINT)

SINT is set with a 1-bit or 8-bit memory manipulation instruction.

$\overline{\text{RESET}}$ input clears SINT to 00H.

Symbol	7	⑥	⑤	④	③	②	1	0	Address	After Reset	R/W
SINT	0	CLD	SIC	SVAM	CLC	WREL	WAT1	WAT0	FF63H	00H	R/W[Note 1]

R	CLD	SCL Pin Level[Note 2]
	0	Low level
	1	High level

R/W	SIC	INTCSI0 Interrupt Cause Selection[Note 3]
	0	CSIIF0 is set to 1 upon termination of serial interface channel 0 transfer
	1	CSIIF0 is set to 1 upon stop condition detection or termination of serial interface channel 0 transfer

R/W	SVAM	SVA Bit to Be Used as Slave Address
	0	Bits 0 to 7
	1	Bits 1 to 7

R/W	CLC	Clock Level Control
	0	Used in I^{2}C bus mode. Make output level of SCL pin low unless serial transfer is being performed.
	1	Used in I^{2}C bus mode. Make SCL pin enter high-impedance state unless serial transfer is being performed (except for clock line which is kept high). Used to enable master device to generate start condition and stop condition signals.

R/W	WREL	Wait Sate Cancellation Control
	0	Wait state has been cancelled.
	1	Cancels wait state. Automatically cleared to 0 when the state is cancelled. (Used to cancel wait state by means of WAT0 and WAT1.)

(continued)

Notes 1. Bit 6 (CLD) is a read-only bit.

2. When CSIE0 = 0, CLD becomes 0.

3. When using the wake-up function in I^{2}C bus mode, set SIC to 1.

Remark SVA: Slave address register

CSIIF0: Interrupt request flag corresponding to INTCSI0

CSIE0: Bit 7 of serial operating mode register 0 (CSIM0)

R/W

WAT1	WAT0	Wait and Interrupt Control[Note]
0	0	Generates interrupt service request at rising edge of 8th $\overline{\text{SCK0}}$ clock cycle (keeping clock output in high impedance).
0	1	Setting prohibited
1	0	Used in I^{2}C bus mode (8-clock wait). Generates interrupt service request at rising edge of 8th SCL clock cycle. (In the case of master device, makes SCL output low to enter wait state after 8 clock pulses are output. In the case of slave device, makes SCL output low to request wait state after 8 clock pulses are input.)
1	1	Used in I^{2}C bus mode (9-clock wait). Generates interrupt service request at rising edge of 9th SCL clock cycle. (In the case of master device, makes SCL output low to enter wait state after 9 clock pulses are output. In the case of slave device, makes SCL output low to request wait state after 9 clock pulses are input.)

Note When the I^{2}C bus mode is used, be sure to set 1 and 0, or 1 and 1 in WAT1 and WAT0, respectively.

(4) Various signals

A list of signals in the I^2C bus mode is given in Table 16-4.

Table 16-4. Signals in I^2C Bus Mode

Signal Name	Description	
Start condition	Definition:	SDA0 (SDA1) falling edge when SCL is high[Note 1]
	Function:	Indicates that serial communication starts and subsequent data are address data.
	Signalled by:	Master
	Signalled when:	CMDT is set.
	Affected flag(s):	CMDD (is set.)
Stop condition	Definition:	SDA0 (SDA1) rising edge when SCL is high[Note 1]
	Function:	Indicates end of serial transmission.
	Signalled by:	Master
	Signalled when:	RELT is set.
	Affected flag(s):	RELD (is set) and CMDD (is cleared)
Acknowledge signal ($\overline{\text{ACK}}$)	Definition:	Low level of SDA0(SDA1) pin during one SCL clock cycle after serial reception
	Function:	Indicates completion of reception of 1 byte.
	Signalled by:	Master or slave
	Signalled when:	ACKT is set with ACKE = 1.
	Affected flag(s):	ACKD (is set.)
Wait ($\overline{\text{WAIT}}$)	Definition:	Low-level signal output to SCL
	Function:	Indicates state in which serial reception is not possible.
	Signalled by:	Slave
	Signalled when:	WAT1, WAT0 = 1x.
	Affected flag(s):	None
Serial clock (SCL)	Definition:	Synchronization clock for output of various signals
	Function:	Serial communication synchronization signal.
	Signalled by:	Master
	Signalled when:	See **Note 2** below.
	Affected flag(s):	CSIIF0. Also see **Note 3** below.
Address (A6 to A0)	Definition:	7-bit data synchronized with SCL immediately after start condition signal
	Function:	Indicates address value for specification of slave on serial bus.
	Signalled by:	Master
	Signalled when:	See **Note 2** below.
	Affected flag(s):	CSIIF0. Also see **Note 3** below.
Transfer direction (R/$\overline{\text{W}}$)	Definition:	1-bit data output in synchronization with SCL after address output
	Function:	Indicates whether data transmission or reception is to be performed.
	Signalled by:	Master
	Signalled when:	See **Note 2** below.
	Affected flag(s):	CSIIF0. Also see **Note 3** below.
Data (D7 to D0)	Definition:	8-bit data synchronized with SCL, not immediately after start condition
	Function:	Contains data actually to be sent.
	Signalled by:	Master or slave
	Signalled when:	See **Note 2** below.
	Affected flag(s):	CSIIF0. Also see **Note 3** below.

Notes 1. The level of the serial clock can be controlled by CLC of the interrupt timing specify register (SINT).

2. Execution of instruction to write data to SIO0 when CSIE0 = 1 (serial transfer start directive). In the wait state, the serial transfer operation will be started after the wait state is released.

3. If the 8-clock wait is selected when WUP = 0, CSIIF0 is set at the rising edge of the 8th clock cycle of SCL. If the 9-clock wait is selected when WUP = 0, CSIIF0 is set at the rising edge of the 9th clock cycle of SCL. If WUP = 1, CSIIF0 is set when an address is received and the address matches the slave address register (SVA) value, or when the stop condition is detected.

(5) Pin configurations

The configurations of the serial clock pin (SCL) and the serial data bus pins (SDA0, SDA1) are shown below.

(a) SCL

Serial clock I/O pin.

<1> Master ..... N-ch open-drain output

<2> Slave .... Schmitt input

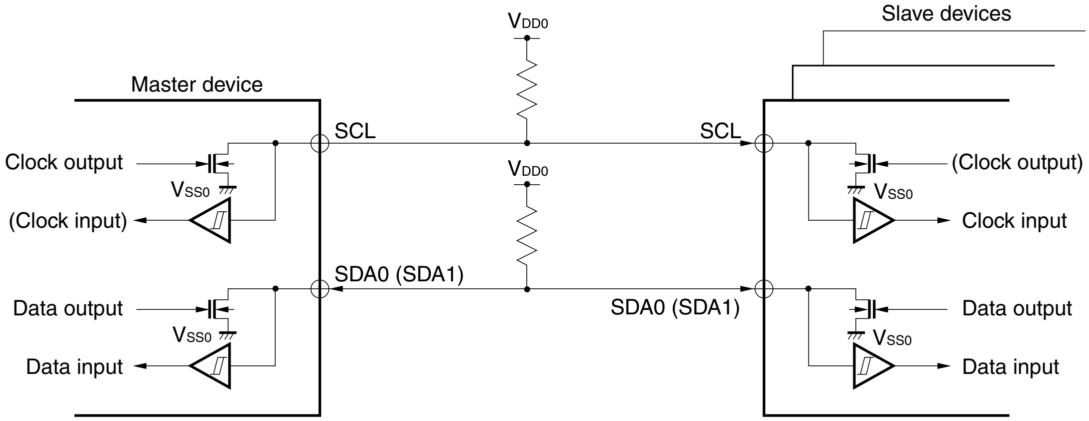
(b) SDA0 (SDA1)

Serial data I/O dual-function pin.

Uses N-ch open-drain output and Schmitt-input buffers for both master and slave devices.

Note that pull-up resistors are required to connect to both serial clock line and serial data bus line, because open-drain buffers are used for the serial clock pin (SCL) and the serial data bus pin (SDA0 or SDA1) on the I^{2}C bus.

Figure 16-21. Pin Configuration



Caution When data is received, the N-ch open-drain output pin must go into a high-impedance state. Therefore, set bit 7 (BSYE) of the serial bus interface control register (SBIC) to 1, and write FFH to serial I/O shift register 0 (SIO0).

However, do not write FFH to SIO0 before reception when the wake-up function is used (when bit 5 (WUP) of serial operating mode register 0 (CSIM0) is set). Even if FFH is not written to SIO0, the N-ch open-drain output pin always goes into a high-impedance state.

(6) Address match detection method

In the I^{2}C bus mode, the master can select a specific slave device by sending slave address data.

Address match detection is performed automatically by the slave device hardware. A slave device has a slave address register (SVA), and compares its contents and the slave address sent from the master device. If they match and the wake-up function specify (WUP) bit is then 1, interrupt request flag (CSIIF0) is set (CSIIF0 is also set when the stop condition is detected).

Set SIC to 1 when using the wake-up function.

Caution Status detection of slave selection/non-selection is performed by a match detection of reception data (address) after the start condition. This address match signal interrupt (INTCSI0) generated during WUP = 1 is used as a match signal. Thus, perform the slave selection/non-selection detection during WUP = 1.

(7) Error detection

In the I^{2}C bus mode, transmission error detection can be performed by the following methods because the serial bus SDA0 (SDA1) status during transmission is also taken into serial I/O shift register 0 (SIO0) of the transmitting device.

(a) Comparison of SIO0 data before and after transmission

In this case, a transmission error is judged to have occurred if the two data values are different.

(b) Using the slave address register (SVA)

Transmit data is set in SIO0 and SVA before transmission is performed. After transmission, the COI bit (match signal from the address comparator) of serial operating mode register 0 (CSIM0) is tested: "1" indicates normal transmission, and "0" indicates a transmission error.

(8) Communication operation

In the I^{2}C bus mode, the master selects the slave device to be communicated with from among multiple devices by outputting address data onto the serial bus.

After the slave address data, the master sends the R/W bit which indicates the data transfer direction, and starts serial communication with the selected slave device.

Data communication timing charts are shown in Figures 16-22 and 16-23.

In the transmitting device, serial I/O shift register 0 (SIO0) shifts transmission data to the SO latch in synchronization with the falling edge of the serial clock (SCL), the SO0 latch outputs the data on an MSB-first basis from the SDA0 or SDA1 pin to the receiving device.

In the receiving device, the data input from the SDA0 or SDA1 pin is taken into the SIO0 in synchronization with the rising edge of SCL.

Figure 16-22. Data Transmission from Master to Slave (Both Master and Slave Selected 9-Clock Wait) (1/3)

(a) Start condition to address

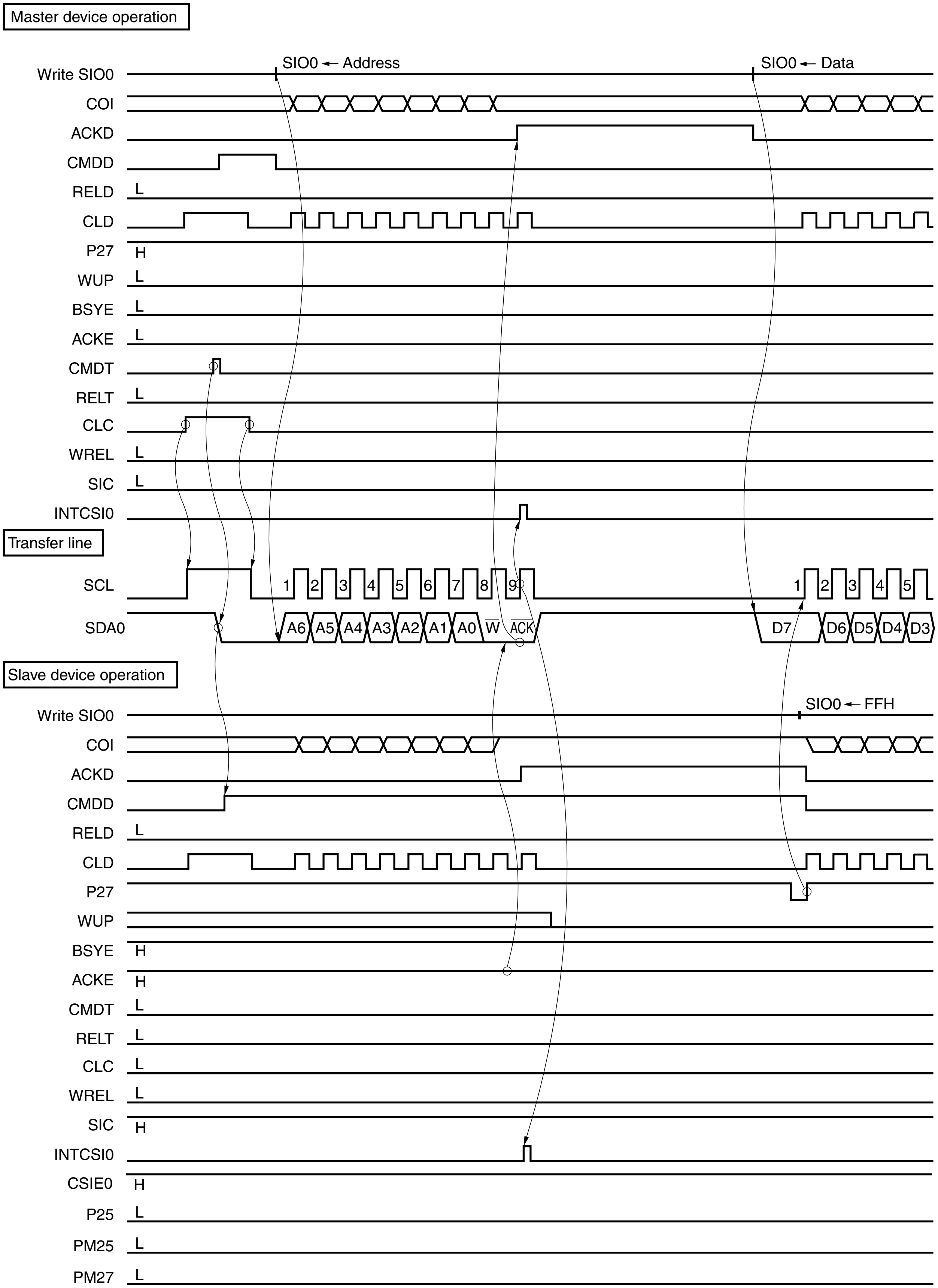


Figure 16-22. Data Transmission from Master to Slave (Both Master and Slave Selected 9-Clock Wait) (2/3)

(b) Data

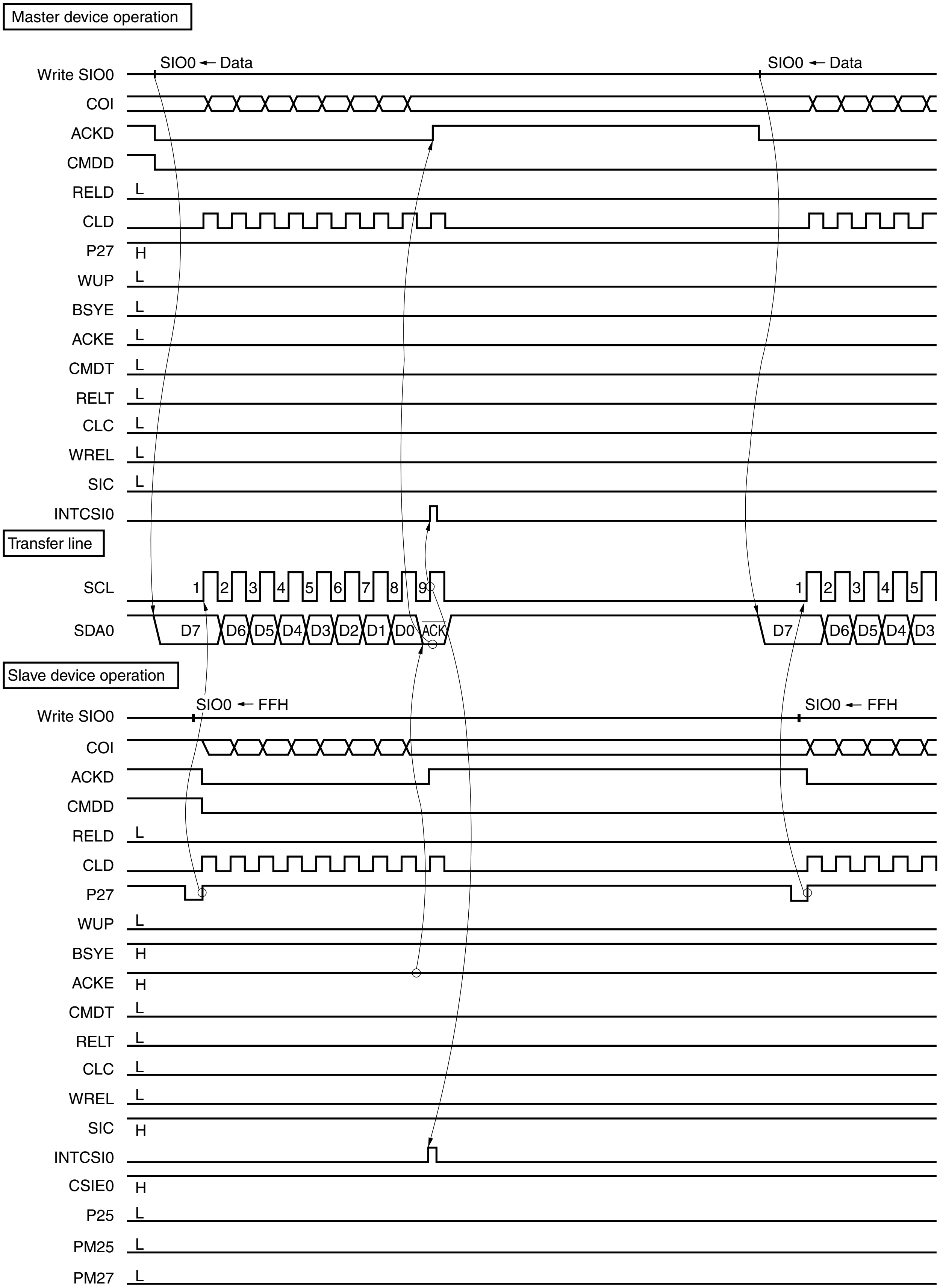


Figure 16-22. Data Transmission from Master to Slave (Both Master and Slave Selected 9-Clock Wait) (3/3)

(c) Stop condition

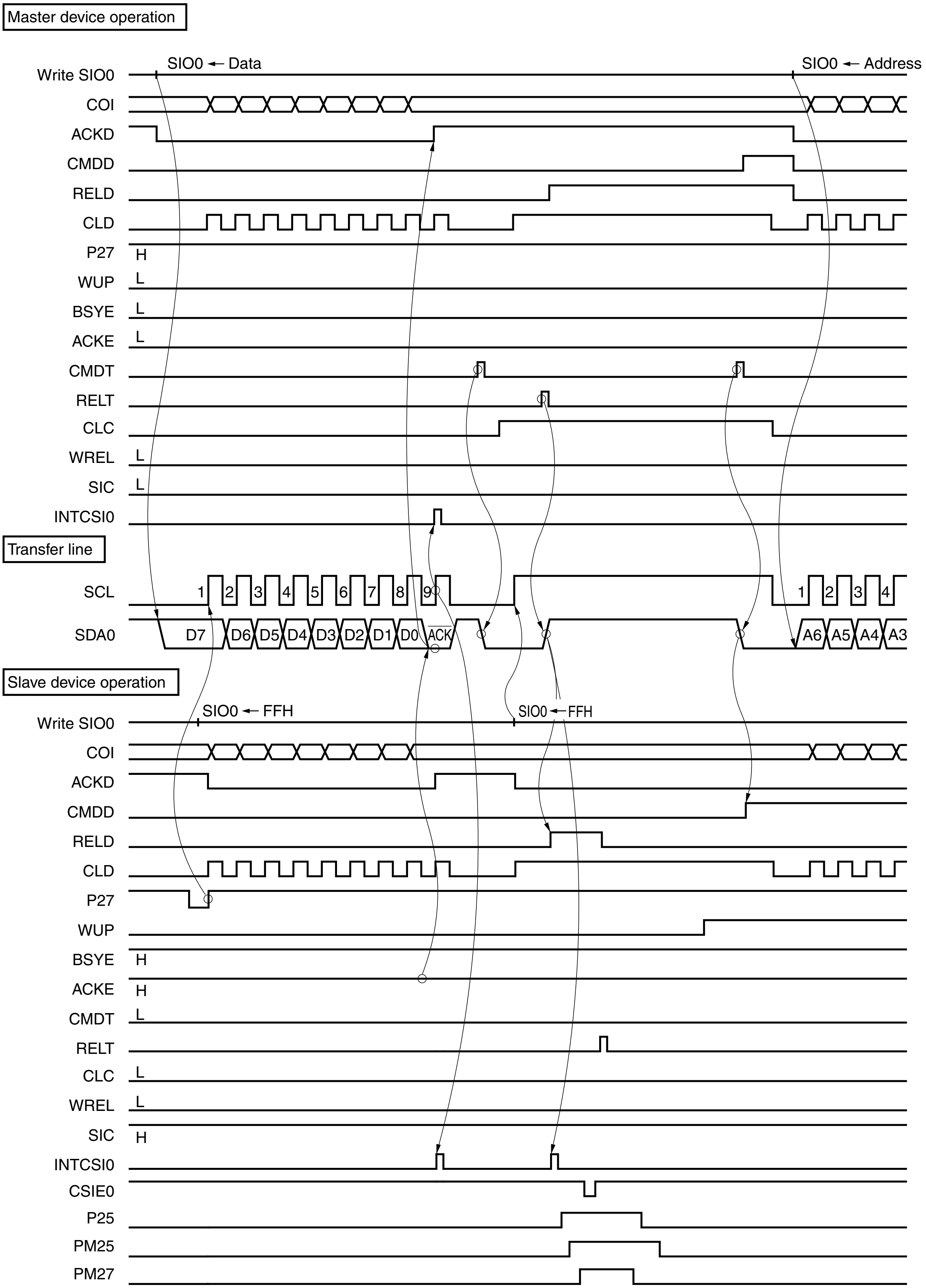


Figure 16-23. Data Transmission from Slave to Master (Both Master and Slave Selected 9-Clock Wait) (1/3)

(a) Start condition to address

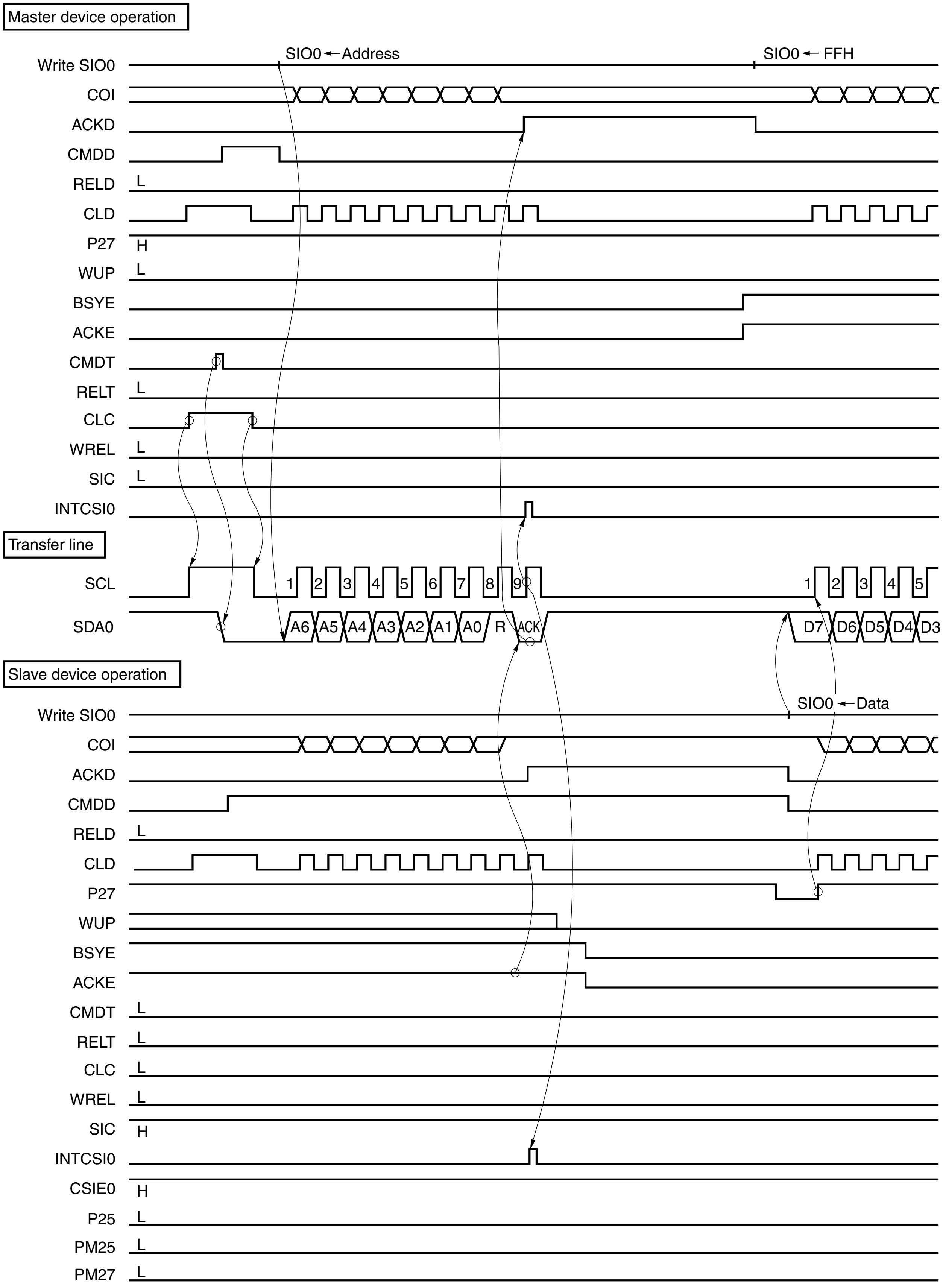


Figure 16-23. Data Transmission from Slave to Master (Both Master and Slave Selected 9-Clock Wait) (2/3)

(b) Data

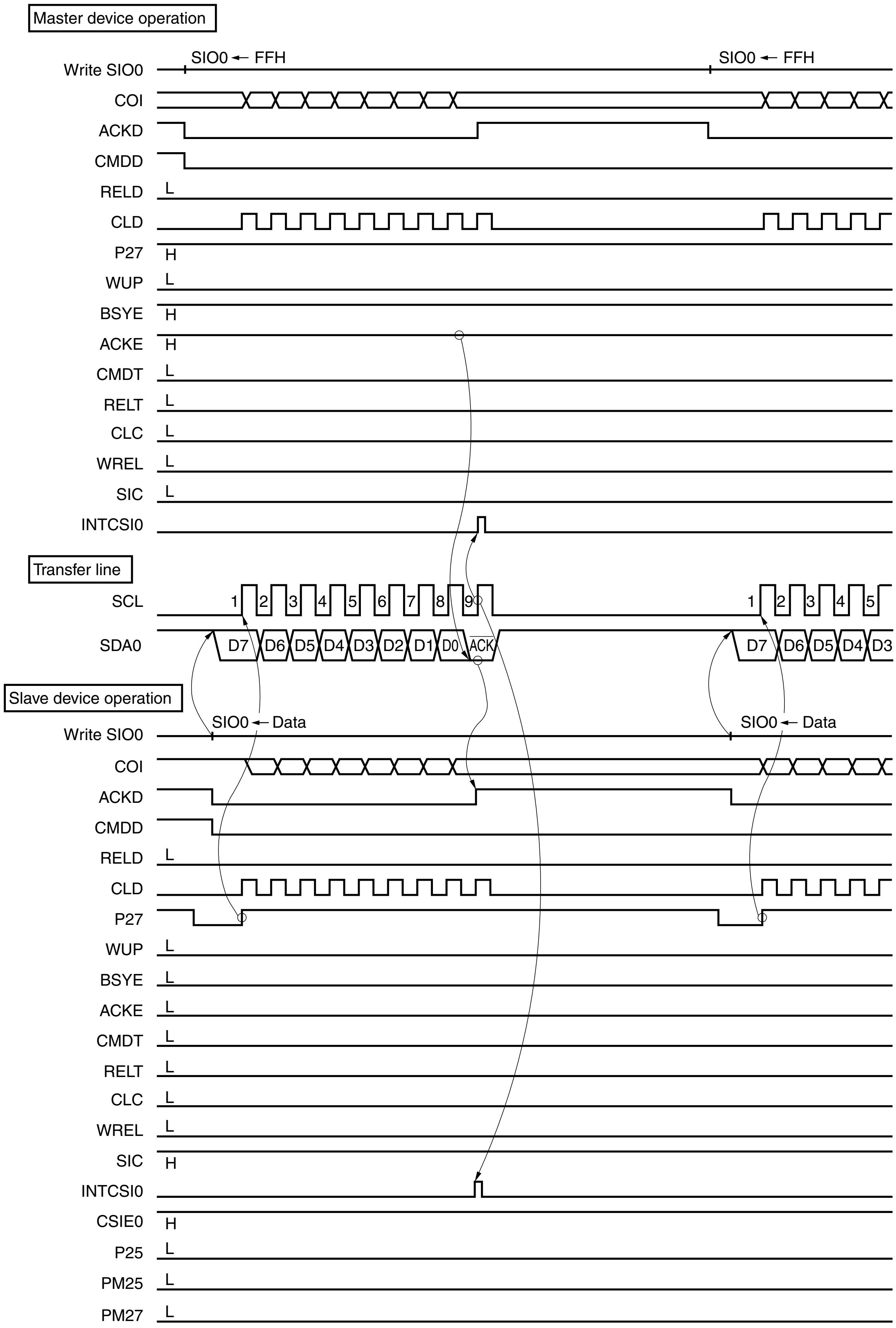
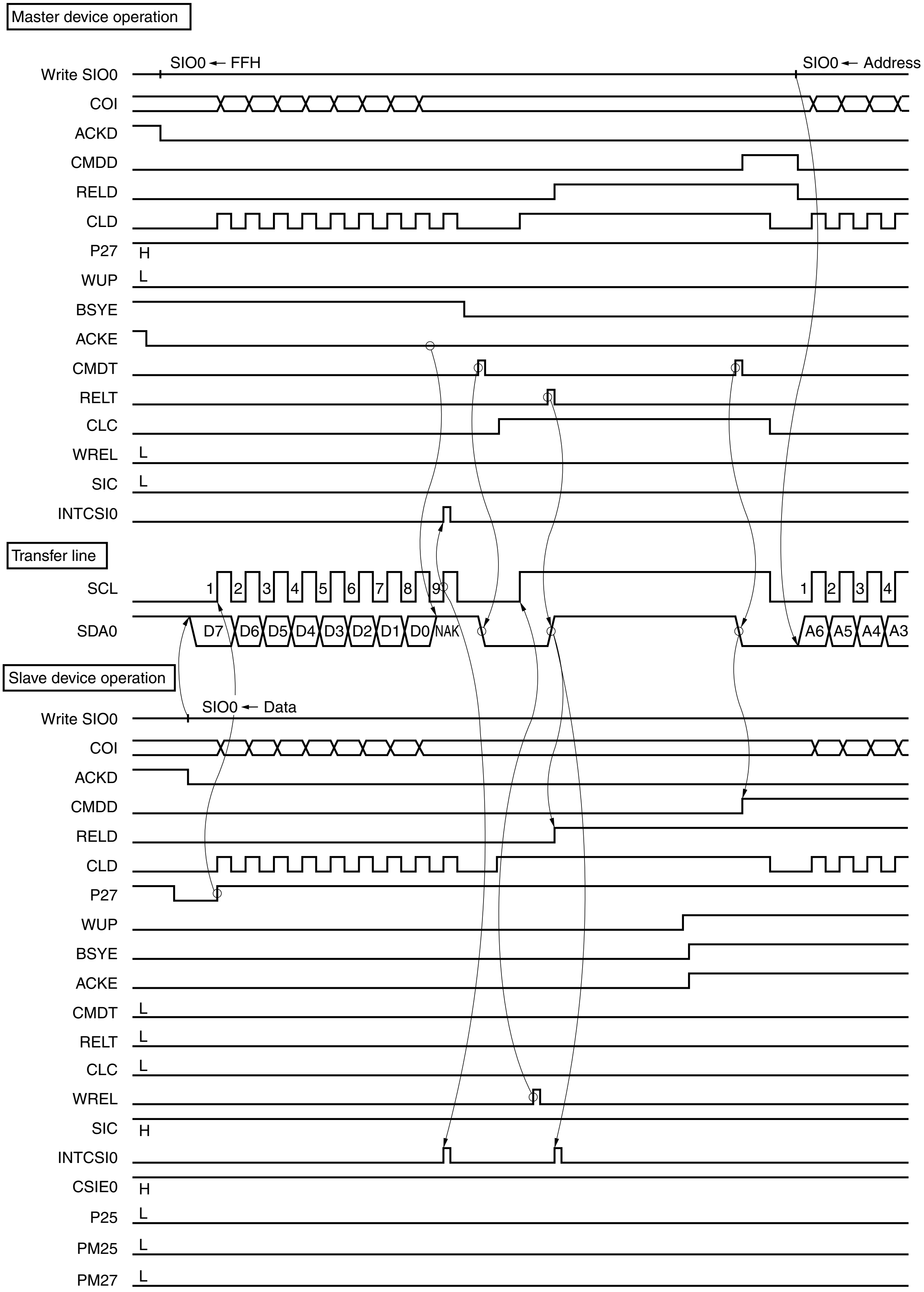


Figure 16-23. Data Transmission from Slave to Master (Both Master and Slave Selected 9-Clock Wait) (3/3)

(c) Stop condition



(9) Start of transfer

A serial transfer is started by setting transfer data in serial I/O shift register 0 (SIO0) if the following two conditions have been satisfied:

- Serial interface channel 0 operation control bit (CSIE0) = 1
- After an 8-bit serial transfer, the internal serial clock is stopped or SCL is low.

Cautions 1. Be sure to set CSIE0 to 1 before writing data in SIO0. Setting CSIE0 to 1 after writing data in SIO0 does not initiate transfer operation.

2. When data is received, the N-ch open-drain output pin must go into a high-impedance state. Therefore, set BSYE of the serial bus interface control register (SBIC) to 1, and write FFH to SIO0.

However, do not write FFH to SIO0 before reception when the wake-up function is used (when bit 5 (WUP) of serial operating mode register 0 (CSIM0) is set). Even if FFH is not written to SIO0, the N-ch open-drain output pin always goes into a high-impedance state.

3. If data is written to SIO0 while the slave is in the wait state, that data is held. Transfer is started when SCL is output after the wait state is cleared.

When an 8-bit data transfer ends, serial transfer is stopped automatically and the interrupt request flag (CSIIF0) is set.

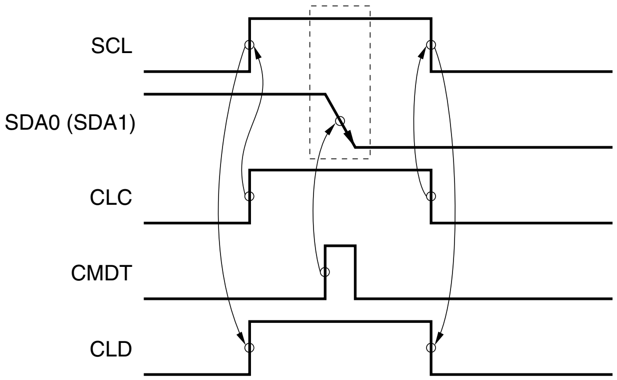
16.4.5 Cautions on use of I^{2}C bus mode

(1) Start condition output (master)

The SCL pin normally outputs a low-level signal when no serial clock is output. It is necessary to change the SCL pin to high in order to output a start condition signal. Set 1 in CLC of the interrupt timing specify register (SINT) to drive the SCL pin high.

After setting CLC, clear CLC to 0 and return the SCL pin to low. If CLC remains 1, no serial clock is output. If it is the master device which outputs the start condition and stop condition signals, confirm that CLD is set to 1 after setting CLC to 1; a slave device may have set SCL to low (wait state).

Figure 16-24. Start Condition Output



(2) Slave wait release (slave transmission)

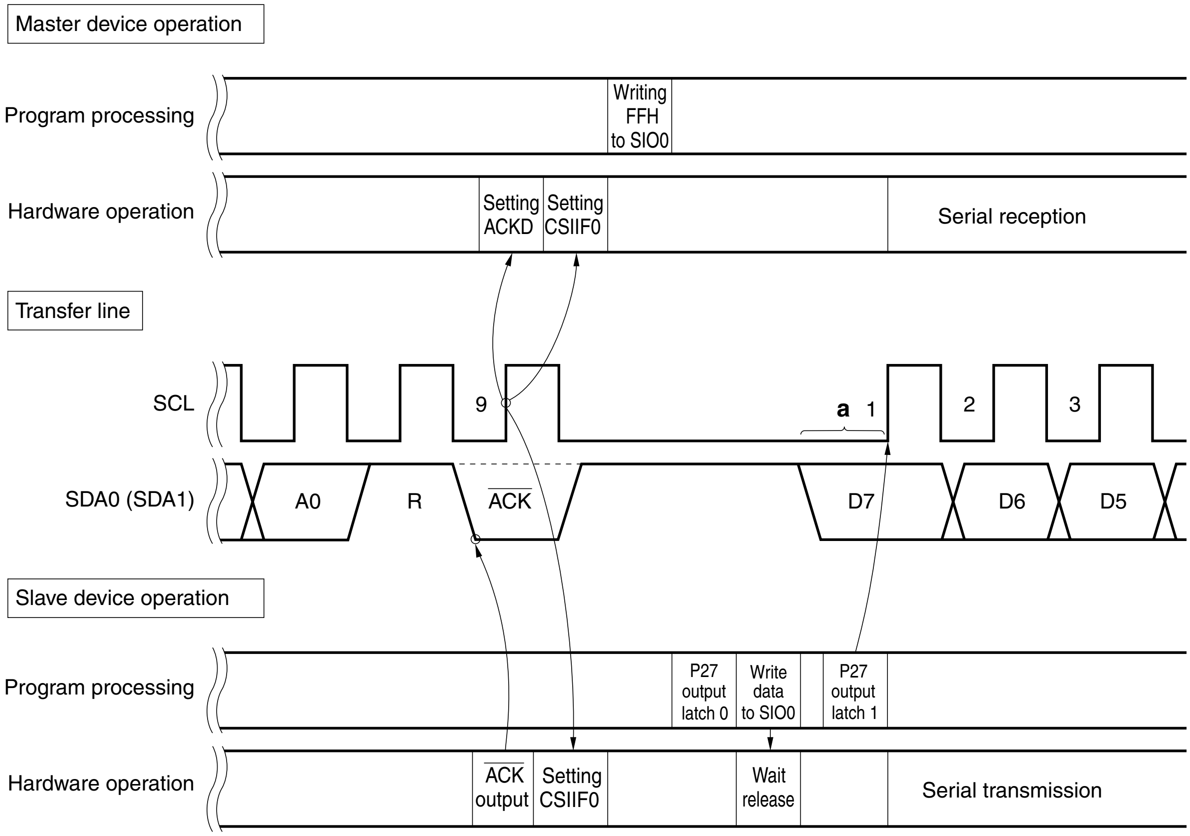
The wait status of a slave is released by setting the WREL flag, which is bit 2 of the interrupt timing specify register (SINT), or by executing a serial I/O shift register 0 (SIO0) write instruction.

If the slave sends data, the wait is immediately released by execution of an SIO0 write instruction and the clock rises without the start transmission bit being output in the data line. Therefore, manipulate the P27 output latch through the program as shown in Figure 16-25 to transmit data correctly. At this time, control the low-level width ("**a**" in Figure 16-25) of the first serial clock at the timing used for setting the P27 output latch to 1 after execution of an SIO0 write instruction.

In addition, if the acknowledge signal from the master is not output (if data transmission from the slave is completed), set 1 in the WREL flag of SINT and release the wait.

For these timings, see Figure 16-23.

Figure 16-25. Slave Wait Release (Transmission)



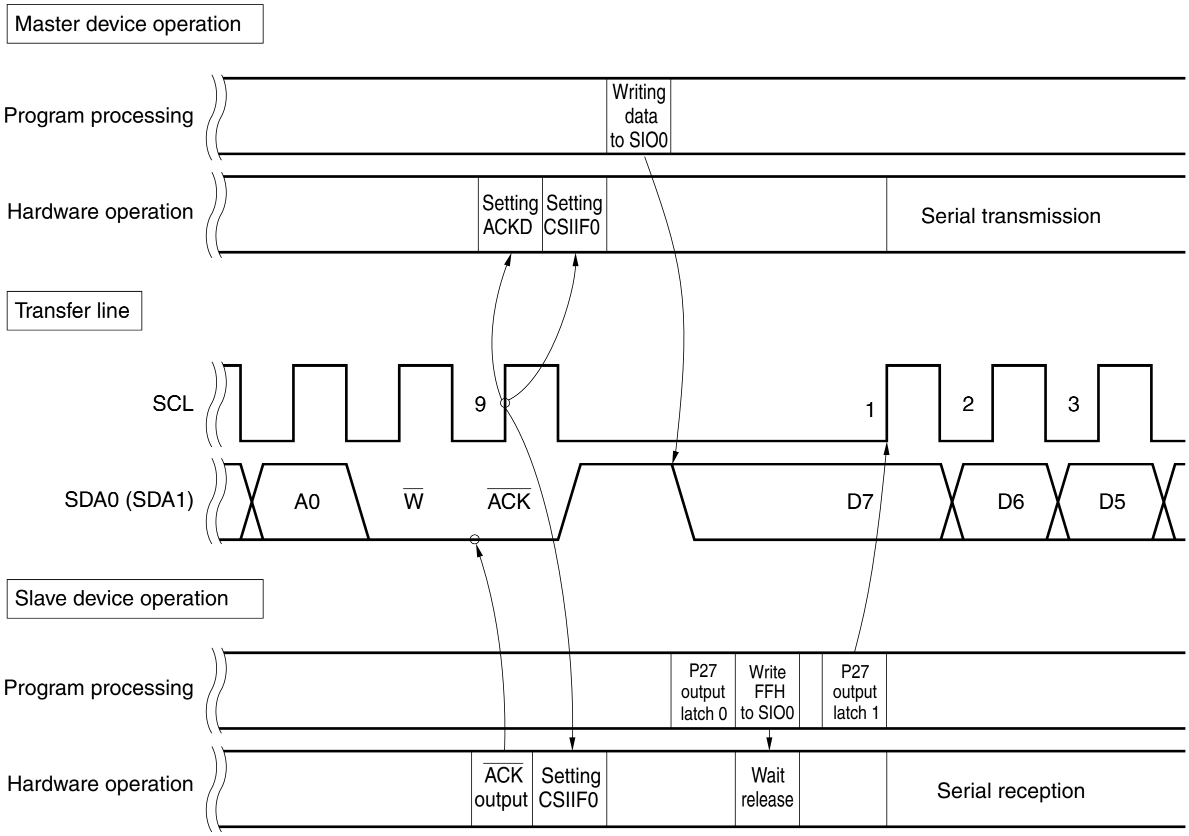
(3) Slave wait release (slave reception)

The wait status of a slave is released by setting the WREL flag, which is bit 2 of the interrupt timing specify register (SINT), or by executing a serial I/O shift register 0 (SIO0) write instruction.

When a slave receives data, if the SCL line immediately enters a high-impedance state due to a write to SIO0, the slave may not receive the first bit of the data sent from the master. This is because SIO0 cannot start operation if the SCL line is in a high-impedance state during execution of a write instruction to SIO0 (until the next instruction execution is started). Therefore, manipulate the P27 output latch through the program as shown in Figure 16-26 to receive data correctly.

For these timings, see Figure 16-22.

Figure 16-26. Slave Wait Release (Reception)



(4) Reception completion of slave

During processing of reception completion by a slave device, confirm the statuses of bit 3 (CMDD) of the serial bus interface control register (SBIC) and bit 6 (COI) of serial operating mode register 0 (CSIM0) (if CMDD = 1). This procedure is necessary to use the wake-up function normally. If an uncertain amount of data is sent from the master device, the slave device cannot determine whether the start condition signal or the data will be sent from the master. This may disable use of the wake-up function.

16.4.6 Restrictions when using I^2C bus mode

The following restrictions are applied to the μPD780308Y Subseries.

- Restrictions when μPD780308Y Subseries is used as slave device in I^2C bus mode

Subject: μPD780306Y, 780308Y, 78P0308Y, IE-780308-R-EM

Description: If the wake-up function is executed in the serial transfer status[Note] (by setting the WUP flag (bit 5 of serial operating mode register 0 (CSIM0)) to 1), the data between the other slave devices and the master device is identified as an address. If that data matches the slave address of the μPD780308Y Subseries, therefore, the μPD780308Y participates in communication, destroying the communication data.

Note The serial transfer status is the status where the interrupt request flag (CSIIF0) is set to 1 on completion of serial transfer after data has been written to serial I/O shift register 0 (SIO0).

Preventive measures: The above problem can be avoided by modifying the program.

Before executing the wake-up function, execute the program shown below that releases the serial transfer status. When executing the wake-up function, do not execute the instruction that writes data to SIO0. Even if this instruction is not executed, data can be received while the wake-up function is executed.

The program shown below is to release the serial transfer status. To release the serial transfer status, it is necessary to stop serial interface channel 0 once (by clearing the CSIE0 flag (bit 7 of serial operating mode register 0 (CSIM0)) to 0). If serial interface channel 0 is stopped in the I^2C bus mode, however, the SCL pin outputs the high level and the SDA0 (SDA1) pin outputs the low level, affecting communication of the I^2C bus. To prevent the I^2C bus from being influenced, therefore, this program makes the SCL and SDA0 (SDA1) pins go into a high-impedance state.

In this example, SDA0 (/P25) is used as a serial data input/output pin. To use SDA1 (/P26) as the serial data input/output pin, change P2.5 and PM2.5 in the program below to P2.6 and PM2.6, respectively.

For the timing of each signal when this program is executed, refer to Figure 16-22.

- Example of program to release serial transfer status

```
SET1   P2.5;     <1>
SET1   PM2.5;    <2>
SET1   PM2.7;    <3>
CLR1   CSIE0;    <4>
SET1   CSIE0;    <5>
SET1   RELT;     <6>
CLR1   PM2.7;    <7>
CLR1   P2.5;     <8>
CLR1   PM2.5;    <9>
```

<1> Prevents the SDA0 pin from outputting the low level when the I^2C bus mode is restored by instruction **<5>**. The SDA0 pin goes into a high-impedance state.

<2> Sets the P25(/SDA0) pin in the input mode to prevent the SDA0 line from being affected when the port mode is set by instruction **<4>**. The input mode is set when instruction **<2>** is executed.

<3> Sets the P27(/SCL) pin in the input mode to prevent the SCL line from being affected when the port mode is set by instruction **<4>**. The input mode is set when instruction **<3>** is executed.

<4> Changes the mode from the I^2C bus mode to the port mode.

<5> Restores the I^2C bus mode from the port mode.

<6> Prevents instruction **<8>** from causing the SDA0 pin to output the low level.

<7> Because the P27 pin must be set in the output mode in the I^2C bus mode, sets the P27 pin in the output mode.

<8> Because the output latch of the P25 pin must be cleared to 0 in the I^2C bus mode, clears the output latch of the P25 pin to 0.

<9> Because, in the I^2C bus mode, the P25 pin must be set in the output mode, sets the P25 pin in the output mode.

Remark RELT: bit 0 of serial bus interface control register (SBIC)

16.4.7 SCK0/SCL/P27 pin output manipulation

The SCK0/SCL/P27 pin enables static output by manipulating software in addition to normal serial clock output.

The value of serial clocks can be set by software (SI0/SB0/SDA0 and SO0/SB1/SDA1 pins are controlled with the RELT and CMDT bits of the serial bus interface control register (SBIC)).

The SCK0/SCL/P27 pin output should be manipulated as described below.

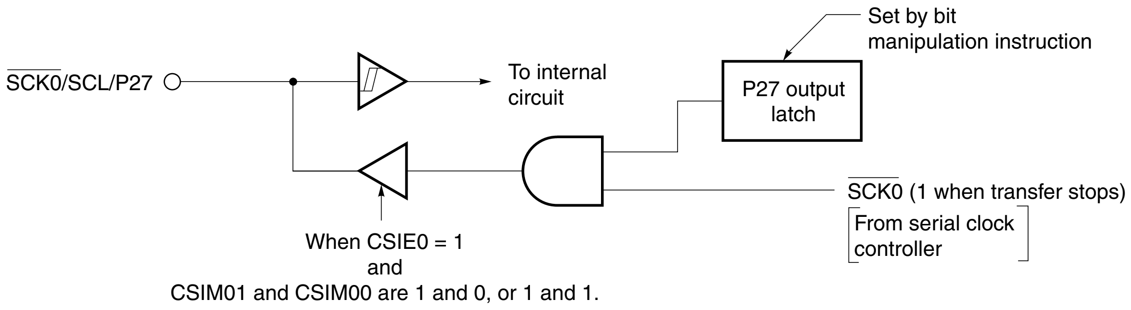
(1) In 3-wire serial I/O mode and 2-wire serial I/O mode

The SCK0/SCL/P27 pin output level is manipulated by the P27 output latch.

<1> Set serial operating mode register 0 (CSIM0) (SCK0 pin is set in the output mode and serial operation is enabled). While serial transfer is suspended, SCK0 is set to 1.

<2> Manipulate the content of the P27 output latch by executing the bit manipulation instruction.

Figure 16-27. SCK0/SCL/P27 Pin Configuration


SCK0/SCL/P27
To internal circuit
Set by bit manipulation instruction
P27 output latch
SCK0 (1 when transfer stops)
From serial clock controller
When CSIE0 = 1
and
CSIM01 and CSIM00 are 1 and 0, or 1 and 1.

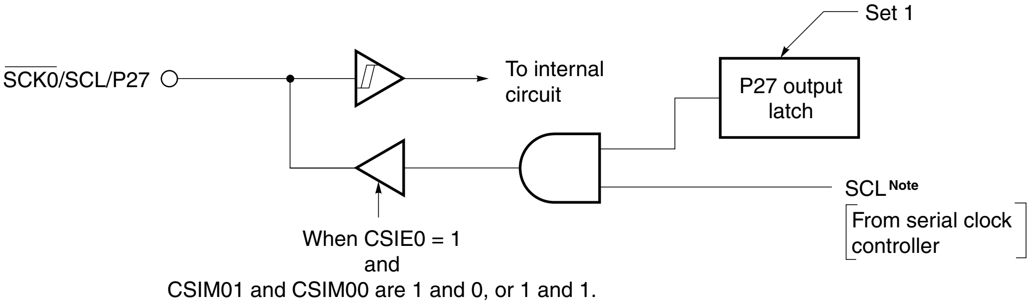
(2) In I²C bus mode

The $\overline{\text{SCK0}}$/SCL/P27 pin output level is manipulated by the CLC bit of the interrupt timing specify register (SINT).

<1> Set serial operating mode register 0 (CSIM0) (SCL pin is set in the output mode and serial operation is enabled). Set 1 to the P27 output latch. While serial transfer is suspended, SCL is set to 0.

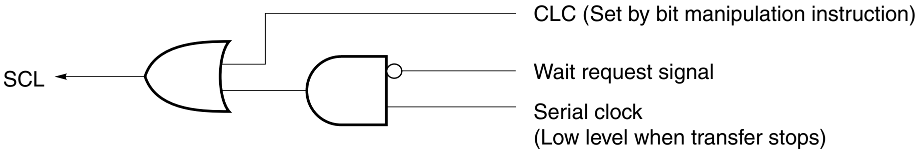
<2> Manipulate the content of the CLC bit of SINT by executing the bit manipulation instruction.

Figure 16-28. $\overline{\text{SCK0}}$/SCL/P27 Pin Configuration



Note The level of SCL signal follows the contents of logic circuit shown in Figure 16-29.

Figure 16-29. Logic Circuit of SCL Signal


CLC (Set by bit manipulation instruction)
SCL
Wait request signal
Serial clock
(Low level when transfer stops)

Remarks 1. This figure shows the relationship of each signal, and does not show the internal circuit.

2. CLC: Bit 3 of interrupt timing specify register (SINT)

CHAPTER 17 SERIAL INTERFACE CHANNEL 2

17.1 Serial Interface Channel 2 Functions

Serial interface channel 2 has the following three modes.

- Operation stop mode
- Asynchronous serial interface (UART) mode
- 3-wire serial I/O mode

(1) Operation stop mode

This mode is used when serial transfer is not carried out to reduce power consumption.

(2) Asynchronous serial interface (UART) mode

In this mode, one byte of data is transmitted/received following the start bit, and full-duplex operation is possible.

A dedicated UART baud rate generator is incorporated, allowing communication over a wide range of baud rates. In addition, the baud rate can be defined by scaling the input clock to the ASCK pin.

The MIDI standard baud rate (31.25 kbps) can be used by employing the dedicated UART baud rate generator.

Serial interface channel 2 has two data I/O pins (RxD and TxD) which can be selected by software. Note that only one data I/O pin can be used at one time.

Caution When it is not necessary to change the data I/O pin, using the RxD/SI2/P70 and TxD/SO2/P71 is recommended. If only port 11 (RxD/P114 and TxD/P113) is used as data I/O pin, the function of port 7 is limited.

(3) 3-wire serial I/O mode (MSB-first/LSB-first switchable)

In this mode, 8-bit data transfer is performed using three lines: the serial clock ($\overline{\text{SCK2}}$), and serial data lines (SI2, SO2).

In the 3-wire serial I/O mode, simultaneous transmission and reception is possible, increasing the data transfer processing speed.

Either the MSB or LSB can be specified as the start bit for an 8-bit data serial transfer, allowing connection to devices using either as the start bit.

The 3-wire serial I/O mode is useful for connection to peripheral I/Os and display controllers, etc., which incorporate a conventional synchronous clocked serial interface, such as the 75X/XL Series, 78K Series, 17K Series, etc.

17.2 Serial Interface Channel 2 Configuration

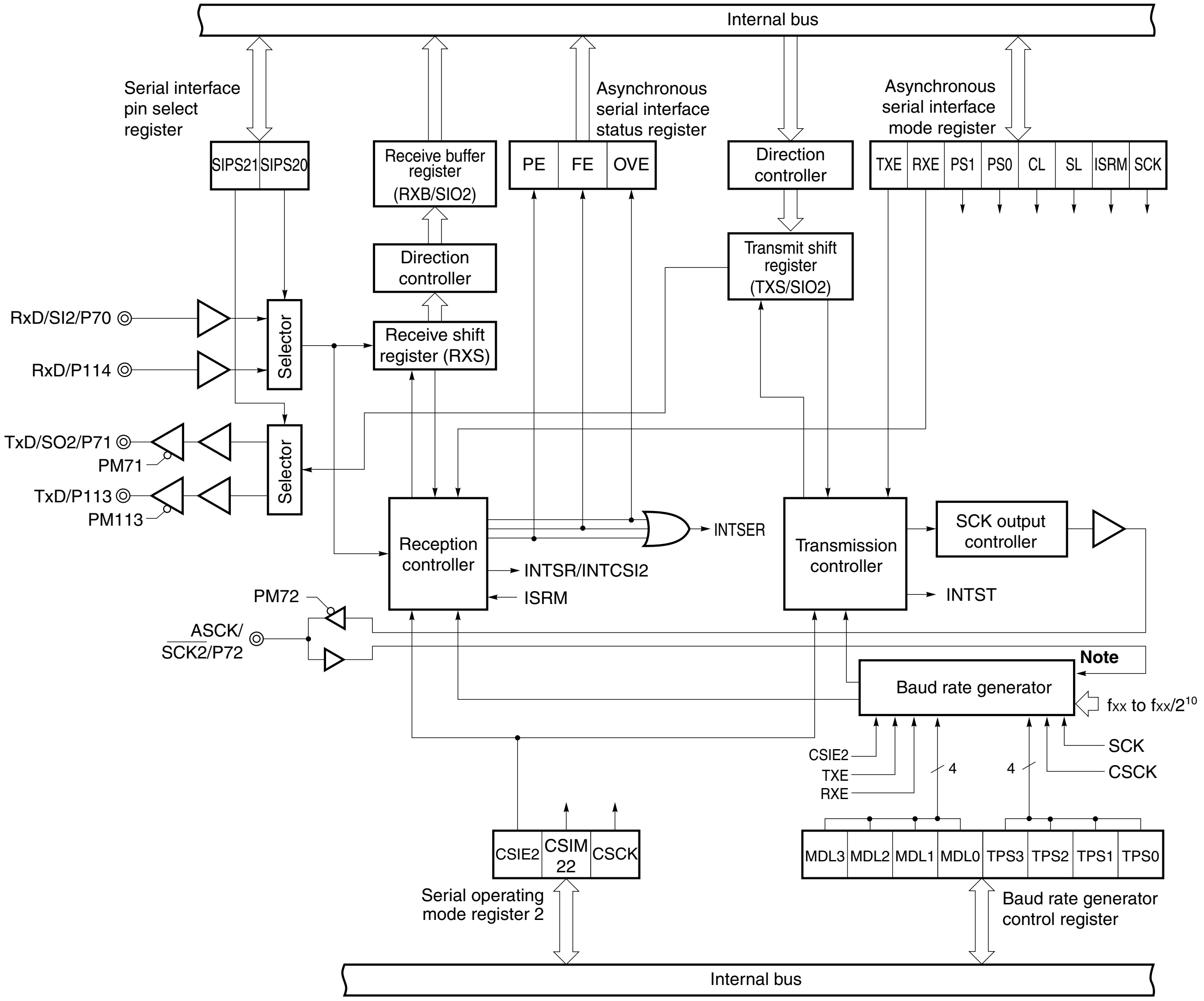
Serial interface channel 2 consists of the following hardware.

Table 17-1. Serial Interface Channel 2 Configuration

Item	Configuration
Register	Transmit shift register (TXS) Receive shift register (RXS) Receive buffer register (RXB)
Control register	Serial operating mode register 2 (CSIM2) Asynchronous serial interface mode register (ASIM) Asynchronous serial interface status register (ASIS) Baud rate generator control register (BRGC) Serial interface pin select register (SIPS) Port mode register 7 (PM7)[Note]

Note Refer to **Figure 6-10 P70 Block Diagram** and **Figure 6-11 P71 and P72 Block Diagram**.

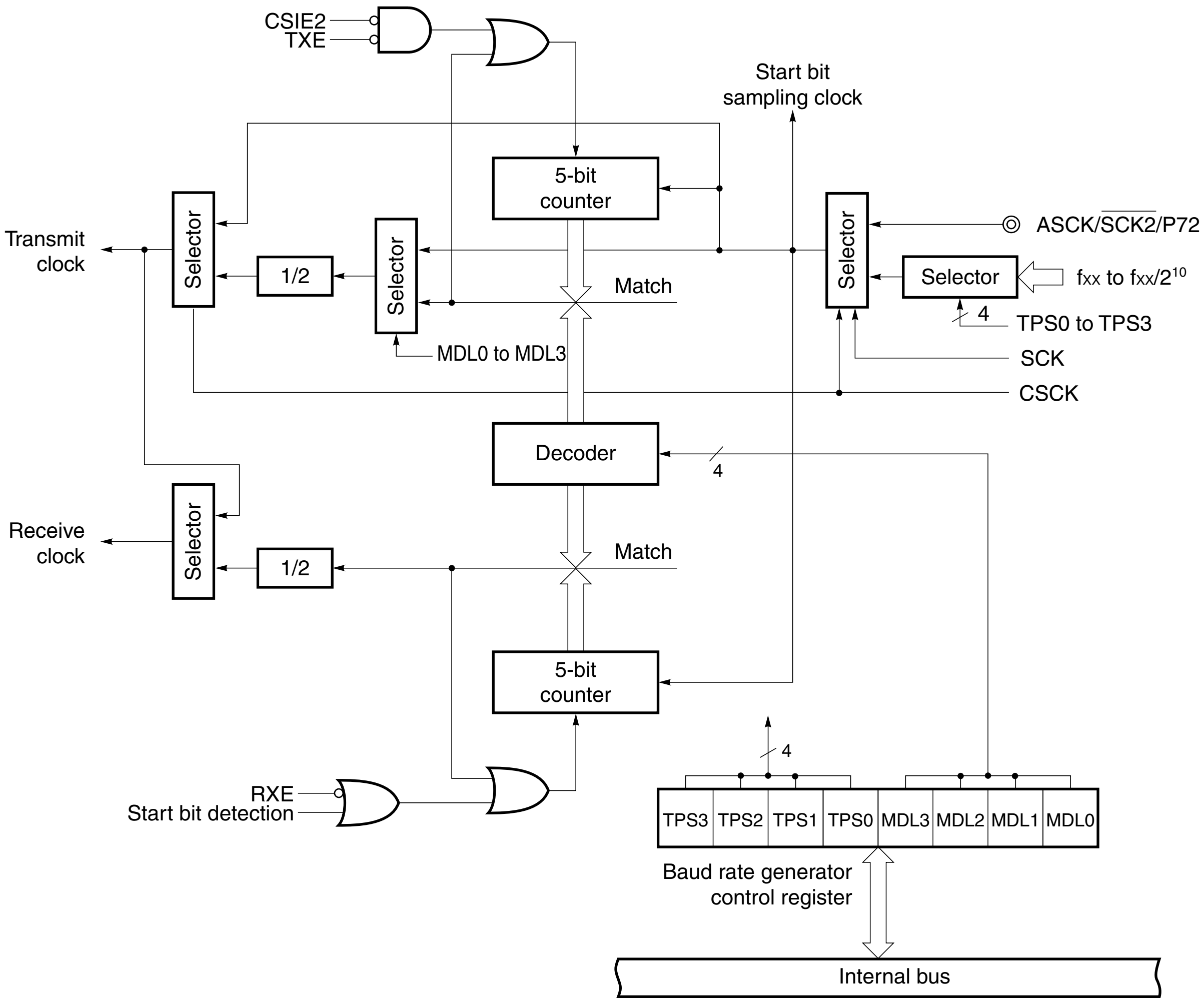
Figure 17-1. Serial Interface Channel 2 Block Diagram



Note See **Figure 17-2** for the baud rate generator configuration.

Remark fxx = fx/2 (MCS = 0), fxx = fx (MCS = 1)

Figure 17-2. Baud Rate Generator Block Diagram



(1) Transmit shift register (TXS)

This register is used to set the transmit data. The data written in TXS is transmitted as serial data.

If the data length is specified as 7 bits, bits 0 to 6 of the data written in TXS are transferred as transmit data.

Writing data to TXS starts the transmit operation.

TXS is written to with an 8-bit memory manipulation instruction. It cannot be read.

TXS value is FFH after $\overline{\text{RESET}}$ input.

Caution TXS must not be written to during a transmit operation. TXS and the receive buffer register (RXB) are allocated to the same address, and when a read is performed, the value of RXB is read.

(2) Receive shift register (RXS)

This register is used to convert serial data input to the RxD pin to parallel data. When one byte of data is received, the receive data is transferred to the receive buffer register (RXB).

RXS cannot be directly manipulated by a program.

(3) Receive buffer register (RXB)

This register holds receive data. Each time one byte of data is received, new receive data is transferred from the receive shift register (RXS).

If the data length is specified as 7 bits, the receive data is transferred to bits 0 to 6 of RXB, and the MSB of RXB is always set to 0.

RXB is read with an 8-bit memory manipulation instruction. It cannot be written to.

RXB value is FFH after $\overline{\text{RESET}}$ input.

Caution RXB and the transmit shift register (TXS) are allocated to the same address, and when a write is performed, the value is written to TXS.

(4) Transmission controller

This circuit performs transmit operation control such as the addition of a start bit, parity bit and stop bit to data written in the transmit shift register (TXS) in accordance with the contents set in the asynchronous serial interface mode register (ASIM).

(5) Reception controller

This circuit controls receive operations in accordance with the contents set in the asynchronous serial interface mode register (ASIM). It performs error checks for parity errors, etc., during a receive operation, and if an error is detected, sets a value in the asynchronous serial interface status register (ASIS) in accordance with the error contents.

17.3 Serial Interface Channel 2 Control Registers

Serial interface channel 2 is controlled by the following five registers.

- Serial operating mode register 2 (CSIM2)
- Asynchronous serial interface mode register (ASIM)
- Asynchronous serial interface status register (ASIS)
- Baud rate generator control register (BRGC)
- Serial interface pin select register (SIPS)

(1) Serial operating mode register 2 (CSIM2)

This register is set when serial interface channel 2 is used in the 3-wire serial I/O mode.
CSIM2 is set with a 1-bit or 8-bit memory manipulation instruction.
$\overline{\text{RESET}}$ input clears CSIM2 to 00H.

Figure 17-3. Serial Operating Mode Register 2 Format

Symbol	⑦	6	5	4	3	2	1	0	Address	After Reset	R/W
CSIM2	CSIE2	0	0	0	0	CSIM 22	CSCK	0	FF72H	00H	R/W

CSIE2	Operation Control in 3-wire Serial I/O Mode
0	Operation stopped
1	Operation enabled

CSIM22	First Bit Specification
0	MSB
1	LSB

CSCK	Clock Selection in 3-wire Serial I/O Mode
0	Input clock from off-chip to $\overline{\text{SCK2}}$ pin
1	Dedicated baud rate generator output

Cautions 1. Ensure that bit 0 and bit 3 to bit 6 are set to 0.

2. When UART mode is selected, CSIM2 should be set to 00H.

(2) Asynchronous serial interface mode register (ASIM)

This register is set when serial interface channel 2 is used in the asynchronous serial interface mode.

ASIM is set with a 1-bit or 8-bit memory manipulation instruction.

$\overline{\text{RESET}}$ input clears ASIM to 00H.

Figure 17-4. Asynchronous Serial Interface Mode Register Format

Symbol	⑦	⑥	5	4	3	2	1	0	Address	After Reset	R/W
ASIM	TXE	RXE	PS1	PS0	CL	SL	ISRM	SCK	FF70H	00H	R/W

TXE	Transmit Operation Control
0	Transmit operation stopped
1	Transmit operation enabled

RXE	Receive Operation Control
0	Receive operation stopped
1	Receive operation enabled

PS1	PS0	Parity Bit Specification
0	0	No parity
0	1	0 parity always added in transmission No parity test in reception (parity errors do not occur)
1	0	Odd parity
1	1	Even parity

CL	Character Length Specification
0	7 bits
1	8 bits

SL	Transmit Data Stop Bit Length Specification
0	1 bit
1	2 bits

ISRM	Control of Reception Completion Interrupt When Error Occurs
0	Reception completion interrupt generated when an error occurs
1	Reception completion interrupt not generated when an error occurs

SCK	Clock Selection in Asynchronous Serial Interface Mode
0	Input clock from off-chip to ASCK pin
1	Dedicated baud rate generator output[Note]

Note When SCK is set to 1 and the baud rate generator output is selected, the ASCK pin can be used as an I/O port.

Cautions 1. When the 3-wire serial I/O mode is selected, 00H should be set in ASIM.

2. The serial transmit/receive operation must be stopped before changing the operating mode.

Table 17-2. Serial Interface Channel 2 Operating Mode Settings (1/2)

(1) Operation stop mode

ASIM			CSIM2			SIPS		PM70	P70	PM71	P71	PM113	P113	PM114	P114	PM72	P72	Start Bit	Shift Clock	P70/SI2 /RxD Pin Functions	P71/SO2 /TxD Pin Functions	P113/TxD Pin Functions	P114/RxD Pin Functions	P72/$\overline{\text{SCK2}}$ /ASCK Pin Functions
TXE	RXE	SCK	CSIE2	CSIM22	CSCK	SIPS21	SIPS20																	
0	0	×	0	×	×	×	×	×[Note 1]	×[Note 1]	×[Note 1]	×[Note 1]	×[Note 1]	×[Note 1]	×[Note 1]	×[Note 1]	×[Note1]	×[Note 1]	—	—	P70	P71	P113	P114	P72
Other than above																		Setting prohibited						

(2) 3-wire serial I/O mode

ASIM			CSIM2			SIPS		PM70	P70	PM71	P71	PM113	P113	PM114	P114	PM72	P72	Start Bit	Shift Clock	P70/SI2 /RxD Pin Functions	P71/SO2 /TxD Pin Functions	P113/TxD Pin Functions	P114/RxD Pin Functions	P72/$\overline{\text{SCK2}}$ /ASCK Pin Functions
TXE	RXE	SCK	CSIE2	CSIM22	CSCK	SIPS21	SIPS20																	
0	0	0	1	0	0	×	×	×[Note 2]	×[Note 2]	0	1	×[Note 1]	×[Note 1]	×[Note 1]	×[Note 1]	1	×	MSB	External clock	SI2[Note 2]	SO2 (CMOS output)	P113	P114	$\overline{\text{SCK2}}$ input
					1											0	1		Internal clock					$\overline{\text{SCK2}}$ output
			1	1	0											1	×	LSB	External clock	SI2[Note 2]	SO2 (CMOS output)			$\overline{\text{SCK2}}$ input
					1											0	1		Internal clock					$\overline{\text{SCK2}}$ output
Other than above																		Setting prohibited						

Notes 1. Can be used freely as port function.

2. Can be used as P70 (CMOS I/O) when only transmitter is used.

Remark ×: don't care

Table 17-2. Serial Interface Channel 2 Operating Mode Settings (2/2)

(3) Asynchronous serial interface mode

ASIM			CSIM2			SIPS		PM70	P70	PM71	P71	PM113	P113	PM114	P114	PM72	P72	Start Bit	Shift Clock	P70/SI2 /RxD Pin Functions	P71/SO2 /TxD Pin Functions	P113/TxD Pin Functions	P114/RxD Pin Functions	P72/$\overline{\text{SCK2}}$ /ASCK Pin Functions
TXE	RXE	SCK	CSIE2	CSIM22	CSCK	SIPS21	SIPS20																	
1	0	0	0	0	0	0	0	×Note	×Note	0	1	×Note	×Note	×Note	×Note	1	×	LSB	External clock	P70	TxD (CMOS output)	P113	P114	ASCK input
		1														×Note	×Note		Internal clock					P72
0	1	0	0	0	0	0	0	1	×	×Note	×Note	×Note	×Note	×Note	×Note	1	×		External clock	RxD	P71			ASCK input
		1														×Note	×Note		Internal clock					P72
1	1	0	0	0	0	0	0	1	×	0	1	×Note	×Note	×Note	×Note	1	×		External clock		TxD (CMOS output)			ASCK input
		1														×Note	×Note		Internal clock					P72
1	0	0	0	0	0	1	0	×Note	×Note	0	1	0	1	×Note	×Note	1	×		External clock	P70	High output	TxD	P114	ASCK input
		1														×Note	×Note		Internal clock					P72
0	1	0	0	0	0	0	1	1	×	×Note	×Note	×Note	×Note	1	×	1	×		External clock	P70 (Input)	P71	P113	RxD	ASCK input
		1														×Note	×Note		Internal clock					P72
1	1	0	0	0	0	1	1	1	×	0	1	0	1	1	×	1	×		External clock	P70 (Input)	High output	TxD	RxD	ASCK input
		1														×Note	×Note		Internal clock					P72
Other than above																		Setting prohibited						

Note Can be used freely as port function.

Remark ×: don't care
PM××: Port mode register
P××: Port output latch

(3) Asynchronous serial interface status register (ASIS)

This is a register which displays the type of error when a receive error occurs in the asynchronous serial interface mode.

ASIS is read with an 8-bit memory manipulation instruction.

In 3-wire serial I/O mode, the contents of ASIS are undefined.

$\overline{\text{RESET}}$ input clears ASIS to 00H.

Figure 17-5. Asynchronous Serial Interface Status Register Format

Symbol	7	6	5	4	3	2	1	0	Address	After Reset	R/W
ASIS	0	0	0	0	0	PE	FE	OVE	FF71H	00H	R

PE	Parity Error Flag
0	Parity error does not occur
1	Parity error occurs (When transmit data parity does not match)

FE	Framing Error Flag
0	Framing error does not occur
1	Framing error occurs (When stop bit is not detected)$^{\text{Note 1}}$

OVE	Overrun Error Flag
0	Overrun error does not occur
1	Overrun error occurs (When next receive operation is completed before data from receive buffer register is read)$^{\text{Note 2}}$

Notes 1. Even if the stop bit length has been set as 2 bits by bit 2 (SL) of the asynchronous serial interface mode register (ASIM), only single stop bit detection is performed during reception.

2. The receive buffer register (RXB) must be read when an overrun error occurs. Overrun errors will continue to occur until RXB is read.

(4) Baud rate generator control register (BRGC)

This register sets the serial clock for serial interface channel 2.

BRGC is set with an 8-bit memory manipulation instruction.

$\overline{\text{RESET}}$ input clears BRGC to 00H.

Figure 17-6. Baud Rate Generator Control Register Format (1/2)

Symbol	7	6	5	4	3	2	1	0	Address	After Reset	R/W
BRGC	TPS3	TPS2	TPS1	TPS0	MDL3	MDL2	MDL1	MDL0	FF73H	00H	R/W

TPS3	TPS2	TPS1	TPS0	Selects Source Clock of 5-bit Counter		n
				MCS = 1	MCS = 0	
0	0	0	0	$f_X/2^{10}$ (4.9 kHz)	$f_X/2^{11}$ (2.4 kHz)	11
0	1	0	1	f_X (5.0 MHz)	$f_X/2$ (2.5 MHz)	1
0	1	1	0	$f_X/2$ (2.5 MHz)	$f_X/2^2$ (1.25 MHz)	2
0	1	1	1	$f_X/2^2$ (1.25 MHz)	$f_X/2^3$ (625 kHz)	3
1	0	0	0	$f_X/2^3$ (625 kHz)	$f_X/2^4$ (313 kHz)	4
1	0	0	1	$f_X/2^4$ (313 kHz)	$f_X/2^5$ (156 kHz)	5
1	0	1	0	$f_X/2^5$ (156 kHz)	$f_X/2^6$ (78.1 kHz)	6
1	0	1	1	$f_X/2^6$ (78.1 kHz)	$f_X/2^7$ (39.1 kHz)	7
1	1	0	0	$f_X/2^7$ (39.1 kHz)	$f_X/2^8$ (19.5 kHz)	8
1	1	0	1	$f_X/2^8$ (19.5 kHz)	$f_X/2^9$ (9.8 kHz)	9
1	1	1	0	$f_X/2^9$ (9.8 kHz)	$f_X/2^{10}$ (4.9 kHz)	10
Other than above				Setting prohibited		

Remarks 1. f_X: Main system clock oscillation frequency

2. MCS: Oscillation mode select register bit 0

3. n: Value set in TPS0 to TPS3 ($1 \leq n \leq 11$)

4. Figures in parentheses apply to operation with f_X = 5.0 MHz

Figure 17-6. Baud Rate Generator Control Register Format (2/2)

MDL3	MDL2	MDL1	MDL0	Baud Rate Generator Input Clock Selection	k
0	0	0	0	$f_{SCK}/16$	0
0	0	0	1	$f_{SCK}/17$	1
0	0	1	0	$f_{SCK}/18$	2
0	0	1	1	$f_{SCK}/19$	3
0	1	0	0	$f_{SCK}/20$	4
0	1	0	1	$f_{SCK}/21$	5
0	1	1	0	$f_{SCK}/22$	6
0	1	1	1	$f_{SCK}/23$	7
1	0	0	0	$f_{SCK}/24$	8
1	0	0	1	$f_{SCK}/25$	9
1	0	1	0	$f_{SCK}/26$	10
1	0	1	1	$f_{SCK}/27$	11
1	1	0	0	$f_{SCK}/28$	12
1	1	0	1	$f_{SCK}/29$	13
1	1	1	0	$f_{SCK}/30$	14
1	1	1	1	f_{SCK}Note	—

Note Can only be used in 3-wire serial I/O mode.

Caution When a write is performed to BRGC during a communication operation, baud rate generator output is disrupted and communication cannot be performed normally.
Therefore, BRGC must not be written to during a communication operation.

Remarks 1. f_{SCK}: 5-bit counter source clock

2. k: Value set in MDL0 to MDL3 ($0 \leq k \leq 14$)

The baud rate transmit/receive clock generated is either a signal scaled from the main system clock, or a signal scaled from the clock input from the ASCK pin.

(a) Generation of baud rate transmit/receive clock by means of main system clock

The transmit/receive clock is generated by scaling the main system clock. The baud rate generated from the main system clock is obtained with the following expression.

$$[\text{Baud rate}] = \frac{f_{XX}}{2^n \times (k + 16)} \ [\text{Hz}]$$

fx: Main system clock oscillation frequency

fxx: Main system clock frequency (fx or fx/2)

n: Value set in TPS0 to TPS3 ($1 \leq n \leq 11$)

k: Value set in MDL0 to MDL3 ($0 \leq k \leq 14$)

Table 17-3. Relationships Between Main System Clock and Baud Rate

Baud Rate (bps)	fx = 5.0 MHz				fx = 4.19 MHz			
	MCS = 1		MCS = 0		MCS = 1		MCS = 0	
	BRGC Set Value	Error (%)	BRGC Set Value	Error (%)	BRGC Set Value	Error (%)	BRGC Set Value	Error (%)
75	–		00H	1.73	0BH	1.14	EBH	1.14
110	06H	0.88	E6H	0.88	03H	–2.01	E3H	–2.01
150	00H	1.73	E0H	1.73	EBH	1.14	DBH	1.14
300	E0H	1.73	D0H	1.73	DBH	1.14	CBH	1.14
600	D0H	1.73	C0H	1.73	CBH	1.14	BBH	1.14
1200	C0H	1.73	B0H	1.73	BBH	1.14	ABH	1.14
2400	B0H	1.73	A0H	1.73	ABH	1.14	9BH	1.14
4800	A0H	1.73	90H	1.73	9BH	1.14	8BH	1.14
9600	90H	1.73	80H	1.73	8BH	1.14	7BH	1.14
19200	80H	1.73	70H	1.73	7BH	1.14	6BH	1.14
31250	74H	0	64H	0	71H	–1.31	61H	–1.31
38400	70H	1.73	60H	1.73	6BH	1.14	5BH	1.14
76800	60H	1.73	50H	1.73	5BH	1.14	—	—

MCS: Oscillation mode select register (CSMS) bit 0

(b) Generation of baud rate transmit/receive clock by means of external clock from ASCK pin

The transmit/receive clock is generated by scaling the clock input from the ASCK pin. The baud rate generated from the clock input from the ASCK pin is obtained with the following expression.

$$[\text{Baud rate}] = \frac{f_{ASCK}}{2 \times (k + 16)} \text{ [Hz]}$$

f_{ASCK}: Frequency of clock input to ASCK pin

k: Value set in MDL0 to MDL3 ($0 \leq k \leq 14$)

Table 17-4. Relationships Between ASCK Pin Input Frequency and Baud Rate (When BRGC Is Set to 00H)

Baud Rate (bps)	ASCK Pin Input Frequency
75	2.4 kHz
110	3.52 kHz
150	4.8 kHz
300	9.6 kHz
600	19.2 kHz
1200	38.4 kHz
2400	76.8 kHz
4800	153.6 kHz
9600	307.2 kHz
19200	614.4 kHz
31250	1000.0 kHz
38400	1228.8 kHz

(5) Serial interface pin select register (SIPS)

This register selects I/O pins when serial interface channel 2 is used in the asynchronous serial interface mode.

SIPS is set with a 1-bit or 8-bit memory manipulation instruction.

$\overline{\text{RESET}}$ input clears SIPS to 00H.

To select I/O pins, the port mode register and the output latch of the port must be set. For details, refer to **Table 17-2 Serial Interface Channel 2 Operating Mode Settings**.

Figure 17-7. Serial Interface Pin Select Register Format

Symbol	7	6	5	4	3	2	1	0	Address	After Reset	R/W
SIPS	0	0	SIPS21	SIPS20	0	0	0	0	FF75H	00H	R/W

SIPS21	SIPS20	Selects I/O Pin of Asynchronous Serial Interface
0	0	Input pin: RxD/SI2/P70 Output pin: TxD/SO2/P71
0	1	Input pin: RxD/P114 Output pin: TxD/SO2/P71
1	0	Input pin: RxD/SI2/P70 Output pin: TxD/P113
1	1	Input pin: RxD/P114 Output pin: TxD/P113

Cautions 1. Select I/O pins after stopping serial transmission/reception.

2. Port 11 has a falling edge detection function. Do not specify the pin of this port used in a mode other than port mode to input the falling edge. For how to set to input the falling edge, refer to Figure 6-21 Key Return Mode Register Format.

17.4 Serial Interface Channel 2 Operation

The operating mode of serial interface channel 2 has the following three types.

- Operation stop mode
- Asynchronous serial interface (UART) mode
- 3-wire serial I/O mode

17.4.1 Operation stop mode

In the operation stop mode, serial transfer is not performed, and therefore power consumption can be reduced.

In the operation stop mode, the P70/SI2/RxD, P71/SO2/TxD, P72/$\overline{\text{SCK2}}$/ASCK, P113/TxD, and P114/RxD pins can be used as normal I/O ports.

(1) Register setting

Operation stop mode settings are performed using serial operating mode register 2 (CSIM2) and the asynchronous serial interface mode register (ASIM).

(a) Serial operating mode register 2 (CSIM2)

CSIM2 is set with a 1-bit or 8-bit memory manipulation instruction.

$\overline{\text{RESET}}$ input clears CSIM2 to 00H.

Symbol	⑦	6	5	4	3	2	1	0	Address	After Reset	R/W
CSIM2	CSIE2	0	0	0	0	CSIM22	CSCK	0	FF72H	00H	R/W

CSIE2	Operation Control in 3-wire Serial I/O Mode
0	Operation stopped
1	Operation enabled

Caution Ensure that bit 0 and bit 3 to bit 6 are set to 0.

(b) Asynchronous serial interface mode register (ASIM)

ASIM is set with a 1-bit or 8-bit memory manipulation instruction.

$\overline{\text{RESET}}$ input clears ASIM to 00H.

Symbol	⑦	⑥	5	4	3	2	1	0	Address	After Reset	R/W
ASIM	TXE	RXE	PS1	PS0	CL	SL	ISRM	SCK	FF70H	00H	R/W

TXE	Transmit Operation Control
0	Transmit operation stopped
1	Transmit operation enabled

RXE	Receive Operation Control
0	Receive operation stopped
1	Receive operation enabled

17.4.2 Asynchronous serial interface (UART) mode

In this mode, one byte of data is transmitted/received following the start bit, and full-duplex operation is possible.

A dedicated UART baud rate generator is incorporated, allowing communication over a wide range of baud rates. In addition, the baud rate can be defined by scaling the input clock to the ASCK pin.

The MIDI standard baud rate (31.25 kbps) can be used by employing the dedicated UART baud rate generator.

Serial interface channel 2 has two data I/O pins (RxD and TxD) which can be selected by software. Note that only one data I/O pin can be used at one time.

Caution When it is not necessary to change the data I/O pin, using the RxD/SI2/P70 and TxD/SO2/P71 is recommended. If only port 11 (RxD/P114 and TxD/P113) is used as data I/O pin, the function of port 7 is limited.

(1) Register setting

UART mode settings are performed using serial operating mode register 2 (CSIM2), the asynchronous serial interface mode register (ASIM), the asynchronous serial interface status register (ASIS), the baud rate generator control register (BRGC), and the serial interface pin select register (SIPS).

(a) Serial operating mode register 2 (CSIM2)

CSIM2 is set with a 1-bit or 8-bit memory manipulation instruction.

$\overline{\text{RESET}}$ input clears CSIM2 to 00H.

When the UART mode is selected, 00H should be set in CSIM2.

Symbol	⑦	6	5	4	3	2	1	0	Address	After Reset	R/W
CSIM2	CSIE2	0	0	0	0	CSIM 22	CSCK	0	FF72H	00H	R/W

CSIE2	Operation Control in 3-wire Serial I/O Mode
0	Operation stopped
1	Operation enabled

CSIM22	First Bit Specification
0	MSB
1	LSB

CSCK	Clock Selection in 3-wire Serial I/O Mode
0	Input clock from off-chip to $\overline{\text{SCK2}}$ pin
1	Dedicated baud rate generator output

Caution Ensure that bit 0 and bit 3 to bit 6 are set to 0.

(b) Asynchronous serial interface mode register (ASIM)

ASIM is set with a 1-bit or 8-bit memory manipulation instruction.

$\overline{\text{RESET}}$ input clears ASIM to 00H.

Symbol	⑦	⑥	5	4	3	2	1	0	Address	After Reset	R/W
ASIM	TXE	RXE	PS1	PS0	CL	SL	ISRM	SCK	FF70H	00H	R/W

TXE	Transmit Operation Control
0	Transmit operation stopped
1	Transmit operation enabled

RXE	Receive Operation Control
0	Receive operation stopped
1	Receive operation enabled

PS1	PS0	Parity Bit Specification
0	0	No parity
0	1	0 parity always added in transmission No parity test in reception (parity errors do not occur)
1	0	Odd parity
1	1	Even parity

CL	Character Length Specification
0	7 bits
1	8 bits

SL	Transmit Data Stop Bit Length Specification
0	1 bit
1	2 bits

ISRM	Control of Reception Completion Interrupt When Error Occurs
0	Reception completion interrupt generated when an error occurs
1	Reception completion interrupt not generated when an error occurs

SCK	Clock Selection in Asynchronous Serial Interface Mode
0	Input clock from off-chip to ASCK pin
1	Dedicated baud rate generator output[Note]

Note When SCK is set to 1 and the baud rate generator output is selected, the ASCK pin can be used as an I/O port.

Caution The serial transmit/receive operation must be stopped before changing the operating mode.

(c) Asynchronous serial interface status register (ASIS)

ASIS is read with an 8-bit memory manipulation instruction.

$\overline{\text{RESET}}$ input clears ASIS to 00H.

Symbol	7	6	5	4	3	2	1	0	Address	After Reset	R/W
ASIS	0	0	0	0	0	PE	FE	OVE	FF71H	00H	R

PE	Parity Error Flag
0	Parity error does not occur
1	Parity error occurs (When transmit data parity does not match)

FE	Framing Error Flag
0	Framing error does not occur
1	Framing error occurs (When stop bit is not detected)[Note 1]

OVE	Overrun Error Flag
0	Overrun error does not occur
1	Overrun error occurs (When next receive operation is completed before data from receive buffer register is read)[Note 2]

Notes 1. Even if the stop bit length has been set as 2 bits by bit 2 (SL) of the asynchronous serial interface mode register (ASIM), only single stop bit detection is performed during reception.

2. The receive buffer register (RXB) must be read when an overrun error occurs. Overrun errors will continue to occur until RXB is read.

(d) Baud rate generator control register (BRGC)

BRGC is set with an 8-bit memory manipulation instruction.

$\overline{\text{RESET}}$ input clears BRGC to 00H.

Symbol	7	6	5	4	3	2	1	0	Address	After Reset	R/W
BRGC	TPS3	TPS2	TPS1	TPS0	MDL3	MDL2	MDL1	MDL0	FF73H	00H	R/W

TPS3	TPS2	TPS1	TPS0	Selects Source Clock of 5-bit Counter		n
				MCS = 1	MCS = 0	
0	0	0	0	fx/2^{10} (4.9 kHz)	fx/2^{11} (2.4 kHz)	11
0	1	0	1	fx (5.0 MHz)	fx/2 (2.5 MHz)	1
0	1	1	0	fx/2 (2.5 MHz)	fx/2^2 (1.25 MHz)	2
0	1	1	1	fx/2^2 (1.25 MHz)	fx/2^3 (625 kHz)	3
1	0	0	0	fx/2^3 (625 kHz)	fx/2^4 (313 kHz)	4
1	0	0	1	fx/2^4 (313 kHz)	fx/2^5 (156 kHz)	5
1	0	1	0	fx/2^5 (156 kHz)	fx/2^6 (78.1 kHz)	6
1	0	1	1	fx/2^6 (78.1 kHz)	fx/2^7 (39.1 kHz)	7
1	1	0	0	fx/2^7 (39.1 kHz)	fx/2^8 (19.5 kHz)	8
1	1	0	1	fx/2^8 (19.5 kHz)	fx/2^9 (9.8 kHz)	9
1	1	1	0	fx/2^9 (9.8 kHz)	fx/2^{10} (4.9 kHz)	10
Other than above				Setting prohibited		

(continued)

Remarks 1. fx: Main system clock oscillation frequency

2. MCS: Oscillation mode select register bit 0

3. n: Value set in TPS0 to TPS3 (1 ≤ n ≤ 11)

4. Figures in parentheses apply to operation with fx = 5.0 MHz.

MDL3	MDL2	MDL1	MDL0	Baud Rate Generator Input Clock Selection	k
0	0	0	0	$f_{SCK}/16$	0
0	0	0	1	$f_{SCK}/17$	1
0	0	1	0	$f_{SCK}/18$	2
0	0	1	1	$f_{SCK}/19$	3
0	1	0	0	$f_{SCK}/20$	4
0	1	0	1	$f_{SCK}/21$	5
0	1	1	0	$f_{SCK}/22$	6
0	1	1	1	$f_{SCK}/23$	7
1	0	0	0	$f_{SCK}/24$	8
1	0	0	1	$f_{SCK}/25$	9
1	0	1	0	$f_{SCK}/26$	10
1	0	1	1	$f_{SCK}/27$	11
1	1	0	0	$f_{SCK}/28$	12
1	1	0	1	$f_{SCK}/29$	13
1	1	1	0	$f_{SCK}/30$	14

Caution When a write is performed to BRGC during a communication operation, baud rate generator output is disrupted and communication cannot be performed normally. Therefore, BRGC must not be written to during a communication operation.

Remark f_{SCK}: 5-bit counter source clock
k: Value set in MDL0 to MDL3 ($0 \leq k \leq 14$)

The baud rate transmit/receive clock generated is either a signal scaled from the main system clock, or a signal scaled from the clock input from the ASCK pin.

(i) Generation of baud rate transmit/receive clock by means of main system clock

The transmit/receive clock is generated by scaling the main system clock. The baud rate generated from the main system clock is obtained with the following expression.

$$[\text{Baud rate}] = \frac{f_{XX}}{2^n \times (k + 16)} \ [\text{Hz}]$$

f_X: Main system clock oscillation frequency
f_{XX}: Main system clock frequency (f_X or $f_X/2$)
n: Value set in TPS0 to TPS3 ($1 \leq n \leq 11$)
k: Value set in MDL0 to MDL3 ($0 \leq k \leq 14$)

Table 17-5. Relationships Between Main System Clock and Baud Rate

Baud Rate (bps)	fx = 5.0 MHz				fx = 4.19 MHz			
	MCS = 1		MCS = 0		MCS = 1		MCS = 0	
	BRGC Set Value	Error (%)	BRGC Set Value	Error (%)	BRGC Set Value	Error (%)	BRGC Set Value	Error (%)
75	—		00H	1.73	0BH	1.14	EBH	1.14
110	06H	0.88	E6H	0.88	03H	–2.01	E3H	–2.01
150	00H	1.73	E0H	1.73	EBH	1.14	DBH	1.14
300	E0H	1.73	D0H	1.73	DBH	1.14	CBH	1.14
600	D0H	1.73	C0H	1.73	CBH	1.14	BBH	1.14
1200	C0H	1.73	B0H	1.73	BBH	1.14	ABH	1.14
2400	B0H	1.73	A0H	1.73	ABH	1.14	9BH	1.14
4800	A0H	1.73	90H	1.73	9BH	1.14	8BH	1.14
9600	90H	1.73	80H	1.73	8BH	1.14	7BH	1.14
19200	80H	1.73	70H	1.73	7BH	1.14	6BH	1.14
31250	74H	0	64H	0	71H	–1.31	61H	–1.31
38400	70H	1.73	60H	1.73	6BH	1.14	5BH	1.14
76800	60H	1.73	50H	1.73	5BH	1.14	—	—

MCS: Oscillation mode select register (OSMS) bit 0

(ii) Generation of baud rate transmit/receive clock by means of external clock from ASCK pin

The transmit/receive clock is generated by scaling the clock input from the ASCK pin. The baud rate generated from the clock input from the ASCK pin is obtained with the following expression.

$$[\text{Baud rate}] = \frac{f_{ASCK}}{2 \times (k + 16)} \text{ [Hz]}$$

f_{ASCK}: Frequency of clock input to ASCK pin

k: Value set in MDL0 to MDL3 ($0 \leq k \leq 14$)

Table 17-6. Relationships Between ASCK Pin Input Frequency and Baud Rate (When BRGC Is Set to 00H)

Baud Rate (bps)	ASCK Pin Input Frequency
75	2.4 kHz
110	3.52 kHz
150	4.8 kHz
300	9.6 kHz
600	19.2 kHz
1200	38.4 kHz
2400	76.8 kHz
4800	153.6 kHz
9600	307.2 kHz
19200	614.4 kHz
31250	1000.0 kHz
38400	1228.8 kHz

(e) Serial interface pin select register (SIPS)

SIPS is set with a 1-bit or 8-bit memory manipulation instruction.

$\overline{\text{RESET}}$ input clears SIPS to 00H.

To select I/O pins, the port mode register and the output latch of the port must be set. For details, refer to **Table 17-2 Serial Interface Channel 2 Operating Mode Settings**.

Symbol	7	6	5	4	3	2	1	0	Address	After Reset	R/W
SIPS	0	0	SIPS21	SIPS20	0	0	0	0	FF75H	00H	R/W

SIPS21	SIPS20	Selects I/O Pin of Asynchronous Serial Interface
0	0	Input pin: RxD/SI2/P70 Output pin: TxD/SO2/P71
0	1	Input pin: RxD/P114 Output pin: TxD/SO2/P71
1	0	Input pin: RxD/SI2/P70 Output pin: TxD/P113
1	1	Input pin: RxD/P114 Output pin: TxD/P113

Cautions 1. Select I/O pins after stopping serial transmission/reception.

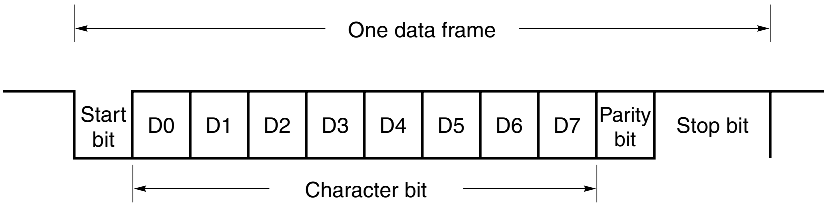
2. Port 11 has a function to detect the falling edge. To use the TxD/P113 or RxD/P114 pin as the I/O pin of serial interface channel 2, the falling edge detection function must be disabled by using the key return mode register (KRM). For details, refer to Figure 6-21 Key Return Mode Register Format.

(2) Communication operation

(a) Data format

Figure 17-8 shows the format of the transmit/receive data.

Figure 17-8. Asynchronous Serial Interface Transmit/Receive Data Format



One data frame consists of the following bits:

• Start bit	1 bit
• Character bits	7 bits/8 bits
• Parity bits	Even parity/odd parity/0 parity/no parity
• Stop bit(s)	1 bit/2 bits

The specification of character bit length, parity selection, and specification of stop bit length for each data frame is carried out with the asynchronous serial interface mode register (ASIM).

When 7 bits are selected as the number of character bits, only the lower 7 bits (bits 0 to 6) are valid; in transmission the most significant bit (bit 7) is ignored, and in reception the most significant bit (bit 7) is always "0".

The serial transfer rate is selected by means of the ASIM and the baud rate generator control register (BRGC).

If a serial data receive error occurs, the receive error contents can be determined by reading the status of the asynchronous serial interface status register (ASIS).

(b) Parity types and operation

The parity bit is used to detect a bit error in the communication data. Normally, the same kind of parity bit is used on the transmitting side and the receiving side. With even parity and odd parity, a one-bit (odd number) error can be detected. With 0 parity and no parity, an error cannot be detected.

(i) Even parity

- **At transmission**

  Control is executed so that the number of bits with a value of "1" contained in the transmit data including parity bit is an even number. The parity bit value should be as follows.

  The number of bits with a value of "1" is an odd number in transmit data: 1
  The number of bits with a value of "1" is an even number in transmit data: 0

- **At reception**

  The number of bits with a value of "1" contained in the receive data including parity bit are counted, and if this is an odd number, a parity error occurs.

(ii) Odd parity

- **At transmission**

  Conversely to the situation with even parity, control is executed so that the number of bits with a value of "1" contained in the transmit data including parity bit is an odd number. The parity bit value should be as follows.

  The number of bits with a value of "1" is an odd number in transmit data: 0
  The number of bits with a value of "1" is an even number in transmit data: 1

- **At reception**

  The number of bits with a value of "1" contained in the receive data including parity bit are counted, and if this is an even number, a parity error occurs.

(iii) 0 Parity

When transmitting, the parity bit is set to "0" irrespective of the transmit data.
At reception, a parity bit check is not performed. Therefore, no parity errors will occur, irrespective of whether the parity bit is set to "0" or "1".

(iv) No parity

A parity bit is not added to the transmit data. At reception, data is received assuming that there is no parity bit. Since there is no parity bit, no parity errors will occur.

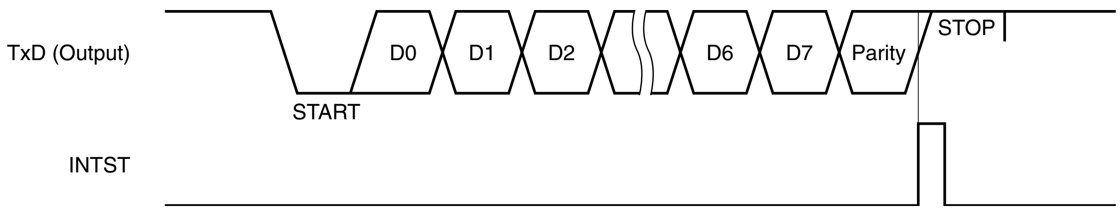
(c) Transmission

A transmit operation is started by writing transmit data to the transmit shift register (TXS). The start bit, parity bit and stop bit(s) are added automatically.

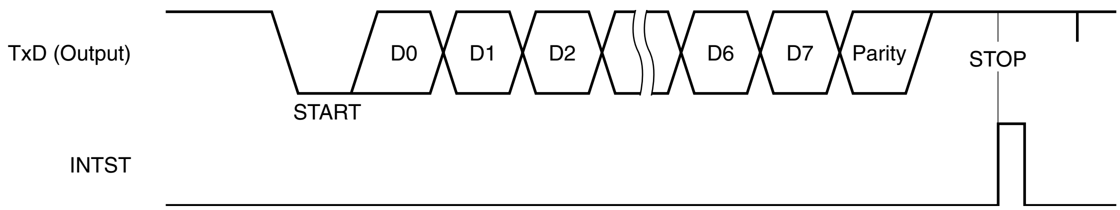
When the transmit operation starts, the data in the TXS is shifted out, and when the TXS is empty, a transmit completion interrupt request (INTST) is generated.

Figure 17-9. Asynchronous Serial Interface Transmit Completion Interrupt Request Timing

(a) Stop bit length: 1



(b) Stop bit length: 2



Caution Rewriting of the asynchronous serial interface mode register (ASIM) should not be performed during a transmit operation. If rewriting of the ASIM register is performed during transmission, subsequent transmit operations may not be possible (the normal state is restored by $\overline{\text{RESET}}$ input).

It is possible to determine whether transmission is in progress by software by using a transmit completion interrupt request (INTST) or the interrupt request flag (STIF) set by the INTST.

(d) Reception

When the RXE bit of the asynchronous serial interface mode register (ASIM) is set (to 1), a receive operation is enabled and sampling of the RxD pin input is performed.

RxD pin input sampling is performed using the serial clock specified by ASIM.

When the RxD pin goes low, the 5-bit counter of the baud rate generator (refer to **Figure 17-2**) starts counting. When the time half the set baud rate has elapsed, a signal to start data sampling is output. If the RxD pin input sampled again as a result of this start timing signal is low, it is identified as a start bit, the 5-bit counter is initialized and starts counting, and data sampling is performed. When character data, a parity bit and one stop bit are detected after the start bit, reception of one frame of data ends.

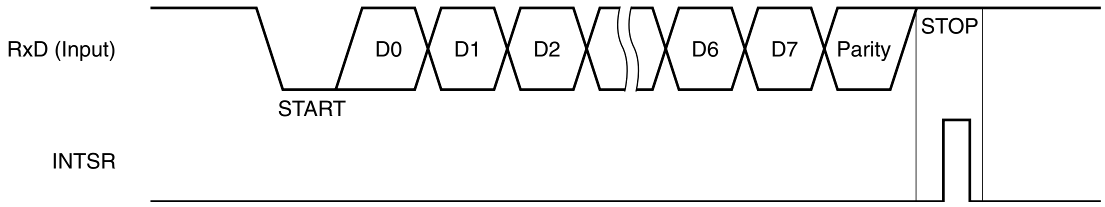
When one frame of data has been received, the receive data in the shift register is transferred to the receive buffer register (RXB), and a receive completion interrupt request (INTSR) is generated.

Even if an error occurs, the receive data responsible for the error is transferred to RXB. If bit 1 (ISRM) of ASIM is cleared to 0 on occurrence of the error, INTSR is generated.

If the ISRM bit is set to 1, INTSR is not generated.

If the RXE bit is reset (to 0) during the receive operation, the receive operation is stopped immediately. In this case, the contents of RXB and asynchronous serial interface status register (ASIS) are not changed, and INTSR and INTSER are not generated.

Figure 17-10. Asynchronous Serial Interface Receive Completion Interrupt Request Timing



Caution The receive buffer register (RXB) must be read even if a receive error occurs. If RXB is not read, an overrun error will occur when the next data is received, and the receive error state will continue indefinitely.

(e) Receive errors

Three kinds of errors can occur during a receive operation: a parity error, framing error, or overrun error. The data reception result error flag is set in the asynchronous serial interface status register (ASIS) and a receive error interrupt (INTSER) is generated. The receive error interrupt occurs earlier than the receive completion interrupt (INTSR). Receive error causes are shown in Table 17-7.

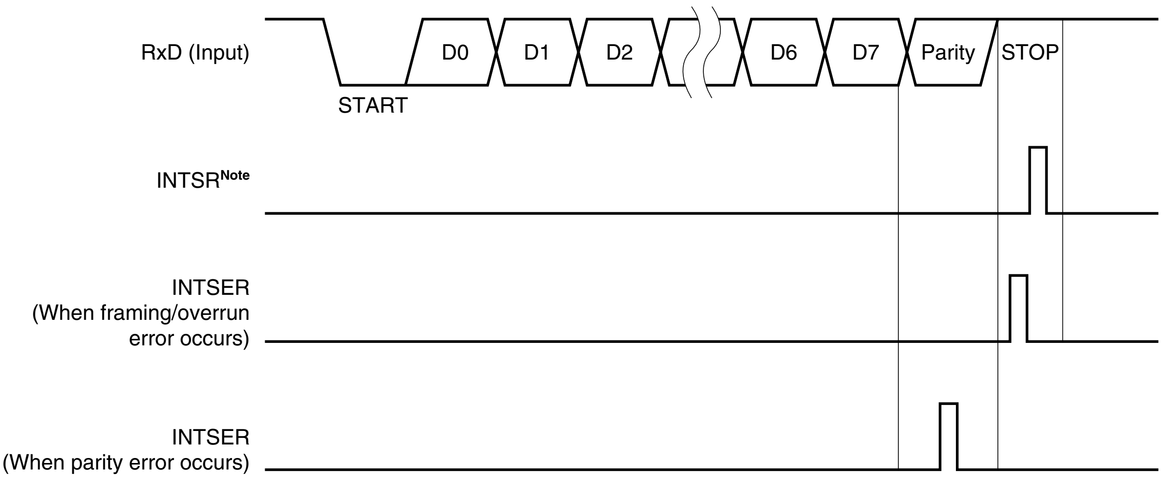
It is possible to determine what kind of error occurred during reception by reading the contents of the ASIS in the receive error interrupt servicing (INTSER) (see **Figures 17-10** and **17-11**).

The contents of ASIS are reset (to 0) by reading the receive buffer register (RXB) or receiving the next data (if there is an error in the next data, the corresponding error flag is set).

Table 17-7. Receive Error Causes

Receive Errors	Cause
Parity error	Transmission-time parity specification and reception data parity do not match
Framing error	Stop bit not detected
Overrun error	Reception of next data is completed before data is read from receive register buffer

Figure 17-11. Receive Error Timing



Note INTSR does not occur if a receive error occurs while bit 1 (ISRM) of the asynchronous serial interface mode register (ASIM) is set to 1.

Cautions 1. The contents of the asynchronous serial interface status register (ASIS) are reset (to 0) by reading the receive buffer register (RXB) or receiving the next data. To ascertain the error contents, ASIS must be read before reading RXB.

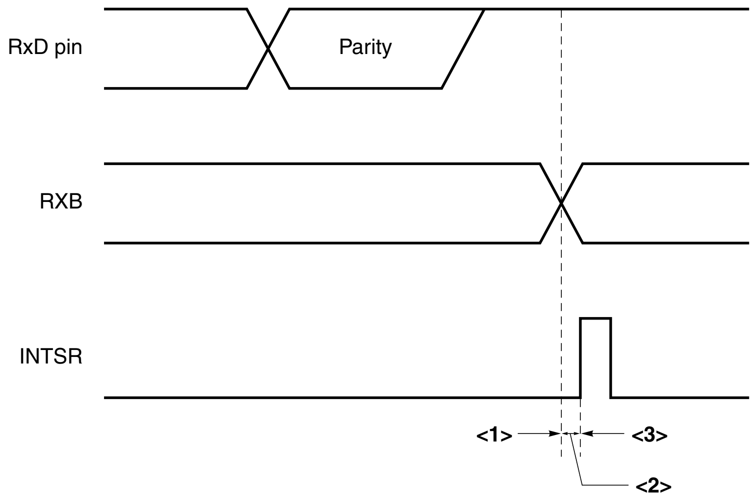
2. The receive buffer register (RXB) must be read even if a receive error occurs. If RXB is not read, an overrun error will occur when the next data is received, and the receive error state will continue indefinitely.

(3) UART mode cautions

(a) If the transmission under execution has been stopped by clearing bit 7 of the asynchronous serial interface mode register (ASIM) to 0, be sure to set the transmit shift register (TXS) to FFH and TXE to 1 before executing the next transmission.

(b) If the reception under execution has been stopped by clearing bit 6 (REX) of the asynchronous serial interface mode register (ASIM) to 0, the status of the receive buffer register (RXB) and whether the receive completion interrupt request (INTSR) occurs differ depending on the reception stop timing.

Figure 17-12. Status of Receive Buffer Register (RXB) and Generation of Interrupt Request (INTSR) When Reception is Stopped



When RXE is set to 0 at a time indicated by **<1>**, RXB holds the previous data and does not generate INTSR.
When RXE is set to 0 at a time indicated by **<2>**, RXB renews the data and does not generate INTSR.
When RXE is set to 0 at a time indicated by **<3>**, RXB renews the data and generates INTSR.

17.4.3 3-wire serial I/O mode

The 3-wire serial I/O mode is useful for connection of peripheral I/Os and display controllers, etc., which incorporate a conventional synchronous clocked serial interface, such as the 75X/XL Series, 78K Series, 17K Series, etc.

Communication is performed using three lines: the serial clock ($\overline{\mathrm{SCK2}}$), serial output (SO2), and serial input (SI2).

In the 3-wire serial I/O mode, the P113/TxD and P114/RxD pins can be used as ordinary I/O port pins.

(1) Register setting

3-wire serial I/O mode settings are performed using serial operating mode register 2 (CSIM2), the asynchronous serial interface mode register (ASIM), and the baud rate generator control register (BRGC).

(a) Serial operating mode register 2 (CSIM2)

CSIM2 is set with a 1-bit or 8-bit memory manipulation instruction.

$\overline{\mathrm{RESET}}$ input clears CSIM2 to 00H.

Symbol	⑦	6	5	4	3	2	1	0	Address	After Reset	R/W
CSIM2	CSIE2	0	0	0	0	CSIM 22	CSCK	0	FF72H	00H	R/W

CSIE2	Operation Control in 3-wire Serial I/O Mode
0	Operation stopped
1	Operation enabled

CSIM22	First Bit Specification
0	MSB
1	LSB

CSCK	Clock Selection in 3-wire Serial I/O Mode
0	Input clock from off-chip to $\overline{\mathrm{SCK2}}$ pin
1	Dedicated baud rate generator output

Caution Ensure that bit 0 and bit 3 to bit 6 are set to 0.

(b) Asynchronous serial interface mode register (ASIM)

ASIM is set with a 1-bit or 8-bit memory manipulation instruction.

$\overline{\text{RESET}}$ input clears ASIM to 00H.

When the 3-wire serial I/O mode is selected, 00H should be set in ASIM.

Symbol	⑦	⑥	5	4	3	2	1	0	Address	After Reset	R/W
ASIM	TXE	RXE	PS1	PS0	CL	SL	ISRM	SCK	FF70H	00H	R/W

TXE	Transmit Operation Control
0	Transmit operation stopped
1	Transmit operation enabled

RXE	Receive Operation Control
0	Receive operation stopped
1	Receive operation enabled

PS1	PS0	Parity Bit Specification
0	0	No parity
0	1	0 parity always added in transmission No parity test in reception (parity errors do not occur)
1	0	Odd parity
1	1	Even parity

CL	Character Length Specification
0	7 bits
1	8 bits

SL	Transmit Data Stop Bit Length Specification
0	1 bit
1	2 bits

ISRM	Control of Receive Completion Interrupt When Error Occurs
0	Receive completion interrupt generated when an error occurs
1	Receive completion interrupt not generated when an error occurs

SCK	Clock Selection in Asynchronous Serial Interface Mode
0	Input clock from off-chip to ASCK pin
1	Dedicated baud rate generator output

(c) Baud rate generator control register (BRGC)

BRGC is set with an 8-bit memory manipulation instruction.

$\overline{\text{RESET}}$ input clears BRGC to 00H.

Symbol	7	6	5	4	3	2	1	0	Address	After Reset	R/W
BRGC	TPS3	TPS2	TPS1	TPS0	MDL3	MDL2	MDL1	MDL0	FF73H	00H	R/W

TPS3	TPS2	TPS1	TPS0	Selects Source Clock of 5-bit Counter		n
				MCS = 1	MCS = 0	
0	0	0	0	$f_x/2^{10}$ (4.9 kHz)	$f_x/2^{11}$ (2.4 kHz)	11
0	1	0	1	f_x (5.0 MHz)	$f_x/2$ (2.5 MHz)	1
0	1	1	0	$f_x/2$ (2.5 MHz)	$f_x/2^2$ (1.25 MHz)	2
0	1	1	1	$f_x/2^2$ (1.25 MHz)	$f_x/2^3$ (625 kHz)	3
1	0	0	0	$f_x/2^3$ (625 kHz)	$f_x/2^4$ (313 kHz)	4
1	0	0	1	$f_x/2^4$ (313 kHz)	$f_x/2^5$ (156 kHz)	5
1	0	1	0	$f_x/2^5$ (156 kHz)	$f_x/2^6$ (78.1 kHz)	6
1	0	1	1	$f_x/2^6$ (78.1 kHz)	$f_x/2^7$ (39.1 kHz)	7
1	1	0	0	$f_x/2^7$ (39.1 kHz)	$f_x/2^8$ (19.5 kHz)	8
1	1	0	1	$f_x/2^8$ (19.5 kHz)	$f_x/2^9$ (9.8 kHz)	9
1	1	1	0	$f_x/2^9$ (9.8 kHz)	$f_x/2^{10}$ (4.9 kHz)	10
Other than above				Setting prohibited		

(continued)

Remarks 1. f_x: Main system clock oscillation frequency

2. MCS: Oscillation mode select register bit 0

3. n: Value set in TPS0 to TPS3 ($1 \le n \le 11$)

4. Figures in parentheses apply to operation with f_x = 5.0 MHz.

MDL3	MDL2	MDL1	MDL0	Baud Rate Generator Input Clock Selection	k
0	0	0	0	$f_{SCK}/16$	0
0	0	0	1	$f_{SCK}/17$	1
0	0	1	0	$f_{SCK}/18$	2
0	0	1	1	$f_{SCK}/19$	3
0	1	0	0	$f_{SCK}/20$	4
0	1	0	1	$f_{SCK}/21$	5
0	1	1	0	$f_{SCK}/22$	6
0	1	1	1	$f_{SCK}/23$	7
1	0	0	0	$f_{SCK}/24$	8
1	0	0	1	$f_{SCK}/25$	9
1	0	1	0	$f_{SCK}/26$	10
1	0	1	1	$f_{SCK}/27$	11
1	1	0	0	$f_{SCK}/28$	12
1	1	0	1	$f_{SCK}/29$	13
1	1	1	0	$f_{SCK}/30$	14
1	1	1	1	f_{SCK}	—

Caution When a write is performed to BRGC during a communication operation, baud rate generator output is disrupted and communication cannot be performed normally. Therefore, BRGC must not be written to during a communication operation.

Remark f_{SCK}: 5-bit counter source clock
k: Value set in MDL0 to MDL3 ($0 \leq k \leq 14$)

When the internal clock is used as the serial clock in the 3-wire serial I/O mode, set BRGC as described below. BRGC setting is not required if an external serial clock is used.

(i) When the baud rate generator is not used:

Select a serial clock frequency with TPS0 to TPS3. Be sure then to set MDL0 to MDL3 to 1, 1, 1, 1. The serial clock frequency is half the source clock frequency of the 5-bit counter.

(ii) When the baud rate generator is used:

Select a serial clock frequency with MDL0 to MDL3 and TPS0 to TPS3. Be sure then to set MDL0 to MDL3 to a value other than 1, 1, 1, 1.

The serial clock frequency is calculated by the following formula:

$$\text{Serial clock frequency} = \frac{f_{XX}}{2^n \times (k + 16)} \text{ [Hz]}$$

f_X:	Main system clock oscillation frequency
f_{XX}:	Main system clock frequency (f_X or $f_X/2$)
n:	Value set in TPS0 to TPS3 ($1 \leq n \leq 11$)
k:	Value set in MDL0 to MDL3 ($0 \leq k \leq 14$)

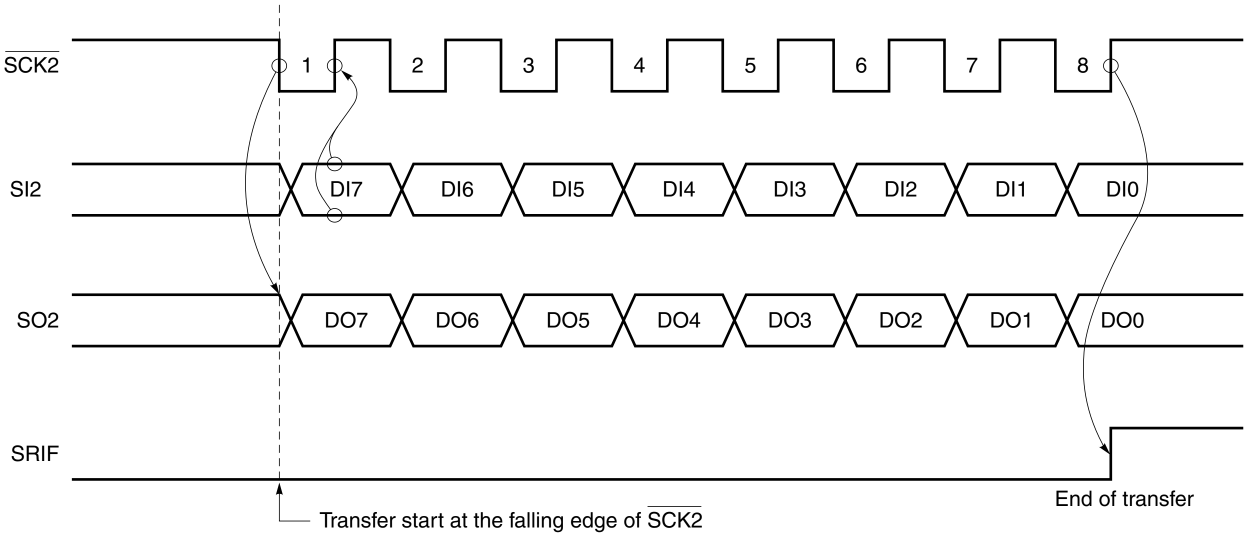
(2) Communication operation

In the 3-wire serial I/O mode, data transmission/reception is performed in 8-bit units. Each bit of data is transmitted or received in synchronization with the serial clock.

Transmit shift register (TXS/SIO2) and receive shift register (RXS) shift operations are performed in synchronization with the falling edge of the serial clock ($\overline{\text{SCK2}}$). Then transmit data is held in the SO2 latch and output from the SO2 pin. Also, receive data input to the SI2 pin is latched in the receive buffer register (RXB/SIO2) at the rising edge of $\overline{\text{SCK2}}$.

At the end of an 8-bit transfer, the operation of the transmit shift register (TXS/SIO2) or receive shift register (RXS) stops automatically, and the interrupt request flag (SRIF) is set.

Figure 17-13. 3-Wire Serial I/O Mode Timing



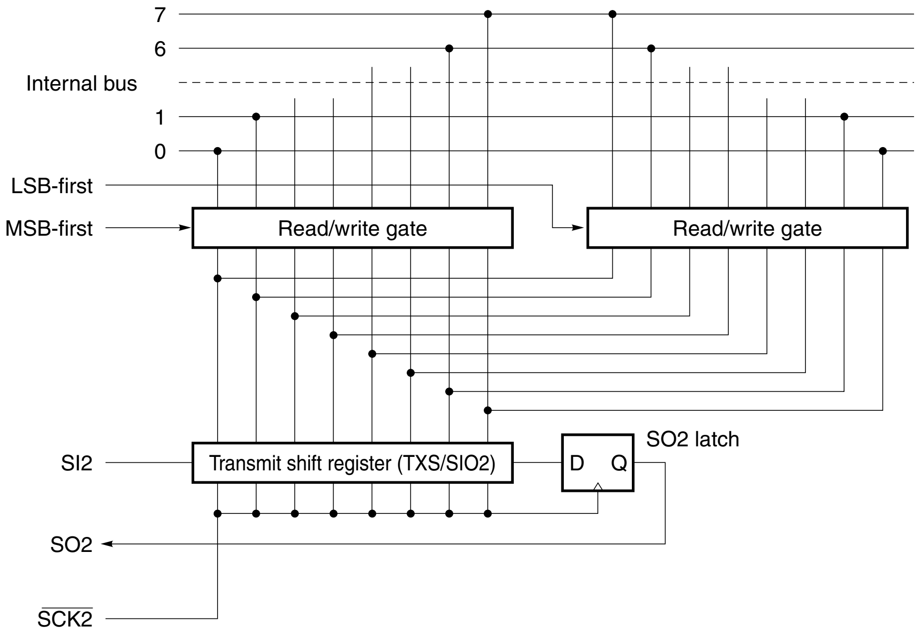
(3) MSB/LSB switching as the start bit

The 3-wire serial I/O mode enables to select transfer to start from MSB or LSB.

Figure 17-14 shows the configuration of the transmit shift register (TXS/SIO2) and internal bus. As shown in the figure, MSB/LSB can be read/written in reverse form.

MSB/LSB switching as the start bit can be specified with bit 2 (CSIM22) of serial operating mode register 2 (CSIM2).

Figure 17-14. Circuit of Switching in Transfer Bit Order



Start bit switching is realized by switching the bit order for data write to SIO2. The SIO2 shift order remains unchanged.

Thus, switching between MSB-first and LSB-first must be performed before writing data to the shift register.

(4) Transfer start

Serial transfer is started by setting transfer data to the transmit shift register (TXS/SIO2) when the following two conditions are satisfied.

- Serial interface channel 2 operation control bit (CSIE2) = 1
- Internal serial clock is stopped or $\overline{\text{SCK2}}$ is at high level after 8-bit serial transfer.

Caution   If CSIE2 is set to "1" after data write to TXS/SIO2, transfer does not start.

Upon termination of 8-bit transfer, serial transfer automatically stops and the interrupt request flag (SRIF) is set.

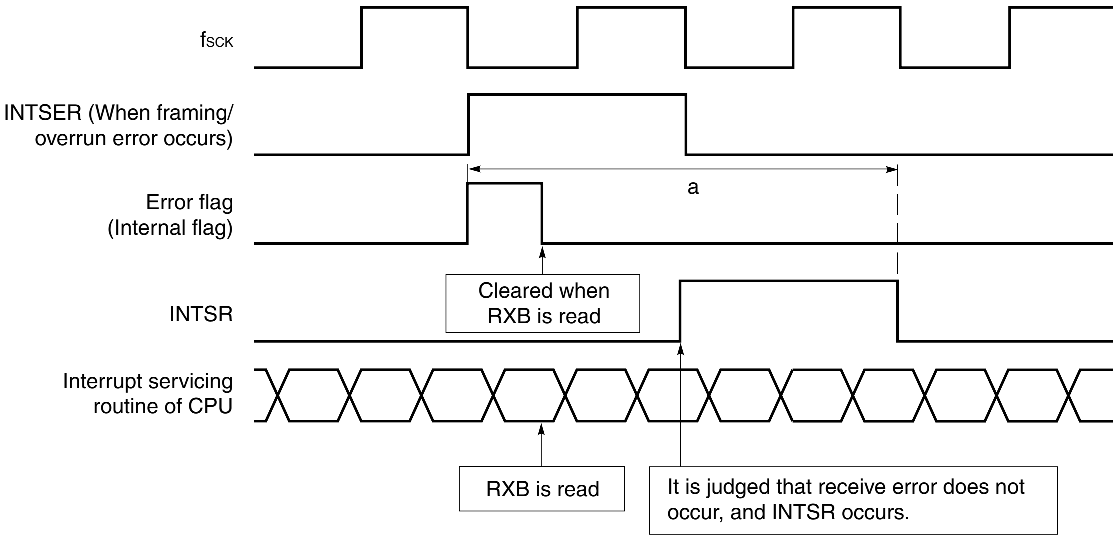
17.4.4 Limitations of UART mode

In the UART mode, the receive completion interrupt (INTSR) occurs a certain time after the receive error interrupt (INTSER) occurred and cleared. Consequently, the following phenomenon may take place.

- **Description**

  If bit 1 (ISRM) of the asynchronous serial interface mode register (ASIM) is set to 1, the receive completion interrupt (INTSR) does not occur when a receive error occurs. If the receive buffer register (RXB) is read at certain timing (a in Figure 17-15) while the receive error interrupt (INTSER) is serviced, the internal error flag is cleared to 0. Therefore, it is judged that the receive error does not occur, and INTSR, which must not occur, occurs. This is illustrated in Figure 17-15.

Figure 17-15. Receive Completion Interrupt Generation Timing (When ISRM = 1)


f_{SCK}
INTSER (When framing/ overrun error occurs)
a
Error flag (Internal flag)
Cleared when RXB is read
INTSR
Interrupt servicing routine of CPU
RXB is read
It is judged that receive error does not occur, and INTSR occurs.

Remark ISRM: Bit 1 of asynchronous serial interface mode register (ASIM)

f_{SCK}: Source clock of 5-bit counter of baud rate generator

RXB: Receive buffer register

To prevent this phenomenon, take the following measures:

- **Preventive measures**

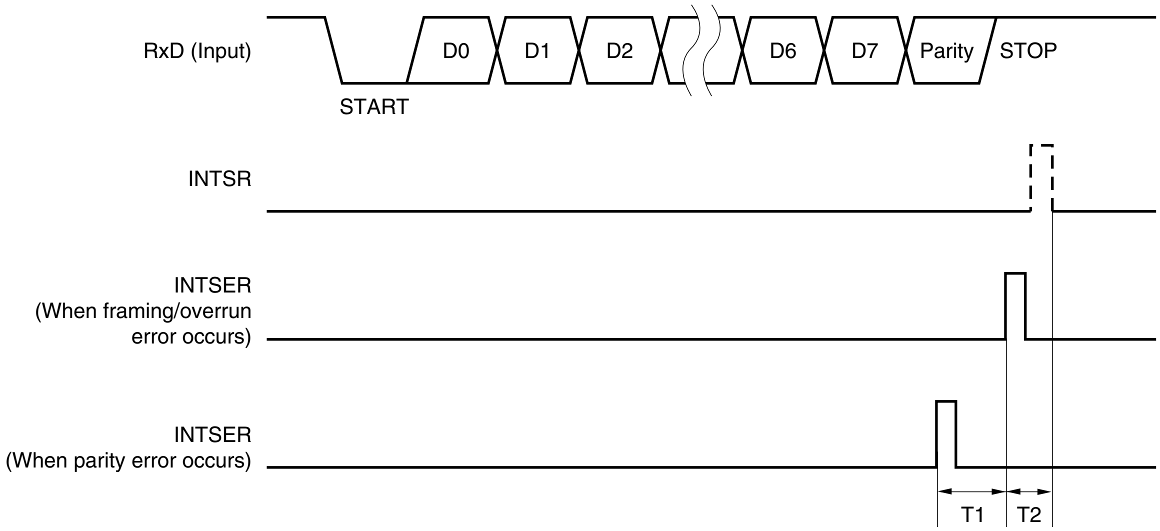
  - **In case of framing error or overrun error**

    Disable reading the receive buffer register (RXB) for a certain time (T2 in Figure 17-16) after the receive error interrupt (INTSER) has occurred.

  - **In case of parity error**

    Disable reading the receive buffer register (RXB) for a certain time (T1 + T2 in Figure 17-16) after the receive error interrupt (INTSER) has occurred.

Figure 17-16. Disabling Reading Receive Buffer Register


RxD (Input)
START
D0
D1
D2
D6
D7
Parity
STOP
INTSR
INTSER
(When framing/overrun error occurs)
INTSER
(When parity error occurs)
T1
T2

T1: Time of one data of baud rate selected by baud rate generator control register (BRGC) (1/baud rate)

T2: Time of two clocks of source clock (f_{SCK}) of 5-bit counter selected by BRGC

- **Example of preventive measures**

  Here is an example of preventive measures.

[Condition]

f_X = 5.0 MHz

Processor clock control register (PCC) = 00H

Oscillation mode select register (OSMS) = 01H

Baud rate generator control register (BRGC) = B0H (baud rate: 2400 bps)

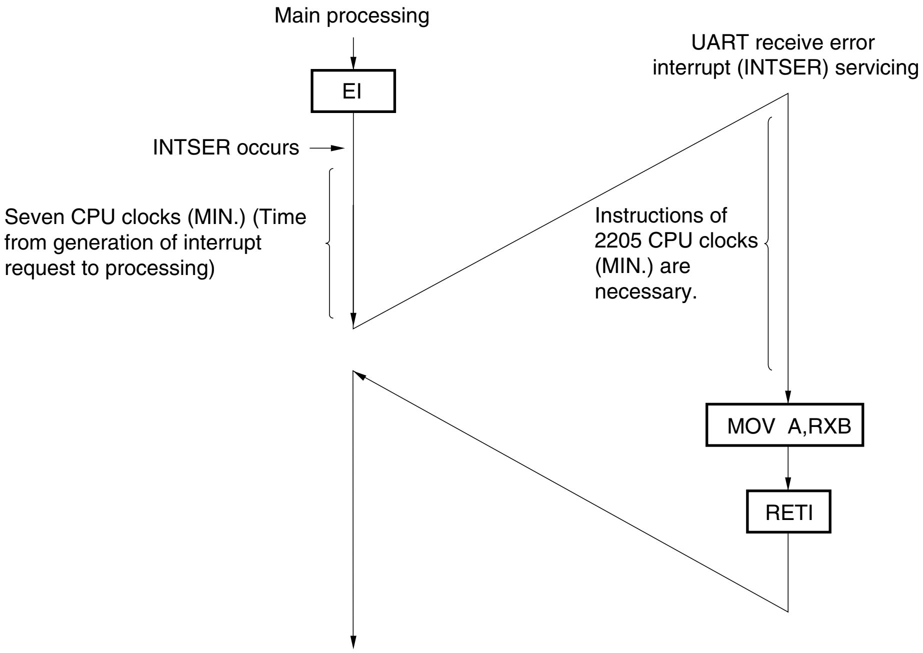
$T_{CY} = 0.4\ \mu s$ ($t_{CY} = 0.2\ \mu s$)

$$T1 = \frac{1}{2400} = 416.7\ \mu s$$

$$T2 = 12.8 \times 2 = 25.6\ \mu s$$

$$\frac{T1 + T2}{t_{CY}} = 2212 \text{ (clock)}$$

[Example]


Main processing
EI
INTSER occurs
Seven CPU clocks (MIN.) (Time from generation of interrupt request to processing)
UART receive error interrupt (INTSER) servicing
Instructions of 2205 CPU clocks (MIN.) are necessary.
MOV A,RXB
RETI

CHAPTER 18 SERIAL INTERFACE CHANNEL 3

18.1 Serial Interface Channel 3 Functions

Serial interface channel 3 operates in the following two modes.

- Operation stop mode
- 3-wire serial I/O mode

(1) Operation stop mode

This mode is used when serial interface channel 3 does not perform serial transfer, to reduce the power consumption.

(2) 3-wire serial I/O mode (MSB/LSB first selectable)

In this mode, 8-bit data are transferred by using three lines: serial clock ($\overline{\text{SCK3}}$), serial output (SO3), and serial input (SI3).

Because transmission and reception can be carried out simultaneously in this mode, the data transfer time can be shortened.

Because whether the 8-bit data to be transferred is transferred starting from its MSB or LSB can be selected in this mode, serial interface channel 3 can be connected to any device.

The 3-wire serial I/O mode is effective for connecting peripheral I/Os and display controllers that have the conventional clocked serial interface, such as the 75X/XL Series, 78K Series, and 17K Series.

18.2 Serial Interface Channel 3 Configuration

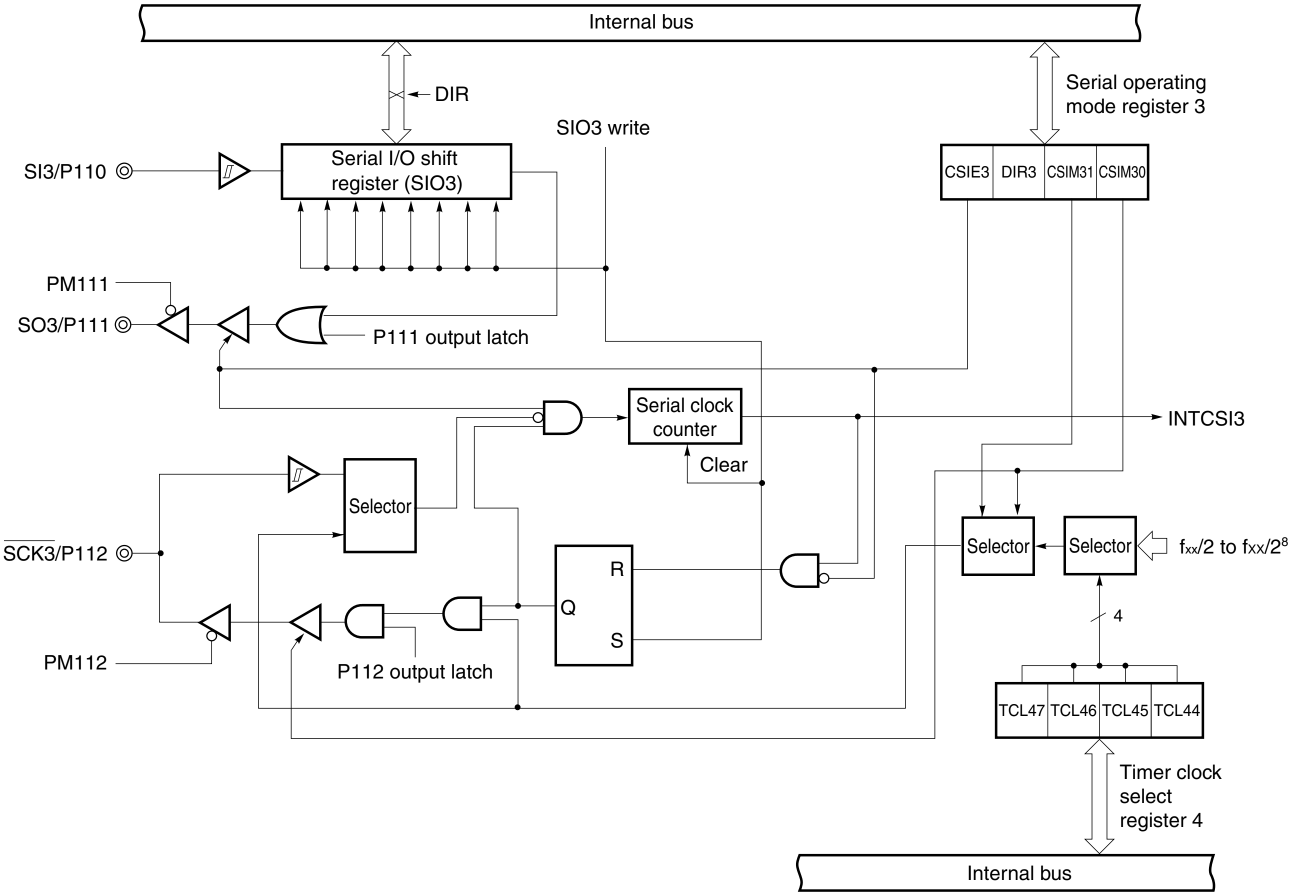
Serial interface channel 3 consists of the following hardware.

Table 18-1. Configuration of Serial Interface Channel 3

Item	Configuration
Register	Serial I/O shift register 3 (SIO3)
Control register	Timer clock select register 4 (TCL4) Serial operating mode register 3 (CSIM3) Port mode register 11 (PM11)[Note]

Note Refer to **Figure 6-15 P110, P114 to P117 Block Diagram** and **Figure 6-16 P111 Block Diagram**.

Figure 18-1. Serial Interface Channel 3 Block Diagram



Remark fxx = fx/2 (MCS = 0), fxx = fx (MCS = 1)

(1) Serial I/O shift register 3 (SIO3)

This is an 8-bit register that performs parallel-to-serial conversion and serial transmission/reception (shift operation) in synchronization with the serial clock.

SIO3 is set with an 8-bit memory manipulation instruction.

Serial operation is started by writing data to SIO3 when bit 7 (CSIE3) of serial operating mode register 3 (CSIM3) is 1.

During transmission, the data written to SIO3 is output to the serial output line (SO3). During reception, data is read from the serial input line (SI3) to SIO3.

$\overline{\text{RESET}}$ input makes SIO3 undefined.

(2) Serial clock counter

This counter counts the serial clock output or input during transmission or reception to check whether 8-bit data has been transmitted or received.

18.3 Serial Interface Channel 3 Control Registers

Serial interface channel 3 is controlled by the following two registers.

- Timer clock select register 4 (TCL4)
- Serial operating mode register 3 (CSIM3)

(1) Timer clock select register 4 (TCL4)

This register sets the serial clock of serial interface channel 3.

TCL4 is set with an 8-bit memory manipulation instruction.

$\overline{\text{RESET}}$ input sets TCL4 to 88H.

Figure 18-2. Timer Clock Select Register 4 Format

Symbol	7	6	5	4	3	2	1	0	Address	After Reset	R/W
TCL4	TCL47	TCL46	TCL45	TCL44	1	0	0	0	FF44H	88H	R/W

TCL47	TCL46	TCL45	TCL44	Selects Serial Clock of Serial Interface Channel 3	
				MCS = 1	MCS = 0
0	1	1	0	Setting prohibited	$f_X/2^2$ (1.25 MHz)
0	1	1	1	$f_X/2^2$ (1.25 MHz)	$f_X/2^3$ (625 kHz)
1	0	0	0	$f_X/2^3$ (625 kHz)	$f_X/2^4$ (313 kHz)
1	0	0	1	$f_X/2^4$ (313 kHz)	$f_X/2^5$ (156 kHz)
1	0	1	0	$f_X/2^5$ (156 kHz)	$f_X/2^6$ (78.1 kHz)
1	0	1	1	$f_X/2^6$ (78.1 kHz)	$f_X/2^7$ (39.1 kHz)
1	1	0	0	$f_X/2^7$ (39.1 kHz)	$f_X/2^8$ (19.5 kHz)
1	1	0	1	$f_X/2^8$ (19.5 kHz)	$f_X/2^9$ (9.8 kHz)
Other than above				Setting prohibited	

Cautions 1. Set bit 0 to bit 2 to 0, and bit 3 to 1.

2. When rewriting TCL4 to other data, stop the serial transfer operation beforehand.

Remarks 1. f_X: Main system clock oscillation frequency

2. MCS: Oscillation mode select register bit 0

3. Figures in parentheses apply to operation with f_X = 5.0 MHz.

(2) Serial operating mode register 3 (CSIM3)

This register sets the serial clock of serial interface channel 3, and enables or disables the operation of the interface channel.

CSIM3 is set with a 1-bit or 8-bit memory manipulation instruction.

$\overline{\text{RESET}}$ input clears CSIM3 to 00H.

Figure 18-3. Serial Operating Mode Register 3 Format

Symbol	⑦	6	5	4	3	2	1	0	Address	After Reset	R/W
CSIM3	CSIE3	DIR3	Note 1 0	0	0	0	CSIM31	CSIM30	FF6CH	00H	R/W

CSIE3	CSIM31	PM110	P110	PM111	P111	PM112	P112	Controls Operation of Serial Interface Channel 3	Controls Operation of Serial Clock Counter	SI3/P110 Pin Functions	SO3/P111 Pin Functions	$\overline{\text{SCK3}}$/P112 Pin Functions
0	×	Note 2 ×	Note 2 ×	Note 2 ×	Note 2 ×	Note 2 ×	Note 2 ×	Stops operation	Clear	P110 (CMOS I/O)	P111 (CMOS I/O)	P112 (CMOS I/O)
1	0	Note 3 1	Note 3 ×	0	0	1	×	Enables operation	Count operation	SI3[Note 3] (Input)	SO3 (CMOS output)	$\overline{\text{SCK3}}$ (Input)
	1					0	1					$\overline{\text{SCK3}}$ (CMOS output)

DIR3	First Bit	SI3/P110 Pin Functions	SO3/P111 Pin Functions
0	MSB	SI3[Note 3] (input)	SO3 (CMOS output)
1	LSB		

CSIM31	CSIM30	Selects Clock of Serial Interface Channel 3
0	×	Input clock to $\overline{\text{SCK3}}$ pin from off-chip
1	1	Clock specified with bits 0 to 3 of timer clock select register 4 (TCL4)
Other than above		Setting prohibited

Notes **1.** Be sure to clear bit 5 to 0.

2. These pins can be used freely as port pins.

3. This pin can be used as P110 (CMOS I/O) when only transmitting data.

Caution Port 11 has a function to detect the falling edge. To use the SI3/P110, SO3/P111, and $\overline{\text{SCK3}}$/P112 pins as the I/O pins of serial interface channel 3, the falling edge detection function must be disabled by using the key return mode register (KRM). For details, refer to Figure 6-21 Key Return Mode Register Format.

Remark ×: don’t care

PM××: Port mode register

P××: Port output latch

18.4 Serial Interface Channel 3 Operation

The operating mode of serial interface channel 3 has the following two types.

- Operation stop mode
- 3-wire serial I/O mode

18.4.1 Operation stop mode

In the operation stop mode, serial transfer is not performed, and therefore current consumption can be reduced.

In addition, serial I/O shift register 3 (SIO3) does not perform the shift operation, and therefore, this register can be used as an ordinary 8-bit register.

In the operation stop mode, the P110/SI3, P111/SO3, and P112/$\overline{\text{SCK3}}$ pins can be used as normal I/O ports.

(1) Register setting

Operation stop mode settings are performed using serial operating mode register 3 (CSIM3).

CSIM3 is set with a 1-bit or 8-bit memory manipulation instruction.

$\overline{\text{RESET}}$ input clears CSIM3 to 00H.

Symbol	⑦	6	5	4	3	2	1	0	Address	After Reset	R/W
CSIM3	CSIE3	DIR3	Note 1 0	0	0	0	CSIM31	CSIM30	FF6CH	00H	R/W

CSIE3	CSIM31	PM110	P110	PM111	P111	PM112	P112	Controls Operation of Serial Interface Channel 3	Controls Operation of Serial Clock Counter	SI3/P110 Pin Functions	SO3/P111 Pin Functions	$\overline{\text{SCK3}}$/P112 Pin Functions
0	×	Note 2 ×	Note 2 ×	Note 2 ×	Note 2 ×	Note 2 ×	Note 2 ×	Stops operation	Clear	P110 (CMOS I/O)	P111 (CMOS I/O)	P112 (CMOS I/O)
1	0	Note 3 1	Note 3 ×	0	0	1	×	Enables operation	Count operation	SI3[Note 3] (Input)	SO3 (CMOS output)	$\overline{\text{SCK3}}$ (Input)
	1					0	1					$\overline{\text{SCK3}}$ (CMOS output)

Notes 1. Be sure to clear bit 5 to 0.

2. These pins can be used freely as port pins.

3. This pin can be used as P110 (CMOS I/O) when only transmitting data.

Caution Port 11 has a function to detect the falling edge. To use the SI3/P110, SO3/P111, and $\overline{\text{SCK3}}$/P112 pins as the I/O pins of serial interface channel 3, the falling edge detection function must be disabled by using the key return mode register (KRM). For details, refer to Figure 6-21 Key Return Mode Register Format.

Remark ×: don’t care

PM××: Port mode register

P××: Port output latch

18.4.2 3-wire serial I/O mode

The 3-wire serial I/O mode is useful for connection of peripheral I/Os and display controllers, etc., which incorporate a conventional synchronous clocked serial interface, such as the 75X/XL Series, 78K Series, and 17K Series, etc.

In this mode, communication is performed by using three lines: serial clock ($\overline{\text{SCK3}}$), serial output (SO3), and serial input (SI3).

(1) Register setting

The 3-wire serial I/O mode is set by using serial operating mode register 3 (CSIM3).

CSIM3 is set with a 1-bit or 8-bit memory manipulation instruction.

$\overline{\text{RESET}}$ input clears CSIM3 to 00H.

Symbol	⑦	6	5	4	3	2	1	0	Address	After Reset	R/W
CSIM3	CSIE3	DIR3	Note 1 0	0	0	0	CSIM31	CSIM30	FF6CH	00H	R/W

CSIE3	CSIM31	PM110	P110	PM111	P111	PM112	P112	Controls Operation of Serial Interface Channel 3	Controls Operation of Serial Clock Counter	SI3/P110 Pin Functions	SO3/P111 Pin Functions	$\overline{\text{SCK3}}$/P112 Pin Functions
0	×	Note 2 ×	Note 2 ×	Note 2 ×	Note 2 ×	Note 2 ×	Note 2 ×	Stops operation	Clear	P110 (CMOS I/O)	P111 (CMOS I/O)	P112 (CMOS I/O)
1	0	Note 3 1	Note 3 ×	0	0	1	×	Enables operation	Count operation	SI3[Note 3] (Input)	SO3 (CMOS output)	$\overline{\text{SCK3}}$ (Input)
	1					0	1					$\overline{\text{SCK3}}$ (CMOS output)

DIR3	First Bit	SI3/P110 Pin Functions	SO3/P111 Pin Functions
0	MSB	SI3[Note 3] (Input)	SO3 (CMOS output)
1	LSB		

CSIM31	CSIM30	Selects Clock of Serial Interface Channel 3
0	×	Input clock to $\overline{\text{SCK3}}$ pin from off-chip
1	1	Clock specified with bits 0 to 3 of timer clock select register 4 (TCL4)
Other than above		Setting prohibited

Notes **1.** Be sure to clear bit 5 to 0.
2. These pins can be used freely as port pins.
3. This pin can be used as P110 (CMOS I/O) when only transmitting data.

Caution Port 11 has a function to detect the falling edge. To use the SI3/P110, SO3/P111, and $\overline{\text{SCK3}}$/P112 pins as the I/O pins of serial interface channel 3, the falling edge detection function must be disabled by using the key return mode register (KRM). For details, refer to Figure 6-21 Key Return Mode Register Format.

Remark ×: don't care
PM××: Port mode register
P××: Port output latch

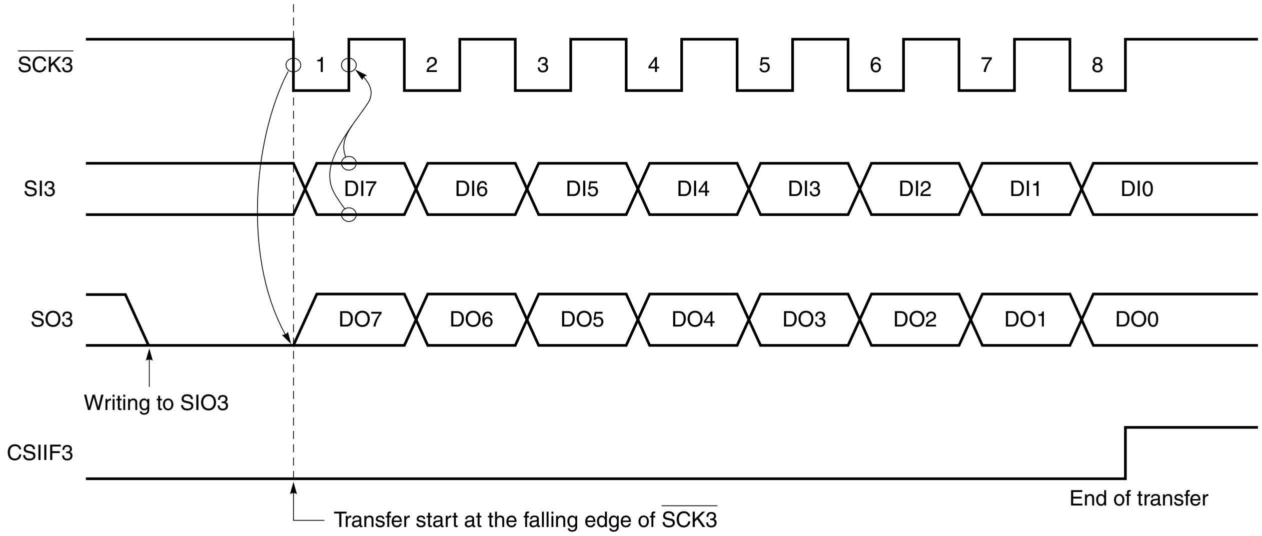
(2) Communication operation

In the 3-wire serial I/O mode, data transmission/reception is performed in 8-bit units. Each bit of data is transmitted or received in synchronization with the serial clock.

Serial I/O shift register 3 (SIO3) performs its shift operation in synchronization with the falling edge of the serial clock ($\overline{\text{SCK3}}$). The transmit data is held in the SO3 latch, and output from the SO3 pin. Also, receive data input to the SI3 pin is latched to SIO3 at the rising edge of $\overline{\text{SCK3}}$.

At the end of an 8-bit transfer, the operation of the SIO3 stops automatically, and the interrupt request flag (CSIIF3) is set.

Figure 18-4. 3-Wire Serial I/O Mode Timing



Caution Do not set 0 to CSIE3 during serial transfer; otherwise, an undefined value will be output.

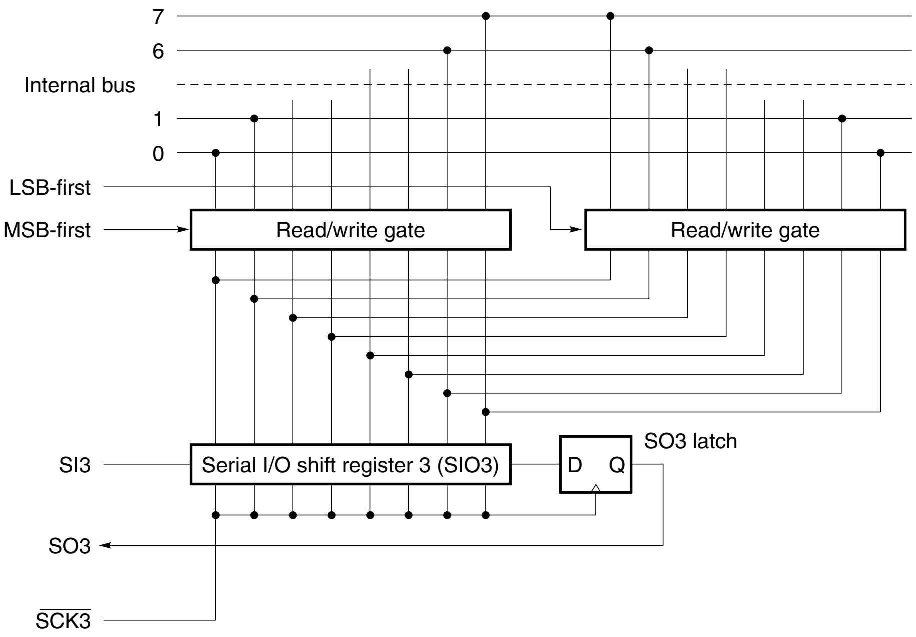
(3) MSB/LSB switching as the start bit

The 3-wire serial I/O mode enables to select transfer to start from MSB or LSB.

Figure 18-5 shows the configuration of serial I/O shift register 3 (SIO3) and internal bus. As shown in the figure, MSB/LSB can be read/written in reverse form.

MSB/LSB switching as the start bit can be specified with bit 2 (CSIM32) of serial operating mode register 3 (CSIM3).

Figure 18-5. Circuit of Switching in Transfer Bit Order



Start bit switching is realized by switching the bit order for data write to SIO3. The SIO3 shift order remains unchanged.

Thus, switching between MSB-first and LSB-first must be performed before writing data to the shift register.

(4) Transfer start

Serial transfer is started by setting transfer data to serial I/O shift register 3 (SIO3) when the following two conditions are satisfied.

- Serial interface channel 3 operation control bit (CSIE3) = 1
- Internal serial clock is stopped or $\overline{SCK3}$ is at high level after 8-bit serial transfer.

Caution   If CSIE3 is set to "1" after data write to SIO3, transfer does not start.

Upon termination of 8-bit transfer, serial transfer automatically stops and the interrupt request flag (CSIE3) is set.

CHAPTER 19 LCD CONTROLLER/DRIVER

19.1 LCD Controller/Driver Functions

The functions of the LCD controller/driver incorporated in the μPD780308, 780308Y Subseries are shown below.

(1) Automatic output of segment signals and common signals is possible by automatic reading of the display data memory.

(2) Any of five display modes can be selected.
- Static
- 1/2 duty (1/2 bias)
- 1/3 duty (1/2 bias)
- 1/3 duty (1/3 bias)
- 1/4 duty (1/3 bias)

(3) Any of four frame frequencies can be selected in each display mode.

(4) Maximum of 40 segment signal outputs (S0 to S39); 4 common signal outputs (COM0 to COM3). Sixteen of the segment signal outputs can be switched to I/O ports in units of 2 (P80/S39 to P87/S32, P90/S31 to P97/S24).

(5) In mask ROM versions, split resistors for LCD drive voltage generation can be incorporated by mask option.

(6) Operation on the subsystem clock is also possible.

The maximum number of displayable pixels in each display mode is shown in Table 19-1.

Table 19-1. Maximum Number of Display Pixels

Bias Method	Time Division	Common Signals Used	Maximum Number of Pixels	Note
—	Static	COM0 (COM1, 2, 3)	40 (40 segments × 1 common)	1
1/2	2	COM0, COM1	80 (40 segments × 2 commons)	2
	3	COM0 to COM2	120 (40 segments × 3 commons)	3
1/3	3			
	4	COM0 to COM3	160 (40 segments × 4 commons)	4

Notes 1. 5 digits on 8. type LCD panel with 8 segments/digit.
2. 10 digits on 8. type LCD panel with 4 segments/digit.
3. 13 digits on 8. type LCD panel with 3 segments/digit.
4. 20 digits on 8. type LCD panel with 2 segments/digit.

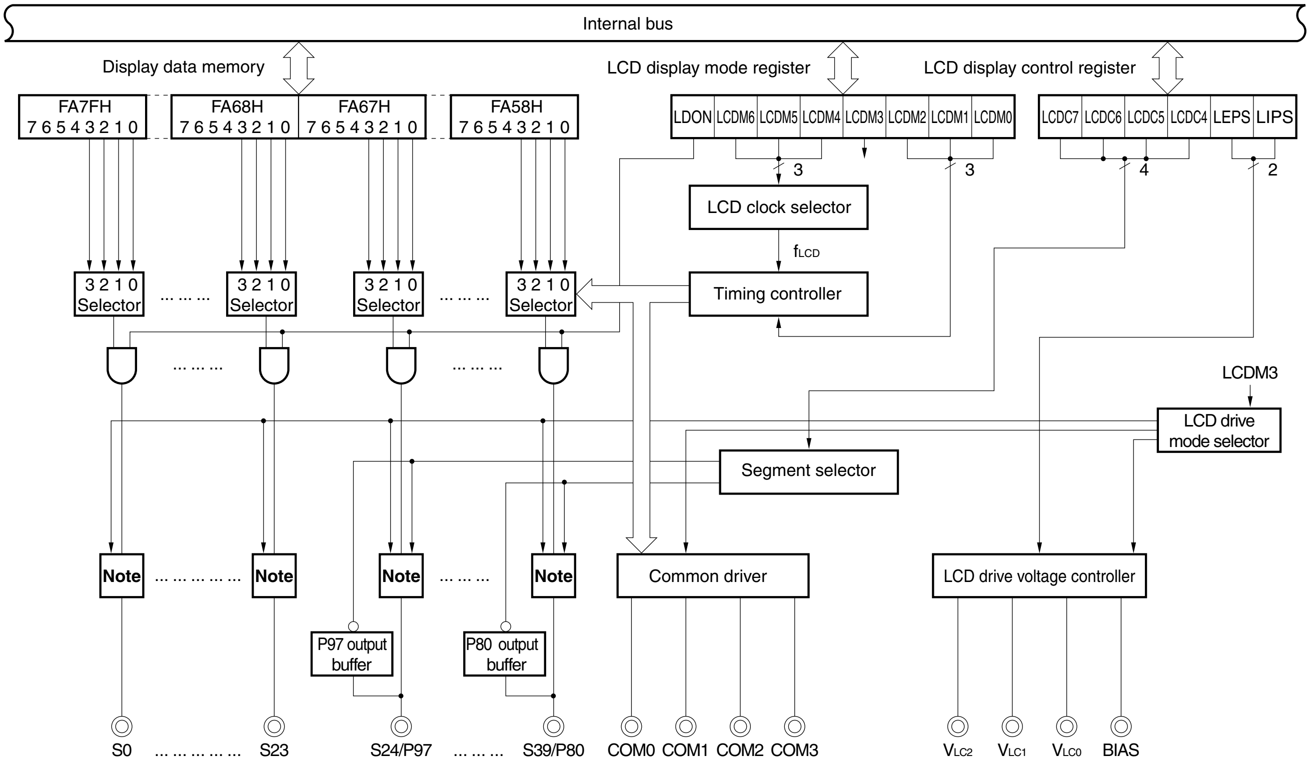
19.2 LCD Controller/Driver Configuration

The LCD controller/driver consists of the following hardware.

Table 19-2. LCD Controller/Driver Configuration

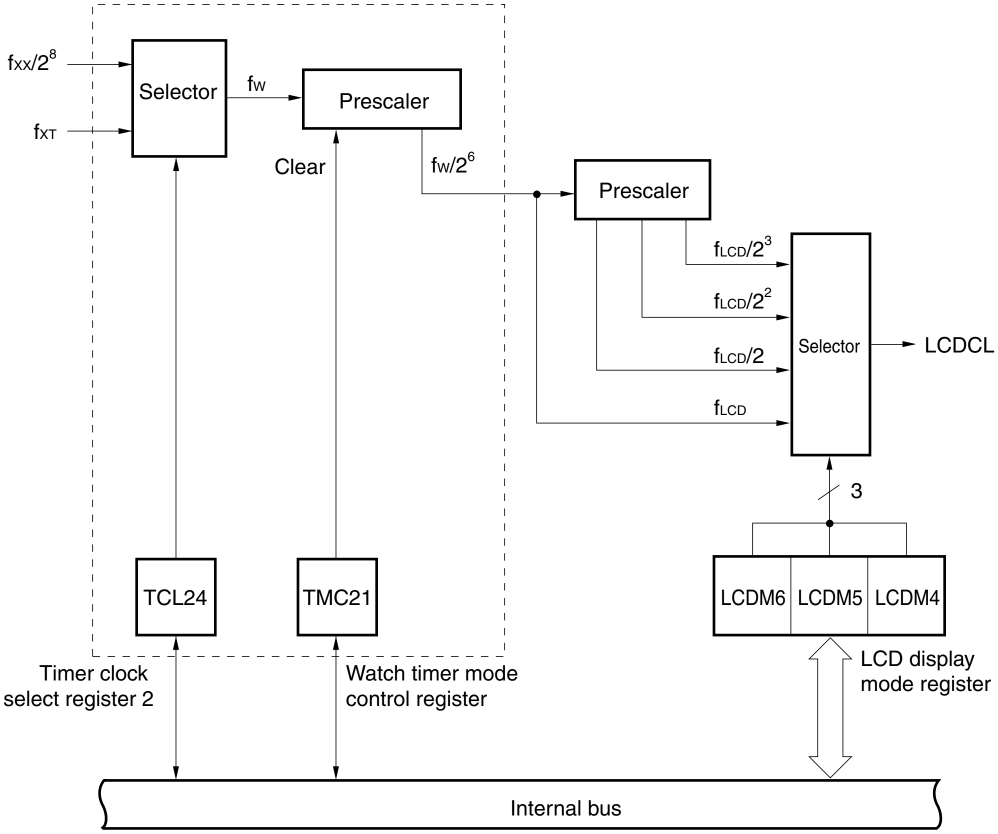
Item	Configuration
Display outputs	Segment signals: 40 Dedicated segment signals: 24 Segment signal/I/O port dual function: 16 Common signals: 4 (COM0 to COM3)
Control registers	LCD display mode register (LCDM) LCD display control register (LCDC)

Figure 19-1. LCD Controller/Driver Block Diagram



Note Segment driver

Figure 19-2. LCD Clock Selector Block Diagram



Remarks **1.** The watch timer includes the circuit enclosed with the dotted line.

2. LCDCL: LCD clock

3. f_{LCD}: LCD clock frequency

4. $f_{XX} = f_X/2$ (MCS = 0), $f_{XX} = f_X$ (MCS = 1)

19.3 LCD Controller/Driver Control Registers

The LCD controller/driver is controlled by the following two registers.

- LCD display mode register (LCDM)
- LCD display control register (LCDC)

(1) LCD display mode register (LCDM)

This register sets display operation enabling/disabling, the LCD clock, frame frequency, display mode, and operating mode.

LCDM is set with a 1-bit or 8-bit memory manipulation instruction.

$\overline{\text{RESET}}$ input clears LCDM to 00H.

Figure 19-3. LCD Display Mode Register Format

Symbol	⑦	6	5	4	3	2	1	0	Address	After Reset	R/W
LCDM	LDON	LCDM6	LCDM5	LCDM4	LCDM3	LCDM2	LCDM1	LCDM0	FFB0H	00H	R/W

LDON	Enables/Disables LCD Display
0	Display OFF (all segment outputs are unselect signals)
1	Display ON

LCDM6	LCDM5	LCDM4	Selects LCD Clock[Note 1]		
			f_{XX} = 5.0 MHz	f_{XX} = 4.19 MHz	f_{XT} = 32.768 kHz
0	0	0	$f_W/2^9$ (76 Hz)	$f_W/2^9$ (64 Hz)	$f_W/2^9$ (64 Hz)
0	0	1	$f_W/2^8$ (153 Hz)	$f_W/2^8$ (128 Hz)	$f_W/2^8$ (128 Hz)
0	1	0	$f_W/2^7$ (305 Hz)	$f_W/2^7$ (256 Hz)	$f_W/2^7$ (256 Hz)
0	1	1	$f_W/2^6$ (610 Hz)	$f_W/2^6$ (512 Hz)	$f_W/2^6$ (512 Hz)
Other than above			Setting prohibited		

Note 2 LCDM3	Operating Mode of LCD Controller/Driver	Supply Voltage of LCD Controller/Driver		
		Static Display Mode	1/3 Bias Mode	1/2 Bias Mode
0	Normal operation	2.0 to 5.5 V	2.5 to 5.5 V	2.7 to 5.5 V
1	Low-voltage operation	2.0 to 3.4 V		

LCDM2	LCDM1	LCDM0	Selects Display Mode of LCD Controller/Driver	
			Time Division	Bias Mode
0	0	0	4	1/3
0	0	1	3	1/3
0	1	0	2	1/2
0	1	1	3	1/2
1	0	0	Static display mode	
Other than above			Setting prohibited	

Notes 1. The LCD clock is supplied from the watch timer. When LCD display is performed, 1 should be set in bit 1 (TMC21) of the watch timer mode control register (TMC2).

2. To reduce the power consumption, clear LCDM3 to 0 when LCD display is not performed. Before manipulating LCDM3, be sure to turn off the LCD display.

If TMC21 is cleared to 0 during LCD display, the LCD clock supply will be stopped and the display will be disrupted.

Remarks 1. f_W: Watch timer clock frequency ($f_{XX}/2^7$ or f_{XT})

2. f_{XX}: Main system clock frequency (f_X or $f_X/2$)

3. f_X: Main system clock oscillation frequency

4. f_{XT}: Subsystem clock oscillation frequency

Table 19-3. Frame Frequencies (Hz)

LCDCL Duty	$f_W/2^9$ (64 Hz)	$f_W/2^8$ (128 Hz)	$f_W/2^7$ (256 Hz)	$f_W/2^6$ (512 Hz)
Static	64	128	256	512
1/2	32	64	128	256
1/3	21	43	85	171
1/4	16	32	64	128

Remarks 1. Figures in parentheses apply to operation with f_{XX} = 4.19 MHz or f_{XT} = 32.768 kHz.

2. f_W: Watch timer clock frequency ($f_{XX}/2^7$ or f_{XT})

3. f_{XX}: Main system clock frequency (f_X or $f_X/2$)

4. f_X: Main system clock oscillation frequency

5. f_{XT}: Subsystem clock oscillation frequency

(2) LCD display control register (LCDC)

This register sets cutoff of the current flowing to split resistors for LCD drive voltage generation and switchover between segment output and I/O port functions.

LCDC is set with a 1-bit or 8-bit memory manipulation instruction.

$\overline{\text{RESET}}$ input clears LCDC to 00H.

Figure 19-4. LCD Display Control Register Format

Symbol	7	6	5	4	3	2	①	⓪	Address	After Reset	R/W
LCDC	LCDC7	LCDC6	LCDC5	LCDC4	0	0	LEPS	LIPS	FFB2H	00H	R/W

LCDC7	LCDC6	LCDC5	LCDC4	P80/S39 to P97/S24 Pin Functions	
				Port Pins	Segment Pins
0	0	0	0	P80 to P97	None
0	0	0	1	P80 to P95	S24, S25
0	0	1	0	P80 to P93	S24 to S27
0	0	1	1	P80 to P91	S24 to S29
0	1	0	0	P80 to P87	S24 to S31
0	1	0	1	P80 to P85	S24 to S33
0	1	1	0	P80 to P83	S24 to S35
0	1	1	1	P80 to P81	S24 to S37
1	0	0	0	None	S24 to S39
Other than above				Setting prohibited	

LEPS	LIPS	LCD Driving Power Supply Selection
0	0	Does not supply power to LCD.
0	1	Supplies power to LCD from V_{DD} pin.
1	0	Supplies power to LCD from BIAS pin. (Shorts BIAS and V_{LC0} pins internally.)
1	1	Setting prohibited

Cautions 1. Pins which perform segment output cannot be used as output port pins even if 0 is set in the port mode register.

2. If a pin which performs segment output is read as a port, its value will be 0.

3. Pins set as segment outputs by LCDC cannot have an internal pull-up resistor used regardless of the value of bits 0 and 1 (PUO8 and PUO9) of pull-up resistor option register H.

19.4 LCD Controller/Driver Settings

LCD controller/driver settings should be performed as shown below. When the LCD controller/driver is used, the watch timer should be set to the operational state beforehand.

<1> Set “watch operation enabled” in timer clock select register 2 (TCL2) and the watch timer mode control register (TMC2).

<2> Set the initial value in the display data memory (FA58H to FA7FH).

<3> Set the pins to be used as segment outputs in the LCD display control register (LCDC).

<4> Set the display mode, operating mode, and the LCD clock in the LCD display mode register (LCDM).

Next, set data in the display data memory according to the display contents.

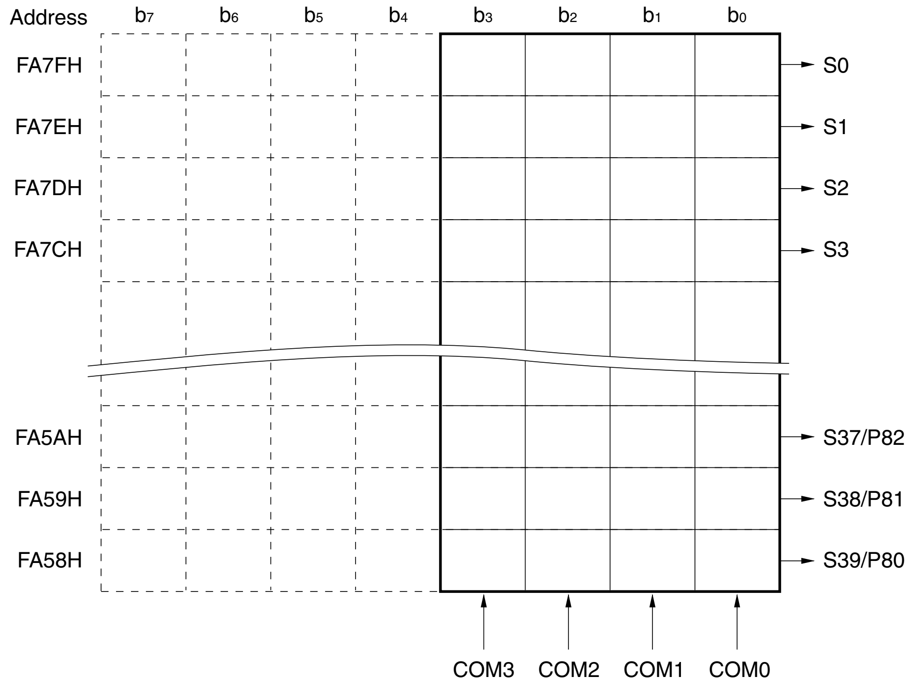
19.5 LCD Display Data Memory

The LCD display data memory is mapped onto addresses FA58H to FA7FH. The data stored in the LCD display data memory can be displayed on an LCD panel by the LCD controller/driver.

Figure 19-5 shows the relationship between the LCD display data memory contents and the segment outputs/common outputs.

Any area not used for display can be used as normal RAM.

Figure 19-5. Relationship Between LCD Display Data Memory Contents and Segment/Common Outputs



Caution The higher 4 bits of the LCD display data memory do not incorporate memory. Be sure to set them to 0.

19.6 Common Signals and Segment Signals

An individual pixel on an LCD panel lights when the potential difference of the corresponding common signal and segment signal reaches or exceeds a given voltage (the LCD drive voltage V_{LCD}).

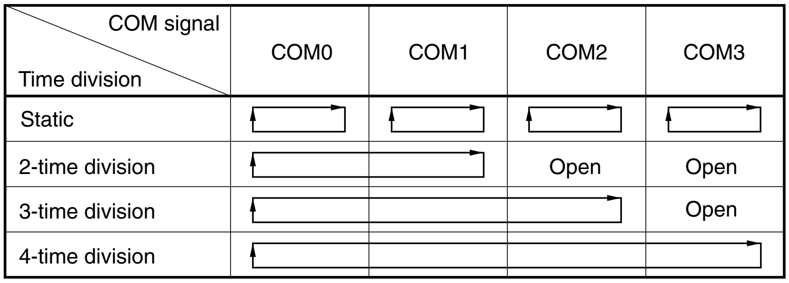
As an LCD panel deteriorates if a DC voltage is applied in the common signals and segment signals, it is driven by AC voltage.

(1) Common signals

For common signals, the selection timing order is as shown in Table 19-4 according to the number of time divisions set, and operations are repeated with these as the cycle. In the static display mode, the same signal is output to COM0 to COM3.

With 2-time-division operation, pins COM2 and COM3 are left open, and with 3-time-division operation, the COM3 pin is left open.

Table 19-4. COM Signals



COM signal / Time division	COM0	COM1	COM2	COM3
Static				
2-time division			Open	Open
3-time division				Open
4-time division				

(2) Segment signals

Segment signals correspond to a 40-byte LCD display data memory (FA58H to FA7FH). Each display data memory bit 0, bit 1, bit 2, and bit 3 is read in synchronization with the COM0, COM1, COM2 and COM3 timings respectively, and if the value of the bit is 1, it is converted to the selection voltage. If the value of the bit is 0, it is converted to the non-selection voltage and output to a segment pin (S0 to S39) (S24 to S39 have a dual function as I/O port pins).

Consequently, it is necessary to check what combination of front surface electrodes (corresponding to the segment signals) and rear surface electrodes (corresponding to the common signals) of the LCD display to be used form the display pattern, and then write bit data corresponding on a one-to-one basis with the pattern to be displayed.

In addition, because LCD display data memory bits 1 and 2 are not used with the static display mode, bits 2 and 3 are not used with the 2-time-division method, and bit 3 is not used with the 3-time-division method, these can be used for other than display purposes.

Bits 4 to 7 are fixed at 0.

(3) Common signal and segment signal output waveforms

The voltages shown in Table 19-5 are output in the common signals and segment signals.

The ±V_{LCD} ON voltage is only produced when the common signal and segment signal are both at the selection voltage; other combinations produce the OFF voltage.

Table 19-5. LCD Drive Voltages

(a) Static display mode

Segment	Select	Non-select
Common	V_{SS1}, V_{LC0}	V_{LC0}, V_{SS1}
V_{LC0}, V_{SS1}	$-V_{LCD}$, $+V_{LCD}$	0 V, 0 V

(b) 1/2 bias method

	Segment	Select	Non-select
Common		V_{SS1}, V_{LC0}	V_{LC0}, V_{SS1}
Select level	V_{LC0}, V_{SS1}	$-V_{LCD}$, $+V_{LCD}$	0 V, 0 V
Non-select level	$V_{LC1} = V_{LC2}$	$-1/2V_{LCD}$, $+1/2V_{LCD}$	$+1/2V_{LCD}$, $-1/2V_{LCD}$

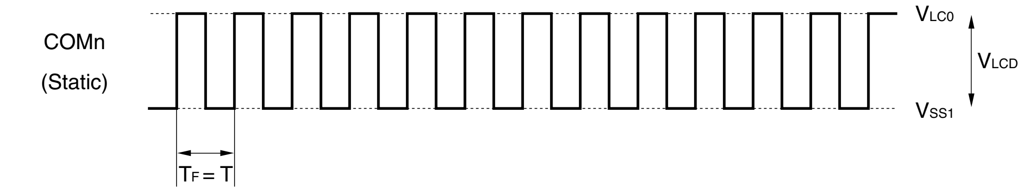
(c) 1/3 bias method

	Segment	Select	Non-select
Common		V_{SS1}, V_{LC0}	V_{LC1}, V_{LC2}
Select level	V_{LC0}, V_{SS1}	$-V_{LCD}$, $+V_{LCD}$	$-1/3V_{LCD}$, $+1/3V_{LCD}$
Non-select level	V_{LC2}, V_{LC1}	$-1/3V_{LCD}$, $+1/3V_{LCD}$	$-1/3V_{LCD}$, $+1/3V_{LCD}$

Figure 19-6 shows the common signal waveform, and Figure 19-7 shows the common signal and segment signal voltages and phases.

Figure 19-6. Common Signal Waveform

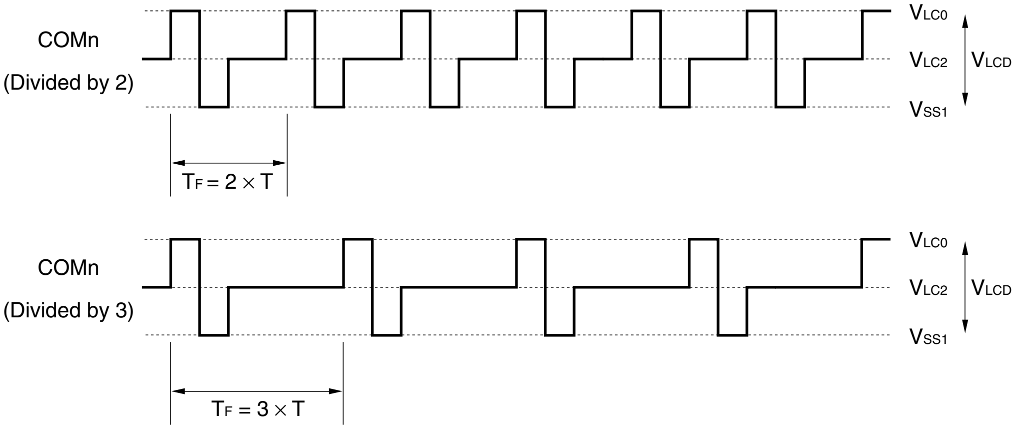
(a) Static display mode



Remarks 1. T: One LCDCL cycle

2. T_F: Frame frequency

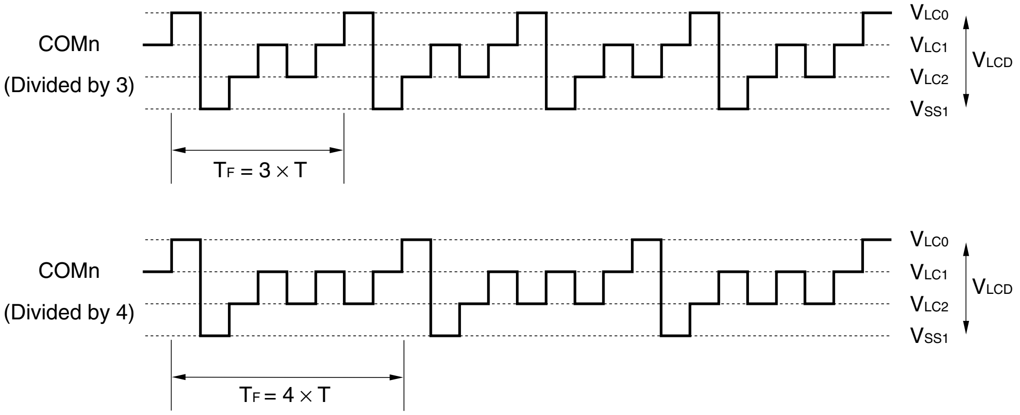
(b) 1/2 bias method



Remarks 1. T: One LCDCL cycle

2. T_F: Frame frequency

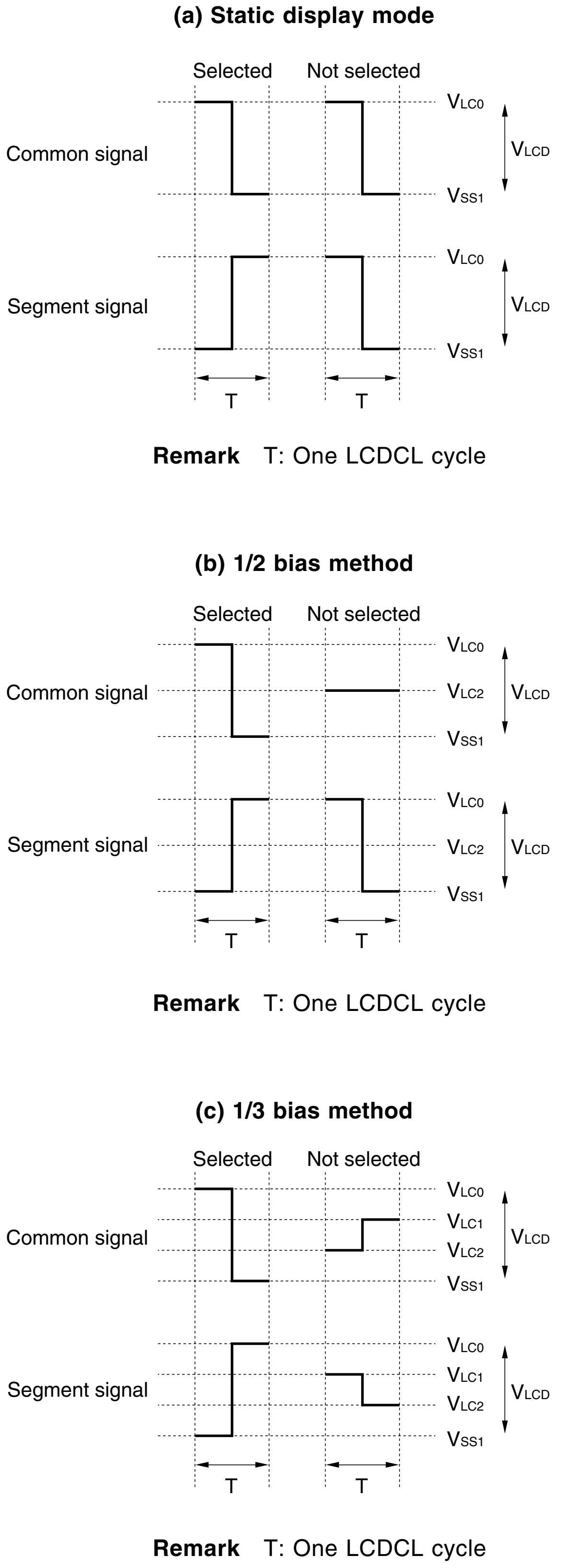
(c) 1/3 bias method



Remarks 1. T: One LCDCL cycle

2. T_F: Frame frequency

Figure 19-7. Common Signal and Static Signal Voltages and Phases



(a) Static display mode

Remark T: One LCDCL cycle

(b) 1/2 bias method

Remark T: One LCDCL cycle

(c) 1/3 bias method

Remark T: One LCDCL cycle

19.7 Supply of LCD Drive Voltages V_{LC0}, V_{LC1}, V_{LC2}

Split resistors for producing the LCD drive voltages can be incorporated in the mask ROM versions (μPD780306, 780308, 780306Y, and 780308Y) by mask option (the PROM versions (μPD78P0308, 78P0308Y) do not incorporate split resistors). Incorporating the split resistors makes it possible to produce LCD drive voltages appropriate to the various bias methods shown in Table 19-6 without using external split resistors.

Also, an LCD drive voltage can be externally supplied from the BIAS pin to produce other LCD drive voltages.

Table 19-6. LCD Drive Voltages (with On-Chip Split Resistor)

Bias Method / LCD Drive Voltage	No Bias (Static Mode)	1/2 Bias	1/3 Bias
V_{LC0}	V_{LCD}	V_{LCD}	V_{LCD}
V_{LC1}	2/3 V_{LCD}	1/2 V_{LCD} Note	2/3 V_{LCD}
V_{LC2}	1/3 V_{LCD}		1/3 V_{LCD}

Note With the 1/2 bias method, the V_{LC1} pin and V_{LC2} pin must be connected externally.

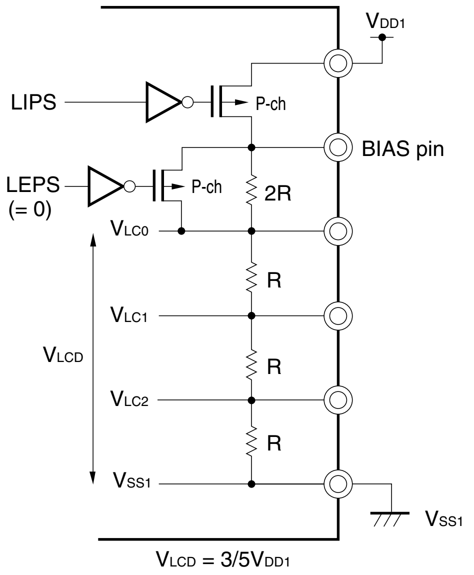
Remarks 1. When the BIAS pin and V_{LC0} pin are open, $V_{LCD} = 3/5\ V_{DD}$ (with on-chip split resistor).

2. When the BIAS pin and V_{LC0} pin are connected, $V_{LCD} = V_{DD1}$.

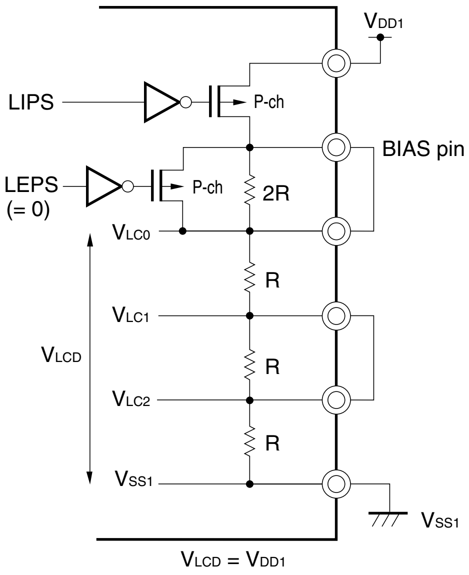
Examples of internal supply of the LCD drive voltage in accordance with Table 19-6 are shown in Figures 19-8 and 19-9. An example of supply of the LCD drive voltage from off-chip is shown in Figure 19-10. Stepless LCD drive voltages can be supplied by means of variable resistor r.

Figure 19-8. LCD Drive Power Supply Connection Examples (with On-Chip Split Resistor)

(a) 1/3 bias method and static display mode
(Example with V_{DD1} = 5 V, V_{LCD} = 3 V)



(b) 1/2 bias method mode
(Example with V_{DD1} = 5 V, V_{LCD} = 5 V)



(c) 1/3 bias method and static display mode
(Example with V_{DD1} = 5 V, V_{LCD} = 5 V)

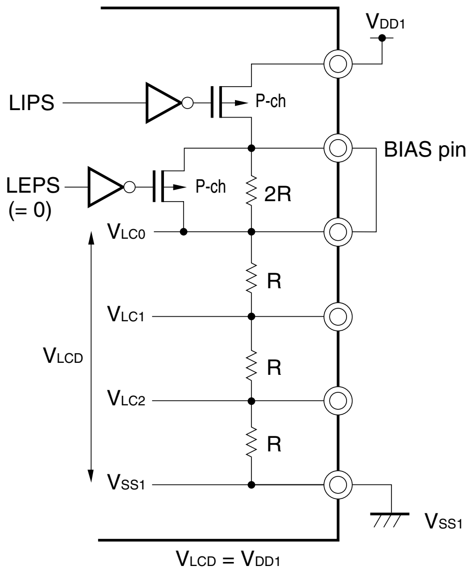
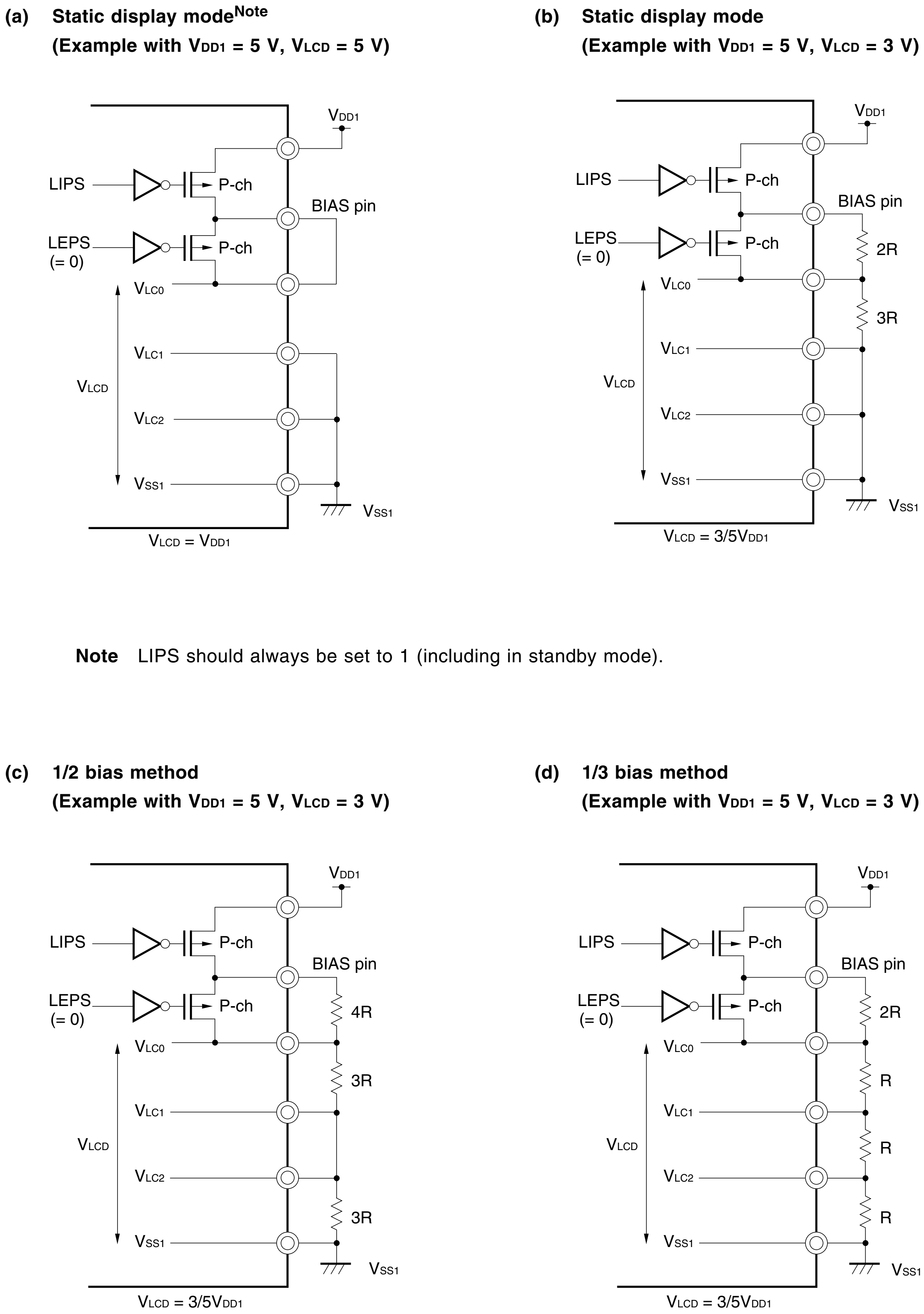


Figure 19-9. LCD Drive Power Supply Connection Examples (with External Split Resistor)



(a) **Static display mode**Note **(Example with V_{DD1} = 5 V, V_{LCD} = 5 V)**

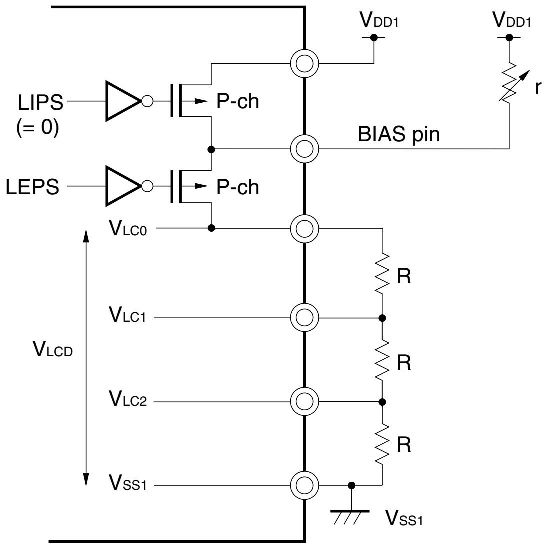
(b) **Static display mode (Example with V_{DD1} = 5 V, V_{LCD} = 3 V)**

(c) **1/2 bias method (Example with V_{DD1} = 5 V, V_{LCD} = 3 V)**

(d) **1/3 bias method (Example with V_{DD1} = 5 V, V_{LCD} = 3 V)**

Note LIPS should always be set to 1 (including in standby mode).

Figure 19-10. Example of LCD Drive Voltage Supply from Off-Chip



$$V_{LCD} = \frac{3R}{3R + r} V_{DD1}$$

19.8 Display Modes

19.8.1 Static display example

Figure 19-12 shows the connection of a static type 5-digit LCD panel with the display pattern shown in Figure 19-11 with the μPD780308, 780308Y Subseries segment (S0 to S39) and common (COM0) signals. The display example is “123.45,” and the display data memory contents (addresses FA58H to FA7FH) correspond to this.

An explanation is given here taking the example of the third digit “3.” (3.). In accordance with the display pattern in Figure 19-11, selection and non-selection voltages must be output to pins S16 to S23 as shown in Table 19-7 at the COM0 common signal timing.

Table 19-7. Selection and Non-Selection Voltages (COM0)

Segment / Common	S16	S17	S18	S19	S20	S21	S22	S23
COM0	S	S	S	S	NS	S	NS	S

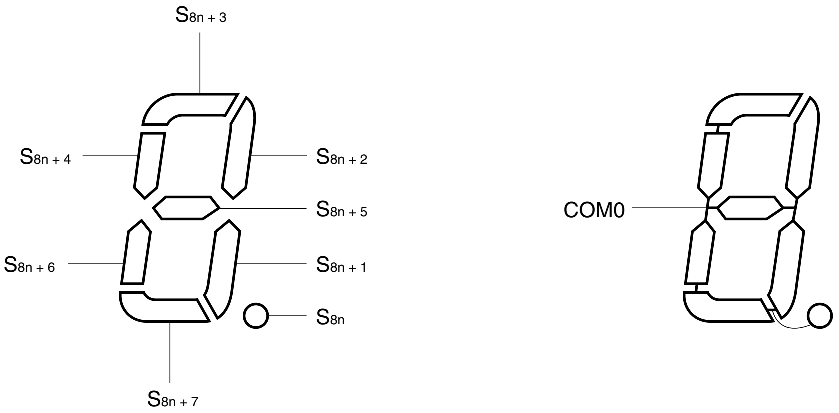
S: Selection, NS: Non-selection

From this, it can be seen that 10101111 must be prepared in bit 0 of the display data memory (addresses FA68H to FA6FH) corresponding to S16 to S23.

The LCD drive waveforms for S19, S20, and COM0 are shown in Figure 19-13. When S19 is at the selection voltage at the timing for selection with COM0, it can be seen that the $+V_{LCD}/-V_{LCD}$ AC square wave, which is the LCD illumination (ON) level, is generated.

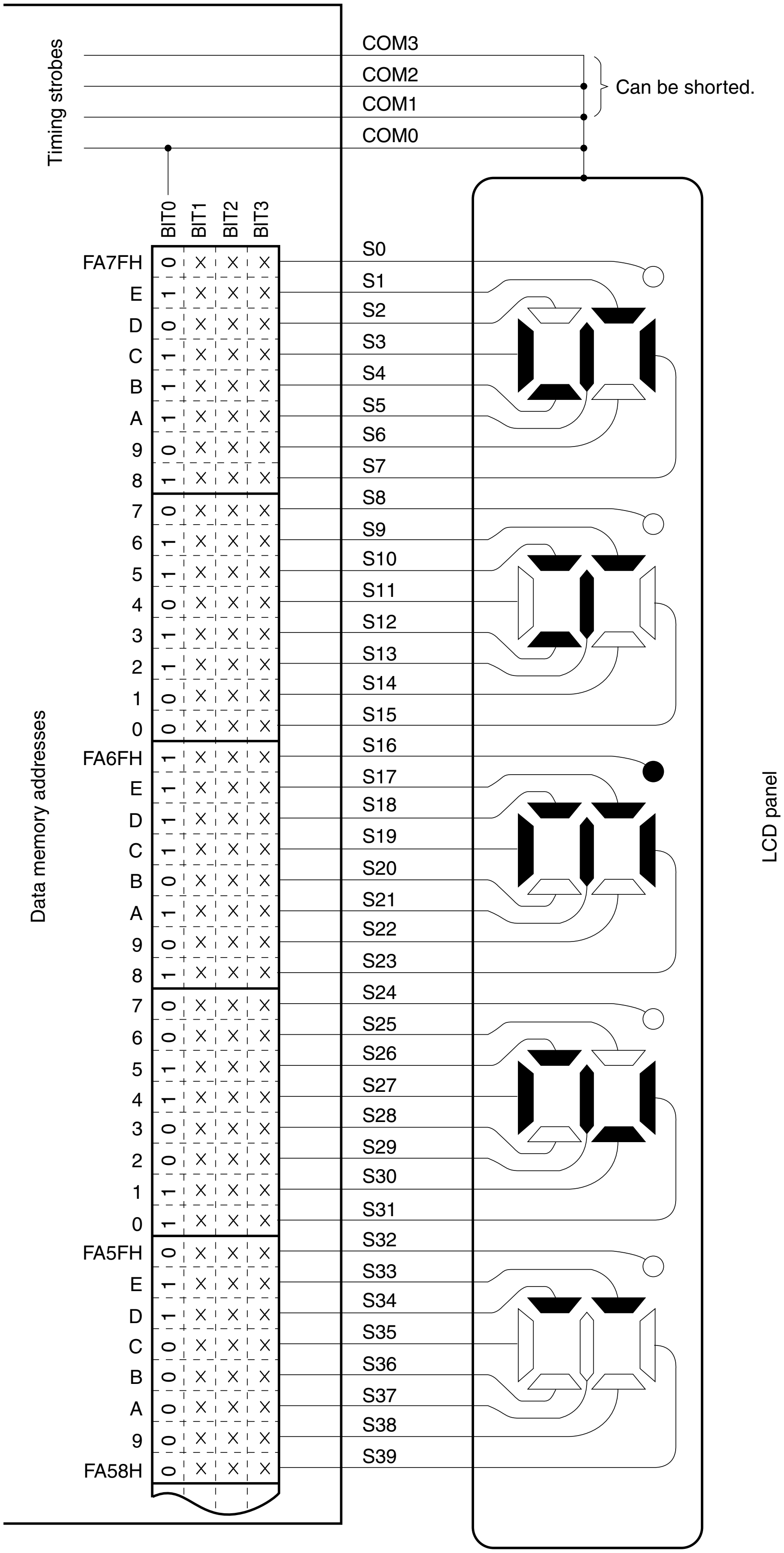
Shorting the COM0 to COM3 lines increases the current drive capability because the same waveform as COM0 is output to COM1 to COM3.

Figure 19-11. Static LCD Display Pattern and Electrode Connections



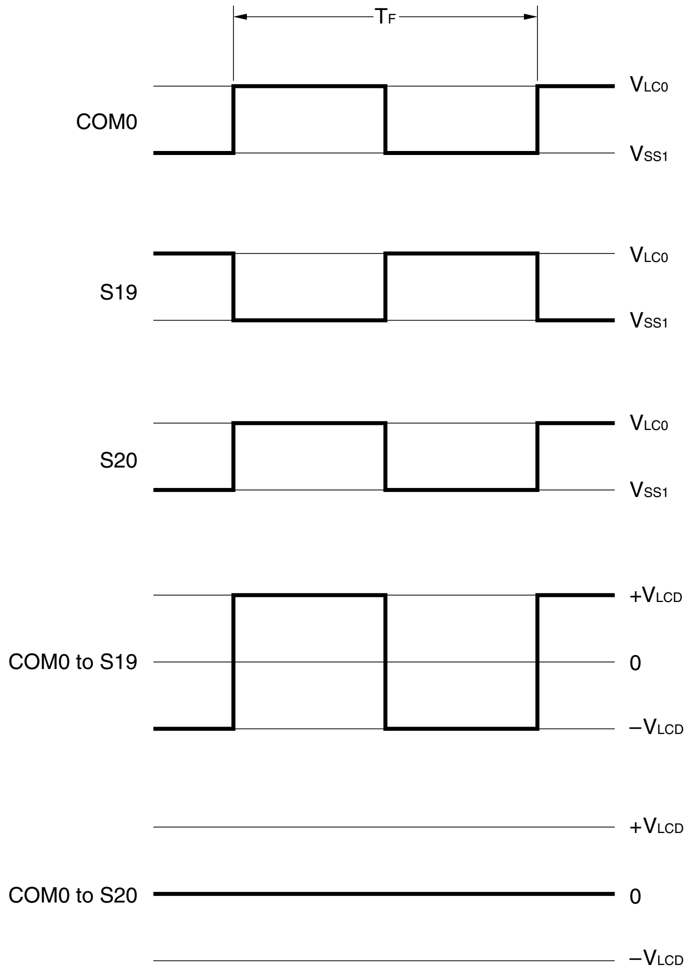
n = 0 to 4

Figure 19-12. Static LCD Panel Connection Example



	BIT0	BIT1	BIT2	BIT3
FA7FH	0	×	×	×
E	1	×	×	×
D	0	×	×	×
C	1	×	×	×
B	1	×	×	×
A	1	×	×	×
9	0	×	×	×
8	1	×	×	×
7	0	×	×	×
6	1	×	×	×
5	1	×	×	×
4	0	×	×	×
3	1	×	×	×
2	1	×	×	×
1	0	×	×	×
0	0	×	×	×
FA6FH	1	×	×	×
E	1	×	×	×
D	1	×	×	×
C	1	×	×	×
B	0	×	×	×
A	1	×	×	×
9	0	×	×	×
8	1	×	×	×
7	0	×	×	×
6	0	×	×	×
5	1	×	×	×
4	1	×	×	×
3	0	×	×	×
2	0	×	×	×
1	1	×	×	×
0	1	×	×	×
FA5FH	0	×	×	×
E	1	×	×	×
D	1	×	×	×
C	0	×	×	×
B	0	×	×	×
A	0	×	×	×
9	0	×	×	×
FA58H	0	×	×	×

Figure 19-13. Static LCD Drive Waveform Examples



19.8.2 2-time-division display example

Figure 19-15 shows the connection of a 2-time-division type 10-digit LCD panel with the display pattern shown in Figure 19-14 with the μPD780308, 780308Y Subseries segment signals (S0 to S39) and common signals (COM0, COM1). The display example is "123456.7890," and the display data memory contents (addresses FA58H to FA7FH) correspond to this.

An explanation is given here taking the example of the eighth digit "3" (3). In accordance with the display pattern in Figure 19-14, selection and non-selection voltages must be output to pins S28 to S31 as shown in Table 19-8 at the COM0 and COM1 common signal timings.

Table 19-8. Selection and Non-Selection Voltages (COM0, COM1)

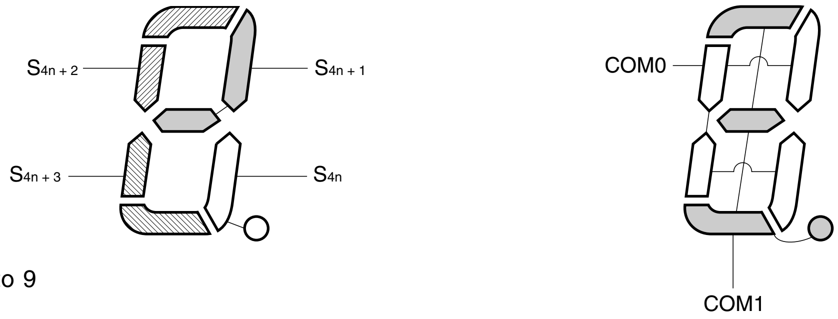
Segment / Common	S28	S29	S30	S31
COM0	S	S	NS	NS
COM1	NS	S	S	S

S: Selection, NS: Non-selection

From this, it can be seen that, for example, ××10 must be prepared in the display data memory (address FA60H) corresponding to S31.

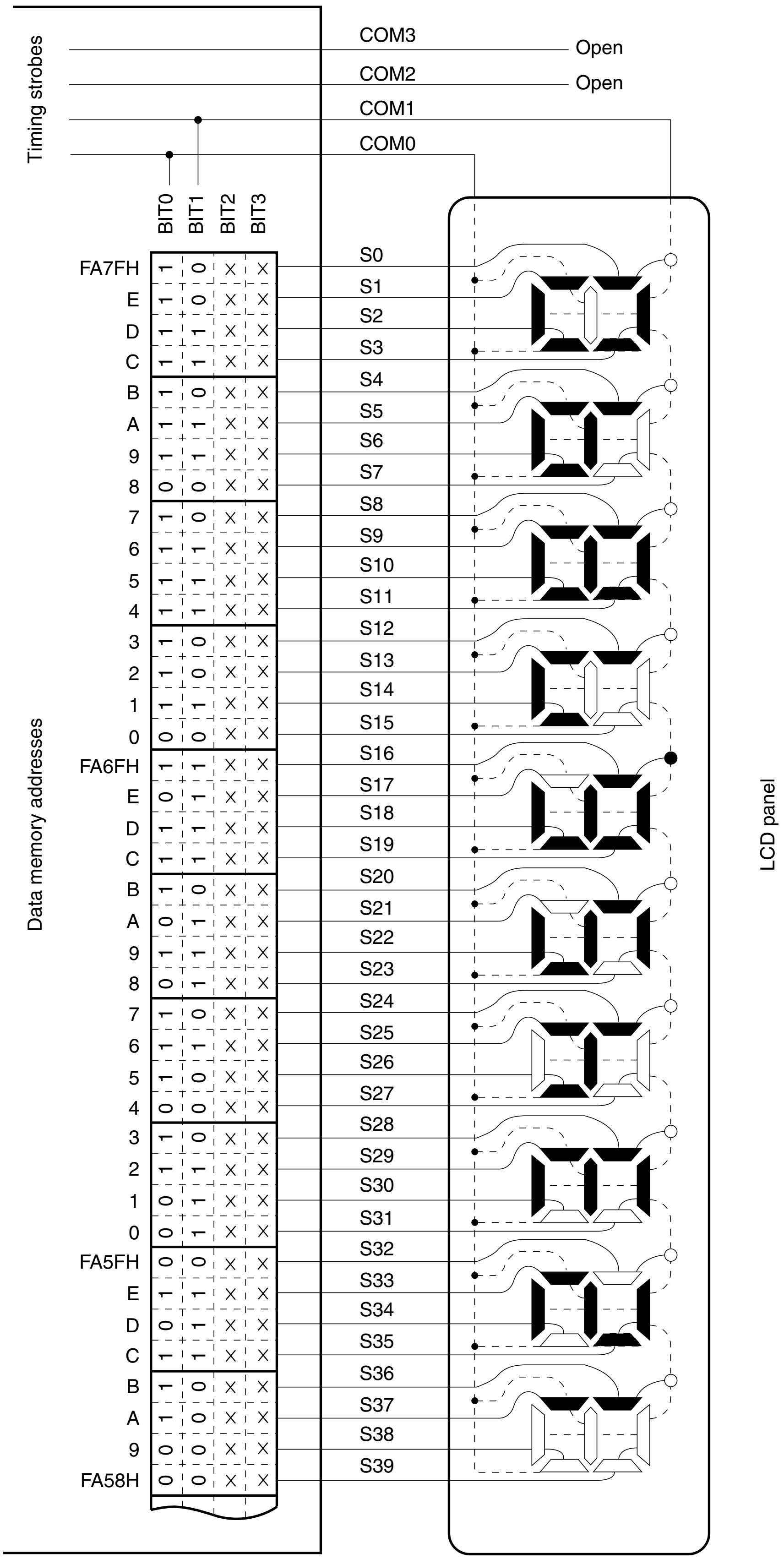
Examples of the LCD drive waveforms between S31 and the common signals are shown in Figure 19-16. When S31 is at the selection voltage at the COM1 selection timing, it can be seen that the $+V_{LCD}$/$-V_{LCD}$ AC square wave, which is the LCD illumination (ON) level, is generated.

Figure 19-14. 2-Time-Division LCD Display Pattern and Electrode Connections



n = 0 to 9

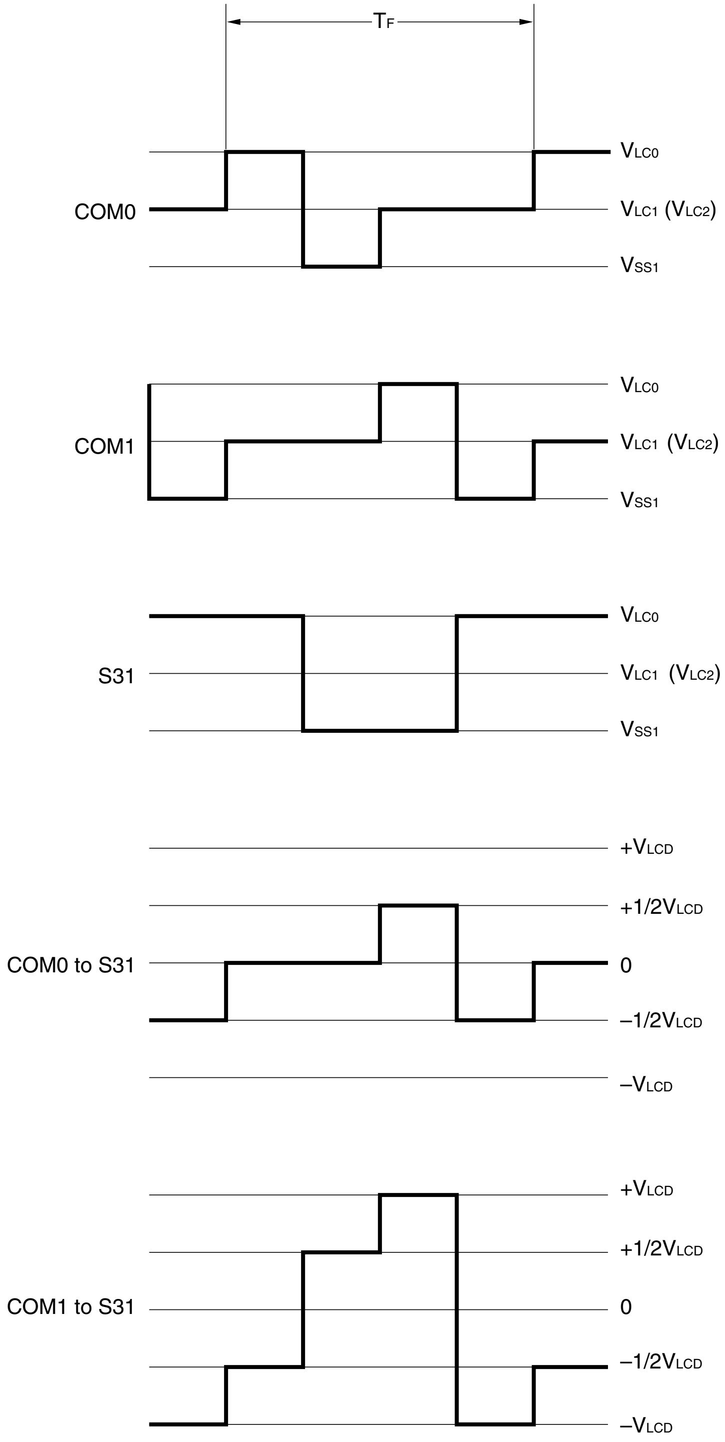
Figure 19-15. 2-Time-Division LCD Panel Connection Example



	BIT0	BIT1	BIT2	BIT3
FA7FH	1	0	×	×
E	1	0	×	×
D	1	1	×	×
C	1	1	×	×
B	1	0	×	×
A	1	1	×	×
9	1	1	×	×
8	0	0	×	×
7	1	0	×	×
6	1	1	×	×
5	1	1	×	×
4	1	1	×	×
3	1	0	×	×
2	1	0	×	×
1	1	1	×	×
0	0	0	×	×
FA6FH	1	1	×	×
E	0	1	×	×
D	1	1	×	×
C	1	1	×	×
B	1	0	×	×
A	0	1	×	×
9	1	1	×	×
8	0	1	×	×
7	1	0	×	×
6	1	1	×	×
5	1	0	×	×
4	0	0	×	×
3	1	0	×	×
2	1	1	×	×
1	0	1	×	×
0	0	1	×	×
FA5FH	0	0	×	×
E	1	1	×	×
D	0	1	×	×
C	1	1	×	×
B	1	0	×	×
A	1	0	×	×
9	0	0	×	×
FA58H	0	0	×	×

Remark ×: Any data can be stored because this is a 2-time-division display.

Figure 19-16. 2-Time-Division LCD Drive Waveform Examples (1/2 Bias Method)



19.8.3 3-time-division display example

Figure 19-18 shows the connection of a 3-time-division type 13-digit LCD panel with the display pattern shown in Figure 19-17 with the μPD780308, 780308Y Subseries segment signals (S0 to S38) and common signals (COM0 to COM2). The display example is “123456.7890123,” and the display data memory contents (addresses FA59H to FA7FH) correspond to this.

An explanation is given here taking the example of the eighth digit “6.” ( ). In accordance with the display pattern in Figure 19-17, selection and non-selection voltages must be output to pins S21 to S23 as shown in Table 19-9 at the COM0 to COM2 common signal timings.

Table 19-9. Selection and Non-Selection Voltages (COM0 to COM2)

Segment / Common	S21	S22	S23
COM0	NS	S	S
COM1	S	S	S
COM2	S	S	—

S: Selection, NS: Non-selection

From this, it can be seen that ×110 must be prepared in the display data memory (address FA6AH) corresponding to S21.

Examples of the LCD drive waveforms between S21 and the common signals are shown in Figure 19-19 (1/2 bias method) and Figure 19-20 (1/3 bias method). When S21 is at the selection voltage at the COM1 selection timing, and S21 is at the selection voltage at the COM2 selection timing, it can be seen that the +VLCD/–VLCD AC square wave, which is the LCD illumination (ON) level, is generated.

Figure 19-17. 3-Time-Division LCD Display Pattern and Electrode Connections

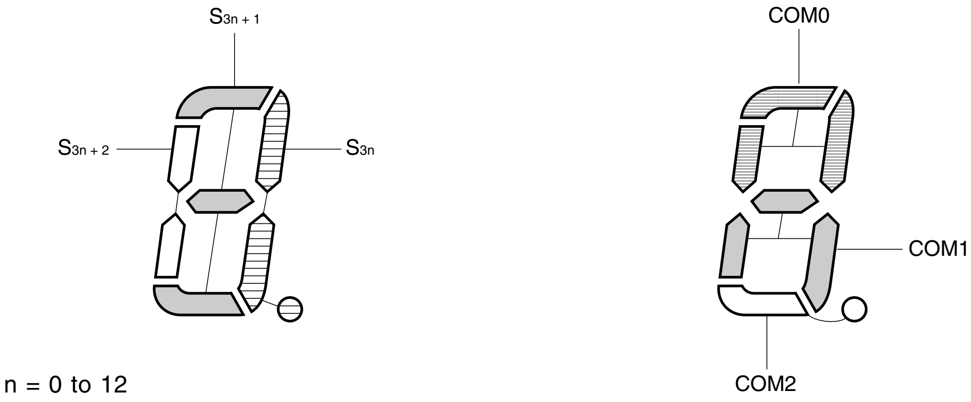
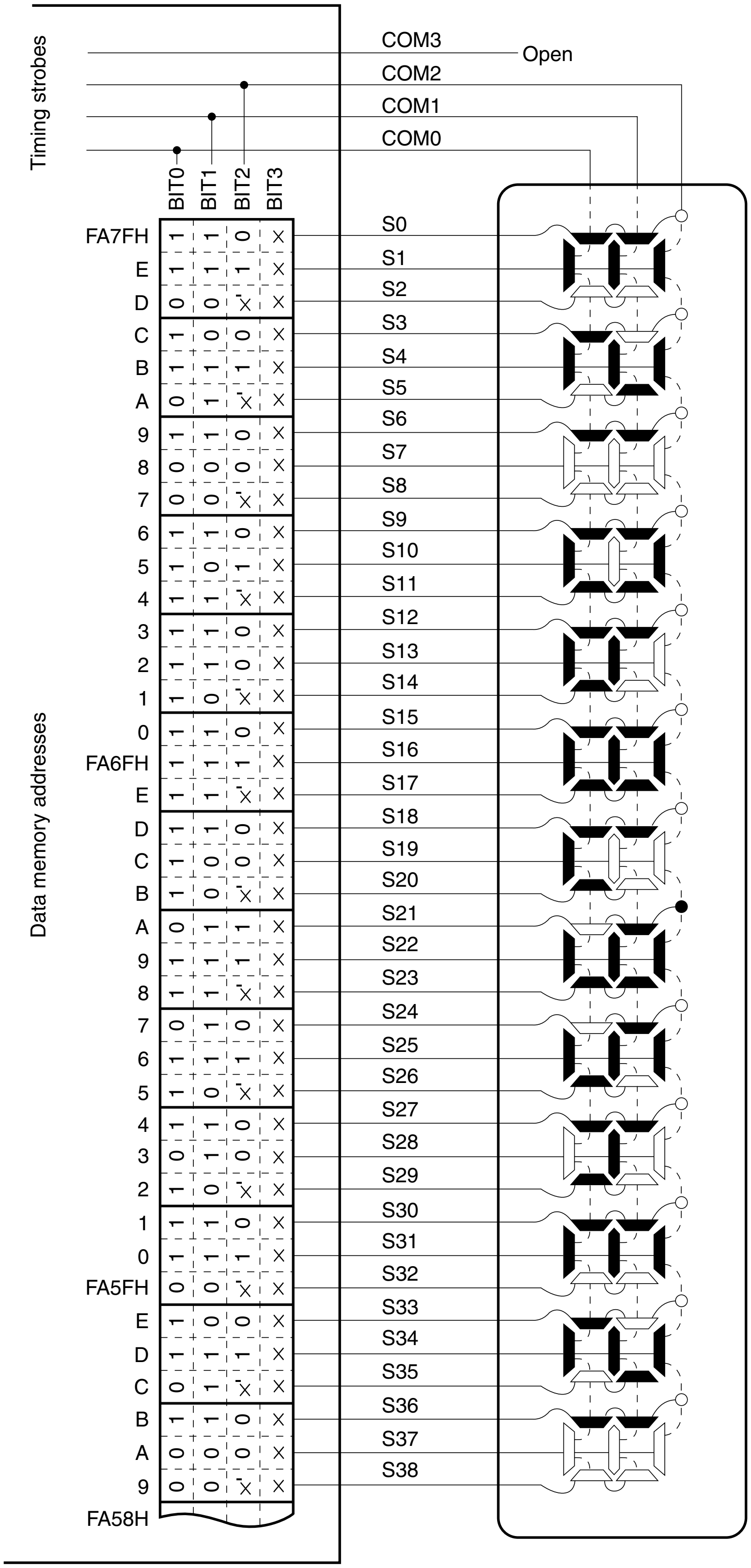


Figure 19-18. 3-Time-Division LCD Panel Connection Example



Remarks 1. ×: Any data can be stored because they have no corresponding segment in the LCD panel.

2. ×: Any data can be stored because this is a 3-time-division display.

Figure 19-19. 3-Time-Division LCD Drive Waveform Examples (1/2 Bias Method)

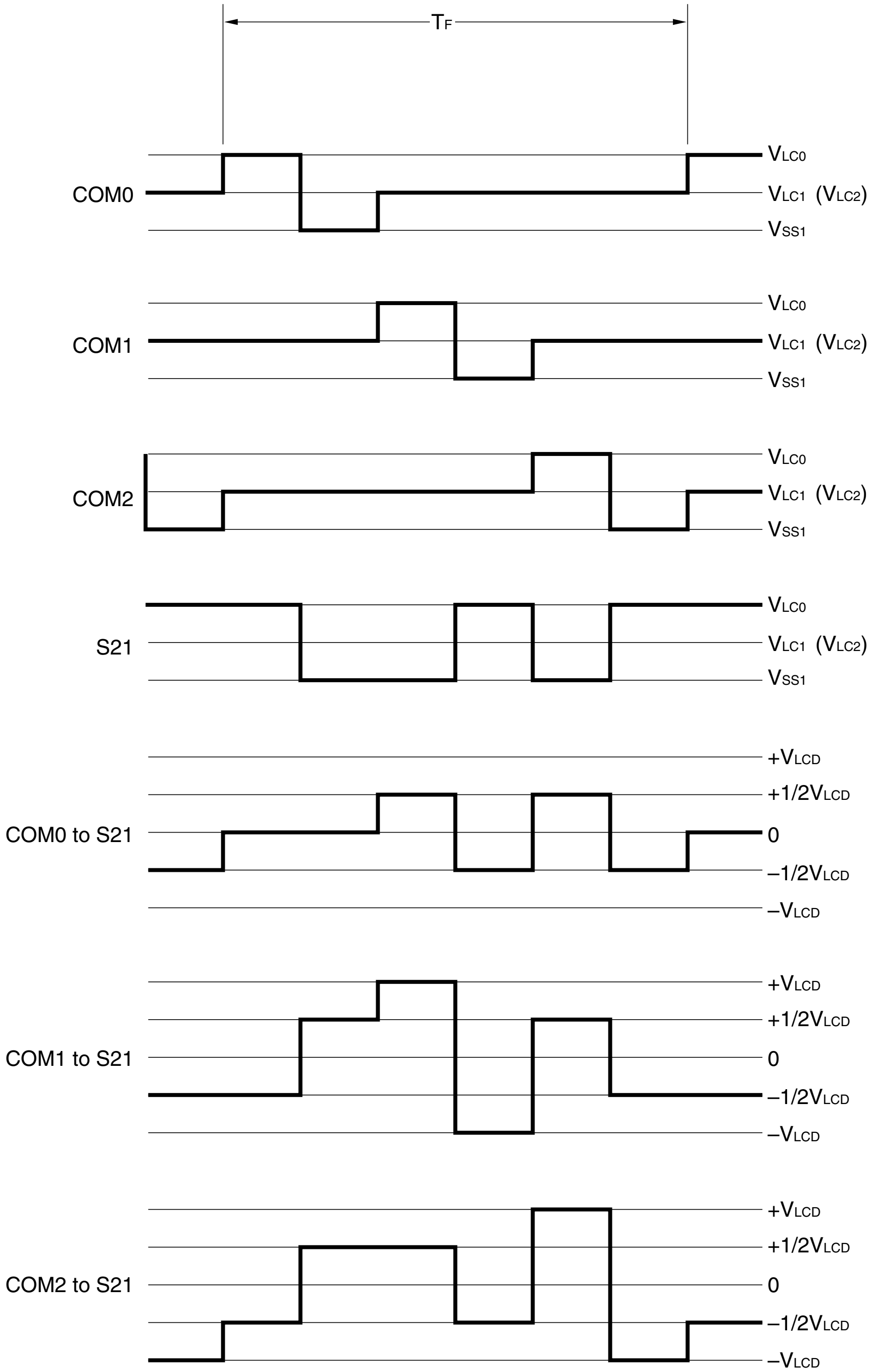
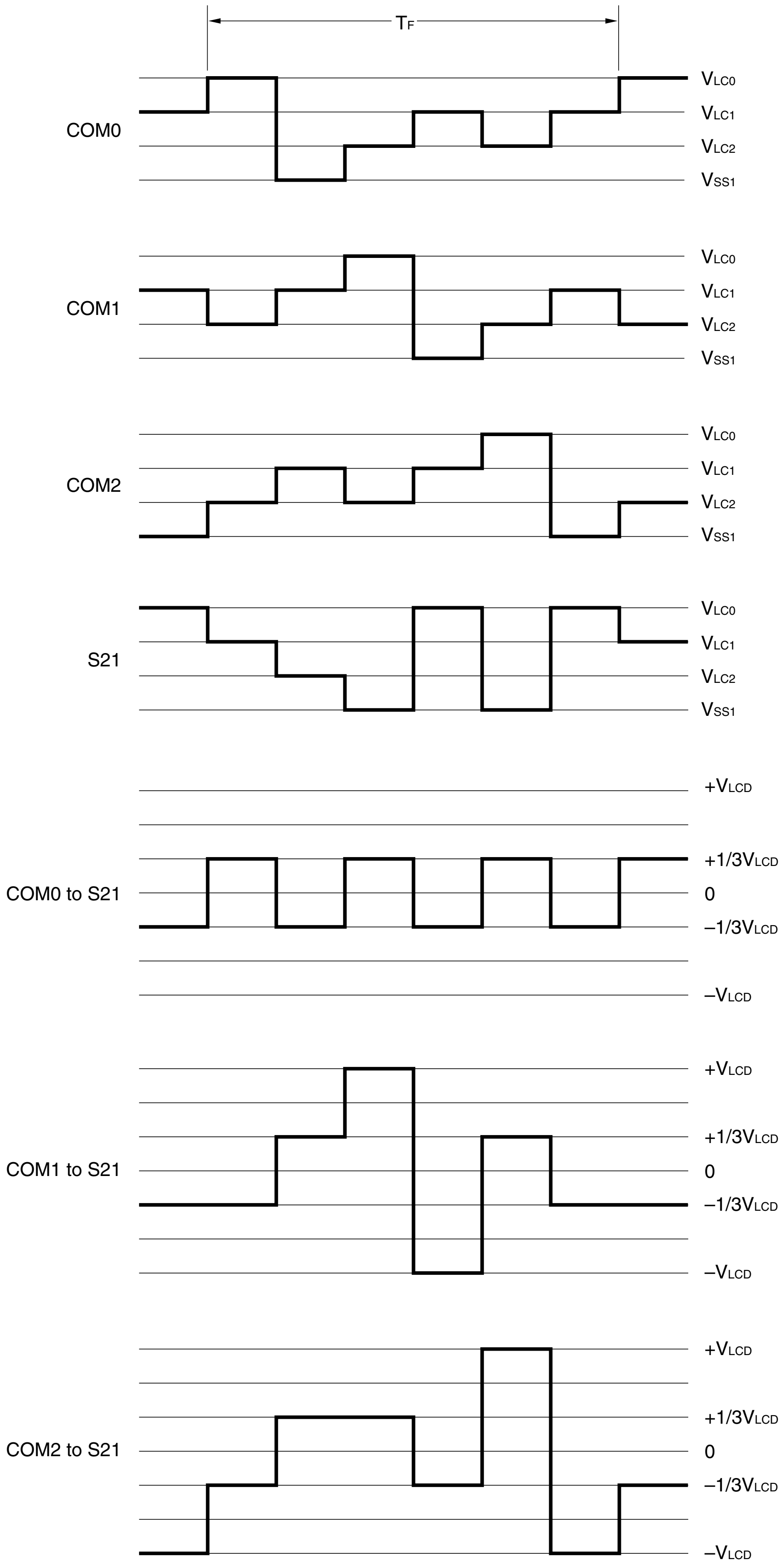


Figure 19-20. 3-Time-Division LCD Drive Waveform Examples (1/3 Bias Method)



19.8.4 4-time-division display example

Figure 19-22 shows the connection of a 4-time-division type 20-digit LCD panel with the display pattern shown in Figure 19-21 with the μPD780308, 780308Y Subseries segment signals (S0 to S39) and common signals (COM0 to COM3). The display example is “123456.78901234567890,” and the display data memory contents (addresses FA58H to FA7FH) correspond to this.

An explanation is given here taking the example of the 15th digit “6.” (6.). In accordance with the display pattern in Figure 19-21, selection and non-selection voltages must be output to pins S28 and S29 as shown in Table 19-10 at the COM0 to COM3 common signal timings.

Table 19-10. Selection and Non-Selection Voltages (COM0 to COM3)

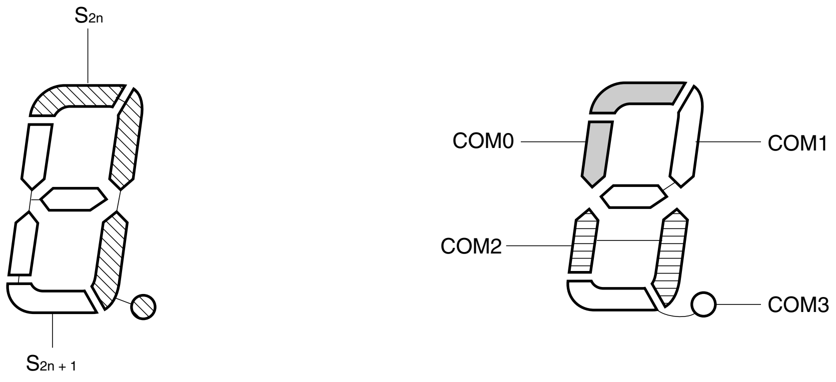
Segment Common	S28	S29
COM0	S	S
COM1	NS	S
COM2	S	S
COM3	S	S

S: Selection, NS: Non-selection

From this, it can be seen that 1101 must be prepared in the display data memory (address FA63H) corresponding to S28.

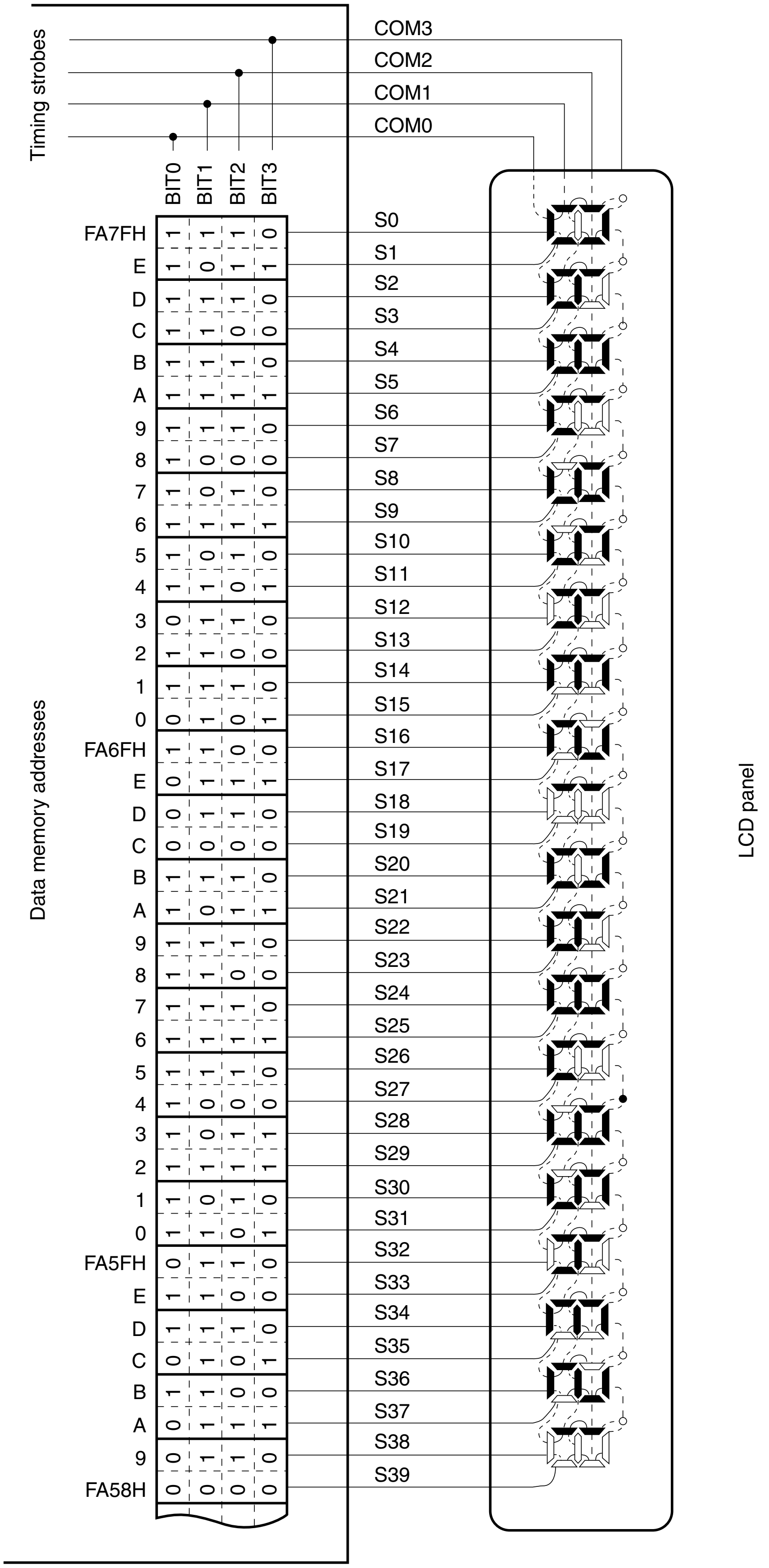
Examples of the LCD drive waveforms between S28 and the COM0 and COM1 signals are shown in Figure 19-23 (for the sake of simplicity, waveforms for COM2 and COM3 have been omitted). When S28 is at the selection voltage at the COM0 selection timing, it can be seen that the +VLCD/–VLCD AC square wave, which is the LCD illumination (ON) level, is generated.

Figure 19-21. 4-Time-Division LCD Display Pattern and Electrode Connections



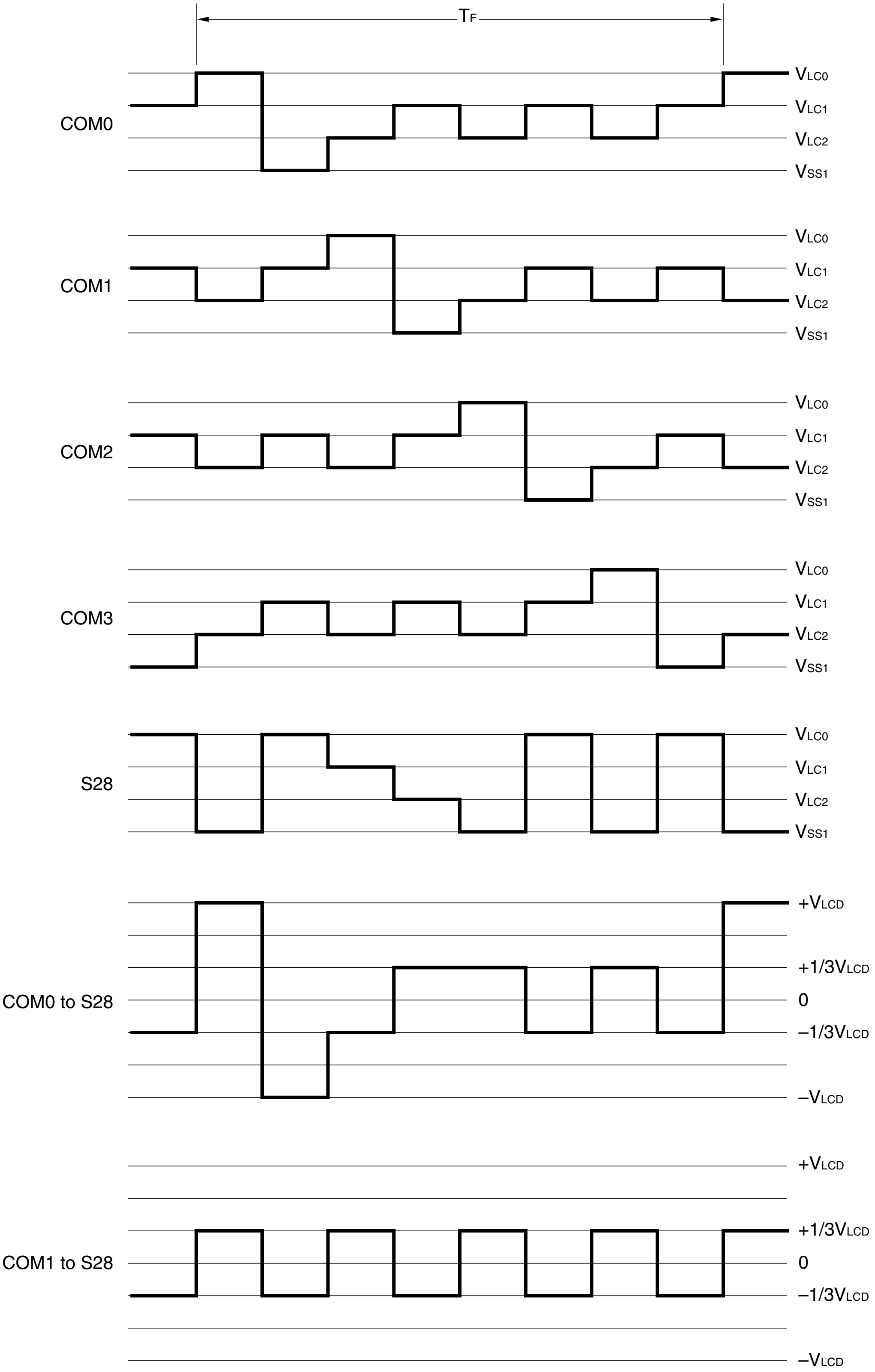
n = 0 to 18

Figure 19-22. 4-Time-Division LCD Panel Connection Example



	BIT0	BIT1	BIT2	BIT3
FA7FH	1	1	1	0
E	1	0	1	1
D	1	1	1	0
C	1	1	0	0
B	1	1	1	0
A	1	1	1	1
9	1	1	1	0
8	1	0	0	0
7	1	0	1	0
6	1	1	1	1
5	1	0	1	0
4	1	1	0	1
3	0	1	1	0
2	1	1	0	0
1	1	1	1	0
0	0	1	0	1
FA6FH	1	1	0	0
E	0	1	1	1
D	0	1	1	0
C	0	0	0	0
B	1	1	1	0
A	1	0	1	1
9	1	1	1	0
8	1	1	0	0
7	1	1	1	0
6	1	1	1	1
5	1	1	1	0
4	1	0	0	0
3	1	0	1	1
2	1	1	1	1
1	1	0	1	0
0	1	1	0	1
FA5FH	0	1	1	0
E	1	1	0	0
D	1	1	1	0
C	0	1	0	1
B	1	1	0	0
A	0	1	1	1
9	0	1	1	0
FA58H	0	0	0	0

Figure 19-23. 4-Time-Division LCD Drive Waveform Examples (1/3 Bias Method)



CHAPTER 20 INTERRUPT AND TEST FUNCTIONS

20.1 Interrupt Function Types

The following three types of interrupt functions are used.

(1) Non-maskable interrupt

This interrupt is acknowledged unconditionally (that is, even in interrupt disabled state). It does not undergo interrupt priority control and is given top priority over all other interrupt requests.
It generates a standby release signal.
One interrupt request from the watchdog timer is provided as a non-maskable interrupt.

(2) Maskable interrupts

These interrupts undergo mask control. Maskable interrupts can be divided into a high interrupt priority group and a low interrupt priority group by setting the priority specify flag register (PR0L, PR0H, and PR1L).
Multiple high priority interrupts can be applied to low priority interrupts. If two or more interrupts with the same priority are simultaneously generated, each interrupt has a predetermined priority (see **Table 20-1**).
A standby release signal is generated.
Six external interrupt requests and 13 internal interrupt requests are provided as maskable interrupts.

(3) Software interrupt

This is a vectored interrupt to be generated by executing the BRK instruction. It is acknowledged even in interrupt disabled state. The software interrupt does not undergo interrupt priority control.

20.2 Interrupt Sources and Configuration

Twenty-one non-maskable, maskable, and software interrupts are provided as interrupt sources (see **Table 20-1**).

Table 20-1. Interrupt Source List

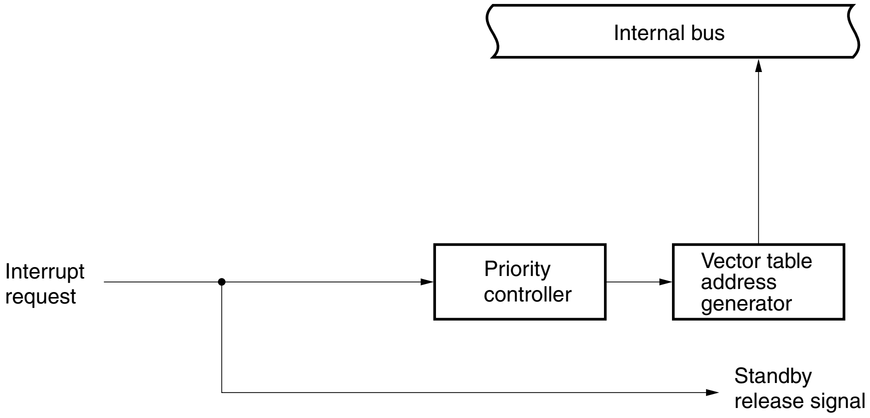
Interrupt Type	Default Priority[Note 1]	Interrupt Source		Internal/ External	Vector Table Address	Basic Configuration Type[Note 2]
		Name	Trigger			
Non-maskable	—	INTWDT	Watchdog timer overflow (with watchdog timer mode 1 selected)	Internal	0004H	(A)
Maskable	0	INTWDT	Watchdog timer overflow (with interval timer mode selected)			(B)
	1	INTP0	Pin input edge detection	External	0006H	(C)
	2	INTP1			0008H	(D)
	3	INTP2			000AH	
	4	INTP3			000CH	
	5	INTP4			000EH	
	6	INTP5			0010H	
	7	INTCSI0	End of serial interface channel 0 transfer	Internal	0014H	(B)
	8	INTSER	Serial interface channel 2 UART reception error occurrence		0018H	
	9	INTSR	End of serial interface channel 2 UART reception		001AH	
		INTCSI2	End of serial interface channel 2 3-wire transfer			
	10	INTST	End of serial interface channel 2 UART transfer		001CH	
	11	INTTM3	Reference time interval signal from watch timer		001EH	
	12	INTTM00	Generation of 16-bit timer register, capture/compare register 00 (CR00) match signal		0020H	
	13	INTTM01	Generation of 16-bit timer register, capture/compare register 01 (CR01) match signal		0022H	
	14	INTTM1	Generation of 8-bit timer/event counter 1 match signal		0024H	
	15	INTTM2	Generation of 8-bit timer/event counter 2 match signal		0026H	
	16	INTAD	End of A/D converter conversion		0028H	
	17	INTCSI3	End of serial interface channel 3 transfer		002AH	
Software	—	BRK	BRK instruction execution	–	003EH	(E)

Notes 1. Default priorities are intended for two or more simultaneously generated maskable interrupt requests. 0 is the highest priority and 17 is the lowest priority.

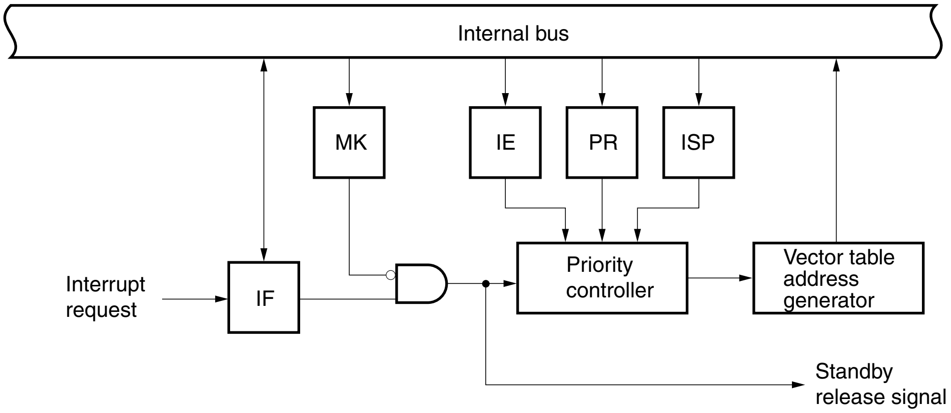
2. Basic configuration types (A) to (E) correspond to (A) to (E) of Figure 20-1.

Figure 20-1. Basic Configuration of Interrupt Function (1/2)

(A) Internal non-maskable interrupt



(B) Internal maskable interrupt



(C) External maskable interrupt (INTP0)

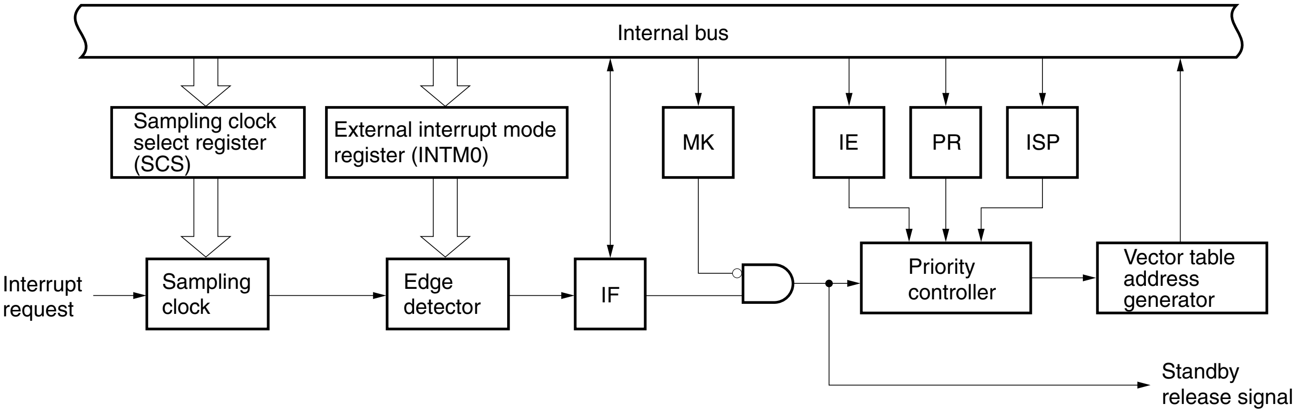
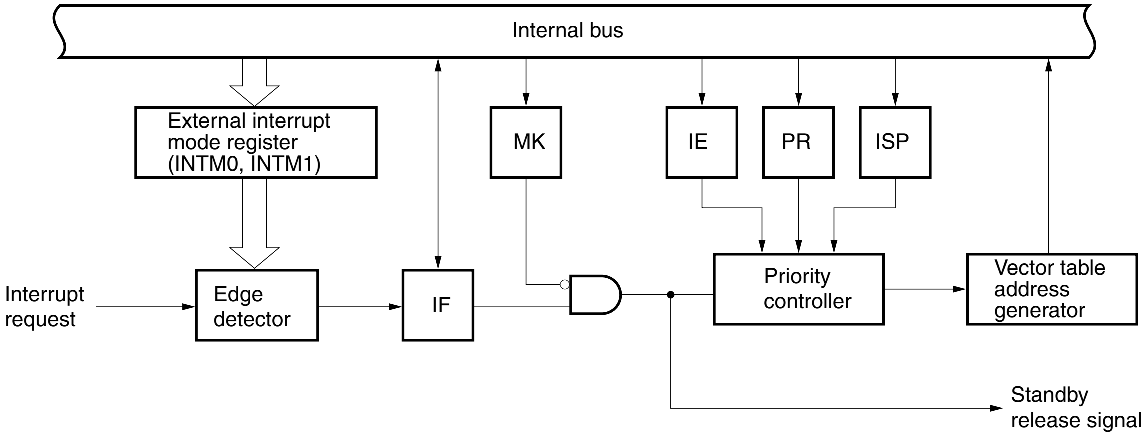
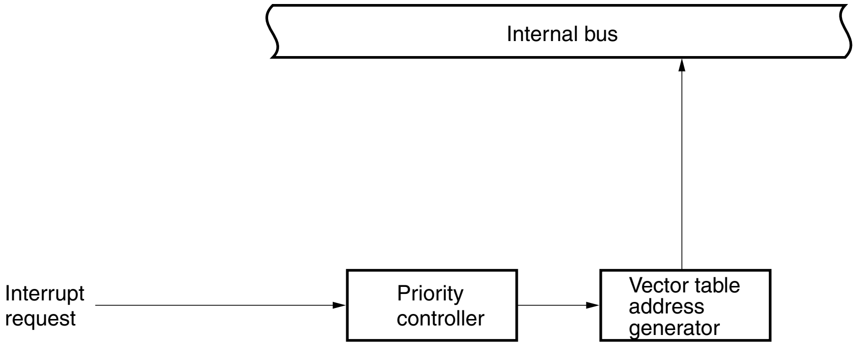


Figure 20-1. Basic Configuration of Interrupt Function (2/2)

(D) External maskable interrupt (except INTP0)



(E) Software interrupt



IF:	Interrupt request flag
IE:	Interrupt enable flag
ISP:	In-service priority flag
MK:	Interrupt mask flag
PR:	Priority specify flag

20.3 Interrupt Function Control Registers

The following six types of registers are used to control the interrupt functions.

- Interrupt request flag register (IF0L, IF0H, IF1L)
- Interrupt mask flag register (MK0L, MK0H, MK1L)
- Priority specify flag register (PR0L, PR0H, PR1L)
- External interrupt mode register (INTM0, INTM1)
- Sampling clock select register (SCS)
- Program status word (PSW)

Table 20-2 gives a listing of interrupt request flags, interrupt mask flags, and priority specify flags corresponding to interrupt request sources.

Table 20-2. Various Flags Corresponding to Interrupt Request Sources

Interrupt Source	Interrupt Request Flag		Interrupt Mask Flag		Priority Specify Flag	
		Register		Register		Register
INTWDT	TMIF4	IF0L	TMMK4	MK0L	TMPR4	PR0L
INTP0	PIF0		PMK0		PPR0	
INTP1	PIF1		PMK1		PPR1	
INTP2	PIF2		PMK2		PPR2	
INTP3	PIF3		PMK3		PPR3	
INTP4	PIF4		PMK4		PPR4	
INTP5	PIF5		PMK5		PPR5	
INTCSI0	CSIIF0	IF0H	CSIMK0	MK0H	CSIPR0	PR0H
INTSER	SERIF		SERMK		SERPR	
INTSR/INTCSI2	SRIF		SRMK		SRPR	
INTST	STIF		STMK		STPR	
INTTM3	TMIF3		TMMK3		TMPR3	
INTTM00	TMIF00		TMMK00		TMPR00	
INTTM01	TMIF01		TMMK01		TMPR01	
INTTM1	TMIF1	IF1L	TMMK1	MK1L	TMPR1	PR1L
INTTM2	TMIF2		TMMK2		TMPR2	
INTAD	ADIF		ADMK		ADPR	
INTCSI3	CSIIF3		CSIMK3		CSIPR3	

(1) Interrupt request flag registers (IF0L, IF0H, IF1L)

The interrupt request flag is set to 1 when the corresponding interrupt request is generated or an instruction is executed. It is cleared to 0 when an instruction is executed upon acknowledgment of an interrupt request or upon application of $\overline{\text{RESET}}$ input.

IF0L, IF0H, and IF1L are set with a 1-bit or 8-bit memory manipulation instruction. If IF0L and IF0H are used as a 16-bit register IF0, use a 16-bit memory manipulation instruction for the setting.

$\overline{\text{RESET}}$ input clears these registers to 00H.

Figure 20-2. Interrupt Request Flag Register Format

Symbol	7	⑥	⑤	④	③	②	①	⓪	Address	After Reset	R/W
IF0L	0	PIF5	PIF4	PIF3	PIF2	PIF1	PIF0	TMIF4	FFE0H	00H	R/W
	⑦	⑥	⑤	④	③	②	1	⓪			
IF0H	TMIF01	TMIF00	TMIF3	STIF	SRIF	SERIF	0	CSIIF0	FFE1H	00H	R/W
	⑦	6	5	4	③	②	①	⓪			
IF1L	WTIF[Note]	0	0	0	CSIIF3	ADIF	TMIF2	TMIF1	FFE2H	00H	R/W

××IF	Interrupt Request Flag
0	No interrupt request signal
1	Interrupt request signal is generated; Interrupt request state

Note WTIF is test input flag. Vectored interrupt request is not generated.

Cautions 1. TMIF4 flag is R/W enabled only when a watchdog timer is used as an interval timer. If a watchdog timer is used in watchdog timer mode 1, set TMIF4 flag to 0.

2. Set always 0 in IF1L bits 4 to 6, IF0L bit 7, and IF0H bit 1.

<R> **3. When manipulating a flag of the interrupt request flag register, use a 1-bit memory manipulation instruction (CLR1). When describing in C language, use a bit manipulation instruction such as "IF0L.0 = 0;" or "_asm("clr1 IF0L, 0");" because the compiled assembler must be a 1-bit memory manipulation instruction (CLR1).**

If a program is described in C language using an 8-bit memory manipulation instruction such as "IF0L &= 0xfe;" and compiled, it becomes the assembler of three instructions.

```
mov a, IF0L
and a, #0FEH
mov IF0L, a
```

In this case, even if the request flag of another bit of the same interrupt request flag register (IF0L) is set to 1 at the timing between "mov a, IF0L" and "mov IF0L, a", the flag is cleared to 0 at "mov IF0L, a". Therefore, care must be exercised when using an 8-bit memory manipulation instruction in C language.

(2) Interrupt mask flag registers (MK0L, MK0H, MK1L)

The interrupt mask flag is used to enable/disable the corresponding maskable interrupt service.

MK0L, MK0H, and MK1L are set with a 1-bit or 8-bit memory manipulation instruction. If MK0L and MK0H are used as a 16-bit register MK0, use a 16-bit memory manipulation instruction for the setting.

$\overline{\text{RESET}}$ input sets these registers to FFH.

Figure 20-3. Interrupt Mask Flag Register Format

Symbol	7	⑥	⑤	④	③	②	①	⓪	Address	After Reset	R/W
MK0L	1	PMK5	PMK4	PMK3	PMK2	PMK1	PMK0	TMMK4	FFE4H	FFH	R/W
	⑦	⑥	⑤	④	③	②	1	⓪			
MK0H	TMMK01	TMMK00	TMMK3	STMK	SRMK	SERMK	1	CSIMK0	FFE5H	FFH	R/W
	⑦	6	5	4	③	②	①	⓪			
MK1L	WTMK[Note]	1	1	1	CSIMK3	ADMK	TMMK2	TMMK1	FFE6H	FFH	R/W

××MK	Interrupt Servicing Control
0	Interrupt servicing enabled
1	Interrupt servicing disabled

Note WTMK controls standby mode release enable/disable. This bit does not control the interrupt function.

Cautions 1. If TMMK4 flag is read when a watchdog timer is used in watchdog timer mode 1, MK0 value becomes undefined.

2. Because port 0 has a dual function as the external interrupt request input, when the output level is changed by specifying the output mode of the port function, an interrupt request flag is set. Therefore, 1 should be set in the interrupt mask flag before using the output mode.

3. Set always 1 in MK1L bits 4 to 6, MK0L bit 7, and MK0H bit 1.

(3) Priority specify flag registers (PR0L, PR0H, and PR1L)

The priority specify flag is used to set the corresponding maskable interrupt priority orders.

PR0L, PR0H, and PR1L are set with a 1-bit or 8-bit memory manipulation instruction. If PR0L and PR0H are used as a 16-bit register PR0, use a 16-bit memory manipulation instruction for the setting.

$\overline{\text{RESET}}$ input sets these registers to FFH.

Figure 20-4. Priority Specify Flag Register Format

Symbol	7	⑥	⑤	④	③	②	①	⓪	Address	After Reset	R/W
PR0L	1	PPR5	PPR4	PPR3	PPR2	PPR1	PPR0	TMPR4	FFE8H	FFH	R/W
	⑦	⑥	⑤	④	③	②	1	⓪			
PR0H	TMPR01	TMPR00	TMPR3	STPR	SRPR	SERPR	1	CSIPR0	FFE9H	FFH	R/W
	7	6	5	4	③	②	①	⓪			
PR1L	1	1	1	1	CSIPR3	ADPR	TMPR2	TMPR1	FFEAH	FFH	R/W

××PR	Priority Level Selection
0	High priority level
1	Low priority level

Cautions 1. When a watchdog timer is used in watchdog timer mode 1, set 1 in TMPR4 flag.

2. Set always 1 in PR1L bits 4 to 7, PR0L bit 7, and PR0H bit 1.

(4) External interrupt mode register (INTM0, INTM1)

These registers set the valid edge for INTP0 to INTP5.

INTM0 and INTM1 are set with an 8-bit memory manipulation instruction.

$\overline{\text{RESET}}$ input clears these registers to 00H.

Figure 20-5. External Interrupt Mode Register 0 Format

Symbol	7	6	5	4	3	2	1	0	Address	After Reset	R/W
INTM0	ES31	ES30	ES21	ES20	ES11	ES10	0	0	FFECH	00H	R/W

ES31	ES30	INTP2 Valid Edge Selection
0	0	Falling edge
0	1	Rising edge
1	0	Setting prohibited
1	1	Both falling and rising edges

ES21	ES20	INTP1 Valid Edge Selection
0	0	Falling edge
0	1	Rising edge
1	0	Setting prohibited
1	1	Both falling and rising edges

ES11	ES10	INTP0 Valid Edge Selection
0	0	Falling edge
0	1	Rising edge
1	0	Setting prohibited
1	1	Both falling and rising edges

Caution Set the valid edges of the INTP0/TI00 pin after setting 16-bit timer mode control register bit 1 to bit 3 (TMC01 to TMC03) to 0, 0, 0 and stopping the timer operation.

Figure 20-6. External Interrupt Mode Register 1 Format

Symbol	7	6	5	4	3	2	1	0	Address	After Reset	R/W
INTM1	0	0	ES61	ES60	ES51	ES50	ES41	ES40	FFEDH	00H	R/W

ES61	ES60	INTP5 Valid Edge Selection
0	0	Falling edge
0	1	Rising edge
1	0	Setting prohibited
1	1	Both falling and rising edges

ES51	ES50	INTP4 Valid Edge Selection
0	0	Falling edge
0	1	Rising edge
1	0	Setting prohibited
1	1	Both falling and rising edges

ES41	ES40	INTP3 Valid Edge Selection
0	0	Falling edge
0	1	Rising edge
1	0	Setting prohibited
1	1	Both falling and rising edges

(5) Sampling clock select register (SCS)

This register is used to set the valid edge clock sampling clock to be input to INTP0. When remote controlled data reception is carried out using INTP0, digital noise is eliminated with sampling clocks.

SCS is set with an 8-bit memory manipulation instruction.

$\overline{\text{RESET}}$ input clears SCS to 00H.

Figure 20-7. Sampling Clock Select Register Format

Symbol	7	6	5	4	3	2	1	0	Address	After Reset	R/W
SCS	0	0	0	0	0	0	SCS1	SCS0	FF47H	00H	R/W

SCS1	SCS0	INTP0 Sampling Clock Selection	
		MCS = 1	MCS = 0
0	0	fxx/2^{N}	
0	1	fx/2^{7} (39.1 kHz)	fx/2^{8} (19.5 kHz)
1	0	fx/2^{5} (156.3 kHz)	fx/2^{6} (78.1 kHz)
1	1	fx/2^{6} (78.1 kHz)	fx/2^{7} (39.1 kHz)

Caution fxx/2^{N} is a clock to be supplied to the CPU and fxx/2^{5}, fxx/2^{6} and fxx/2^{7} are clocks to be supplied to the peripheral hardware. fxx/2^{N} stops in the HALT mode.

Remarks 1. N: Value (N = 0 to 4) at bits 0 to 2 (PCC0 to PCC2) of processor clock control register

2. fxx: Main system clock frequency (fx or fx/2)

3. fx: Main system clock oscillation frequency

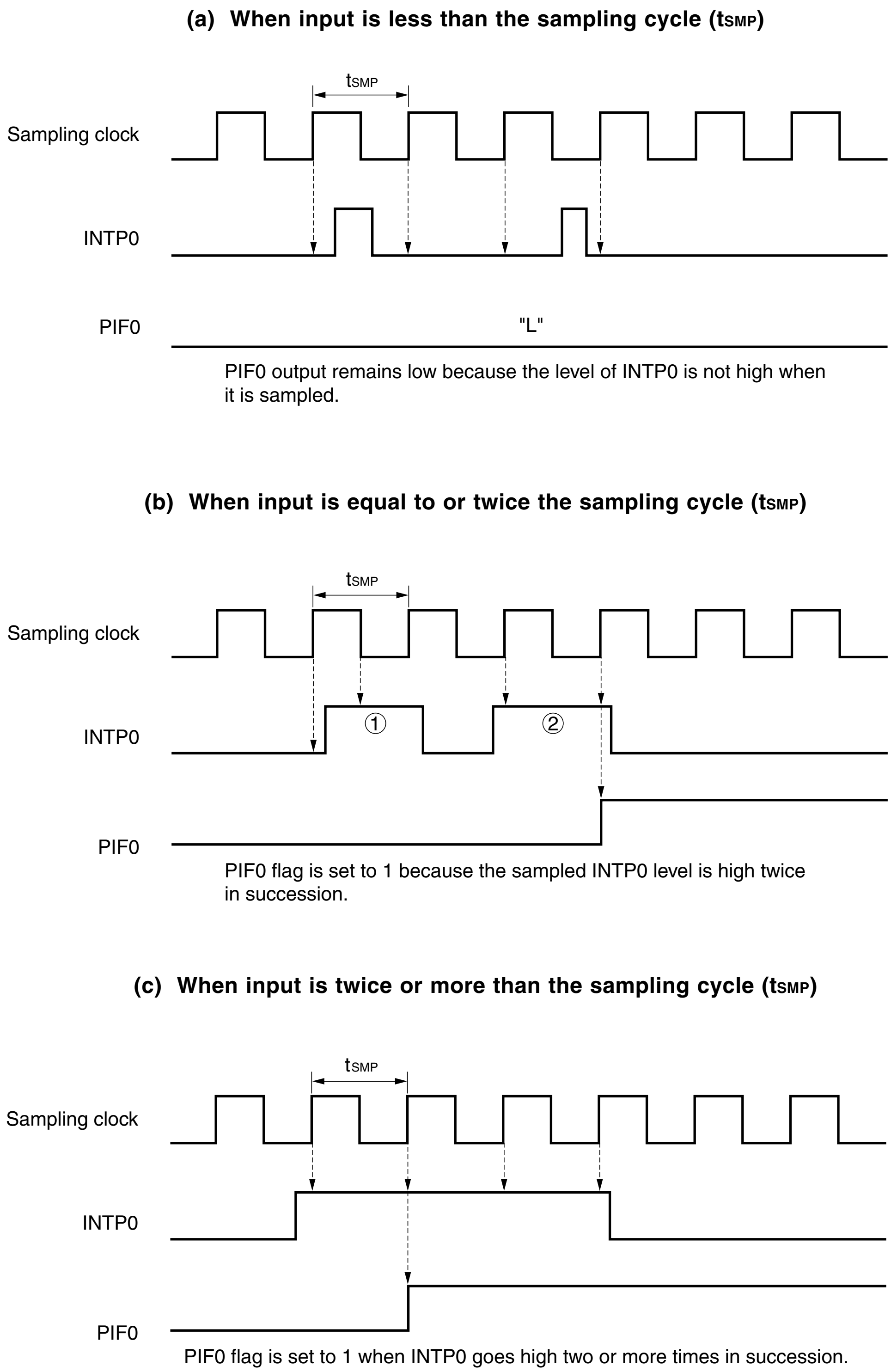
4. MCS: Oscillation mode select register bit 0

5. Values in parentheses when operated with fx = 5.0 MHz.

The noise eliminator sets the interrupt request flag (PIF0) to 1 if the input level of the sampled INTP0 is active twice in succession.

Figure 20-8 shows the noise eliminator I/O timing.

Figure 20-8. Noise Eliminator I/O Timing (During Rising Edge Detection)



(a) When input is less than the sampling cycle (t_{SMP})

PIF0 output remains low because the level of INTP0 is not high when it is sampled.

(b) When input is equal to or twice the sampling cycle (t_{SMP})

PIF0 flag is set to 1 because the sampled INTP0 level is high twice in succession.

(c) When input is twice or more than the sampling cycle (t_{SMP})

PIF0 flag is set to 1 when INTP0 goes high two or more times in succession.

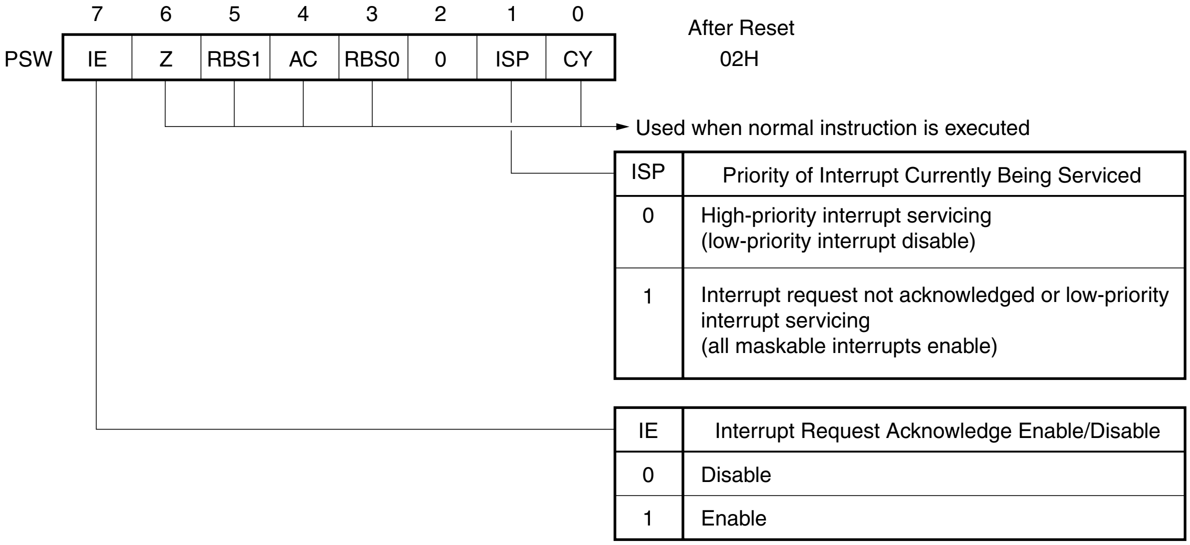
(6) Program status word (PSW)

The program status word is a register to hold the instruction execution result and the current status for interrupt request. The IE flag to set maskable interrupt request enable/disable and the ISP flag to control multiple interrupt servicing are mapped.

Besides 8-bit unit read/write, this register can carry out operations with a bit manipulation instruction and dedicated instructions (EI and DI). When a vectored interrupt request is acknowledged or when the BRK instruction is executed, the contents of the PSW are automatically saved into the stack, and the IE flag is reset to 0. If a maskable interrupt request is acknowledged contents of the priority specify flag of the acknowledged interrupt are transferred to the ISP flag. The contents of the PSW are also saved to the stack by the PUSH PSW instruction. It is reset from the stack with the RETI, RETB, and POP PSW instructions.

$\overline{\text{RESET}}$ input sets PSW to 02H.

Figure 20-9. Program Status Word Format



	7	6	5	4	3	2	1	0	After Reset
PSW	IE	Z	RBS1	AC	RBS0	0	ISP	CY	02H

ISP	Priority of Interrupt Currently Being Serviced
0	High-priority interrupt servicing (low-priority interrupt disable)
1	Interrupt request not acknowledged or low-priority interrupt servicing (all maskable interrupts enable)

IE	Interrupt Request Acknowledge Enable/Disable
0	Disable
1	Enable

20.4 Interrupt Request Servicing Operations

20.4.1 Non-maskable interrupt request acknowledge operation

A non-maskable interrupt request is unconditionally acknowledged even if in an interrupt request acknowledge disable state. It does not undergo interrupt priority control and has highest priority over all other interrupts.

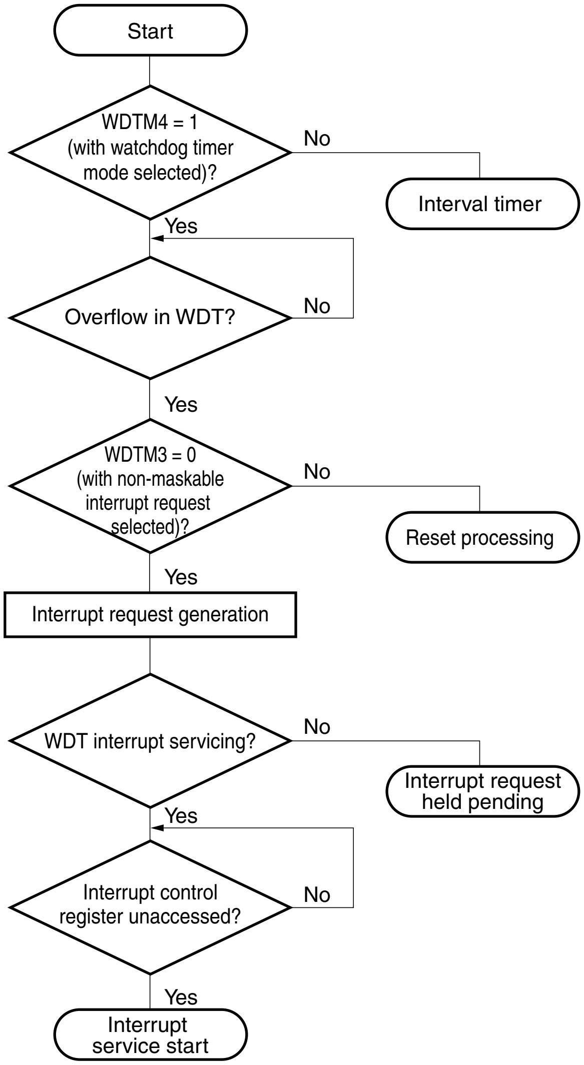
<R> If a non-maskable interrupt request is acknowledged, the contents are saved into the stacks in the order of PSW, then PC, the IE flag and ISP flag are reset (0), and the contents of the vector table are loaded into the PC and branched. Due to this, acknowledgment of multiple interrupts is prohibited.

A new non-maskable interrupt request generated during execution of a non-maskable interrupt servicing program is acknowledged after the current execution of the non-maskable interrupt servicing program is terminated (following RETI instruction execution) and one main routine instruction is executed. If a new non-maskable interrupt request is generated twice or more during non-maskable interrupt service program execution, only one non-maskable interrupt request is acknowledged after termination of the non-maskable interrupt service program execution.

Figure 20-10 shows the flowchart illustrating generation and acknowledgment of the non-maskable interrupt request. Figure 20-11 shows the timing of acknowledging the non-maskable interrupt request. Figure 20-12 illustrates how nested non-maskable interrupt requests are acknowledged.

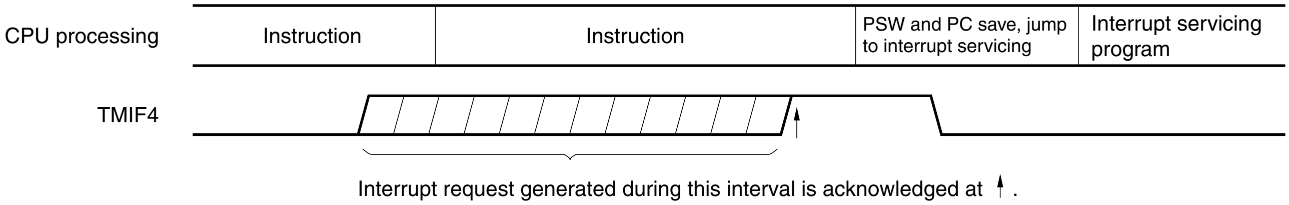
<R> **Caution Be sure to use the RETI instruction to return from a non-maskable interrupt.**

Figure 20-10. Non-Maskable Interrupt Request Acknowledge Flowchart


Start
WDTM4 = 1 (with watchdog timer mode selected)?
No
Interval timer
Yes
Overflow in WDT?
No
Yes
WDTM3 = 0 (with non-maskable interrupt request selected)?
No
Reset processing
Yes
Interrupt request generation
WDT interrupt servicing?
No
Interrupt request held pending
Yes
Interrupt control register unaccessed?
No
Yes
Interrupt service start

WDTM: Watchdog timer mode register
WDT: Watchdog timer

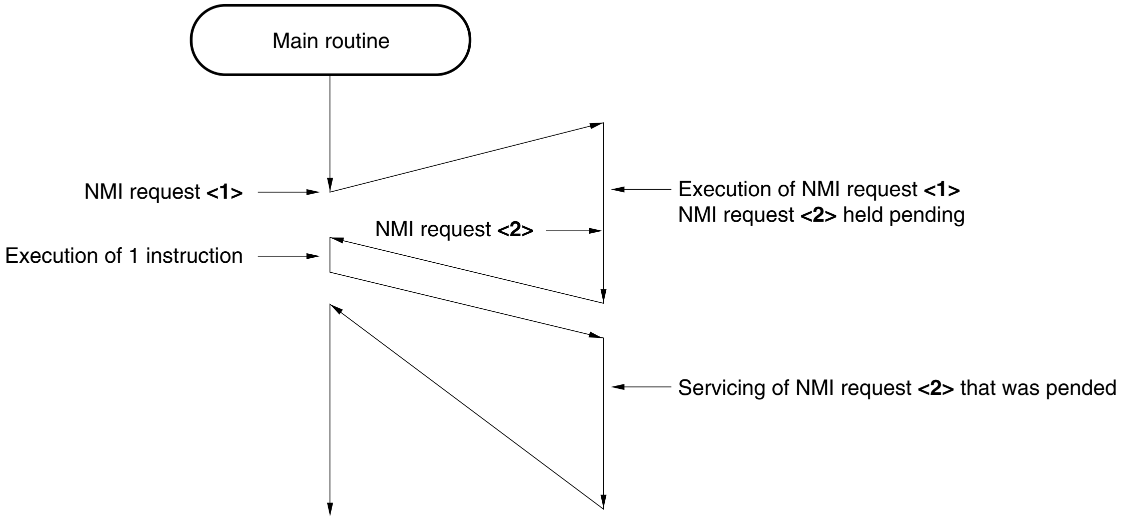
Figure 20-11. Non-Maskable Interrupt Request Acknowledge Timing


CPU processing
Instruction
Instruction
PSW and PC save, jump to interrupt servicing
Interrupt servicing program
TMIF4
Interrupt request generated during this interval is acknowledged at ↑.

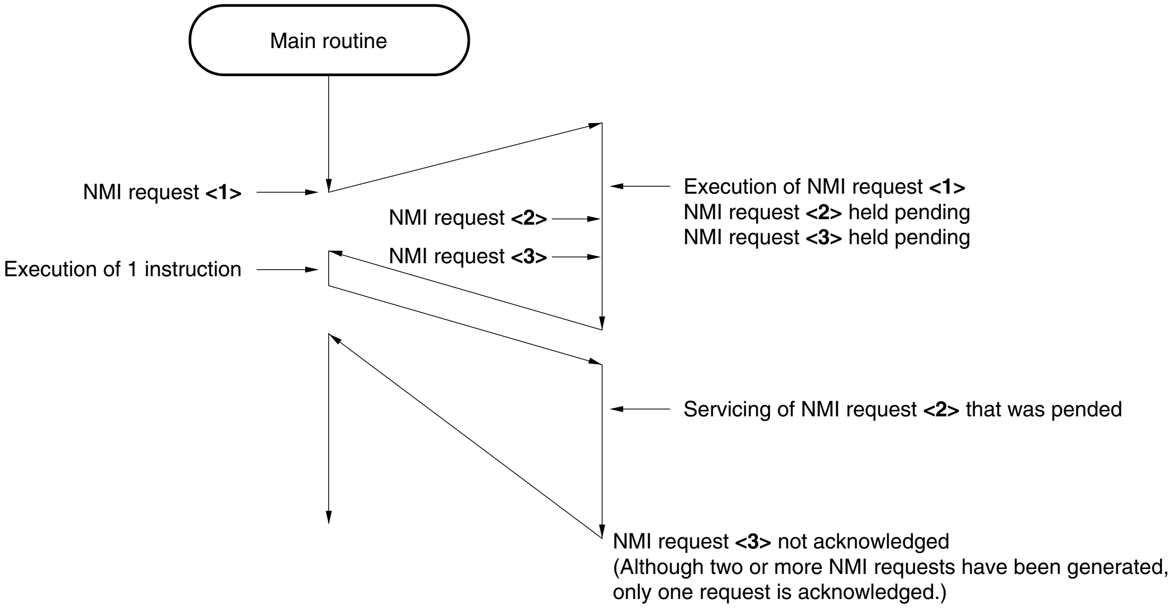
TMIF4: Watchdog timer interrupt request flag

Figure 20-12. Non-Maskable Interrupt Request Acknowledge Operation

(a) If a new non-maskable interrupt request is generated during non-maskable interrupt servicing program execution


Main routine
NMI request <1>
Execution of NMI request <1>
NMI request <2> held pending
NMI request <2>
Execution of 1 instruction
Servicing of NMI request <2> that was pended

(b) If two non-maskable interrupt requests are generated during non-maskable interrupt servicing program execution


Main routine
NMI request <1>
Execution of NMI request <1>
NMI request <2> held pending
NMI request <3> held pending
NMI request <2>
NMI request <3>
Execution of 1 instruction
Servicing of NMI request <2> that was pended
NMI request <3> not acknowledged
(Although two or more NMI requests have been generated,
only one request is acknowledged.)

20.4.2 Maskable interrupt request acknowledge operation

A maskable interrupt request becomes acknowledgeable when an interrupt request flag is set to 1 and a mask (MK) flag of the interrupt is cleared to 0. A vectored interrupt request is acknowledged in an interrupt enable state (with IE flag set to 1). However, a low-priority interrupt is not acknowledged during high-priority interrupt request service (with ISP flag reset to 0).

<R> Moreover, even if the EI instruction is executed during execution of a non-maskable interrupt servicing program, neither non-maskable interrupt requests nor maskable interrupt requests are acknowledged.

Table 20-3 shows the time required until interrupt servicing is executed since a maskable interrupt request has been generated.

For the interrupt request acknowledge timing, refer to Figures 20-14 and 20-15.

Table 20-3. Times from Maskable Interrupt Request Generation to Interrupt Service

	Minimum Time	Maximum Time[Note]
When ××PR = 0	7 clock cycles	32 clock cycles
When ××PR = 1	8 clock cycles	33 clock cycles

Note If an interrupt request is generated just before a divide instruction, the wait time is maximized.

Remark 1 clock cycle = 1/f_{CPU} (f_{CPU}: CPU clock)

If two or more maskable interrupt requests are generated simultaneously, the request specified for higher priority with the priority specify flag is acknowledged first. If the same priorities are specified by the priority specify flag, the interrupt with the highest default priority is acknowledged first.

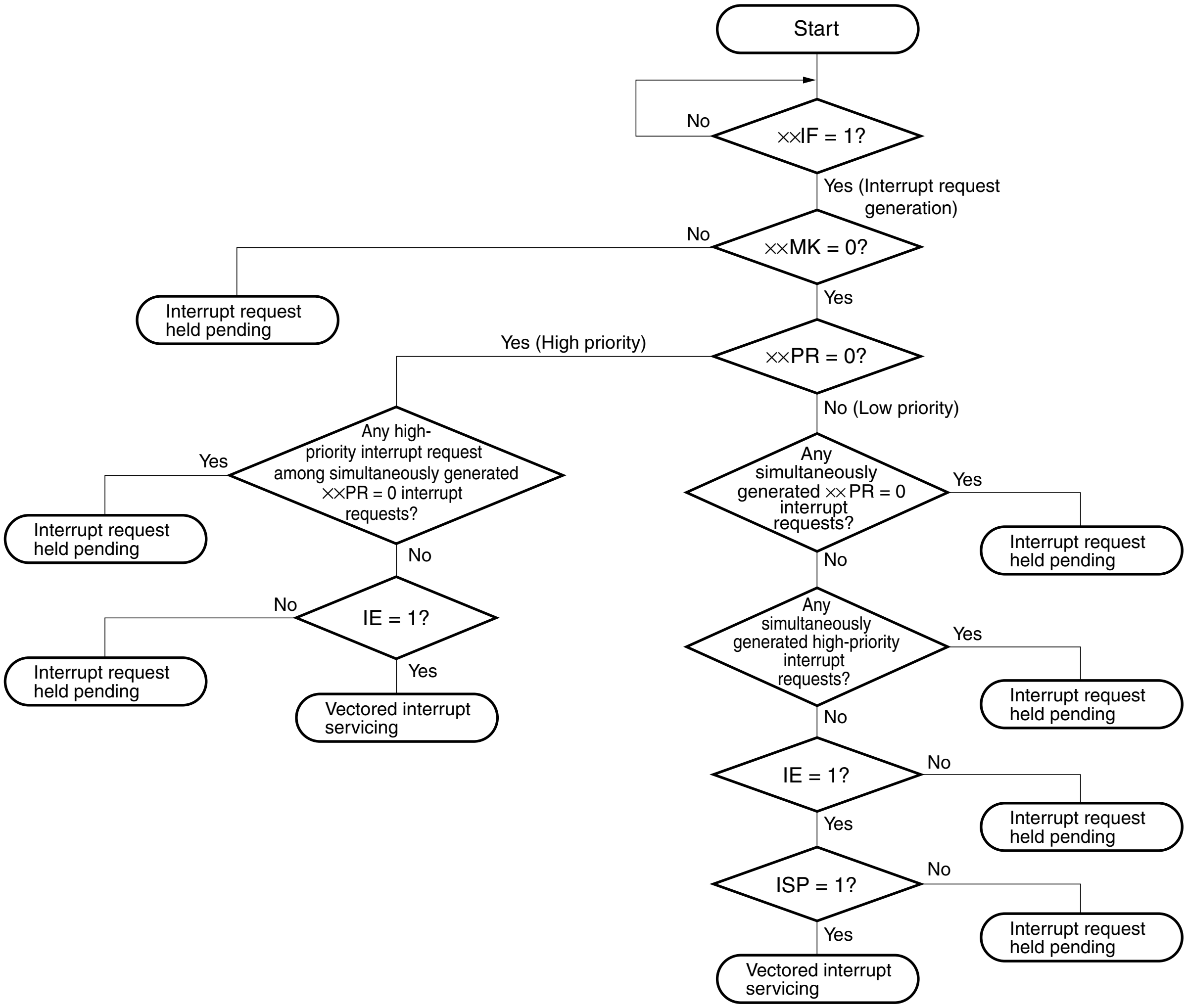
The interrupt requests that are held pending are acknowledged when they become acknowledgeable.

Figure 20-13 shows interrupt request acknowledge algorithms.

If a maskable interrupt request is acknowledged, the contents are saved in the stacks, program status word (PSW) and program counter (PC), in that order, the IE flag is reset to 0, and the acknowledged interrupt priority specify flag contents are transferred to the ISP flag. Further, the vector table data determined for each interrupt request is loaded into PC and branched.

Return from the interrupt is possible with the RETI instruction.

Figure 20-13. Interrupt Request Acknowledge Processing Algorithm


Start
××IF = 1?
No
Yes (Interrupt request generation)
××MK = 0?
No
Interrupt request held pending
Yes
××PR = 0?
Yes (High priority)
No (Low priority)
Any high-priority interrupt request among simultaneously generated ××PR = 0 interrupt requests?
Yes
Interrupt request held pending
No
IE = 1?
No
Interrupt request held pending
Yes
Vectored interrupt servicing
Any simultaneously generated ×× PR = 0 interrupt requests?
Yes
Interrupt request held pending
No
Any simultaneously generated high-priority interrupt requests?
Yes
Interrupt request held pending
No
IE = 1?
No
Interrupt request held pending
Yes
ISP = 1?
No
Interrupt request held pending
Yes
Vectored interrupt servicing

××IF: Interrupt request flag

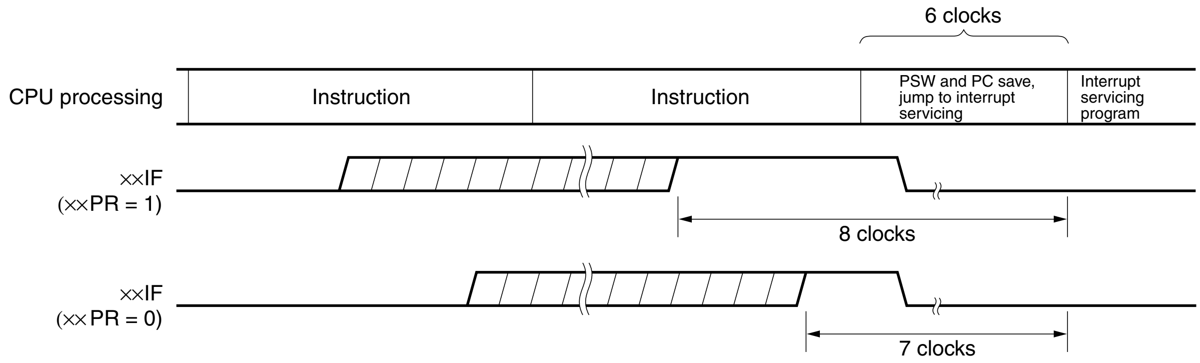
××MK: Interrupt mask flag

××PR: Priority specify flag

IE: Flag controlling acknowledgment of maskable interrupt request (1 = Enabled, 0 = Disabled)

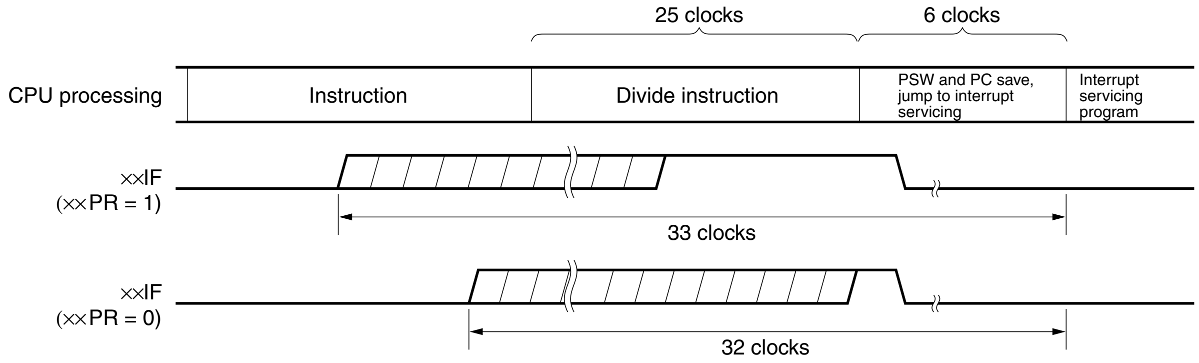
ISP: Flag indicating priority of interrupt currently being serviced (0 = Interrupt with high priority is serviced, 1 = No interrupt request is acknowledged, or interrupt with low priority is serviced).

Figure 20-14. Interrupt Request Acknowledge Timing (Minimum Time)



Remark 1 clock cycle = 1/fCPU (fCPU: CPU clock)

Figure 20-15. Interrupt Request Acknowledge Timing (Maximum Time)



Remark 1 clock cycle = 1/fCPU (fCPU: CPU clock)

20.4.3 Software interrupt request acknowledge operation

A software interrupt request is acknowledged by BRK instruction execution. Software interrupt cannot be disabled.

If a software interrupt request is acknowledged, the contents are saved in the stacks, program status word (PSW) and program counter (PC), in that order, the IE flag is reset to 0 and the contents of the vector tables (003EH and 003FH) are loaded into PC and branched.

Return from the software interrupt is possible with the RETB instruction.

Caution Do not use the RETI instruction for returning from the software interrupt.

20.4.4 Multiple interrupt request servicing

Multiple interrupts occur when another interrupt request is acknowledged during execution of an interrupt.

Multiple interrupts do not occur unless the interrupt request acknowledge enable state is selected (IE = 1) (except non-maskable interrupts). Also, when an interrupt request is received, interrupt requests acknowledge becomes disabled (IE = 0). Therefore, to enable multiple interrupts, it is necessary to set (to 1) the IE flag with the EI instruction during interrupt servicing to enable interrupt acknowledge.

Moreover, even if interrupts are enabled, multiple interrupts may not be enabled, this being subject to interrupt priority control. Two types of priority control are available: default priority control and programmable priority control. Programmable priority control is used for multiple interrupts.

In the interrupt enable state, if an interrupt request with a priority equal to or higher than that of the interrupt currently being serviced is generated, it is acknowledged for multiple interrupt servicing. If an interrupt with a priority lower than that of the interrupt currently being serviced is generated during interrupt servicing, it is not acknowledged for multiple interrupt servicing. Interrupt requests that are not enabled because of the interrupt disable state or they have a lower priority are held pending. When servicing of the current interrupt ends, the pended interrupt request is acknowledged following execution of one main processing instruction execution.

Multiple interrupt servicing is not possible during non-maskable interrupt servicing.

Table 20-4 shows interrupt requests enabled for multiple interrupt servicing, and Figure 20-14 shows multiple interrupt examples.

Table 20-4. Interrupt Request Enabled for Multiple Interrupt during Interrupt Servicing

Multiple Interrupt Request		Non-maskable Interrupt Request	Maskable Interrupt Request			
			PR = 0		PR = 1	
Interrupt Servicing			IE = 1	IE = 0	IE = 1	IE = 0
Non-maskable interrupt		D	D	D	D	D
Maskable interrupt	ISP = 0	E	E	D	D	D
	ISP = 1	E	E	D	E	D
Software interrupt		E	E	D	E	D

Remarks 1. E: Multiple interrupt enable

2. D: Multiple interrupt disable

3. ISP and IE are the flags contained in PSW.

ISP = 0: An interrupt with higher priority is being serviced

ISP = 1: An interrupt request is not acknowledged or an interrupt with lower priority is being serviced

IE = 0: Interrupt request acknowledge is disabled

IE = 1: Interrupt request acknowledge is enabled

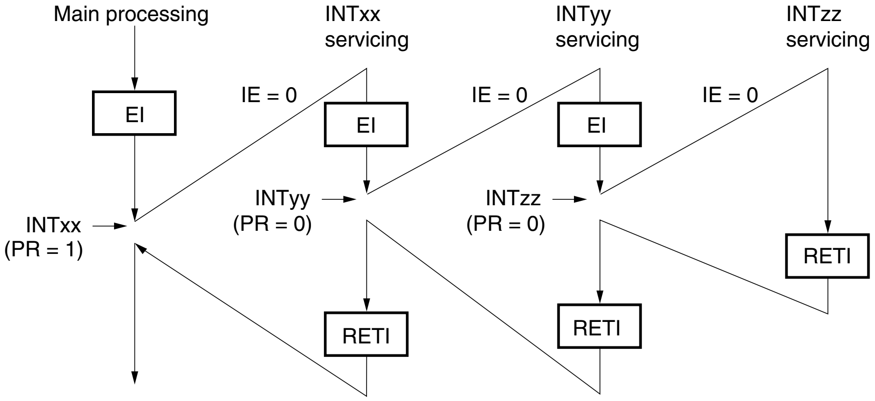
4. PR is a flag contained in PR0L, PR0H, and PR1L.

PR = 0: Higher priority level

PR = 1: Lower priority level

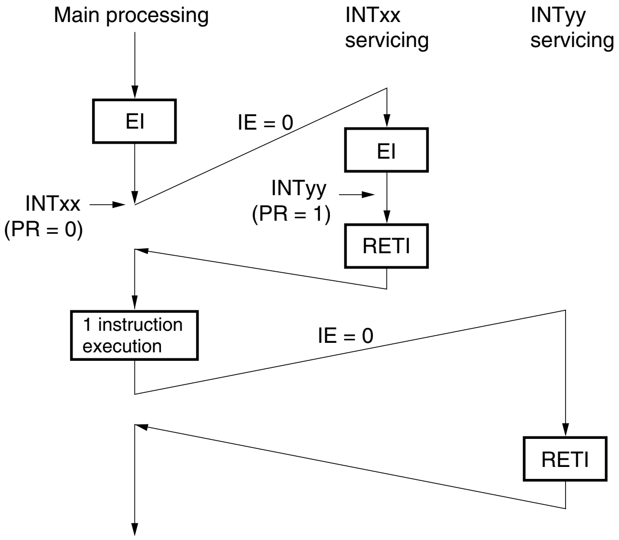
Figure 20-16. Multiple Interrupt Example (1/2)

Example 1. Two multiple interrupts generated



During interrupt INTxx servicing, two interrupt requests, INTyy and INTzz are acknowledged, and a multiple interrupt is generated. An EI instruction is issued before each interrupt request acknowledge, and the interrupt request acknowledge enable state is set.

Example 2. Multiple interrupt is not generated by priority control

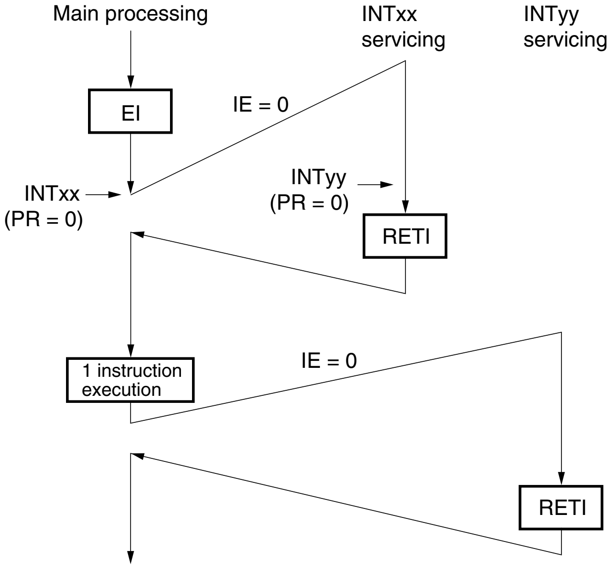


The interrupt request INTyy generated during interrupt INTxx servicing is not acknowledged because the interrupt priority is lower than that of INTxx, and a multiple interrupt is not generated. INTyy request is held pending and acknowledged after 1 instruction execution of the main processing.

PR = 0: Higher priority level
PR = 1: Lower priority level
IE = 0: Interrupt request acknowledge disable

Figure 20-16. Multiple Interrupt Example (2/2)

Example 3. A multiple interrupt is not generated because interrupts are not enabled



Because interrupts are not enabled in interrupt INTxx servicing (an EI instruction is not issued), interrupt request INTyy is not acknowledged, and a multiple interrupt is not generated. The INTyy request is held pending and acknowledged after 1 instruction execution of the main processing.

PR = 0: Higher priority level
IE = 0: Interrupt request acknowledge disable

20.4.5 Interrupt request hold

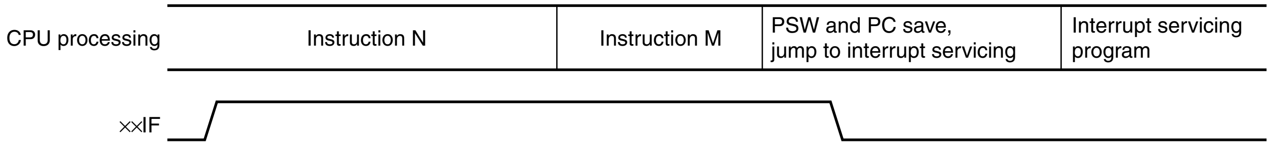
Some instructions keep an interrupt request, if any, pending until the completion of execution of the next instruction. These instructions (that keep an interrupt request pending) are listed below.

- MOV PSW, #byte
- MOV A, PSW
- MOV PSW, A
- MOV1 PSW.bit, CY
- MOV1 CY, PSW.bit
- AND1 CY, PSW.bit
- OR1 CY, PSW.bit
- XOR1 CY, PSW.bit
- SET1 PSW.bit
- CLR1 PSW.bit
- RETB
- RETI
- PUSH PSW
- POP PSW
- BT PSW.bit, $addr16
- BF PSW.bit, $addr16
- BTCLR PSW.bit, $addr16
- EI
- DI
- Manipulate instructions for IF0L, IF0H, IF1L, MK0L, MK0H, MK1L, PR0L, PR0H, PR1L, INTM0, and INTM1 registers

Caution The BRK instruction does not belong to the above group of instructions. However, the software interrupt that is started by execution of the BRK instruction clears the IE flag to 0. Therefore, even if a maskable interrupt request is generated, it is not acknowledged when the BRK instruction is executed. However, a non-maskable interrupt request is acknowledged.

The timing with which interrupt requests are held pending is shown in Figure 20-17.

Figure 20-17. Interrupt Request Hold



Remarks 1. Instruction N: Interrupt request hold instruction

2. Instruction M: Instructions other than interrupt request hold instruction

3. The ××PR (priority level) values do not affect the operation of ××IF (interrupt request).

20.5 Test Functions

The test function sets the corresponding test input flag to 1 and generates a standby release signal when the watch timer overflows and when the falling edge of port 4 is detected.

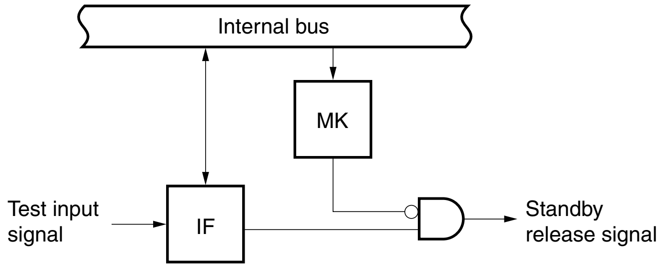
Unlike the interrupt function, this function does not perform vector processing.

There are two test input factors as shown in Table 20-5. The basic configuration is shown in Figure 20-18.

Table 20-5. Test Input Factors

Test Input Factors		Internal/ External
Name	Trigger	
INTWT	Watch timer overflow	Internal
INTPT11	Falling edge detection at port 11	External

Figure 20-18. Basic Configuration of Test Function



IF: Test input flag

MK: Test mask flag

20.5.1 Registers controlling test function

The test function is controlled by the following three registers.

- Interrupt request flag register 1L (IF1L)
- Interrupt mask flag register 1L (MK1L)
- Key return mode register (KRM)

The names of the test input flags and test mask flags corresponding to the test input signals are listed in Table 20-6.

Table 20-6. Flags Corresponding to Test Input Signals

Test Input Signal Name	Test Input Flag	Test Mask Flag
INTWT	WTIF	WTMK
INTPT11	KRIF	KRMK

(1) Interrupt request flag register 1L (IF1L)

This register indicates whether a watch timer overflow is detected or not.

IF1L is set with a 1-bit or 8-bit memory manipulation instruction.

RESET input clears IF1L to 00H.

Figure 20-19. Format of Interrupt Request Flag Register 1L

Symbol	⑦	6	5	4	③	②	①	⓪	Address	After Reset	R/W
IF1L	WTIF	0	0	0	CSIIF3	ADIF	TMIF2	TMIF1	FFE2H	00H	R/W

WTIF	Watch Timer Overflow Detection Flag
0	Not detected
1	Detected

Caution Be sure to set bits 4 to 6 to 0.

(2) Interrupt mask flag register 1L (MK1L)

This register is used to set the standby mode enable/disable at the time the standby mode is released by the watch timer.

MK1L is set with a 1-bit or 8-bit memory manipulation instruction.

RESET input sets MK1L to FFH.

Figure 20-20. Format of Interrupt Mask Flag Register 1L

Symbol	⑦	6	5	4	③	②	①	⓪	Address	After Reset	R/W
MK1L	WTMK	1	1	1	CSIMK3	ADMK	TMMK2	TMMK1	FFE6H	FFH	R/W

WTMK	Standby Mode Control by Watch Timer
0	Enables releasing the standby mode.
1	Disables releasing the standby mode.

Caution Be sure to set bits 4 to 6 to 1.

(3) Key return mode register (KRM)

This register is used to set enable/disable of standby function release by key return signal (port 11 falling edge detection), and selects port 11 falling edge input.

KRM is set with a 1-bit or 8-bit memory manipulation instruction.

$\overline{\text{RESET}}$ input sets KRM to 02H.

Figure 20-21. Key Return Mode Register Format

Symbol	7	6	5	4	3	2	①	⓪	Address	After Reset	R/W
KRM	0	0	0	0	KRM3	KRM2	KRMK	KRIF	FFB8H	02H	R/W

KRM3	KRM2	Selection of Port 11 Falling Edge Input
0	0	P117
0	1	P114 to P117
1	0	P112 to P117
1	1	P110 to P117

KRMK	Standby Mode Control by Key Return Signal
0	Standby mode release enabled
1	Standby mode release disabled

KRIF	Key Return Signal Detection Flag
0	Not detected
1	Detected (port 11 falling edge detection)

Caution When port 11 falling edge detection is used, be sure to clear KRIF to 0 (not cleared to 0 automatically).

20.5.2 Test input signal acknowledge operation

(1) Internal test input signal (INTWT)

The internal test input signal (INTWT) is generated when the watch timer overflows. This signal sets the WTIF flag. At this time, the standby release signal is generated if it is not masked by the interrupt mask flag (WTMK). By checking the WTIF flag in a cycle shorter than the overflow cycle of the watch timer, a watch function can be realized.

(2) External test input signal (INTPT4)

The external test input signal (INTPT4) is generated when the falling edge is input to the pins of port 4 (P40 to P47). As the result, the KRIF flag is set. At this time, the standby release signal is generated if it is not masked by the KRMK flag. By using port 4 to input the key return signal of a key matrix, the presence or absence of key input can be checked according to the status of the KRIF flag.

CHAPTER 21 STANDBY FUNCTION

21.1 Standby Function and Configuration

21.1.1 Standby function

The standby function is designed to decrease power consumption of the system. The following two modes are available.

(1) HALT mode

HALT instruction execution sets the HALT mode. The HALT mode is intended to stop the CPU operation clock. System clock oscillator continues oscillation. In this mode, current consumption cannot be decreased as in the STOP mode. The HALT mode is valid to restart immediately upon interrupt request and to carry out intermittent operations such as in watch applications.

(2) STOP mode

STOP instruction execution sets the STOP mode. In the STOP mode, the main system clock oscillator stops and the whole system stops. CPU current consumption can be considerably decreased.

Data memory low-voltage hold (down to V_{DD} = 1.8 V) is possible. Thus, the STOP mode is effective to hold data memory contents with ultra-low current consumption. Because this mode can be released upon interrupt request, it enables intermittent operations to be carried out.

However, because a wait time is necessary to secure an oscillation stabilization time after the STOP mode is released, select the HALT mode if it is necessary to start processing immediately upon interrupt request.

In any mode, all the contents of the register, flag and data memory just before standby mode setting are held. The I/O port output latch and output buffer statuses are also held.

Cautions 1. The STOP mode can be used only when the system operates with the main system clock (subsystem clock oscillation cannot be stopped). The HALT mode can be used with either the main system clock or the subsystem clock.

2. When proceeding to the STOP mode, be sure to stop the peripheral hardware operation and execute the STOP instruction.

3. The following sequence is recommended for power consumption reduction of the A/D converter when the standby function is used: first clear bit 7 (CS) of A/D converter mode register (ADM) to 0 to stop the A/D conversion operation, and then execute the HALT or STOP instruction.

21.1.2 Standby function control register

A wait time after the STOP mode is released upon interrupt request till the oscillation stabilizes is controlled with the oscillation stabilization time select register (OSTS).

OSTS is set with an 8-bit memory manipulation instruction.

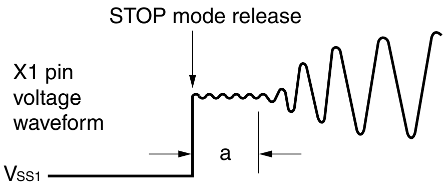
$\overline{\text{RESET}}$ input sets OSTS to 04H. However, it takes 2^{17}/fx, not 2^{18}/fx, until the STOP mode is released by $\overline{\text{RESET}}$ input.

Figure 21-1. Oscillation Stabilization Time Select Register Format

Symbol	7	6	5	4	3	2	1	0	Address	After Reset	R/W
OSTS	0	0	0	0	0	OSTS2	OSTS1	OSTS0	FFFAH	04H	R/W

OSTS2	OSTS1	OSTS0	Selection of Oscillation Stabilization Time when STOP Mode is Released	
			MCS = 1	MCS = 0
0	0	0	2^{12}/fx (819 μs)	2^{13}/fx (1.64 ms)
0	0	1	2^{14}/fx (3.28 ms)	2^{15}/fx (6.55 ms)
0	1	0	2^{15}/fx (6.55 ms)	2^{16}/fx (13.1 ms)
0	1	1	2^{16}/fx (13.1 ms)	2^{17}/fx (26.2 ms)
1	0	0	2^{17}/fx (26.2 ms)	2^{18}/fx (52.4 ms)
Other than above			Setting prohibited	

Caution The wait time after STOP mode release does not include the time (see "a" in the illustration below) from STOP mode release to clock oscillation start, regardless of release by $\overline{\text{RESET}}$ input or by interrupt request generation.



Remarks 1. fx: Main system clock oscillation frequency

2. MCS: Oscillation mode select register bit 0

3. Values in parentheses apply to operating at fx = 5.0 MHz

21.2 Standby Function Operations

21.2.1 HALT mode

(1) HALT mode set and operating status

The HALT mode is set by executing the HALT instruction. It can be set with the main system clock or the subsystem clock.

The operating status in the HALT mode is described below.

Table 21-1. HALT Mode Operating Status

HALT Mode Setting Item		HALT Execution During Main System Clock Operation		HALT Execution During Subsystem Clock Operation	
		Without Subsystem Clock[Note 1]	With Subsystem Clock[Note 2]	Main System Clock Oscillates	Main System Clock Stops
Clock generator		Both main system and subsystem clocks can be oscillated. Clock supply to the CPU stops.			
CPU		Operation stop.			
Port (output latch)		Status before HALT mode setting is held.			
16-bit timer/event counter		Operable.			Operable when watch timer output with fXT selected as count clock (fXT is selected as count clock for watch timer).
8-bit timer/event counter		Operable.			Operable when TI1 or TI2 is selected as count clock.
Watch timer		Operable if fxx/2^7 is selected as count clock.	Operable.		Operable if fXT is selected as count clock.
Watchdog timer		Operable.		Operation stops.	
A/D converter		Operable.			Operation stops.
Serial interface		Operable			Operable at external SCK.
LCD controller/driver		Operable if fxx/2^7 is selected as count clock.	Operable.		Operable if fXT is selected as count clock.
External interrupt	INTP0	Operable when a clock (fxx/2^5, fxx/2^6, fxx/2^7) for the peripheral hardware is selected as sampling clock.			Operation stops.
	INTP1 to INTP5	Operable.			

Notes 1. Including case when external clock is not supplied.

2. Including case when external clock is supplied.

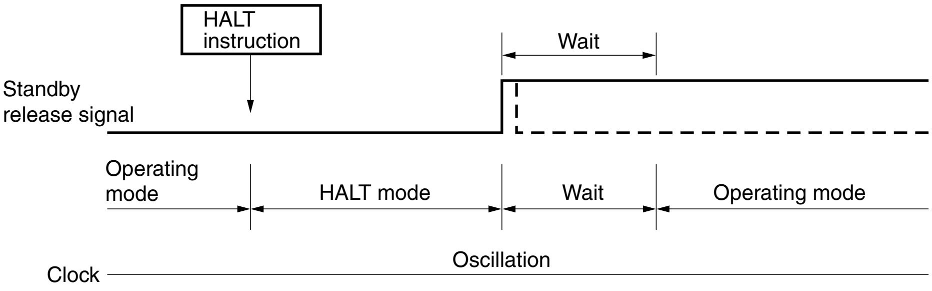
(2) HALT mode release

The HALT mode can be released with the following four types of sources.

(a) Release by unmasked interrupt request

An unmasked interrupt request is used to release the HALT mode. If interrupt request acknowledge is enabled, vectored interrupt request service is carried out. If disabled, the next address instruction is executed.

Figure 21-2. HALT Mode Release by Interrupt Request Generation



Remarks 1. The broken line indicates the case when the interrupt request which has released the standby status is acknowledged.

2. Wait time will be as follows:

- When vectored interrupt service is carried out: 8 to 9 clocks
- When vectored interrupt service is not carried out: 2 to 3 clocks

(b) Release by non-maskable interrupt request

The HALT mode is released and vectored interrupt request service is carried out whether interrupt request acknowledge is enabled or disabled.

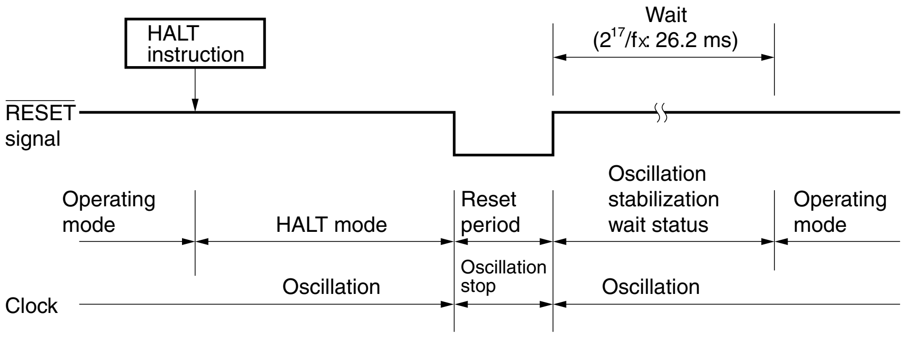
(c) Release by unmasked test input

The HALT mode is released by unmasked test input and the next address instruction of the HALT instruction is executed.

(d) Release by $\overline{\text{RESET}}$ input

As is the case with normal reset operation, a program is executed after branch to the reset vector address.

Figure 21-3. HALT Mode Release by $\overline{\text{RESET}}$ Input



Remarks 1. fx: Main system clock oscillation frequency

2. Time value in parentheses is when fx = 5.0 MHz.

Table 21-2. Operation after HALT Mode Release

Release Source	MK××	PR××	IE	ISP	Operation
Maskable interrupt request	0	0	0	×	Next address instruction execution
	0	0	1	×	Interrupt service execution
	0	1	0	1	Next address instruction execution
	0	1	×	0	
	0	1	1	1	Interrupt service execution
	1	×	×	×	HALT mode hold
Non-maskable interrupt request	–	–	×	×	Interrupt service execution
Test input	0	–	×	×	Next address instruction execution
	1	–	×	×	HALT mode hold
$\overline{\text{RESET}}$ input	–	–	×	×	Reset processing

×: don't care

21.2.2 STOP mode

(1) STOP mode set and operating status

The STOP mode is set by executing the STOP instruction. It can be set only with the main system clock.

Cautions 1. When the STOP mode is set, the X2 pin is internally connected to V_{DD1} via a pull-up resistor to minimize the leakage current at the crystal oscillator. Thus, do not use the STOP mode in a system where an external clock is used for the main system clock.

2. Because the interrupt request signal is used to release the standby mode, if there is an interrupt source with the interrupt request flag set and the interrupt mask flag reset, the standby mode is immediately released if set. Thus, the STOP mode is reset to the HALT mode immediately after execution of the STOP instruction. After the wait set using the oscillation stabilization time select register (OSTS), the operating mode is set.

The operating status in the STOP mode is described below.

Table 21-3. STOP Mode Operating Status

STOP Mode Setting Item		With Subsystem Clock	Without Subsystem Clock
Clock generator		Only main system clock stops oscillation.	
CPU		Operation stop.	
Port (output latch)		Status before STOP mode setting is held.	
16-bit timer/event counter		Operable when watch timer output with f_{XT} selected is selected as count clock (f_{XT} is selected as count clock for watch timer).	Operation stops.
8-bit timer/event counter		Operable when TI1 and TI2 are selected for the count clock.	
Watch timer		Operable when f_{XT} is selected for the count clock.	Operation stops.
Watchdog timer		Operation stops.	
A/D converter		Operation stops.	
Serial interface	Other than UART	Operable when externally supplied clock is specified as the serial clock.	
	UART	Operation stops.	
LCD controller/driver		Operable when f_{XT} is selected for the count clock.	Operation stops.
External interrupt	INTP0	Operation is impossible.	
	INTP1 to INTP5	Operable.	

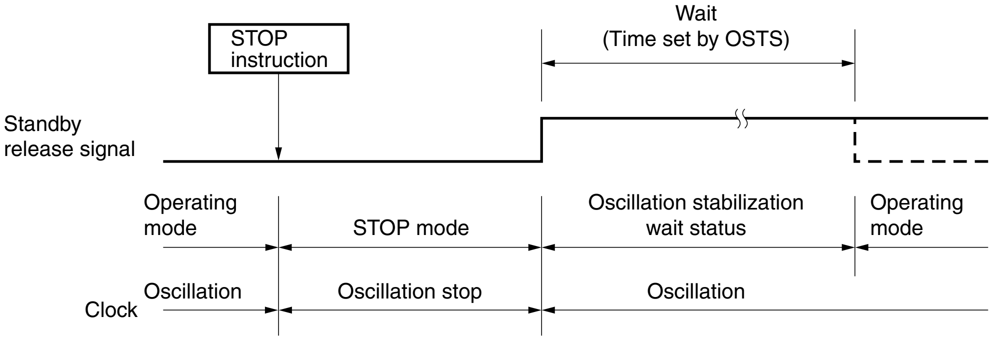
(2) STOP mode release

The STOP mode can be released with the following three types of sources.

(a) Release by unmasked interrupt request

An unmasked interrupt request is used to release the STOP mode. If interrupt request acknowledge is enabled after the lapse of oscillation stabilization time, vectored interrupt service is carried out. If interrupt request acknowledge is disabled, the next address instruction is executed.

Figure 21-4. STOP Mode Release by Interrupt Request Generation



Remark The broken line indicates the case when the interrupt request which has released the standby status is acknowledged.

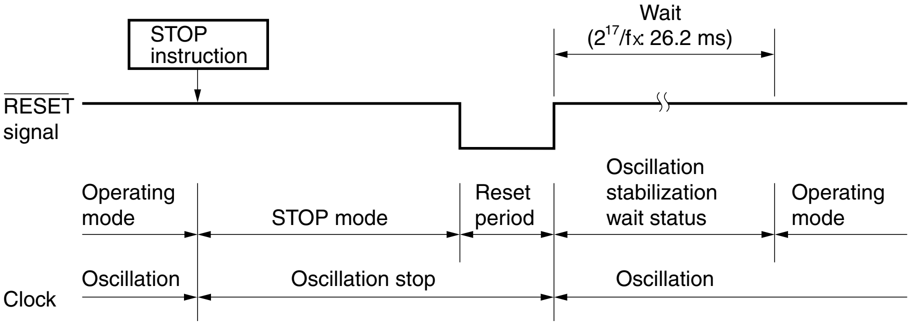
(b) Release by unmasked test input

The STOP mode is released by unmasked test input. After the lapse of oscillation stabilization time, the instruction at the next address of the STOP instruction is executed.

(c) Release by $\overline{\text{RESET}}$ input

The STOP mode is released and after the lapse of oscillation stabilization time, reset operation is carried out.

Figure 21-5. STOP Mode Release by $\overline{\text{RESET}}$ Input



Remarks 1. fx: Main system clock oscillation frequency

2. Time value in parentheses is when fx = 5.0 MHz.

Table 21-4. Operation after STOP Mode Release

Release Source	MK××	PR××	IE	ISP	Operation
Maskable interrupt request	0	0	0	×	Next address instruction execution
	0	0	1	×	Interrupt service execution
	0	1	0	1	Next address instruction execution
	0	1	×	0	
	0	1	1	1	Interrupt service execution
	1	×	×	×	STOP mode hold
Test input	0	–	×	×	Next address instruction execution
	1	–	×	×	STOP mode hold
$\overline{\text{RESET}}$ input	–	–	×	×	Reset processing

×: don't care

CHAPTER 22 RESET FUNCTION

22.1 Reset Function

The following two operations are available to generate the reset signal.

(1) External reset input with $\overline{\text{RESET}}$ pin

(2) Internal reset by watchdog timer program loop time detection

External reset and internal reset have no functional differences. In both cases, program execution starts at the address at 0000H and 0001H by $\overline{\text{RESET}}$ input.

When a low level is input to the $\overline{\text{RESET}}$ pin or the watchdog timer overflows, a reset is applied and each hardware is set to the status as shown in Table 22-1. Each pin has high impedance during reset input or during oscillation stabilization time just after reset release.

When a high level is input to the $\overline{\text{RESET}}$ input, the reset is released and program execution starts after the lapse of oscillation stabilization time (2^{17}/fx). The reset applied by watchdog timer overflow is automatically released after a reset and program execution starts after the lapse of oscillation stabilization time (2^{17}/fx) (see **Figures 22-2** to **22-4**).

Cautions 1. For an external reset, input a low level for 10 μs or more to the $\overline{\text{RESET}}$ pin.

2. During reset input, main system clock oscillation remains stopped but subsystem clock oscillation continues.

3. When the STOP mode is released by reset, the STOP mode contents are held during reset input. However, the port pin becomes high-impedance.

Figure 22-1. Block Diagram of Reset Function

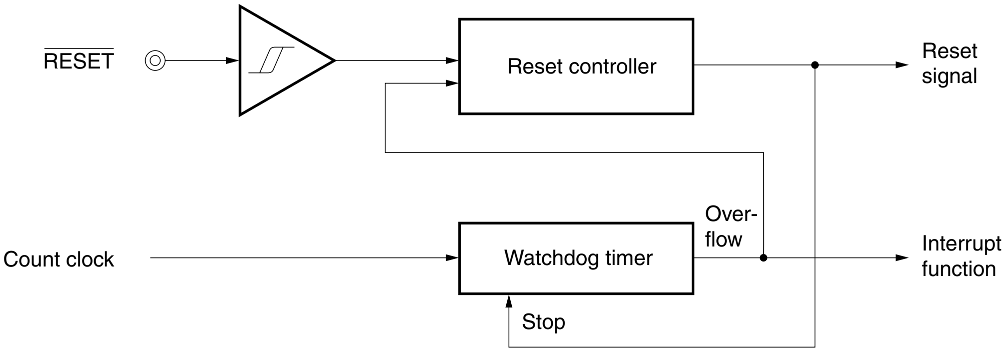


Figure 22-2. Timing of Reset Input by $\overline{\text{RESET}}$ Input

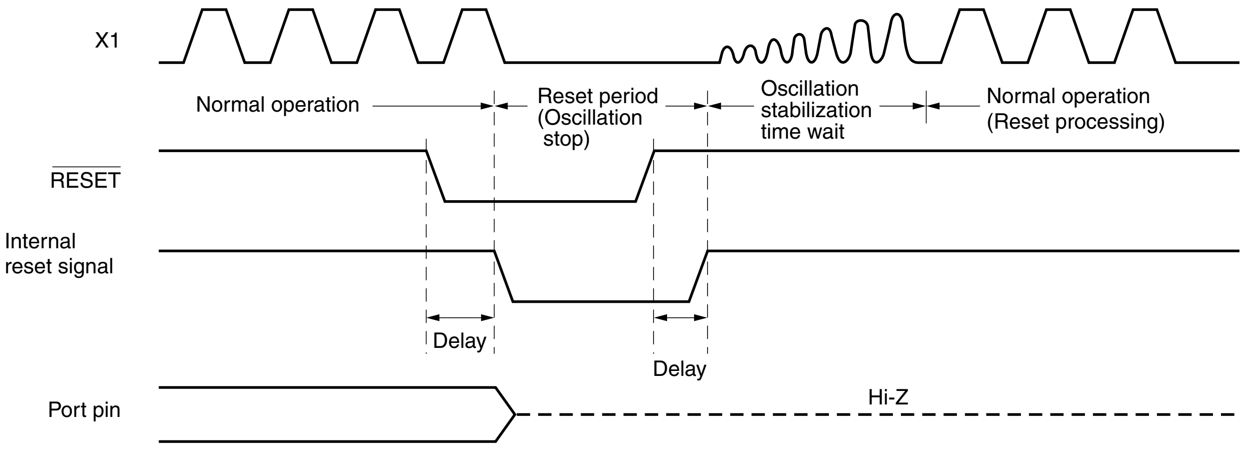

X1
Normal operation
Reset period (Oscillation stop)
Oscillation stabilization time wait
Normal operation (Reset processing)
RESET
Internal reset signal
Delay
Delay
Port pin
Hi-Z

Figure 22-3. Timing of Reset due to Watchdog Timer Overflow

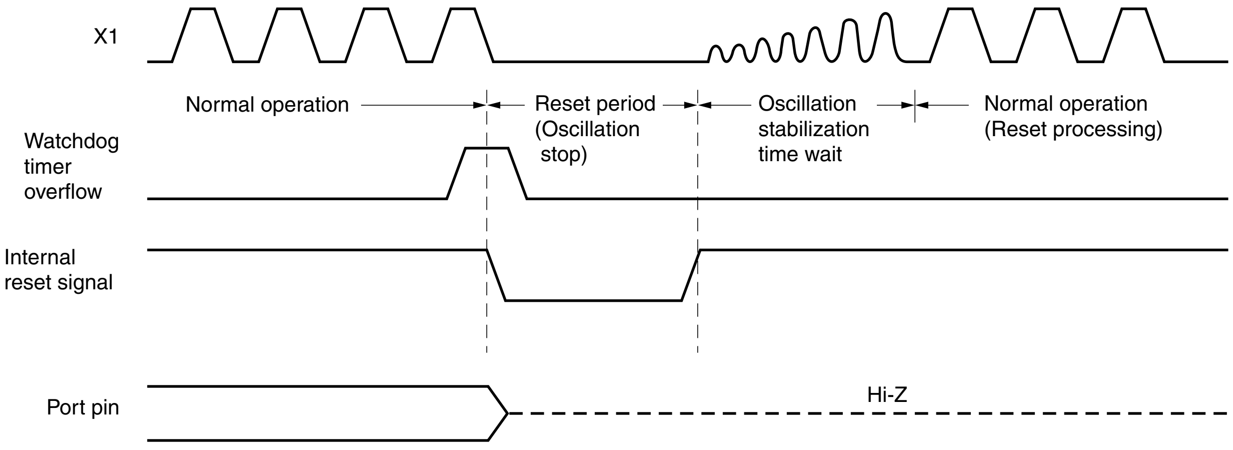

X1
Normal operation
Reset period (Oscillation stop)
Oscillation stabilization time wait
Normal operation (Reset processing)
Watchdog timer overflow
Internal reset signal
Port pin
Hi-Z

Figure 22-4. Timing of Reset Input in STOP Mode by $\overline{\text{RESET}}$ Input

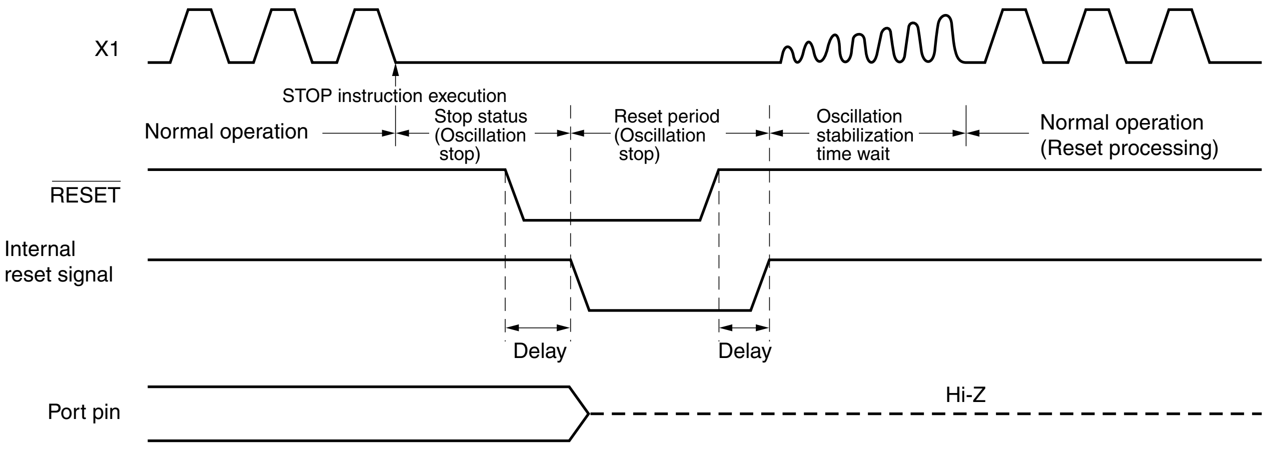

X1
STOP instruction execution
Normal operation
Stop status (Oscillation stop)
Reset period (Oscillation stop)
Oscillation stabilization time wait
Normal operation (Reset processing)
RESET
Internal reset signal
Delay
Delay
Port pin
Hi-Z

Table 22-1. Hardware Status After Reset (1/2)

Hardware		Status After Reset
Program counter (PC)[Note 1]		The contents of reset vector tables (0000H and 0001H) are set.
Stack pointer (SP)		Undefined
Program status word (PSW)		02H
RAM	Data memory	Undefined[Note 2]
	General-purpose register	Undefined[Note 2]
Ports 0 to 3, 7 to 11 (P0 to P3, P7 to P11) (output latch)		00H
Port mode register (PM0 to PM3, PM7 to PM11)		FFH
Pull-up resistor option register (PUOH, PUOL)		00H
Processor clock control register (PCC)		04H
Oscillation mode select register (OSMS)		00H
Internal memory size switching register (IMS)		Note 3
Internal expansion RAM size switching register (IXS)		0AH
Oscillation stabilization time select register (OSTS)		04H
16-bit timer/event counter	Timer register (TM0)	0000H
	Capture/compare register (CR00, CR01)	Undefined
	Clock select register (TCL0)	00H
	Mode control register (TMC0)	00H
	Capture/compare control register 0 (CRC0)	04H
	Output control register (TOC0)	00H
8-bit timer/event counter	Timer register (TM1, TM2)	00H
	Compare register (CR10, CR20)	Undefined
	Clock select register (TCL1)	00H
	Mode control register (TMC1)	00H
	Output control register (TOC1)	00H

Notes 1. During reset input or oscillation stabilization time wait, only the PC contents among the hardware statuses become undefined. All other hardware statuses remains unchanged after reset.

2. The post-reset status is held in the standby mode.

3. The values after reset depend on the product.
μPD780306, 780306Y: CCH, μPD780308, 780308Y: CFH, μPD78P0308, 78P0308Y: CFH

Table 22-1. Hardware Status After Reset (2/2)

Hardware		Status After Reset
Watch timer	Mode control register (TMC2)	00H
Watchdog timer	Clock select register (TCL2)	00H
	Mode register (WDTM)	00H
Serial interface	Clock select register (TCL3, TCL4)	88H
	Shift register (SIO0, SIO3)	Undefined
	Mode register (CSIM0, CSIM2, CSIM3)	00H
	Serial bus interface control register (SBIC)	00H
	Slave address register (SVA)	Undefined
	Asynchronous serial interface mode register (ASIM)	00H
	Asynchronous serial interface status register (ASIS)	00H
	Baud rate generator control register (BRGC)	00H
	Serial interface pin select register (SIPS)	00H
	Transmit shift register (TXS)	FFH
	Receive buffer register (RXB)	
	Interrupt timing specify register (SINT)	00H
A/D converter	Mode register (ADM)	01H
	Conversion result register (ADCR)	Undefined
	Input select register (ADIS)	00H
LCD controller/driver	Display mode register (LCDM)	00H
	Display control register (LCDC)	00H
Interrupt	Request flag register (IF0L, IF0H, IF1L)	00H
	Mask flag register (MK0L, MK0H, MK1L)	FFH
	Priority specify flag register (PR0L, PR0H, PR1L)	FFH
	External interrupt mode register (INTM0, INTM1)	00H
	Key return mode register (KRM)	02H
	Sampling clock select register (SCS)	00H

CHAPTER 23 μPD78P0308, 78P0308Y

The μPD78P0308, 78P0308Y replace the internal mask ROM of the μPD780308, 780308Y with one-time PROM or EPROM. Table 23-1 lists the differences among the μPD78P0308, 78P0308Y and the mask ROM versions (μPD780306, 780306Y, 780308, 780308Y).

Table 23-1. Differences among μPD78P0308, 78P0308Y, and Mask ROM Versions

Item	μPD78P0308, 78P0308Y	Mask ROM Versions
ROM structure	One-time PROM/EPROM	Mask ROM
ROM capacity	60 KB	μPD780306, 780306Y: 48 KB μPD780308, 780308Y: 60 KB
Changing internal ROM capacity by memory size select register	Possible[Note]	Impossible
IC pin	None	Available
V_{PP} pin	Available	None
On-chip mask option split resistors for LCD driving power supply	None	Available
Electrical characteristics	Refer to Data Sheet of individual product.	

Note The internal PROM capacity is set to 60 KB at $\overline{\text{RESET}}$.

Caution There are differences in noise immunity and noise radiation between the PROM and mask ROM versions. When pre-producing an application set with the PROM version and then mass producing it with the mask ROM version, be sure to conduct sufficient evaluations on the commercial samples (CS) (not engineering samples (ES)) of the mask ROM versions.

23.1 Internal Memory Size Switching Register

The µPD78P0308, 78P0308Y allows users to define its internal ROM and high-speed RAM sizes using the internal memory size switching register (IMS), so that the same memory mapping as that of a mask ROM version with a different-size internal ROM and high-speed RAM is possible.

IMS is set with an 8-bit memory manipulation instruction.

$\overline{\text{RESET}}$ input sets IMS to CFH.

Figure 23-1. Internal Memory Size Switching Register Format

Symbol	7	6	5	4	3	2	1	0	Address	After Reset	R/W
IMS	RAM2	RAM1	RAM0	0	ROM3	ROM2	ROM1	ROM0	FFF0H	CFH	R/W

RAM2	RAM1	RAM0	Internal High-Speed RAM Capacity Selection
1	1	0	1024 bytes
Other than above			Setting prohibited

ROM3	ROM2	ROM1	ROM0	Internal ROM Capacity Selection
1	1	0	0	48 KB
1	1	1	1	60 KB
Other than above				Setting prohibited

The IMS settings to give the same memory map as mask ROM versions are shown in Table 23-2.

Table 23-2. Examples of Internal Memory Size Switching Register Settings

Relevant Mask ROM Version	IMS Setting
µPD780306, 780306Y	CCH
µPD780308, 780308Y	CFH

23.2 Internal Expansion RAM Size Switching Register

The μPD78P0308 and 78P0308Y can select the internal expansion RAM size by using the internal expansion RAM size switching register (IXS). By setting IXS, the memory mapping of the μPD78P0308 and 78P0308Y can be made the same as that of the mask ROM versions with a different internal expansion RAM capacity.

IXS is set with an 8-bit memory manipulation instruction.

$\overline{\text{RESET}}$ input sets IXS to 0AH.

Figure 23-2. Internal Expansion RAM Size Switching Register Format

Symbol	7	6	5	4	3	2	1	0	Address	After Reset	R/W
IXS	0	0	0	0	IXRAM3	IXRAM2	IXRAM1	IXRAM0	FFF4H	0AH	W

IXRAM3	IXRAM2	IXRAM1	IXRAM0	Internal Expansion RAM Capacity Selection
1	0	1	0	1024 bytes
Other than above				Setting prohibited

The IXS settings to give the same memory map as mask ROM versions are shown in Table 23-3.

Table 23-3. Examples of Internal Expansion RAM Size Switching Register Settings

Relevant Mask ROM Version	IXS Setting
μPD780306, 780306Y	0AH
μPD780308, 780308Y	

23.3 PROM Programming

The μPD78P0308 and 78P0308Y each incorporate a 60 KB PROM as program memory. To write a program into the μPD78P0308 or 78P0308Y PROM, make the device enter the PROM programming mode by setting the levels of the V_{PP} and $\overline{\text{RESET}}$ pins as specified. For the connection of unused pins, see **(2) PROM programming mode** in **1.5** or **2.5**.

Caution Write the program in the range of addresses 0000H to EFFFH (specify the last address as EFFFH).
The program cannot be correctly written by a PROM programmer which does not have a write address specification function.

23.3.1 Operating modes

When +5 V or +12.5 V is applied to the V_{PP} pin and a low-level signal is applied to the $\overline{\text{RESET}}$ pin, the μPD78P0308 and μPD78P0308Y are set to the PROM programming mode. This is one of the operating modes shown in Table 23-4 below according to the setting of the $\overline{\text{CE}}$, $\overline{\text{OE}}$, and $\overline{\text{PGM}}$ pins.

The PROM contents can be read by setting the read mode.

Table 23-4. PROM Programming Operating Modes

Pin Operating Mode	$\overline{\text{RESET}}$	V_{PP}	V_{DD}	$\overline{\text{CE}}$	$\overline{\text{OE}}$	$\overline{\text{PGM}}$	D0 to D7
Page data latch	L	+12.5 V	+6.5 V	H	L	H	Data input
Page write				H	H	L	High impedance
Byte write				L	H	L	Data input
Program verify				L	L	H	Data output
Program inhibit				×	H	H	High impedance
				×	L	L	
Read		+5 V	+5 V	L	L	H	Data output
Output disabled				L	H	×	High impedance
Standby				H	×	×	High impedance

×: L or H

(1) Read mode

Read mode is set by setting $\overline{\text{CE}}$ to L and $\overline{\text{OE}}$ to L.

(2) Output disable mode

If $\overline{\text{OE}}$ is set to H, data output becomes high impedance and the output disable mode is set.

Therefore, if multiple μPD78P0308s or 78P0308Ys are connected to the data bus, data can be read from any one device by controlling the $\overline{\text{OE}}$ pin.

(3) Standby mode

Setting $\overline{\text{CE}}$ to H sets the standby mode.

In this mode, data output becomes high impedance irrespective of the status of $\overline{\text{OE}}$.

(4) Page data latch mode

Setting $\overline{\text{CE}}$ to H, $\overline{\text{PGM}}$ to H, and $\overline{\text{OE}}$ to L at the start of the page write mode sets the page data latch mode.

In this mode, 1-page 4-byte data is latched in the internal address/data latch circuit.

(5) Page write mode

After a 1-page 4-byte address and data are latched by the page data latch mode, a page write is executed by applying a 0.1 ms program pulse (active-low) to the $\overline{\text{PGM}}$ pin while $\overline{\text{CE}}$ = H and $\overline{\text{OE}}$ = H. After this, program verification can be performed by setting $\overline{\text{CE}}$ to L and $\overline{\text{OE}}$ to L.

If programming is not performed by one program pulse, repeated write and verify operations are executed X times ($X \leq 10$).

(6) Byte write mode

A byte write is executed by applying a 0.1 ms program pulse (active-low) to the $\overline{\text{PGM}}$ pin while $\overline{\text{CE}}$ = L and $\overline{\text{OE}}$ = H. After this, program verification can be performed by setting $\overline{\text{OE}}$ to L.

If programming is not performed by one program pulse, repeated write and verify operations are executed X times ($X \leq 10$).

(7) Program verify mode

Setting $\overline{\text{CE}}$ to L, $\overline{\text{PGM}}$ to H, and $\overline{\text{OE}}$ to L sets the program verify mode.

After writing is performed, this mode should be used to check whether the data was written correctly.

(8) Program inhibit mode

The program inhibit mode is used when the $\overline{\text{OE}}$ pins, V_{PP} pins and pins D0 to D7 of multiple µPD78P0308s or 78P0308Ys are connected in parallel and any one of these devices must be written to.

The page write mode or byte write mode described above is used to perform a write. At this time, the write is not performed on the device which has the $\overline{\text{PGM}}$ pin driven high.

23.3.2 PROM write procedure

Figure 23-3. Page Program Mode Flowchart

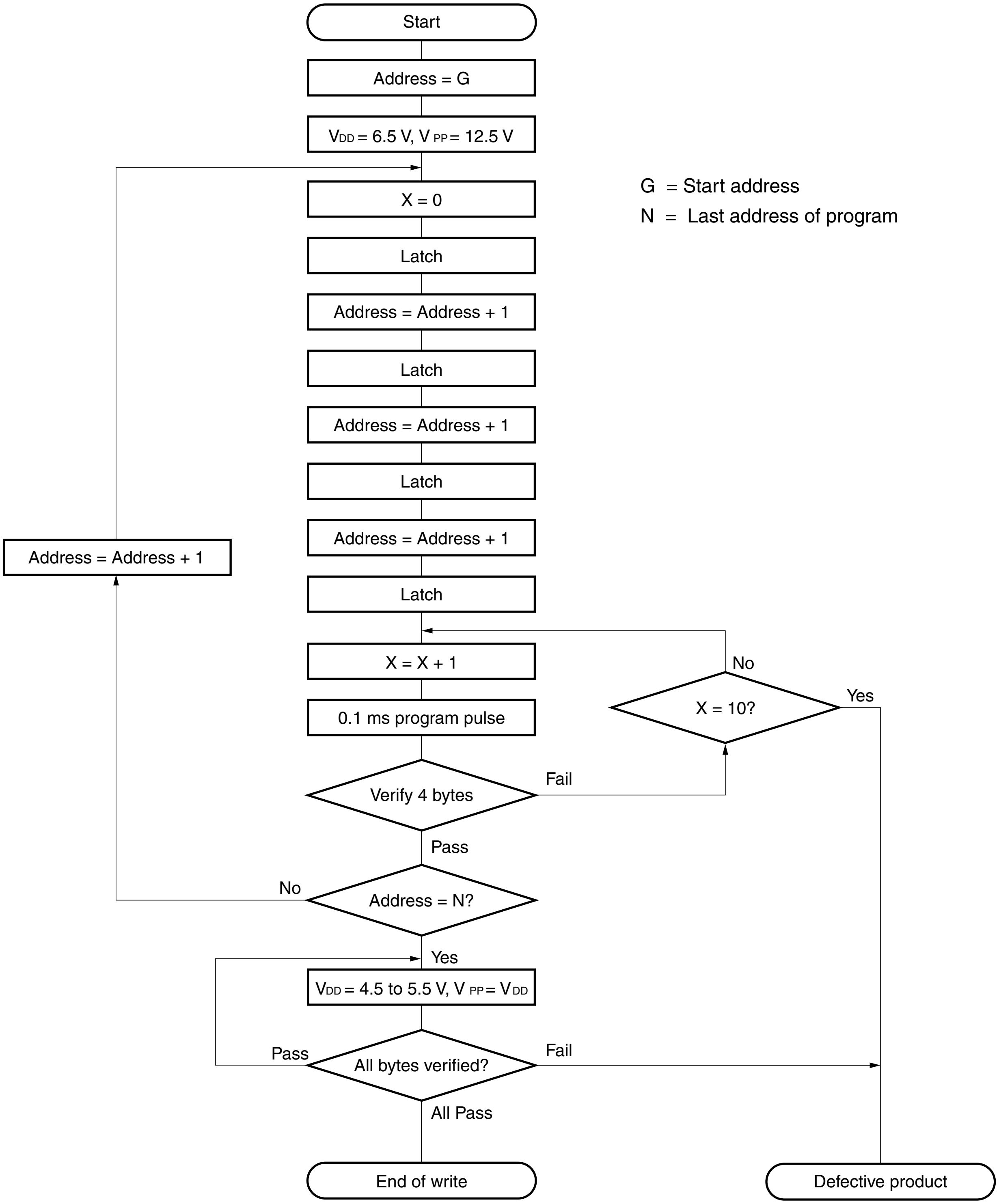


Figure 23-4. Page Program Mode Timing

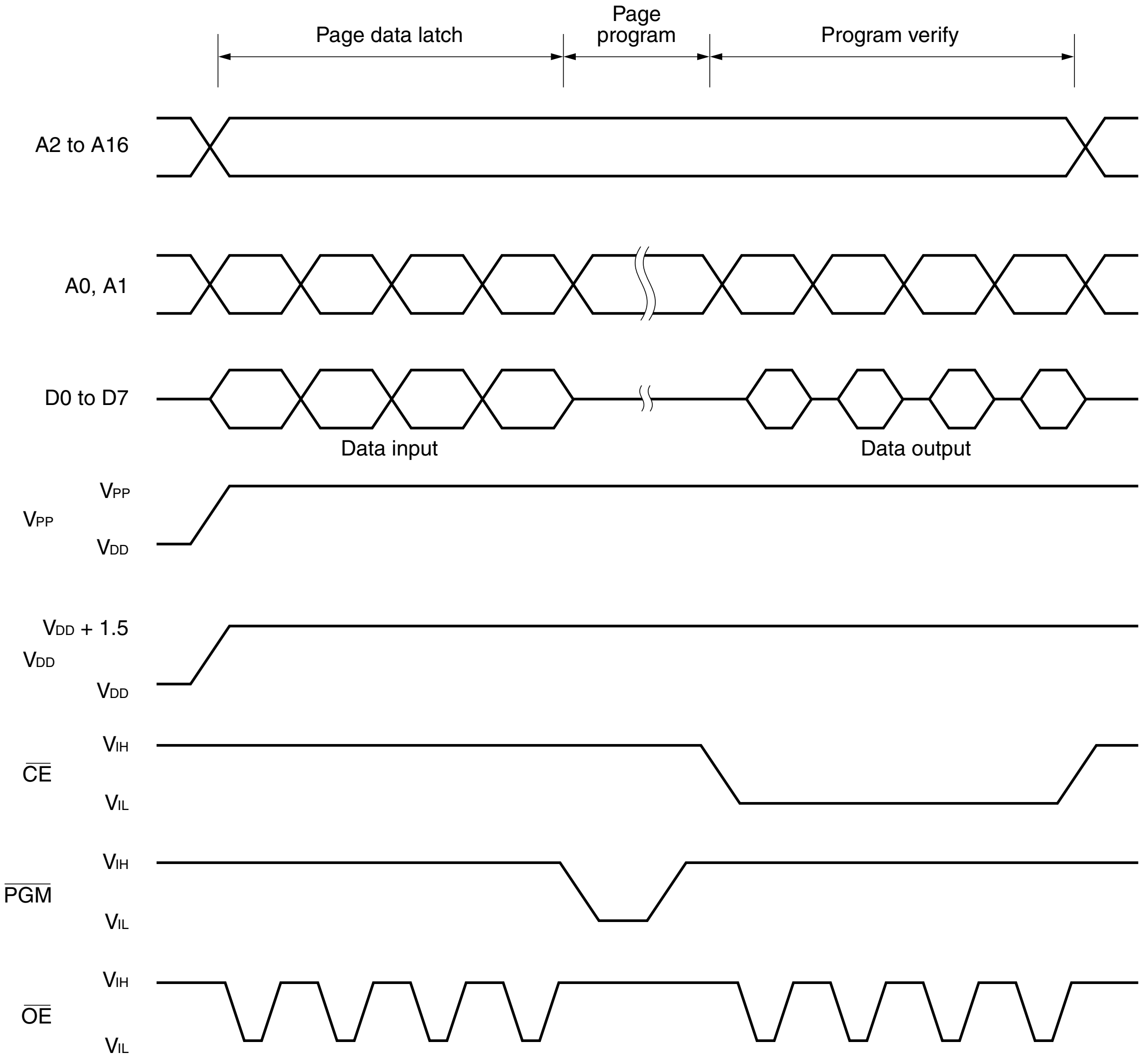


Figure 23-5. Byte Program Mode Flowchart

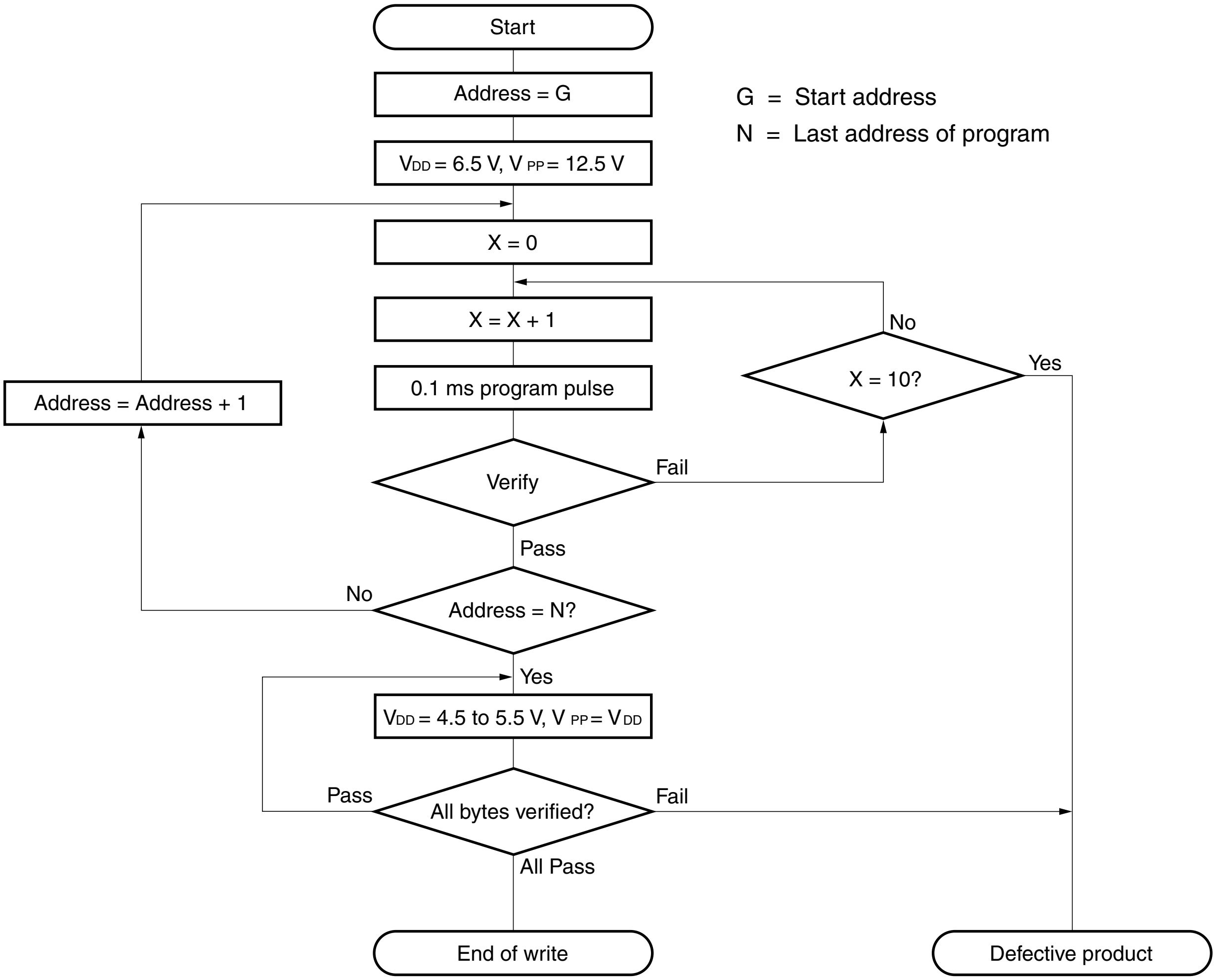
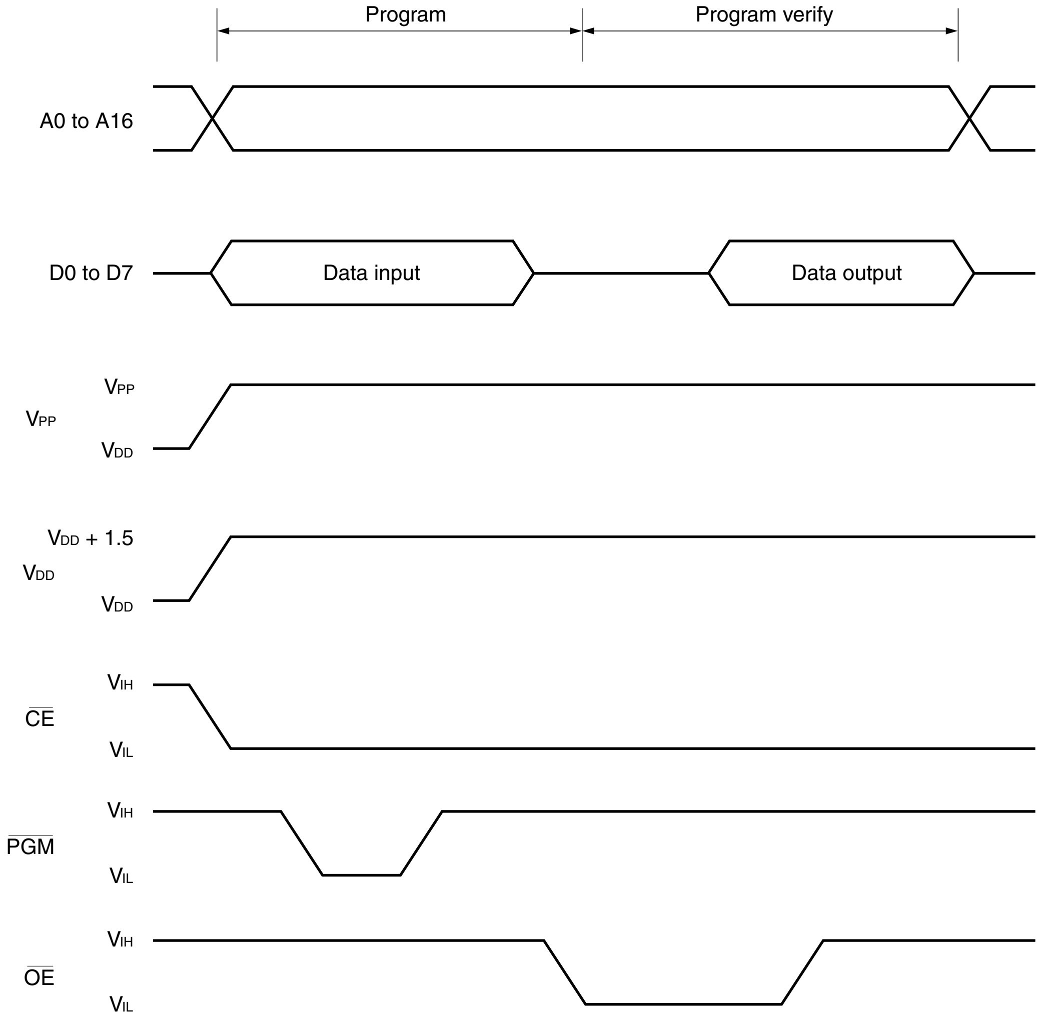


Figure 23-6. Byte Program Mode Timing



Cautions 1. Be sure to apply V_{DD} before applying V_{PP}, and cut it off after cutting V_{PP}.

2. V_{PP} must not exceed +13.5 V including overshoot voltage.

3. Disconnecting/inserting the device from/to the on-board socket while +12.5 V is being applied to the V_{PP} pin may have an adverse affect on device reliability.

23.3.3 PROM reading procedure

PROM contents can be read onto the external data bus (D0 to D7) using the following procedure.

(1) Fix the $\overline{\text{RESET}}$ pin low, and supply +5 V to the V_{PP} pin. Unused pins are handled as shown in **(2) PROM programming mode** in **1.5** or **2.5**.

(2) Supply +5 V to the V_{DD} and V_{PP} pins.

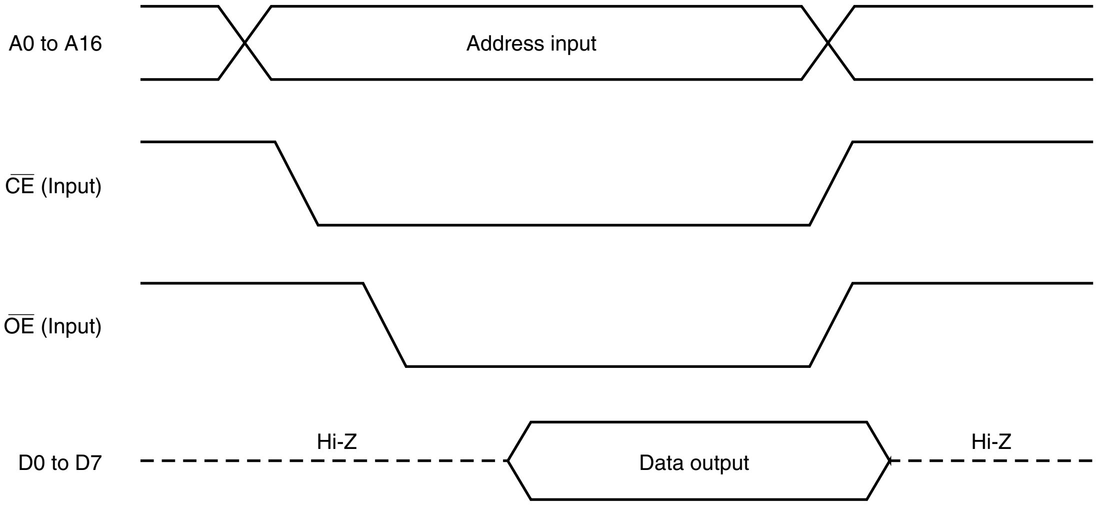
(3) Input the address of data to be read to pins A0 to A16.

(4) Read mode is entered.

(5) Data is output to pins D0 to D7.

The timing for steps (2) to (5) above is shown in Figure 23-7.

Figure 23-7. PROM Read Timing


A0 to A16
Address input
$\overline{\text{CE}}$ (Input)
$\overline{\text{OE}}$ (Input)
D0 to D7
Hi-Z
Data output
Hi-Z

23.4 Screening of One-Time PROM Versions

One-time PROM versions cannot be fully tested by NEC Electronics before shipment due to the structure of one-time PROM. Therefore, after users have written data into the PROM, screening should be implemented by user: that is, store devices at high temperature for one day as specified below, and verify their contents after the devices have returned to room temperature.

Storage Temperature	Storage Time
125°C	24 hours

CHAPTER 24 INSTRUCTION SET

This chapter describes each instruction set of the μPD780308 and 780308Y Subseries as list table. For details of its operation and operation code, refer to the **78K/0 Series Instructions User's Manual (U12326E)**.

24.1 Conventions

24.1.1 Operand identifiers and description methods

Operands are described in “Operand” column of each instruction in accordance with the description method of the instruction operand identifier (refer to the assembler specifications for detail). When there are two or more description methods, select one of them. Alphabetic letters in capitals and symbols, #, !, $ and [ ] are key words and must be described as they are. Each symbol has the following meaning.

- #: Immediate data specification
- !: Absolute address specification
- $: Relative address specification
- [ ]: Indirect address specification

In the case of immediate data, describe an appropriate numeric value or a label. When using a label, be sure to describe the #, !, $, and [ ] symbols.

For operand register identifiers, r and rp, either function names (X, A, C, etc.) or absolute names (names in parentheses in the table below, R0, R1, R2, etc.) can be used for description.

Table 24-1. Operand Identifiers and Description Methods

Identifier	Description Method
r rp sfr sfrp	X (R0), A (R1), C (R2), B (R3), E (R4), D (R5), L (R6), H (R7) AX (RP0), BC (RP1), DE (RP2), HL (RP3) Special-function register symbol**Note** Special-function register symbol (16-bit manipulatable register even addresses only)**Note**
saddr saddrp	FE20H to FF1FH Immediate data or labels FE20H to FF1FH Immediate data or labels (even address only)
addr16 addr11 addr5	0000H to FFFFH Immediate data or labels (Only even addresses for 16-bit data transfer instructions) 0800H to 0FFFH Immediate data or labels 0040H to 007FH Immediate data or labels (even address only)
word byte bit	16-bit immediate data or label 8-bit immediate data or label 3-bit immediate data or label
RBn	RB0 to RB3

Note Addresses from FFD0H to FFDFH cannot be accessed with these operands.

Remark For special-function register symbols, refer to **Table 5-3 Special-Function Register List**.

24.1.2 Description of “operation” column

A:	A register; 8-bit accumulator
X:	X register
B:	B register
C:	C register
D:	D register
E:	E register
H:	H register
L:	L register
AX:	AX register pair; 16-bit accumulator
BC:	BC register pair
DE:	DE register pair
HL:	HL register pair
PC:	Program counter
SP:	Stack pointer
PSW:	Program status word
CY:	Carry flag
AC:	Auxiliary carry flag
Z:	Zero flag
RBS:	Register bank select flag
IE:	Interrupt request enable flag
( ):	Memory contents indicated by address or register contents in parentheses
$\times_H$, $\times_L$:	Higher 8 bits and lower 8 bits of 16-bit register
$\wedge$:	Logical product (AND)
$\vee$:	Logical sum (OR)
$\forall$:	Exclusive logical sum (exclusive OR)
___:	Inverted data
addr16:	16-bit immediate data or label
jdisp8:	Signed 8-bit data (displacement value)

24.1.3 Description of “flag operation” column

(Blank):	Not affected
0:	Cleared to 0
1:	Set to 1
$\times$:	Set/cleared according to the result
R:	Previously saved value is restored

24.2 Operation List

Instruction Group	Mnemonic	Operands	Byte	Clock		Operation	Flag		
				Note 1	Note 2		Z	AC	CY
8-bit data transfer	**MOV**	r, #byte	2	4	–	r ← byte			
		saddr, #byte	3	6	7	(saddr) ← byte			
		sfr, #byte	3	–	7	sfr ← byte			
		A, r Note 3	1	2	–	A ← r			
		r, A Note 3	1	2	–	r ← A			
		A, saddr	2	4	5	A ← (saddr)			
		saddr, A	2	4	5	(saddr) ← A			
		A, sfr	2	–	5	A ← sfr			
		sfr, A	2	–	5	sfr ← A			
		A, !addr16	3	8	9	A ← (addr16)			
		!addr16, A	3	8	9	(addr16) ← A			
		PSW, #byte	3	–	7	PSW ← byte	×	×	×
		A, PSW	2	–	5	A ← PSW			
		PSW, A	2	–	5	PSW ← A	×	×	×
		A, [DE]	1	4	5	A ← (DE)			
		[DE], A	1	4	5	(DE) ← A			
		A, [HL]	1	4	5	A ← (HL)			
		[HL], A	1	4	5	(HL) ← A			
		A, [HL + byte]	2	8	9	A ← (HL + byte)			
		[HL + byte], A	2	8	9	(HL + byte) ← A			
		A, [HL + B]	1	6	7	A ← (HL + B)			
		[HL + B], A	1	6	7	(HL + B) ← A			
		A, [HL + C]	1	6	7	A ← (HL + C)			
		[HL + C], A	1	6	7	(HL + C) ← A			
	XCH	A, r Note 3	1	2	–	A ↔ r			
		A, saddr	2	4	6	A ↔ (saddr)			
		A, sfr	2	–	6	A ↔ sfr			
		A, !addr16	3	8	10	A ↔ (addr16)			
		A, [DE]	1	4	6	A ↔ (DE)			
		A, [HL]	1	4	6	A ↔ (HL)			
		A, [HL + byte]	2	8	10	A ↔ (HL + byte)			
		A, [HL + B]	2	8	10	A ↔ (HL + B)			
		A, [HL + C]	2	8	10	A ↔ (HL + C)			

Notes 1. When the internal high-speed RAM area is accessed or instruction with no data access

2. When an area except the internal high-speed RAM area is accessed.

3. Except "r = A"

Remark One instruction clock cycle is one cycle of the CPU clock (f_{CPU}) selected by the processor clock control register (PCC) register.

Instruction Group	Mnemonic	Operands	Byte	Clock		Operation	Flag		
				Note 1	Note 2		Z	AC	CY
16-bit data transfer	**MOVW**	rp, #word	3	6	–	rp ← word			
		saddrp, #word	4	8	10	(saddrp) ← word			
		sfrp, #word	4	–	10	sfrp ← word			
		AX, saddrp	2	6	8	AX ← (saddrp)			
		saddrp, AX	2	6	8	(saddrp) ← AX			
		AX, sfrp	2	–	8	AX ← sfrp			
		sfrp, AX	2	–	8	sfrp ← AX			
		AX, rp **Note 3**	1	4	–	AX ← rp			
		rp, AX **Note 3**	1	4	–	rp ← AX			
		AX, !addr16	3	10	12	AX ← (addr16)			
		!addr16, AX	3	10	12	(addr16) ← AX			
	XCHW	AX, rp **Note 3**	1	4	–	AX ↔ rp			
8-bit operation	**ADD**	A, #byte	2	4	–	A, CY ← A + byte	×	×	×
		saddr, #byte	3	6	8	(saddr), CY ← (saddr) + byte	×	×	×
		A, r **Note 4**	2	4	–	A, CY ← A + r	×	×	×
		r, A	2	4	–	r, CY ← r + A	×	×	×
		A, saddr	2	4	5	A, CY ← A + (saddr)	×	×	×
		A, !addr16	3	8	9	A, CY ← A + (addr16)	×	×	×
		A, [HL]	1	4	5	A, CY ← A + (HL)	×	×	×
		A, [HL + byte]	2	8	9	A, CY ← A + (HL + byte)	×	×	×
		A, [HL + B]	2	8	9	A, CY ← A + (HL + B)	×	×	×
		A, [HL + C]	2	8	9	A, CY ← A + (HL + C)	×	×	×
	ADDC	A, #byte	2	4	–	A, CY ← A + byte + CY	×	×	×
		saddr, #byte	3	6	8	(saddr), CY ← (saddr) + byte + CY	×	×	×
		A, r **Note 4**	2	4	–	A, CY ← A + r + CY	×	×	×
		r, A	2	4	–	r, CY ← r + A + CY	×	×	×
		A, saddr	2	4	5	A, CY ← A + (saddr) + CY	×	×	×
		A, !addr16	3	8	9	A, CY ← A + (addr16) + CY	×	×	×
		A, [HL]	1	4	5	A, CY ← A + (HL) + CY	×	×	×
		A, [HL + byte]	2	8	9	A, CY ← A + (HL + byte) + CY	×	×	×
		A, [HL + B]	2	8	9	A, CY ← A + (HL + B) + CY	×	×	×
		A, [HL + C]	2	8	9	A, CY ← A + (HL + C) + CY	×	×	×

Notes **1.** When the internal high-speed RAM area is accessed or instruction with no data access

2. When an area except the internal high-speed RAM area is accessed

3. Only when rp = BC, DE or HL

4. Except "r = A"

Remark One instruction clock cycle is one cycle of the CPU clock (f_{CPU}) selected by the processor clock control register (PCC) register.

Instruction Group	Mnemonic	Operands	Byte	Clock		Operation	Flag		
				Note 1	Note 2		Z	AC	CY
8-bit operation	**SUB**	A, #byte	2	4	–	A, CY ← A – byte	×	×	×
		saddr, #byte	3	6	8	(saddr), CY ← (saddr) – byte	×	×	×
		A, r Note 3	2	4	–	A, CY ← A – r	×	×	×
		r, A	2	4	–	r, CY ← r – A	×	×	×
		A, saddr	2	4	5	A, CY ← A – (saddr)	×	×	×
		A, !addr16	3	8	9	A, CY ← A – (addr16)	×	×	×
		A, [HL]	1	4	5	A, CY ← A – (HL)	×	×	×
		A, [HL + byte]	2	8	9	A, CY ← A – (HL + byte)	×	×	×
		A, [HL + B]	2	8	9	A, CY ← A – (HL + B)	×	×	×
		A, [HL + C]	2	8	9	A, CY ← A – (HL + C)	×	×	×
	SUBC	A, #byte	2	4	–	A, CY ← A – byte – CY	×	×	×
		saddr, #byte	3	6	8	(saddr), CY ← (saddr) – byte – CY	×	×	×
		A, r Note 3	2	4	–	A, CY ← A – r – CY	×	×	×
		r, A	2	4	–	r, CY ← r – A – CY	×	×	×
		A, saddr	2	4	5	A, CY ← A – (saddr) – CY	×	×	×
		A, !addr16	3	8	9	A, CY ← A – (addr16) – CY	×	×	×
		A, [HL]	1	4	5	A, CY ← A – (HL) – CY	×	×	×
		A, [HL + byte]	2	8	9	A, CY ← A – (HL + byte) – CY	×	×	×
		A, [HL + B]	2	8	9	A, CY ← A – (HL + B) – CY	×	×	×
		A, [HL + C]	2	8	9	A, CY ← A – (HL + C) – CY	×	×	×
	AND	A, #byte	2	4	–	A ← A∧byte	×		
		saddr, #byte	3	6	8	(saddr) ← (saddr)∧byte	×		
		A, r Note 3	2	4	–	A ← A∧r	×		
		r, A	2	4	–	r ← r∧A	×		
		A, saddr	2	4	5	A ← A∧(saddr)	×		
		A, !addr16	3	8	9	A ← A∧(addr16)	×		
		A, [HL]	1	4	5	A ← A∧(HL)	×		
		A, [HL + byte]	2	8	9	A ← A∧(HL + byte)	×		
		A, [HL + B]	2	8	9	A ← A∧(HL + B)	×		
		A, [HL + C]	2	8	9	A ← A∧(HL + C)	×		

Notes 1. When the internal high-speed RAM area is accessed or instruction with no data access

2. When an area except the internal high-speed RAM area is accessed

3. Except "r = A"

Remark One instruction clock cycle is one cycle of the CPU clock (f_{CPU}) selected by the processor clock control register (PCC) register.

Instruction Group	Mnemonic	Operands	Byte	Clock		Operation	Flag		
				Note 1	Note 2		Z	AC	CY
8-bit operation	**OR**	A, #byte	2	4	–	A ← A∨byte	×		
		saddr, #byte	3	6	8	(saddr) ← (saddr)∨byte	×		
		A, r Note 3	2	4	–	A ← A∨r	×		
		r, A	2	4	–	r ← r∨A	×		
		A, saddr	2	4	5	A ← A∨(saddr)	×		
		A, !addr16	3	8	9	A ← A∨(addr16)	×		
		A, [HL]	1	4	5	A ← A∨(HL)	×		
		A, [HL + byte]	2	8	9	A ← A∨(HL + byte)	×		
		A, [HL + B]	2	8	9	A ← A∨(HL + B)	×		
		A, [HL + C]	2	8	9	A ← A∨(HL + C)	×		
	XOR	A, #byte	2	4	–	A ← A∀byte	×		
		saddr, #byte	3	6	8	(saddr) ← (saddr)∀byte	×		
		A, r Note 3	2	4	–	A ← A∀r	×		
		r, A	2	4	–	r ← r∀A	×		
		A, saddr	2	4	5	A ← A∀(saddr)	×		
		A, !addr16	3	8	9	A ← A∀(addr16)	×		
		A, [HL]	1	4	5	A ← A∀(HL)	×		
		A, [HL + byte]	2	8	9	A ← A∀(HL + byte)	×		
		A, [HL + B]	2	8	9	A ← A∀(HL + B)	×		
		A, [HL + C]	2	8	9	A ← A∀(HL + C)	×		
	CMP	A, #byte	2	4	–	A – byte	×	×	×
		saddr, #byte	3	6	8	(saddr) – byte	×	×	×
		A, r Note 3	2	4	–	A – r	×	×	×
		r, A	2	4	–	r – A	×	×	×
		A, saddr	2	4	5	A – (saddr)	×	×	×
		A, !addr16	3	8	9	A – (addr16)	×	×	×
		A, [HL]	1	4	5	A – (HL)	×	×	×
		A, [HL + byte]	2	8	9	A – (HL + byte)	×	×	×
		A, [HL + B]	2	8	9	A – (HL + B)	×	×	×
		A, [HL + C]	2	8	9	A – (HL + C)	×	×	×

Notes 1. When the internal high-speed RAM area is accessed or instruction with no data access

2. When an area except the internal high-speed RAM area is accessed

3. Except "r = A"

Remark One instruction clock cycle is one cycle of the CPU clock (f_{CPU}) selected by the processor clock control register (PCC) register.

Instruction Group	Mnemonic	Operands	Byte	Clock		Operation	Flag		
				Note 1	Note 2		Z	AC	CY
16-bit operation	**ADDW**	AX, #word	3	6	–	AX, CY ← AX + word	×	×	×
	SUBW	AX, #word	3	6	–	AX, CY ← AX – word	×	×	×
	CMPW	AX, #word	3	6	–	AX – word	×	×	×
Multiply/ divide	**MULU**	X	2	16	–	AX ← A × X			
	DIVUW	C	2	25	–	AX (Quotient), C (Remainder) ← AX ÷ C			
Increment/ decrement	**INC**	r	1	2	–	r ← r + 1	×	×	
		saddr	2	4	6	(saddr) ← (saddr) + 1	×	×	
	DEC	r	1	2	–	r ← r – 1	×	×	
		saddr	2	4	6	(saddr) ← (saddr) – 1	×	×	
	INCW	rp	1	4	–	rp ← rp + 1			
	DECW	rp	1	4	–	rp ← rp – 1			
Rotate	**ROR**	A, 1	1	2	–	(CY, $A_7 \leftarrow A_0$, $A_{m-1} \leftarrow A_m$) × 1 time			×
	ROL	A, 1	1	2	–	(CY, $A_0 \leftarrow A_7$, $A_{m+1} \leftarrow A_m$) × 1 time			×
	RORC	A, 1	1	2	–	(CY ← A_0, A_7 ← CY, $A_{m-1} \leftarrow A_m$) × 1 time			×
	ROLC	A, 1	1	2	–	(CY ← A_7, A_0 ← CY, $A_{m+1} \leftarrow A_m$) × 1 time			×
	ROR4	[HL]	2	10	12	$A_{3-0} \leftarrow (HL)_{3-0}$, $(HL)_{7-4} \leftarrow A_{3-0}$, $(HL)_{3-0} \leftarrow (HL)_{7-4}$			
	ROL4	[HL]	2	10	12	$A_{3-0} \leftarrow (HL)_{7-4}$, $(HL)_{3-0} \leftarrow A_{3-0}$, $(HL)_{7-4} \leftarrow (HL)_{3-0}$			
BCD adjust	**ADJBA**		2	4	–	Decimal Adjust Accumulator after Addition	×	×	×
	ADJBS		2	4	–	Decimal Adjust Accumulator after Subtract	×	×	×
Bit manipulate	**MOV1**	CY, saddr.bit	3	6	7	CY ← (saddr.bit)			×
		CY, sfr.bit	3	–	7	CY ← sfr.bit			×
		CY, A.bit	2	4	–	CY ← A.bit			×
		CY, PSW.bit	3	–	7	CY ← PSW.bit			×
		CY, [HL].bit	2	6	7	CY ← (HL).bit			×
		saddr.bit, CY	3	6	8	(saddr.bit) ← CY			
		sfr.bit, CY	3	–	8	sfr.bit ← CY			
		A.bit, CY	2	4	–	A.bit ← CY			
		PSW.bit, CY	3	–	8	PSW.bit ← CY	×	×	
		[HL].bit, CY	2	6	8	(HL).bit ← CY			

Notes 1. When the internal high-speed RAM area is accessed or instruction with no data access

2. When an area except the internal high-speed RAM area is accessed

Remark One instruction clock cycle is one cycle of the CPU clock (f_{CPU}) selected by the processor clock control register (PCC) register.

Instruction Group	Mnemonic	Operands	Byte	Clock		Operation	Flag		
				Note 1	Note 2		Z	AC	CY
Bit manipulate	**AND1**	CY, saddr.bit	3	6	7	CY ← CY∧(saddr.bit)			×
		CY, sfr.bit	3	–	7	CY ← CY∧sfr.bit			×
		CY, A.bit	2	4	–	CY ← CY∧A.bit			×
		CY, PSW.bit	3	–	7	CY ← CY∧PSW.bit			×
		CY, [HL].bit	2	6	7	CY ← CY∧(HL).bit			×
	OR1	CY, saddr.bit	3	6	7	CY ← CY∨(saddr.bit)			×
		CY, sfr.bit	3	–	7	CY ← CY∨sfr.bit			×
		CY, A.bit	2	4	–	CY ← CY∨A.bit			×
		CY, PSW.bit	3	–	7	CY ← CY∨PSW.bit			×
		CY, [HL].bit	2	6	7	CY ← CY∨(HL).bit			×
	XOR1	CY, saddr.bit	3	6	7	CY ← CY⊻(saddr.bit)			×
		CY, sfr.bit	3	–	7	CY ← CY⊻sfr.bit			×
		CY, A.bit	2	4	–	CY ← CY⊻A.bit			×
		CY, PSW.bit	3	–	7	CY ← CY⊻PSW.bit			×
		CY, [HL].bit	2	6	7	CY ← CY⊻(HL).bit			×
	SET1	saddr.bit	2	4	6	(saddr.bit) ← 1			
		sfr.bit	3	–	8	sfr.bit ← 1			
		A.bit	2	4	–	A.bit ← 1			
		PSW.bit	2	–	6	PSW.bit ← 1	×	×	×
		[HL].bit	2	6	8	(HL).bit ← 1			
	CLR1	saddr.bit	2	4	6	(saddr.bit) ← 0			
		sfr.bit	3	–	8	sfr.bit ← 0			
		A.bit	2	4	–	A.bit ← 0			
		PSW.bit	2	–	6	PSW.bit ← 0	×	×	×
		[HL].bit	2	6	8	(HL).bit ← 0			
	SET1	CY	1	2	–	CY ← 1			1
	CLR1	CY	1	2	–	CY ← 0			0
	NOT1	CY	1	2	–	CY ← $\overline{\text{CY}}$			×

Notes 1. When the internal high-speed RAM area is accessed or instruction with no data access

2. When an area except the internal high-speed RAM area is accessed

Remark One instruction clock cycle is one cycle of the CPU clock (f_{CPU}) selected by the processor clock control register (PCC) register.

Instruction Group	Mnemonic	Operands	Byte	Clock		Operation	Flag		
				Note 1	Note 2		Z	AC	CY
Call/return	**CALL**	!addr16	3	7	–	(SP – 1) ← (PC + 3)$_H$, (SP – 2) ← (PC + 3)$_L$, PC ← addr16, SP ← SP – 2			
	CALLF	!addr11	2	5	–	(SP – 1) ← (PC + 2)$_H$, (SP – 2) ← (PC + 2)$_L$, PC$_{15-11}$ ← 00001, PC$_{10-0}$ ← addr11, SP ← SP – 2			
	CALLT	[addr5]	1	6	–	(SP – 1) ← (PC + 1)$_H$, (SP – 2) ← (PC + 1)$_L$, PC$_H$ ← (00000000, addr5 + 1), PC$_L$ ← (00000000, addr5), SP ← SP – 2			
	BRK		1	6	–	(SP – 1) ← PSW, (SP – 2) ← (PC + 1)$_H$, (SP – 3) ← (PC + 1)$_L$, PC$_H$ ← (003FH), PC$_L$ ← (003EH), SP ← SP – 3, IE ← 0			
	RET		1	6	–	PC$_H$ ← (SP + 1), PC$_L$ ← (SP), SP ← SP + 2			
	RETI		1	6	–	PC$_H$ ← (SP + 1), PC$_L$ ← (SP), PSW ← (SP + 2), SP ← SP + 3	R	R	R
	RETB		1	6	–	PC$_H$ ← (SP + 1), PC$_L$ ← (SP), PSW ← (SP + 2), SP ← SP + 3	R	R	R
Stack manipulate	**PUSH**	PSW	1	2	–	(SP – 1) ← PSW, SP ← SP – 1			
		rp	1	4	–	(SP – 1) ← rp$_H$, (SP – 2) ← rp$_L$, SP ← SP – 2			
	POP	PSW	1	2	–	PSW ← (SP), SP ← SP + 1	R	R	R
		rp	1	4	–	rp$_H$ ← (SP + 1), rp$_L$ ← (SP), SP ← SP + 2			
	MOVW	SP, #word	4	–	10	SP ← word			
		SP, AX	2	–	8	SP ← AX			
		AX, SP	2	–	8	AX ← SP			
Unconditional branch	**BR**	!addr16	3	6	–	PC ← addr16			
		$addr16	2	6	–	PC ← PC + 2 + jdisp8			
		AX	2	8	–	PC$_H$ ← A, PC$_L$ ← X			
Conditional branch	**BC**	$addr16	2	6	–	PC ← PC + 2 + jdisp8 if CY = 1			
	BNC	$addr16	2	6	–	PC ← PC + 2 + jdisp8 if CY = 0			
	BZ	$addr16	2	6	–	PC ← PC + 2 + jdisp8 if Z = 1			
	BNZ	$addr16	2	6	–	PC ← PC + 2 + jdisp8 if Z = 0			

Notes 1. When the internal high-speed RAM area is accessed or instruction with no data access

2. When an area except the internal high-speed RAM area is accessed

Remark One instruction clock cycle is one cycle of the CPU clock (f$_{CPU}$) selected by the processor clock control register (PCC) register.

Instruction Group	Mnemonic	Operands	Byte	Clock		Operation	Flag		
				Note 1	Note 2		Z	AC	CY
Condi-tional branch	**BT**	saddr.bit, $addr16	3	8	9	PC ← PC + 3 + jdisp8 if (saddr.bit) = 1			
		sfr.bit, $addr16	4	–	11	PC ← PC + 4 + jdisp8 if sfr.bit = 1			
		A.bit, $addr16	3	8	–	PC ← PC + 3 + jdisp8 if A.bit = 1			
		PSW.bit, $addr16	3	–	9	PC ← PC + 3 + jdisp8 if PSW.bit = 1			
		[HL].bit, $addr16	3	10	11	PC ← PC + 3 + jdisp8 if (HL).bit = 1			
	BF	saddr.bit, $addr16	4	10	11	PC ← PC + 4 + jdisp8 if (saddr.bit) = 0			
		sfr.bit, $addr16	4	–	11	PC ← PC + 4 + jdisp8 if sfr.bit = 0			
		A.bit, $addr16	3	8	–	PC ← PC + 3 + jdisp8 if A.bit = 0			
		PSW.bit, $addr16	4	–	11	PC ← PC + 4 + jdisp8 if PSW.bit = 0			
		[HL].bit, $addr16	3	10	11	PC ← PC + 3 + jdisp8 if (HL).bit = 0			
	BTCLR	saddr.bit, $addr16	4	10	12	PC ← PC + 4 + jdisp8 if (saddr.bit) = 1 then reset (saddr.bit)			
		sfr.bit, $addr16	4	–	12	PC ← PC + 4 + jdisp8 if sfr.bit = 1 then reset sfr.bit			
		A.bit, $addr16	3	8	–	PC ← PC + 3 + jdisp8 if A.bit = 1 then reset A.bit			
		PSW.bit, $addr16	4	–	12	PC ← PC + 4 + jdisp8 if PSW.bit = 1 then reset PSW.bit	×	×	×
		[HL].bit, $addr16	3	10	12	PC ← PC + 3 + jdisp8 if (HL).bit = 1 then reset (HL).bit			
	DBNZ	B, $addr16	2	6	–	B ← B – 1, then PC ← PC + 2 + jdisp8 if B ≠ 0			
		C, $addr16	2	6	–	C ← C –1, then PC ← PC + 2 + jdisp8 if C ≠ 0			
		saddr, $addr16	3	8	10	(saddr) ← (saddr) – 1, then PC ← PC + 3 + jdisp8 if (saddr) ≠ 0			
CPU control	**SEL**	RBn	2	4	–	RBS1, 0 ← n			
	NOP		1	2	–	No Operation			
	EI		2	–	6	IE ← 1 (Enable Interrupt)			
	DI		2	–	6	IE ← 0 (Disable Interrupt)			
	HALT		2	6	–	Set HALT Mode			
	STOP		2	6	–	Set STOP Mode			

Notes 1. When the internal high-speed RAM area is accessed or instruction with no data access

2. When an area except the internal high-speed RAM area is accessed

Remark One instruction clock cycle is one cycle of the CPU clock (f_{CPU}) selected by the processor clock control register (PCC) register.

24.3 Instructions Listed by Addressing Type

(1) 8-bit instructions

MOV, XCH, ADD, ADDC, SUB, SUBC, AND, OR, XOR, CMP, MULU, DIVUW, INC, DEC, ROR, ROL, RORC, ROLC, ROR4, ROL4, PUSH, POP, DBNZ

Second Operand First Operand	#byte	A	r[Note]	sfr	saddr	!addr16	PSW	[DE]	[HL]	[HL + byte] [HL + B] [HL + C]	$addr16	1	None
A	ADD ADDC SUB SUBC AND OR XOR CMP		MOV XCH ADD ADDC SUB SUBC AND OR XOR CMP	MOV XCH	MOV XCH ADD ADDC SUB SUBC AND OR XOR CMP	MOV XCH ADD ADDC SUB SUBC AND OR XOR CMP	MOV	MOV XCH	MOV XCH ADD ADDC SUB SUBC AND OR XOR CMP	MOV XCH ADD ADDC SUB SUBC AND OR XOR CMP		ROR ROL RORC ROLC	
r	MOV	MOV ADD ADDC SUB SUBC AND OR XOR CMP											INC DEC
B, C											DBNZ		
sfr	MOV	MOV											
saddr	MOV ADD ADDC SUB SUBC AND OR XOR CMP	MOV									DBNZ		INC DEC
!addr16		MOV											
PSW	MOV	MOV											PUSH POP
[DE]		MOV											
[HL]		MOV											ROR4 ROL4
[HL + byte] [HL + B] [HL + C]		MOV											
X													MULU
C													DIVUW

Note Except r = A

(2) 16-bit instructions

MOVW, XCHW, ADDW, SUBW, CMPW, PUSH, POP, INCW, DECW

Second Operand First Operand	#word	AX	rp[Note]	sfrp	saddrp	!addr16	SP	None
AX	ADDW SUBW CMPW		MOVW XCHW	MOVW	MOVW	MOVW	MOVW	
rp	MOVW	MOVW[Note]						INCW DECW PUSH POP
sfrp	MOVW	MOVW						
saddrp	MOVW	MOVW						
!addr16		MOVW						
SP	MOVW	MOVW						

Note Only when rp = BC, DE, HL

(3) Bit manipulation instructions

MOV1, AND1, OR1, XOR1, SET1, CLR1, NOT1, BT, BF, BTCLR

Second Operand First Operand	A.bit	sfr.bit	saddr.bit	PSW.bit	[HL].bit	CY	$addr16	None
A.bit						MOV1	BT BF BTCLR	SET1 CLR1
sfr.bit						MOV1	BT BF BTCLR	SET1 CLR1
saddr.bit						MOV1	BT BF BTCLR	SET1 CLR1
PSW.bit						MOV1	BT BF BTCLR	SET1 CLR1
[HL].bit						MOV1	BT BF BTCLR	SET1 CLR1
CY	MOV1 AND1 OR1 XOR1	MOV1 AND1 OR1 XOR1	MOV1 AND1 OR1 XOR1	MOV1 AND1 OR1 XOR1	MOV1 AND1 OR1 XOR1			SET1 CLR1 NOT1

(4) Call/instructions/branch instructions

CALL, CALLF, CALLT, BR, BC, BNC, BZ, BNZ, BT, BF, BTCLR, DBNZ

Second Operand First Operand	AX	!addr16	!addr11	[addr5]	$addr16
Basic instruction	BR	CALL BR	CALLF	CALLT	BR BC BNC BZ BNZ
Compound instruction					BT BF BTCLR DBNZ

(5) Other instructions

ADJBA, ADJBS, BRK, RET, RETI, RETB, SEL, NOP, EI, DI, HALT, STOP

Absolute Maximum Ratings (T_A = 25°C)

Parameter	Symbol	Conditions		Ratings	Unit
Supply voltage	V_{DD}			–0.3 to +7.0	V
	V_{PP} Note 1			–0.3 to +13.5	V
	AV_{REF}			–0.3 to V_{DD} + 0.3	V
	AV_{SS}			–0.3 to +0.3	V
Input voltage	V_{I1}	P00 to P05, P07, P10 to P17, P25 to P27, P30 to P37, P70 to P72, P80 to P87, P90 to P97, P100 to P103, P110 to P117, X1, X2, XT2, $\overline{\text{RESET}}$		–0.3 to V_{DD} + 0.3	V
	V_{I2} Note 1	A9	PROM programming mode	–0.3 to +13.5	V
Output voltage	V_O			–0.3 to V_{DD} + 0.3	V
Analog input voltage	V_{AN}	P10 to P17	Analog input pin	AV_{SS} – 0.3 to AV_{REF} + 0.3	V
Output current, high	I_{OH}	Per pin		–10	mA
		Total for P01 to P05, P10 to P17, P25 to P27, P70 to P72, P110 to P117		–15	mA
		Total for P30 to P37, P80 to P87, P90 to P97, P100 to P103		–15	mA
Output current, low	I_{OL}	Per pin	Peak value	30	mA
			r.m.s. value	15 Note 2	mA
		Total for P01 to P05, P10 to P17, P110 to P117	Peak value	60	mA
			r.m.s. value	40 Note 2	mA
		Total for P30 to P37, P100 to P103	Peak value	140	mA
			r.m.s. value	100 Note 2	mA
		Total for P25 to P27, P70 to P72, P80 to P87, P90 to P97	Peak value	50	mA
			r.m.s. value	20 Note 2	mA
Operating ambient temperature	T_A			–40 to +85	°C
Storage temperature	T_{stg}			–65 to +150	°C

Notes 1. μPD78P0308 and 78P0308Y only.

2. The root mean square (r.m.s.) value should be calculated as follows:
[r.m.s. value] = [Peak value] $\times \sqrt{\text{Duty}}$

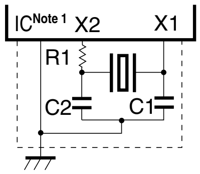
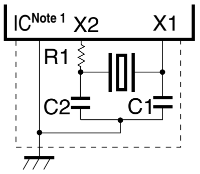
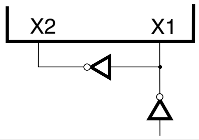
Caution Product quality may suffer if the absolute maximum rating is exceeded even momentarily for any parameter. That is, the absolute maximum ratings are rated values at which the product is on the verge of suffering physical damage, and therefore the product must be used under conditions that ensure that the absolute maximum ratings are not exceeded.

Remark Unless otherwise specified, the characteristics of alternate-function pins are the same as those of port pins.

Capacitance (T_A = 25°C, V_{DD} = V_{SS} = 0 V)

Parameter	Symbol	Conditions	MIN.	TYP.	MAX.	Unit
Input capacitance	C_{IN}	f = 1 MHz Unmeasured pins returned to 0 V.			15	pF
Output capacitance	C_{OUT}				15	pF
I/O capacitance	C_{IO}				15	pF

Main System Clock Oscillator Characteristics (T_A = –40 to +85°C, V_{DD} = 2.0[Note 5] to 5.5 V)

Resonator	Recommended Circuit	Parameter	Conditions	MIN.	TYP.	MAX.	Unit
Ceramic resonator		Oscillation frequency (f_X)[Note 2]	V_{DD} = Oscillation voltage range	1.0		5.0	MHz
		Oscillation stabilization time[Note 3]	After V_{DD} reaches oscillation voltage range MIN.			4	ms
Crystal resonator		Oscillation frequency (f_X)[Note 2]	V_{DD} = Oscillation voltage range	1		5	MHz
		Oscillation stabilization time[Note 3]	4.5 V ≤ V_{DD} ≤ 5.5 V[Note 4]			10	ms
			2.0 V ≤ V_{DD} < 4.5 V[Note 4]			30	
External clock		X1 input frequency (f_X)[Note 2]		1.0		5.0	MHz
		X1 input high-/low-level width (t_{XH}, t_{XL})		85		500	ns

Notes 1. This is V_{PP} pin in the case of μPD78P0308 and 78P0308Y.

2. Indicates only oscillator characteristics. Refer to **AC Characteristics** for instruction execution time.

3. Time required to stabilize oscillation after reset or STOP mode release.

4. After V_{DD} reaches oscillation voltage range MIN.

5. However, oscillation start voltage or higher and V_{DD} = 2.0 V or higher (for external clock, V_{DD} = 2.0 V or higher).

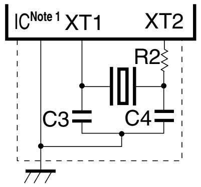
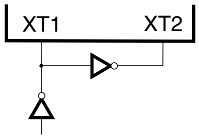
Cautions 1. When using the main system clock oscillator, wire as follows in the area enclosed by the broken lines in the above figures to avoid an adverse effect from wiring capacitance.

- **Keep the wiring length as short as possible.**
- **Do not cross the wiring with the other signal lines.**
- **Do not route the wiring near a signal line through which a high fluctuating current flows.**
- **Always make the ground point of the oscillator capacitor the same potential as V_{SS1}.**
- **Do not ground the capacitor to a ground pattern through which a high current flows.**
- **Do not fetch signals from the oscillator.**

2. When the main system clock is stopped and the system is operating on the subsystem clock, wait until the oscillation stabilization time has been secured by the program before switching back to the main system clock.

Remark For the resonator selection and oscillator constant of the μPD78P0308 and 78P0308Y, customers are required to either evaluate the oscillation themselves or apply to the resonator manufacturer for evaluation.

Subsystem Clock Oscillator Characteristics (T_A = –40 to +85°C, V_{DD} = 2.0[Note 5] to 5.5 V)

Resonator	Recommended Circuit	Parameter	Conditions	MIN.	TYP.	MAX.	Unit
Crystal resonator		Oscillation frequency (f_{XT})[Note 2]	V_{DD} = Oscillation voltage range	32	32.768	35	kHz
		Oscillation stabilization time[Note 3]	4.5 V ≤ V_{DD} ≤ 5.5 V[Ñote 4]		1.2	2	s
			2.0 V ≤ V_{DD} < 4.5 V[Note 4]			10	
External clock		XT1 input frequency (f_{XT})[Note 2]		32		100	kHz
		XT1 input high-/low-level width (t_{XTH}, t_{XTL})		5		15	μs

Notes 1. This is V_{PP} pin in the case of μPD78P0308 and 78P0308Y.

2. Indicates only oscillator characteristics. Refer to **AC Characteristics** for instruction execution time.

3. Time required to stabilize oscillation after V_{DD} reaches oscillation voltage range MIN.

4. After V_{DD} reaches oscillation voltage range MIN.

5. However, oscillation start voltage or higher and V_{DD} = 2.0 V or higher (for external clock, V_{DD} = 2.0 V or higher).

Cautions 1. When using the subsystem clock oscillator, wire as follows in the area enclosed by the broken lines in the above figure to avoid an adverse effect from wiring capacitance.

- **Keep the wiring length as short as possible.**
- **Do not cross the wiring with the other signal lines.**
- **Do not route the wiring near a signal line through which a high fluctuating current flows.**
- **Always make the ground point of the oscillator capacitor the same potential as V_{SS1}.**
- **Do not ground the capacitor to a ground pattern through which a high current flows.**
- **Do not fetch signals from the oscillator.**

2. The subsystem clock oscillator is designed as a low-amplitude circuit for reducing current consumption, and is more prone to malfunction due to noise than the main system clock oscillator. Particular care is therefore required with the wiring method when the subsystem clock is used.

Remark For the resonator selection and oscillator constant, customers are required to either evaluate the oscillation themselves or apply to the resonator manufacturer for evaluation.

Recommended Oscillator Constant

Main system clock: Ceramic resonator (T_A = –40 to +85°C): μPD780306, 780306Y, 780308, 780308Y only

Manufacturer	Product Name	Frequency	Recommended Circuit Constant			Oscillation Voltage Range	
		(MHz)	C1 (pF)	C2 (pF)	R1 (kΩ)	MIN. (V)	MAX. (V)
Matsushita Electronics Components Co., Ltd.	EFOEC2004A5	2.00	On-chip	On-chip	4.7	2.0	5.5
	EFOEC3584A4	3.58	On-chip	On-chip	0	2.0	5.5
	EFOEC4194A4	4.19	On-chip	On-chip	0	2.0	5.5
	EFOEC4914A4	4.91	On-chip	On-chip	0	2.0	5.5
	EFOEC5004A4	5.00	On-chip	On-chip	0	2.0	5.5
TDK Corp.	CCR1000K2	1.00	150	150	0	2.0	5.5
	CCR3.58MC3	3.58	On-chip	On-chip	0	2.0	5.5
	CCR4.19MC3	4.19	On-chip	On-chip	0	2.0	5.5
	CCR4.91MC3	4.91	On-chip	On-chip	0	2.0	5.5
	CCR5.0MC3	5.00	On-chip	On-chip	0	2.0	5.5
Murata Mfg. Co., Ltd.	CSB1000J	1.00	100	100	2.2	2.0	5.5
	CSA2.00MG040	2.00	100	100	0	2.0	5.5
	CST2.00MG040	2.00	On-chip	On-chip	0	2.0	5.5
	CSA3.58MG	3.58	30	30	0	2.0	5.5
	CST3.58MGW	3.58	On-chip	On-chip	0	2.0	5.5
	CSA4.19MG	4.19	30	30	0	2.0	5.5
	CST4.19MGW	4.19	On-chip	On-chip	0	2.0	5.5
	CSA4.91MG	4.91	30	30	0	2.0	5.5
	CST4.91MGW	4.91	On-chip	On-chip	0	2.0	5.5
	CSA5.00MG	5.00	30	30	0	2.0	5.5
	CST5.00MGW	5.00	On-chip	On-chip	0	2.0	5.5

Caution The oscillator constant is a reference value based on evaluation in specific environments by the resonator manufacturer. If the oscillator characteristics need to be optimized in the actual application, request the resonator manufacturer for evaluation on the implementation circuit.

Note that the oscillation voltage and oscillation frequency merely indicate the characteristics of the oscillator. Use the internal operation conditions of the μPD780308, 780308Y Subseries within the specifications of the DC and AC characteristics.

DC Characteristics (T_A = –40 to +85°C, V_{DD} = 2.0 to 5.5 V)

Parameter	Symbol	Conditions		MIN.	TYP.	MAX.	Unit
Input voltage, high	V_{IH1}	P10 to P17, P30 to P32, P35 to P37, P80 to P87, P90 to P97, P100 to P103	2.7 V ≤ V_{DD} ≤ 5.5 V	$0.7V_{DD}$		V_{DD}	V
			2.0 V ≤ V_{DD} < 2.7 V	$0.8V_{DD}$		V_{DD}	V
	V_{IH2}	P00 to P05, P25 to P27, P33, P34, P70 to P72, P110 to P117, $\overline{RESET}$	2.7 V ≤ V_{DD} ≤ 5.5 V	$0.8V_{DD}$		V_{DD}	V
			2.0 V ≤ V_{DD} < 2.7 V	$0.85V_{DD}$		V_{DD}	V
	V_{IH3}	X1, X2	2.7 V ≤ V_{DD} ≤ 5.5 V	V_{DD} – 0.5		V_{DD}	V
			2.0 V ≤ V_{DD} < 2.7 V	V_{DD} – 0.2		V_{DD}	V
	V_{IH4}	XT1/P07, XT2	4.5 V ≤ V_{DD} ≤ 5.5 V	$0.8V_{DD}$		V_{DD}	V
			2.7 V ≤ V_{DD} < 4.5 V	$0.9V_{DD}$		V_{DD}	V
			2.0 V ≤ V_{DD} < 2.7 V[Note]	$0.9V_{DD}$		V_{DD}	V
Input voltage, low	V_{IL1}	P10 to P17, P30 to P32, P35 to P37, P80 to P87, P90 to P97, P100 to P103	2.7 V ≤ V_{DD} ≤ 5.5 V	0		$0.3V_{DD}$	V
			2.0 V ≤ V_{DD} < 2.7 V	0		$0.2V_{DD}$	V
	V_{IL2}	P00 to P05, P25 to P27, P33, P34, P70 to P72, P110 to P117, $\overline{RESET}$	2.7 V ≤ V_{DD} ≤ 5.5 V	0		$0.2V_{DD}$	V
			2.0 V ≤ V_{DD} < 2.7 V	0		$0.15V_{DD}$	V
	V_{IL3}	X1, X2	2.7 V ≤ V_{DD} ≤ 5.5 V	0		0.4	V
			2.0 V ≤ V_{DD} < 2.7 V	0		0.2	V
	V_{IL4}	XT1/P07, XT2	4.5 V ≤ V_{DD} ≤ 5.5 V	0		$0.2V_{DD}$	V
			2.7 V ≤ V_{DD} < 4.5 V	0		$0.1V_{DD}$	V
			2.0 V ≤ V_{DD} < 2.7 V[Note]	0		$0.1V_{DD}$	V
Output voltage, high	V_{OH}	V_{DD} = 4.5 to 5.5 V, I_{OH} = –1 mA		V_{DD} – 1.0		V_{DD}	V
		I_{OH} = –100 μA		V_{DD} – 0.5		V_{DD}	V
Output voltage, low	V_{OL1}	P100 to P103	V_{DD} = 4.5 to 5.5 V, I_{OL} = 15 mA		0.6	2.0	V
		P01 to P05, P10 to P17, P25 to P27, P30 to P37, P70 to P72, P80 to P87, P90 to P97, P110 to P117	V_{DD} = 4.5 to 5.5 V, I_{OL} = 1.6 mA			0.4	V
	V_{OL2}	SB0, SB1, $\overline{SCK0}$	V_{DD} = 4.5 to 5.5 V, open-drain, pulled up (R = 1 kΩ)			$0.2V_{DD}$	V
	V_{OL3}	I_{OL} = 400 μA				0.5	V

Note When the XT1/P07 pin is used as P07, input the inverse phase of P07 to the XT2 pin.

Remark Unless otherwise specified, the characteristics of alternate-function pins are the same as those of port pins.

DC Characteristics (T_A = –40 to +85°C, V_{DD} = 2.0 to 5.5 V)

Parameter	Symbol	Conditions		MIN.	TYP.	MAX.	Unit
Input leakage current, high	I_{LIH1}	$V_{IN} = V_{DD}$	P00 to P05, P10 to P17, P25 to P27, P30 to P37, P70 to P72, P80 to P87, P90 to P97, P100 to P103, P110 to P117, $\overline{\text{RESET}}$			3	μA
	I_{LIH2}		X1, X2, XT1/P07, XT2			20	μA
Input leakage current, low	I_{LIL1}	V_{IN} = 0 V	P00 to P05, P10 to P17, P25 to P27, P30 to P37, P70 to P72, P80 to P87, P90 to P97, P100 to P103, P110 to P117, $\overline{\text{RESET}}$			–3	μA
	I_{LIL2}		X1, X2, XT1/P07, XT2			–20	μA
Output leakage current, high	I_{LOH}	$V_{OUT} = V_{DD}$				3	μA
Output leakage current, low	I_{LOL}	V_{OUT} = 0 V				–3	μA
Software pull-up resistance	R	V_{IN} = 0 V	P01 to P05, P10 to P17, P25 to P27, P30 to P37, P70 to P72, P80 to P87, P90 to P97, P100 to P103, P110 to P117	15	45	90	kΩ

Remark Unless otherwise specified, the characteristics of alternate-function pins are the same as those of port pins.

DC Characteristics (T_A = –40 to +85°C, V_{DD} = 2.0 to 5.5 V): μPD780306, 780306Y, 780308, 780308Y only

Parameter	Symbol	Conditions		MIN.	TYP.	MAX.	Unit
Supply current[Note 1]	I_{DD1}	5.00 MHz crystal oscillation (f_{XX} = 2.5 MHz)[Note 2] operating mode	V_{DD} = 5.0 V ±10%[Note 4]		4	12	mA
			V_{DD} = 3.0 V ±10%[Note 5]		0.6	1.8	mA
			V_{DD} = 2.2 V ±10%[Note 5]		0.35	1.05	mA
		5.00 MHz crystal oscillation (f_{XX} = 5.0 MHz)[Note 3] operating mode	V_{DD} = 5.0 V ±10%[Note 4]		6.5	19.5	mA
			V_{DD} = 3.0 V ±10%[Note 5]		0.8	2.4	mA
	I_{DD2}	5.00 MHz crystal oscillation (f_{XX} = 2.5 MHz)[Note 2] HALT mode	V_{DD} = 5.0 V ±10%		1.4	4.2	mA
			V_{DD} = 3.0 V ±10%		500	1500	μA
			V_{DD} = 2.2 V ±10%		280	840	μA
		5.00 MHz crystal oscillation (f_{XX} = 5.0 MHz)[Note 3] HALT mode	V_{DD} = 5.0 V ±10%		1.6	4.8	mA
			V_{DD} = 3.0 V ±10%		650	1950	μA
	I_{DD3}	32.768 kHz crystal oscillation operating mode[Note 6]	V_{DD} = 5.0 V ±10%		60	120	μA
			V_{DD} = 3.0 V ±10%		32	64	μA
			V_{DD} = 2.2 V ±10%		24	48	μA
	I_{DD4}	32.768 kHz crystal oscillation HALT mode[Note 6]	V_{DD} = 5.0 V ±10%		25	55	μA
			V_{DD} = 3.0 V ±10%		5	15	μA
			V_{DD} = 2.2 V ±10%		2.5	12.5	μA
	I_{DD5}	XT1 = V_{DD} STOP mode When feedback resistor is connected	V_{DD} = 5.0 V ±10%		1	30	μA
			V_{DD} = 3.0 V ±10%		0.5	10	μA
			V_{DD} = 2.2 V ±10%		0.3	10	μA
	I_{DD6}	XT1 = V_{DD} STOP mode When feedback resistor is disconnected	V_{DD} = 5.0 V ±10%		0.1	30	μA
			V_{DD} = 3.0 V ±10%		0.05	10	μA
			V_{DD} = 2.2 V ±10%		0.05	10	μA

Notes 1. Current flowing to the V_{DD} pin. Not including the current flowing to the A/D converter, ports, on-chip pull-up resistors, or LCD dividing resistors.

2. Main system clock f_{XX} = f_X/2 operation (when oscillation mode select register (OSMS) is set to 00H)

3. Main system clock f_{XX} = f_X operation (when OSMS is set to 01H)

4. High-speed mode operation (when processor clock control register (PCC) is set to 00H)

5. Low-speed mode operation (when PCC is set to 04H)

6. When the main system clock is stopped.

DC Characteristics (T_A = –40 to +85°C, V_{DD} = 2.0 to 5.5 V): μPD78P0308, 78P0308Y only

Parameter	Symbol	Conditions		MIN.	TYP.	MAX.	Unit
Supply current[Note 1]	I_{DD1}	5.00 MHz crystal oscillation (f_{XX} = 2.5 MHz)[Note 2] operating mode	V_{DD} = 5.0 V ±10%[Note 4]		5	15	mA
			V_{DD} = 3.0 V ±10%[Note 5]		0.7	2.1	mA
			V_{DD} = 2.2 V ±10%[Note 5]		0.4	1.2	mA
		5.00 MHz crystal oscillation (f_{XX} = 5.0 MHz)[Note 3] operating mode	V_{DD} = 5.0 V ±10%[Note 4]		9	27	mA
			V_{DD} = 3.0 V ±10%[Note 5]		1	3	mA
	I_{DD2}	5.00 MHz crystal oscillation (f_{XX} = 2.5 MHz)[Note 2] HALT mode	V_{DD} = 5.0 V ±10%		1.4	4.2	mA
			V_{DD} = 3.0 V ±10%		500	1500	μA
			V_{DD} = 2.2 V ±10%		280	840	μA
		5.00 MHz crystal oscillation (f_{XX} = 5.0 MHz)[Note 3] HALT mode	V_{DD} = 5.0 V ±10%		1.6	4.8	mA
			V_{DD} = 3.0 V ±10%		650	1950	μA
	I_{DD3}	32.768 kHz crystal oscillation operating mode[Note 6]	V_{DD} = 5.0 V ±10%		135	270	μA
			V_{DD} = 3.0 V ±10%		95	190	μA
			V_{DD} = 2.2 V ±10%		70	140	μA
	I_{DD4}	32.768 kHz crystal oscillation HALT mode[Note 6]	V_{DD} = 5.0 V ±10%		25	55	μA
			V_{DD} = 3.0 V ±10%		5	15	μA
			V_{DD} = 2.2 V ±10%		2.5	12.5	μA
	I_{DD5}	XT1 = V_{DD} STOP mode When feedback resistor is connected	V_{DD} = 5.0 V ±10%		1	30	μA
			V_{DD} = 3.0 V ±10%		0.5	10	μA
			V_{DD} = 2.2 V ±10%		0.3	10	μA
	I_{DD6}	XT1 = V_{DD} STOP mode When feedback resistor is disconnected	V_{DD} = 5.0 V ±10%		0.1	30	μA
			V_{DD} = 3.0 V ±10%		0.05	10	μA
			V_{DD} = 2.2 V ±10%		0.05	10	μA

Notes 1. Current flowing to the V_{DD} pin. Not including the current flowing to the A/D converter, ports, on-chip pull-up resistors, or LCD dividing resistors.

2. Main system clock f_{XX} = f_X/2 operation (when oscillation mode select register (OSMS) is set to 00H)

3. Main system clock f_{XX} = f_X operation (when OSMS is set to 01H)

4. High-speed mode operation (when processor clock control register (PCC) is set to 00H)

5. Low-speed mode operation (when PCC is set to 04H)

6. When the main system clock is stopped.

LCD Controller/Driver Characteristics (at Normal Operation)

(1) Static display mode (T_A = –10 to +85°C, V_{DD} = 2.0 to 5.5 V)

Parameter	Symbol	Conditions		MIN.	TYP.	MAX.	Unit
LCD drive voltage	V_{LCD}			2.0		V_{DD}	V
LCD dividing resistor	R_{LCD}			60	100	150	kΩ
LCD output voltage deviation[Note] (common)	V_{ODC}	I_O = ±5 μA	V_{LCD0} = V_{LCD} 2.0 V ≤ V_{LCD} ≤ V_{DD}	0		±0.2	V
LCD output voltage deviation[Note] (segment)	V_{ODS}	I_O = ±1 μA		0		±0.2	V

Note The voltage deviation is the difference between the output voltage and the corresponding ideal value of the segment or common output (V_{LCDn}; n = 0, 1, 2).

(2) 1/3 bias method (T_A = –10 to +85°C, V_{DD} = 2.5 to 5.5 V)

Parameter	Symbol	Conditions		MIN.	TYP.	MAX.	Unit
LCD drive voltage	V_{LCD}			2.5		V_{DD}	V
LCD dividing resistor	R_{LCD}			60	100	150	kΩ
LCD output voltage deviation[Note] (common)	V_{ODC}	I_O = ±5 μA	V_{LCD0} = V_{LCD} V_{LCD1} = V_{LCD} × 2/3 V_{LCD2} = V_{LCD} × 1/3 2.5 V ≤ V_{LCD} ≤ V_{DD}	0		±0.2	V
LCD output voltage deviation[Note] (segment)	V_{ODS}	I_O = ±1 μA		0		±0.2	V

Note The voltage deviation is the difference between the output voltage and the corresponding ideal value of the segment or common output (V_{LCDn}; n = 0, 1, 2).

(3) 1/2 bias method (T_A = –10 to +85°C, V_{DD} = 2.7 to 5.5 V)

Parameter	Symbol	Conditions		MIN.	TYP.	MAX.	Unit
LCD drive voltage	V_{LCD}			2.7		V_{DD}	V
LCD dividing resistor	R_{LCD}			60	100	150	kΩ
LCD output voltage deviation[Note] (common)	V_{ODC}	I_O = ±5 μA	V_{LCD0} = V_{LCD} V_{LCD1} = V_{LCD} × 1/2 V_{LCD2} = V_{LCD1} 2.7 V ≤ V_{LCD} ≤ V_{DD}	0		±0.2	V
LCD output voltage deviation[Note] (segment)	V_{ODS}	I_O = ±1 μA		0		±0.2	V

Note The voltage deviation is the difference between the output voltage and the corresponding ideal value of the segment or common output (V_{LCDn}; n = 0, 1, 2).

LCD Controller/Driver Characteristics (at Low-Voltage Operation)

(1) Static display mode (T_A = –10 to +85°C, 2.0 V ≤ V_{DD} < 3.4 V)

Parameter	Symbol	Conditions		MIN.	TYP.	MAX.	Unit
LCD drive voltage	V_{LCD}			2.0		V_{DD}	V
LCD dividing resistor	R_{LCD}			60	100	150	kΩ
LCD output voltage deviation[Note] (common)	V_{ODC}	I_O = ±5 μA	$V_{LCD0} = V_{LCD}$ 2.0 V ≤ V_{LCD} ≤ V_{DD}	0		±0.2	V
LCD output voltage deviation[Note] (segment)	V_{ODS}	I_O = ±1 μA		0		±0.2	V

Note The voltage deviation is the difference between the output voltage and the corresponding ideal value of the segment or common output (V_{LCDn}; n = 0, 1, 2).

(2) 1/3 bias method (T_A = –10 to +85°C, 2.0 V ≤ V_{DD} < 3.4 V)

Parameter	Symbol	Conditions		MIN.	TYP.	MAX.	Unit
LCD drive voltage	V_{LCD}			2.0		V_{DD}	V
LCD dividing resistor	R_{LCD}			60	100	150	kΩ
LCD output voltage deviation[Note] (common)	V_{ODC}	I_O = ±5 μA	$V_{LCD0} = V_{LCD}$ $V_{LCD1} = V_{LCD} \times 2/3$ $V_{LCD2} = V_{LCD} \times 1/3$ 2.0 V ≤ V_{LCD} ≤ V_{DD}	0		±0.2	V
LCD output voltage deviation[Note] (segment)	V_{ODS}	I_O = ±1 μA		0		±0.2	V

Note The voltage deviation is the difference between the output voltage and the corresponding ideal value of the segment or common output (V_{LCDn}; n = 0, 1, 2).

(3) 1/2 bias method (T_A = –10 to +85°C, 2.0 V ≤ V_{DD} < 3.4 V)

Parameter	Symbol	Conditions		MIN.	TYP.	MAX.	Unit
LCD drive voltage	V_{LCD}			2.0		V_{DD}	V
LCD dividing resistor	R_{LCD}			60	100	150	kΩ
LCD output voltage deviation[Note] (common)	V_{ODC}	I_O = ±5 μA	$V_{LCD0} = V_{LCD}$ $V_{LCD1} = V_{LCD} \times 1/2$ $V_{LCD2} = V_{LCD1}$ 2.0 V ≤ V_{LCD} ≤ V_{DD}	0		±0.2	V
LCD output voltage deviation[Note] (segment)	V_{ODS}	I_O = ±1 μA		0		±0.2	V

Note The voltage deviation is the difference between the output voltage and the corresponding ideal value of the segment or common output (V_{LCDn}; n = 0, 1, 2).

AC Characteristics

(1) Basic operation (T_A = –40 to +85°C, V_{DD} = 2.0 to 5.5 V)

Parameter	Symbol	Conditions		MIN.	TYP.	MAX.	Unit
Cycle time (Min. instruction execution time)	T_{CY}	Operating on main system clock (f_{XX} = 2.5 MHz)[Note 1]	2.7 V ≤ V_{DD} ≤ 5.5 V	0.8		64	μs
			2.0 V ≤ V_{DD} < 2.7 V	2.0		64	μs
		Operating on main system clock (f_{XX} = 5.0 MHz)[Note 2]	3.5 V ≤ V_{DD} ≤ 5.5 V	0.4		32	μs
			2.7 V ≤ V_{DD} < 3.5 V	0.8		32	μs
		Operating on subsystem clock		40[Note 3]	122	125	μs
TI00 input frequency	f_{TI00}	$t_{TI00} = t_{TIH00} + t_{TIL00}$		0		$1/t_{TI00}$	MHz
TI00 input high-/low-level width	t_{TIH00}, t_{TIL00}	3.5 V ≤ V_{DD} ≤ 5.5 V		$2/f_{sam} + 0.1$[Note 4]			μs
		2.7 V ≤ V_{DD} < 3.5 V		$2/f_{sam} + 0.2$[Note 4]			μs
		2.0 V ≤ V_{DD} < 2.7 V		$2/f_{sam} + 0.5$[Note 4]			μs
TI01 input frequency	f_{TI01}	2.7 V ≤ V_{DD} ≤ 5.5 V		0		100	kHz
		2.0 V ≤ V_{DD} < 2.7 V		0		50	kHz
TI01 input high-/low-level width	t_{TIH01}, t_{TIL01}	2.7 V ≤ V_{DD} ≤ 5.5 V		10			μs
		2.0 V ≤ V_{DD} < 2.7 V		20			μs
TI1, TI2 input frequency	f_{TI1}	4.5 V ≤ V_{DD} ≤ 5.5 V		0		4	MHz
		2.0 V ≤ V_{DD} < 4.5 V		0		275	kHz
TI1, TI2 input high-/low-level width	t_{TIH1}, t_{TIL1}	4.5 V ≤ V_{DD} ≤ 5.5 V		100			ns
		2.0 V ≤ V_{DD} < 4.5 V		1.8			μs
Interrupt request input high-/low-level width	t_{INTH}, t_{INTL}	INTP0	3.5 V ≤ V_{DD} ≤ 5.5 V	$2/f_{sam} + 0.1$[Note 4]			μs
			2.7 V ≤ V_{DD} < 3.5 V	$2/f_{sam} + 0.2$[Note 4]			μs
			2.0 V ≤ V_{DD} < 2.7 V	$2/f_{sam} + 0.5$[Note 4]			μs
		INTP1 to INTP5, P110 to P117	2.7 V ≤ V_{DD} ≤ 5.5 V	10			μs
			2.0 V ≤ V_{DD} < 2.7 V	20			μs
$\overline{\text{RESET}}$ low-level width	t_{RSL}	2.7 V ≤ V_{DD} ≤ 5.5 V		10			μs
		2.0 V ≤ V_{DD} < 2.7 V		20			μs

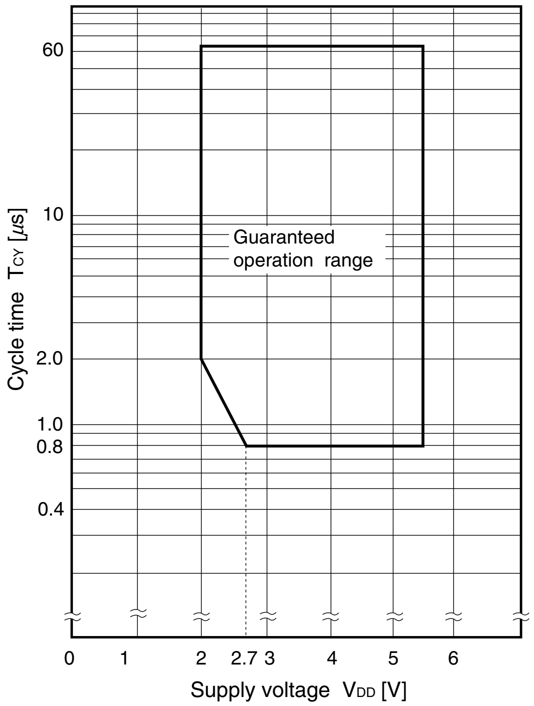
Notes 1. Main system clock f_{XX} = fx/2 operation (when oscillation mode select register (OSMS) is set to 00H)

2. Main system clock f_{XX} = fx operation (when OSMS is set to 01H)

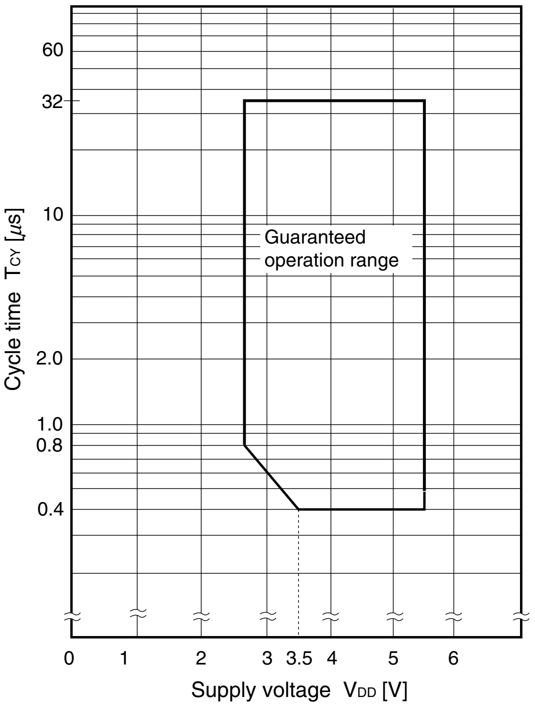
3. This is the value when the external clock is used. The value is 114 μs (min.) when the crystal resonator is used.

4. In combination with bits 0 (SCS0) and 1 (SCS1) of the sampling clock select register (SCS), selection of f_{sam} is possible between $f_{XX}/2^N$, $f_{XX}/32$, $f_{XX}/64$, and $f_{XX}/128$ (when N = 0 to 4).

T_{CY} vs. V_{DD} (at main system clock $f_{XX} = f_X/2$ operation)



T_{CY} vs. V_{DD} (at main system clock $f_{XX} = f_X$ operation)



(2) Serial interface (T_A = –40 to +85°C, V_{DD} = 2.0 to 5.5 V)

(a) Serial interface channel 0

(i) 3-wire serial I/O mode ($\overline{\text{SCK0}}$...internal clock output)

Parameter	Symbol	Conditions	MIN.	TYP.	MAX.	Unit
$\overline{\text{SCK0}}$ cycle time	t_{KCY1}	4.5 V ≤ V_{DD} ≤ 5.5 V	800			ns
		2.7 V ≤ V_{DD} < 4.5 V	1600			ns
		2.0 V ≤ V_{DD} < 2.7 V	3200			ns
$\overline{\text{SCK0}}$ high-/low-level width	t_{KH1}, t_{KL1}	4.5 V ≤ V_{DD} ≤ 5.5 V	t_{KCY1}/2 – 50			ns
		2.0 V ≤ V_{DD} < 4.5 V	t_{KCY1}/2 – 100			ns
SI0 setup time (to $\overline{\text{SCK0}}$↑)	t_{SIK1}	4.5 V ≤ V_{DD} ≤ 5.5 V	100			ns
		2.7 V ≤ V_{DD} < 4.5 V	150			ns
		2.0 V ≤ V_{DD} < 2.7 V	300			ns
SI0 hold time (from $\overline{\text{SCK0}}$↑)	t_{KSI1}		400			ns
SO0 output delay time from $\overline{\text{SCK0}}$↓	t_{KSO1}	C = 100 pF^Note			300	ns

Note C is the load capacitance of $\overline{\text{SCK0}}$ and SO0 output lines.

(ii) 3-wire serial I/O mode ($\overline{\text{SCK0}}$...external clock input)

Parameter	Symbol	Conditions	MIN.	TYP.	MAX.	Unit
$\overline{\text{SCK0}}$ cycle time	t_{KCY2}	4.5 V ≤ V_{DD} ≤ 5.5 V	800			ns
		2.7 V ≤ V_{DD} < 4.5 V	1600			ns
		2.0 V ≤ V_{DD} < 2.7 V	3200			ns
$\overline{\text{SCK0}}$ high-/low-level width	t_{KH2}, t_{KL2}	4.5 V ≤ V_{DD} ≤ 5.5 V	400			ns
		2.7 V ≤ V_{DD} < 4.5 V	800			ns
		2.0 V ≤ V_{DD} < 2.7 V	1600			ns
SI0 setup time (to $\overline{\text{SCK0}}$↑)	t_{SIK2}		100			ns
SI0 hold time (from $\overline{\text{SCK0}}$↑)	t_{KSI2}		400			ns
SO0 output delay time from $\overline{\text{SCK0}}$↓	t_{KSO2}	C = 100 pF^Note			300	ns
$\overline{\text{SCK0}}$ rise, fall time	t_{R2}, t_{F2}				1000	ns

Note C is the load capacitance of SO0 output line.

(iii) SBI mode ($\overline{\text{SCK0}}$...internal clock output): μPD780306, 780308, 78P0308 only

Parameter	Symbol	Conditions		MIN.	TYP.	MAX.	Unit
$\overline{\text{SCK0}}$ cycle time	t_{KCY3}	$4.5\ V \le V_{DD} \le 5.5\ V$		800			ns
		$2.0\ V \le V_{DD} < 4.5\ V$		3200			ns
$\overline{\text{SCK0}}$ high-/low-level width	t_{KH3}, t_{KL3}	$4.5\ V \le V_{DD} \le 5.5\ V$		$t_{KCY3}/2 - 50$			ns
		$2.0\ V \le V_{DD} < 4.5\ V$		$t_{KCY3}/2 - 150$			ns
SB0, SB1 setup time (to $\overline{\text{SCK0}}$↑)	t_{SIK3}	$4.5\ V \le V_{DD} \le 5.5\ V$		100			ns
		$2.0\ V \le V_{DD} < 4.5\ V$		300			ns
SB0, SB1 hold time (from $\overline{\text{SCK0}}$↑)	t_{KSI3}			$t_{KCY3}/2$			ns
SB0, SB1 output delay time from $\overline{\text{SCK0}}$↓	t_{KSO3}	R = 1 kΩ, C = 100 pF[Note]	$4.5\ V \le V_{DD} \le 5.5\ V$	0		250	ns
			$2.0\ V \le V_{DD} < 4.5\ V$	0		1000	ns
SB0, SB1↓ from $\overline{\text{SCK0}}$↑	t_{KSB}			t_{KCY3}			ns
$\overline{\text{SCK0}}$↓ from SB0, SB1↓	t_{SBK}			t_{KCY3}			ns
SB0, SB1 high-level width	t_{SBH}			t_{KCY3}			ns
SB0, SB1 low-level width	t_{SBL}			t_{KCY3}			ns

Note R and C are the load resistance and load capacitance of the $\overline{\text{SCK0}}$, SB0, and SB1 output lines.

(iv) SBI mode ($\overline{\text{SCK0}}$...external clock input): μPD780306, 780308, 78P0308 only

Parameter	Symbol	Conditions		MIN.	TYP.	MAX.	Unit
$\overline{\text{SCK0}}$ cycle time	t_{KCY4}	$4.5\ V \le V_{DD} \le 5.5\ V$		800			ns
		$2.0\ V \le V_{DD} < 4.5\ V$		3200			ns
$\overline{\text{SCK0}}$ high-/low-level width	t_{KH4}, t_{KL4}	$4.5\ V \le V_{DD} \le 5.5\ V$		400			ns
		$2.0\ V \le V_{DD} < 4.5\ V$		1600			ns
SB0, SB1 setup time (to $\overline{\text{SCK0}}$↑)	t_{SIK4}	$4.5\ V \le V_{DD} \le 5.5\ V$		100			ns
		$2.0\ V \le V_{DD} < 4.5\ V$		300			ns
SB0, SB1 hold time (from $\overline{\text{SCK0}}$↑)	t_{KSI4}			$t_{KCY4}/2$			ns
SB0, SB1 output delay time from $\overline{\text{SCK0}}$↓	t_{KSO4}	R = 1 kΩ, C = 100 pF[Note]	$4.5\ V \le V_{DD} \le 5.5\ V$	0		300	ns
			$2.0\ V \le V_{DD} < 4.5\ V$	0		1000	ns
SB0, SB1↓ from $\overline{\text{SCK0}}$↑	t_{KSB}			t_{KCY4}			ns
$\overline{\text{SCK0}}$↓ from SB0, SB1↓	t_{SBK}			t_{KCY4}			ns
SB0, SB1 high-level width	t_{SBH}			t_{KCY4}			ns
SB0, SB1 low-level width	t_{SBL}			t_{KCY4}			ns
$\overline{\text{SCK0}}$ rise, fall time	t_{R4}, t_{F4}					1000	ns

Note R and C are the load resistance and load capacitance of the SB0 and SB1 output lines.

(v) 2-wire serial I/O mode ($\overline{\text{SCK0}}$...internal clock output)

Parameter	Symbol	Conditions		MIN.	TYP.	MAX.	Unit
$\overline{\text{SCK0}}$ cycle time	t_{KCY5}	R = 1 kΩ,	2.7 V ≤ V_{DD} ≤ 5.5 V	1600			ns
		C = 100 pF[Note]	2.0 V ≤ V_{DD} < 2.7 V	3200			ns
$\overline{\text{SCK0}}$ high-level width	t_{KH5}		2.7 V ≤ V_{DD} ≤ 5.5 V	$t_{KCY5}/2 - 160$			ns
			2.0 V ≤ V_{DD} < 2.7 V	$t_{KCY5}/2 - 190$			ns
$\overline{\text{SCK0}}$ low-level width	t_{KL5}		4.5 V ≤ V_{DD} ≤ 5.5 V	$t_{KCY5}/2 - 50$			ns
			2.0 V ≤ V_{DD} < 4.5 V	$t_{KCY5}/2 - 100$			ns
SB0, SB1 setup time (to $\overline{\text{SCK0}}$↑)	t_{SIK5}		4.5 V ≤ V_{DD} ≤ 5.5 V	300			ns
			2.7 V ≤ V_{DD} < 4.5 V	350			ns
			2.0 V ≤ V_{DD} < 2.7 V	400			ns
SB0, SB1 hold time (from $\overline{\text{SCK0}}$↑)	t_{KSI5}			600			ns
SB0, SB1 output delay time from $\overline{\text{SCK0}}$↓	t_{KSO5}					300	ns

Note R and C are the load resistance and load capacitance of the $\overline{\text{SCK0}}$, SB0, and SB1 output lines.

(vi) 2-wire serial I/O mode ($\overline{\text{SCK0}}$...external clock input)

Parameter	Symbol	Conditions		MIN.	TYP.	MAX.	Unit
$\overline{\text{SCK0}}$ cycle time	t_{KCY6}	2.7 V ≤ V_{DD} ≤ 5.5 V		1600			ns
		2.0 V ≤ V_{DD} < 2.7 V		3200			ns
$\overline{\text{SCK0}}$ high-level width	t_{KH6}	2.7 V ≤ V_{DD} ≤ 5.5 V		650			ns
		2.0 V ≤ V_{DD} < 2.7 V		1300			ns
$\overline{\text{SCK0}}$ low-level width	t_{KL6}	2.7 V ≤ V_{DD} ≤ 5.5 V		800			ns
		2.0 V ≤ V_{DD} < 2.7 V		1600			ns
SB0, SB1 setup time (to $\overline{\text{SCK0}}$↑)	t_{SIK6}			100			ns
SB0, SB1 hold time (from $\overline{\text{SCK0}}$↑)	t_{KSI6}			$t_{KCY6}/2$			ns
SB0, SB1 output delay time from $\overline{\text{SCK0}}$↓	t_{KSO6}	R = 1 kΩ,	4.5 V ≤ V_{DD} ≤ 5.5 V	0		300	ns
		C = 100 pF[Note]	2.0 V ≤ V_{DD} < 4.5 V	0		500	ns
$\overline{\text{SCK0}}$ rise, fall time	t_{R6}, t_{F6}					1000	ns

Note R and C are the load resistance and load capacitance of the SB0 and SB1 output lines.

(vii) I^{2}C bus mode (SCL...internal clock output): μPD780306Y, 780308Y, 78P0308Y only

Parameter	Symbol	Conditions		MIN.	TYP.	MAX.	Unit
SCL cycle time	t_{KCY7}	R = 1 kΩ,	2.7 V ≤ V_{DD} ≤ 5.5 V	10			μs
		C = 100 pF[Note]	2.0 V ≤ V_{DD} < 2.7 V	20			μs
SCL high-level width	t_{KH7}		2.7 V ≤ V_{DD} ≤ 5.5 V	t_{KCY7} – 160			ns
			2.0 V ≤ V_{DD} < 2.7 V	t_{KCY7} – 190			ns
SCL low-level width	t_{KL7}		4.5 V ≤ V_{DD} ≤ 5.5 V	t_{KCY7} – 50			ns
			2.0 V ≤ V_{DD} < 4.5 V	t_{KCY7} – 100			ns
SDA0, SDA1 setup time (to SCL↑)	t_{SIK7}		2.7 V ≤ V_{DD} ≤ 5.5 V	200			ns
			2.0 V ≤ V_{DD} < 2.7 V	300			ns
SDA0, SDA1 hold time (from SCL↓)	t_{KSI7}			0			ns
SDA0, SDA1 output delay time from SCL↓	t_{KSO7}		4.5 V ≤ V_{DD} ≤ 5.5 V	0		300	ns
			2.0 V ≤ V_{DD} < 4.5 V	0		500	ns
SDA0, SDA1↓ from SCL↑ or SDA0, SDA1↑ from SCL↑	t_{KSB}			200			ns
SCL↓ from SDA0, SDA1↓	t_{SBK}			400			ns
SDA0, SDA1 high-level width	t_{SBH}			500			ns

Note R and C are the load resistance and load capacitance of SCL, SDA0, and SDA1 output lines.

(viii) I^{2}C bus mode (SCL...external clock input): μPD780306Y, 780308Y, 78P0308Y only

Parameter	Symbol	Conditions		MIN.	TYP.	MAX.	Unit
SCL cycle time	t_{KCY8}			1000			ns
SCL high-/low-level width	t_{KH8}, t_{KL8}			400			ns
SDA0, SDA1 setup time (to SCL↑)	t_{SIK8}			200			ns
SDA0, SDA1 hold time (from SCL↓)	t_{KSI8}			0			ns
SDA0, SDA1 output delay time from SCL↓	t_{KSO8}	R = 1 kΩ,	4.5 V ≤ V_{DD} ≤ 5.5 V	0		300	ns
		C = 100 pF[Note]	2.0 V ≤ V_{DD} < 4.5 V	0		500	ns
SDA0, SDA1↓ from SCL↑ or SDA0, SDA1↑ from SCL↑	t_{KSB}			200			ns
SCL↓ from SDA0, SDA1↓	t_{SBK}			400			ns
SDA0, SDA1 high-level width	t_{SBH}			500			ns
SCL rise, fall time	t_{R8}, t_{F8}					1000	ns

Note R and C are the load resistance and load capacitance of SDA0 and SDA1 output lines.

(b) Serial interface channel 2

(i) 3-wire serial I/O mode ($\overline{SCK2}$...internal clock output)

Parameter	Symbol	Conditions	MIN.	TYP.	MAX.	Unit
$\overline{SCK2}$ cycle time	t_{KCY9}	$4.5\ V \le V_{DD} \le 5.5\ V$	800			ns
		$2.7\ V \le V_{DD} < 4.5\ V$	1600			ns
		$2.0\ V \le V_{DD} < 2.7\ V$	3200			ns
$\overline{SCK2}$ high-/low-level width	t_{KH9},	$4.5\ V \le V_{DD} \le 5.5\ V$	$t_{KCY9}/2 - 50$			ns
	t_{KL9}	$2.0\ V \le V_{DD} < 4.5\ V$	$t_{KCY9}/2 - 100$			ns
SI2 setup time (to $\overline{SCK2}$↑)	t_{SIK9}	$4.5\ V \le V_{DD} \le 5.5\ V$	100			ns
		$2.7\ V \le V_{DD} < 4.5\ V$	150			ns
		$2.0\ V \le V_{DD} < 2.7\ V$	300			ns
SI2 hold time (from $\overline{SCK2}$↑)	t_{KSI9}		400			ns
SO2 output delay time from $\overline{SCK2}$↓	t_{KSO9}	C = 100 pF Note			300	ns

Note C is the load capacitance of $\overline{SCK2}$ and SO2 output lines.

(ii) 3-wire serial I/O mode ($\overline{SCK2}$...external clock input)

Parameter	Symbol	Conditions	MIN.	TYP.	MAX.	Unit
$\overline{SCK2}$ cycle time	t_{KCY10}	$4.5\ V \le V_{DD} \le 5.5\ V$	800			ns
		$2.7\ V \le V_{DD} < 4.5\ V$	1600			ns
		$2.0\ V \le V_{DD} < 2.7\ V$	3200			ns
$\overline{SCK2}$ high-/low-level width	t_{KH10},	$4.5\ V \le V_{DD} \le 5.5\ V$	400			ns
	t_{KL10}	$2.7\ V \le V_{DD} < 4.5\ V$	800			ns
		$2.0\ V \le V_{DD} < 2.7\ V$	1600			ns
SI2 setup time (to $\overline{SCK2}$↑)	t_{SIK10}		100			ns
SI2 hold time (from $\overline{SCK2}$↑)	t_{KSI10}		400			ns
SO2 output delay time from $\overline{SCK2}$↓	t_{KSO10}	C = 100 pF Note			300	ns
$\overline{SCK2}$ rise, fall time	t_{R10}, t_{F10}				1000	ns

Note C is the load capacitance of SO2 output line.

(iii) UART mode (dedicated baud rate generator output)

Parameter	Symbol	Conditions	MIN.	TYP.	MAX.	Unit
Transfer rate		4.5 V ≤ V_{DD} ≤ 5.5 V			78125	bps
		2.7 V ≤ V_{DD} < 4.5 V			39063	bps
		2.0 V ≤ V_{DD} < 2.7 V			19531	bps

(iv) UART mode (external clock input)

Parameter	Symbol	Conditions	MIN.	TYP.	MAX.	Unit
ASCK cycle time	t_{KCY11}	4.5 V ≤ V_{DD} ≤ 5.5 V	800			ns
		2.7 V ≤ V_{DD} < 4.5 V	1600			ns
		2.0 V ≤ V_{DD} < 2.7 V	3200			ns
ASCK high-/low-level width	t_{KH11}, t_{KL11}	4.5 V ≤ V_{DD} ≤ 5.5 V	400			ns
		2.7 V ≤ V_{DD} < 4.5 V	800			ns
		2.0 V ≤ V_{DD} < 2.7 V	1600			ns
Transfer rate		4.5 V ≤ V_{DD} ≤ 5.5 V			39063	bps
		2.7 V ≤ V_{DD} < 4.5 V			19531	bps
		2.0 V ≤ V_{DD} < 2.7 V			9766	bps
ASCK rise, fall time	t_{R11}, t_{F11}				1000	ns

(c) Serial interface channel 3

(i) 3-wire serial I/O mode ($\overline{\text{SCK3}}$...internal clock output)

Parameter	Symbol	Conditions	MIN.	TYP.	MAX.	Unit
$\overline{\text{SCK3}}$ cycle time	t_{KCY12}	4.5 V ≤ V_{DD} ≤ 5.5 V	800			ns
		2.7 V ≤ V_{DD} < 4.5 V	1600			ns
		2.0 V ≤ V_{DD} < 2.7 V	3200			ns
$\overline{\text{SCK3}}$ high-/low-level width	t_{KH12}, t_{KL12}	4.5 V ≤ V_{DD} ≤ 5.5 V	$t_{KCY12}/2 - 50$			ns
		2.0 V ≤ V_{DD} < 4.5 V	$t_{KCY12}/2 - 100$			ns
SI3 setup time (to $\overline{\text{SCK3}}$↑)	t_{SIK12}	4.5 V ≤ V_{DD} ≤ 5.5 V	100			ns
		2.7 V ≤ V_{DD} < 4.5 V	150			ns
		2.0 V ≤ V_{DD} < 2.7 V	300			ns
SI3 hold time (from $\overline{\text{SCK3}}$↑)	t_{KSI12}		400			ns
SO3 output delay time from $\overline{\text{SCK3}}$↓	t_{KSO12}	C = 100 pF[Note]			300	ns

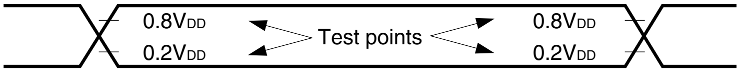
Note C is the load capacitance of $\overline{\text{SCK3}}$ and SO3 output lines.

(ii) 3-wire serial I/O mode ($\overline{\text{SCK3}}$...external clock input)

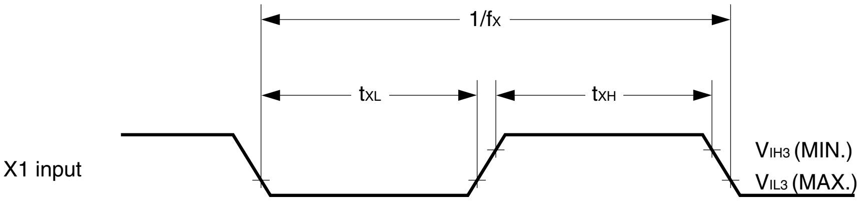
Parameter	Symbol	Conditions	MIN.	TYP.	MAX.	Unit
$\overline{\text{SCK3}}$ cycle time	t_{KCY13}	4.5 V ≤ V_{DD} ≤ 5.5 V	800			ns
		2.7 V ≤ V_{DD} < 4.5 V	1600			ns
		2.0 V ≤ V_{DD} < 2.7 V	3200			ns
$\overline{\text{SCK3}}$ high-/low-level width	t_{KH13}, t_{KL13}	4.5 V ≤ V_{DD} ≤ 5.5 V	400			ns
		2.7 V ≤ V_{DD} < 4.5 V	800			ns
		2.0 V ≤ V_{DD} < 2.7 V	1600			ns
SI3 setup time (to $\overline{\text{SCK3}}$↑)	t_{SIK13}		100			ns
SI3 hold time (from $\overline{\text{SCK3}}$↑)	t_{KSI13}		400			ns
SO3 output delay time from $\overline{\text{SCK3}}$↓	t_{KSO13}	C = 100 pF[Note]			300	ns
$\overline{\text{SCK3}}$ rise, fall time	t_{R13}, t_{F13}				1000	ns

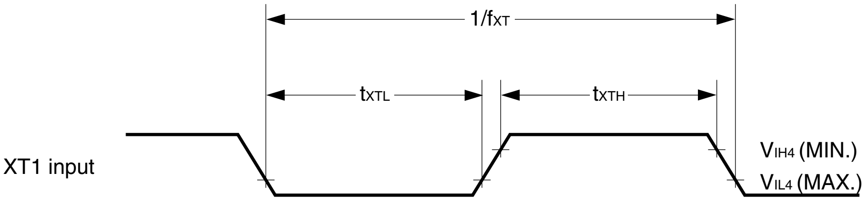
Note C is the load capacitance of SO3 output line.

AC Timing Test Points (Excluding X1, XT1 Input)

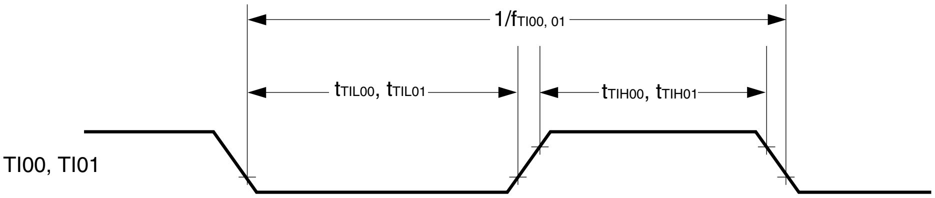


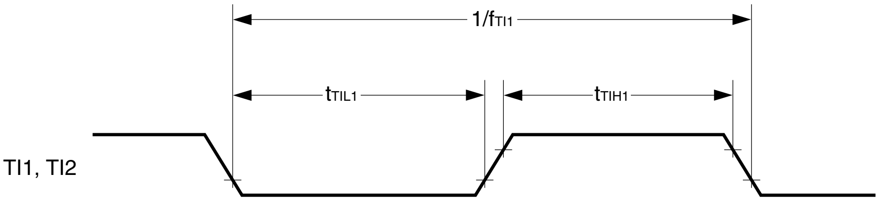
Clock Timing





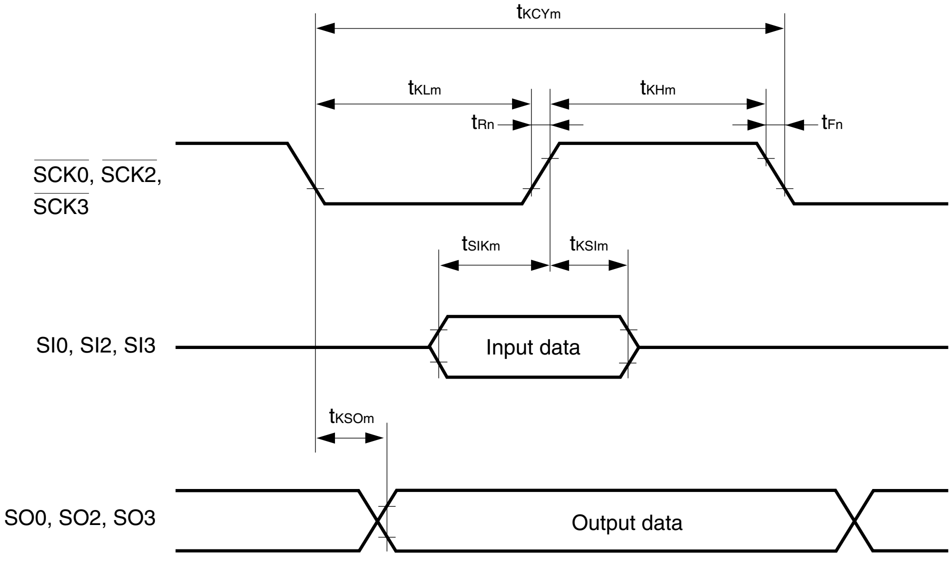
TI Timing





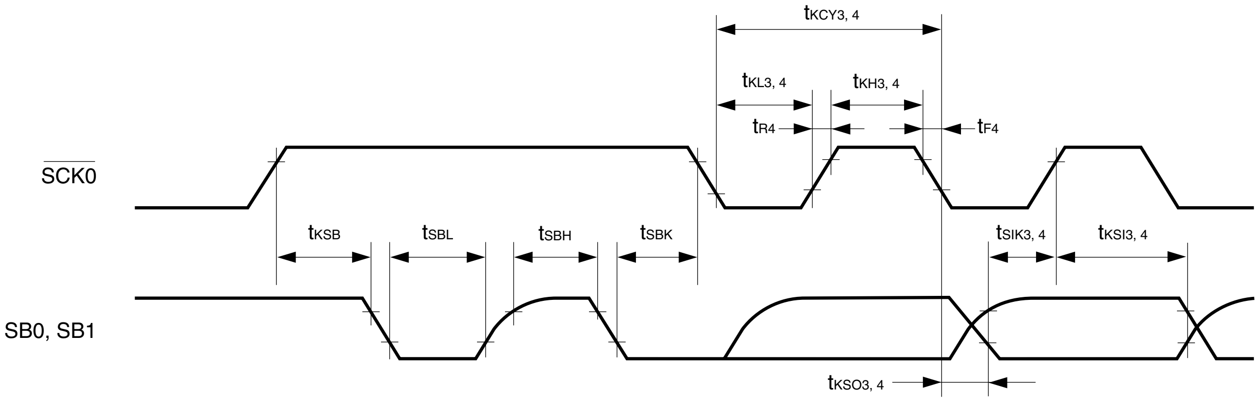
Serial Transfer Timing

3-wire serial I/O mode:

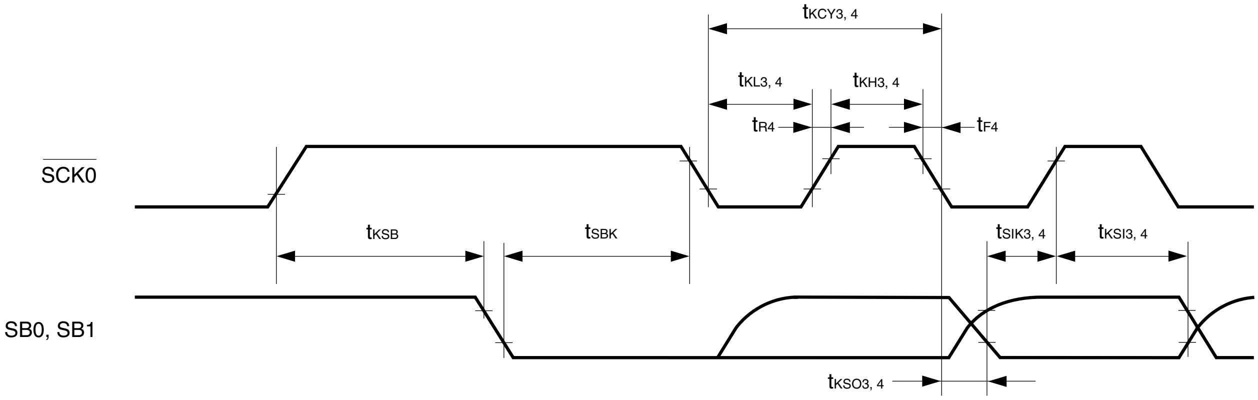


m = 1, 2, 9, 10, 12, 13
n = 2, 10, 13

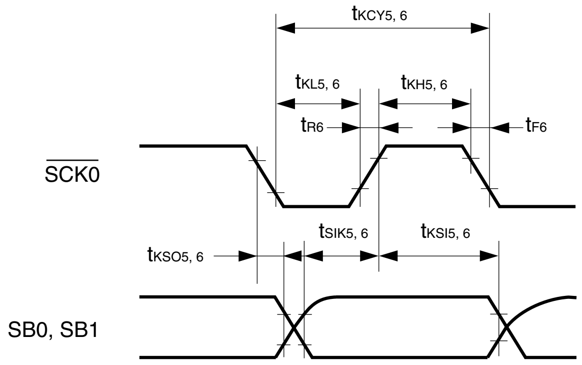
SBI mode (bus release signal transfer, μPD780306, 780308, 78P0308 only):



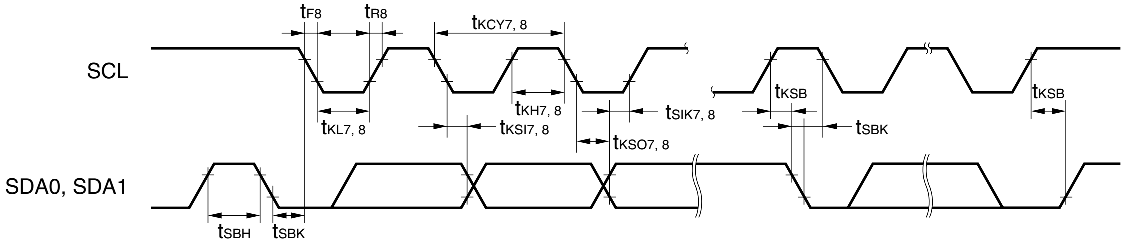
SBI mode (command signal transfer, μPD780306, 780308, 78P0308 only):



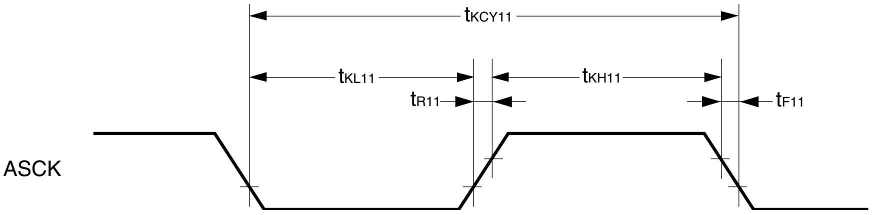
2-wire serial I/O mode:



I^{2}C bus mode (μPD780306Y, 780308Y, 78P0308Y only):



UART mode:



A/D Converter Characteristics (T_A = –40 to +85°C, V_{DD} = 2.0 to 5.5 V, AV_{SS} = V_{SS} = 0 V): μPD780306, 780306Y, 780308, 780308Y only

Parameter	Symbol	Conditions	MIN.	TYP.	MAX.	Unit
Resolution			8	8	8	bit
Overall error$^{\text{Note 1}}$		2.7 V ≤ AV_{REF} ≤ 5.5 V			±0.6	%FSR
		2.0 V ≤ AV_{REF} < 2.7 V			±1.4	%FSR
Conversion time	t_{CONV}		19.1		200	μs
Sampling time	t_{SAMP}		$12/f_{XX}$			μs
Analog input voltage	V_{IAN}		AV_{SS}		AV_{REF}	V
Reference voltage	AV_{REF}		2.0		V_{DD}	V
AV_{REF}-AV_{SS} resistance	R_{REF}	When A/D conversion not operating	4	14		kΩ
AV_{REF} current	AI_{REF}	When A/D conversion operating$^{\text{Note 2}}$		2.5	5.0	mA
		When A/D conversion not operating$^{\text{Note 3}}$		0.5	1.5	mA

Notes 1. Quantization error (±1/2 LSB) is not included. This is expressed as a percentage (%FSR) to the full-scale value.

2. Indicates current flowing to AV_{REF} pin when the CS bit of the A/D converter mode register (ADM) is 1.

3. Indicates current flowing to AV_{REF} pin when the CS bit of ADM is 0.

A/D Converter Characteristics (T_A = –40 to +85°C, V_{DD} = 2.0 to 5.5 V, AV_{SS} = V_{SS} = 0 V): μPD78P0308, 78P0308Y only

Parameter	Symbol	Conditions	MIN.	TYP.	MAX.	Unit
Resolution			8	8	8	bit
Overall error$^{\text{Note 1}}$		2.7 V ≤ AV_{REF} ≤ 5.5 V			±0.6	%FSR
		2.2 V ≤ AV_{REF} < 2.7 V			±1.4	%FSR
Conversion time	t_{CONV}	2.7 V ≤ AV_{REF} ≤ 5.5 V	19.1		200	μs
		2.2 V ≤ AV_{REF} < 2.7 V	38.2		200	μs
Sampling time	t_{SAMP}		$24/f_{XX}$			μs
Analog input voltage	V_{IAN}		AV_{SS}		AV_{REF}	V
Reference voltage	AV_{REF}		2.2		V_{DD}	V
AV_{REF}-AV_{SS} resistance	R_{AIREF}	When A/D conversion not operating	4	14		kΩ
AV_{REF} current	AI_{REF}	When A/D conversion operating$^{\text{Note 2}}$		2.5	5.0	mA
		When A/D conversion not operating$^{\text{Note 3}}$		0.5	1.5	mA

Notes 1. Quantization error (±1/2 LSB) is not included. This is expressed as a percentage (%FSR) to the full-scale value.

2. Indicates current flowing to AV_{REF} pin when the CS bit of the A/D converter mode register (ADM) is 1.

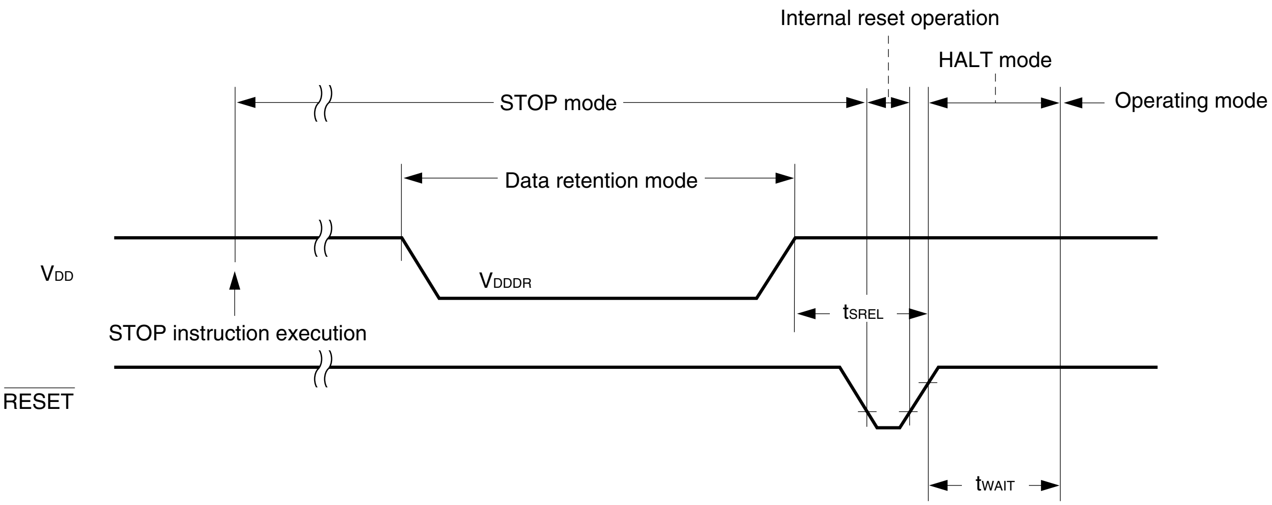
3. Indicates current flowing to AV_{REF} pin when the CS bit of ADM is 0.

Data Memory STOP Mode Low Supply Voltage Data Retention Characteristics (T_A = –40 to +85°C)

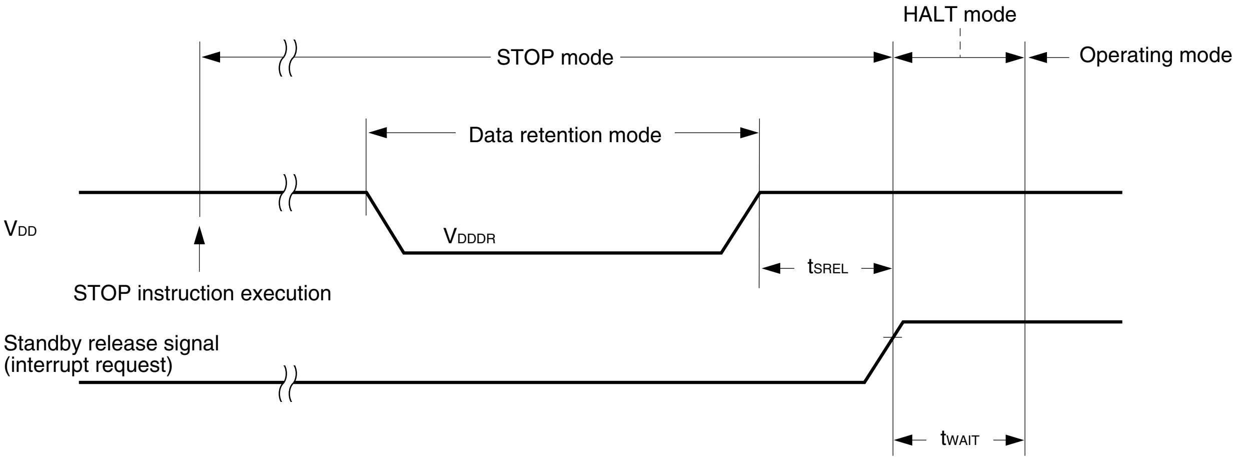
Parameter	Symbol	Conditions	MIN.	TYP.	MAX.	Unit
Data retention supply voltage	V_{DDDR}		1.6		5.5	V
Data retention supply current	I_{DDDR}	V_{DDDR} = 1.6 V Subsystem clock stop and feedback resistor disconnected.		0.1	10	μA
Release signal set time	t_{SREL}		0			μs
Oscillation stabilization wait time	t_{WAIT}	Release by $\overline{\text{RESET}}$		2^{17}/fx		s
		Release by interrupt request		**Note**		s

Note In combination with bits 0 to 2 (OSTS0 to OSTS2) of the oscillation stabilization time select register (OSTS), selection of 2^{12}/fxx and 2^{14}/fxx to 2^{17}/fxx is possible.

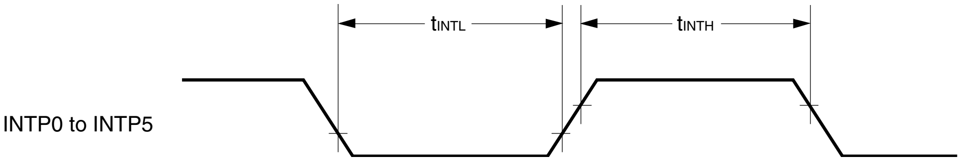
Data Retention Timing (STOP Mode Release by $\overline{\text{RESET}}$)



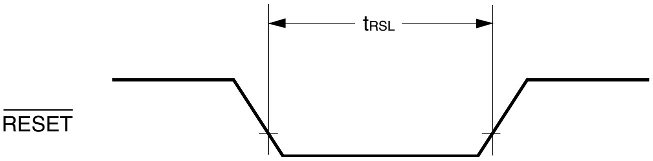
Data Retention Timing (Standby Release Signal: STOP Mode Release by Interrupt Request Signal)



Interrupt Request Input Timing

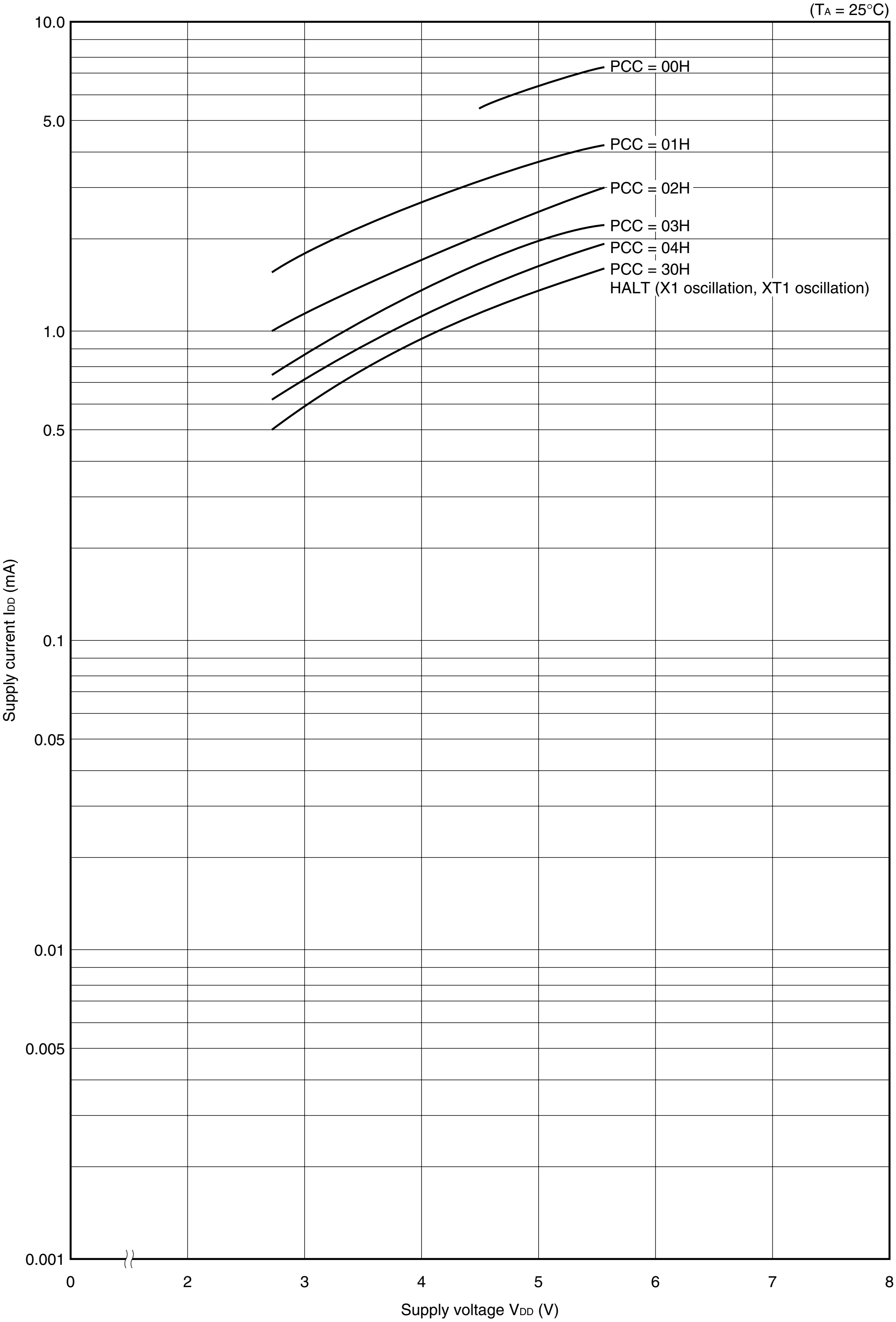


$\overline{\text{RESET}}$ Input Timing

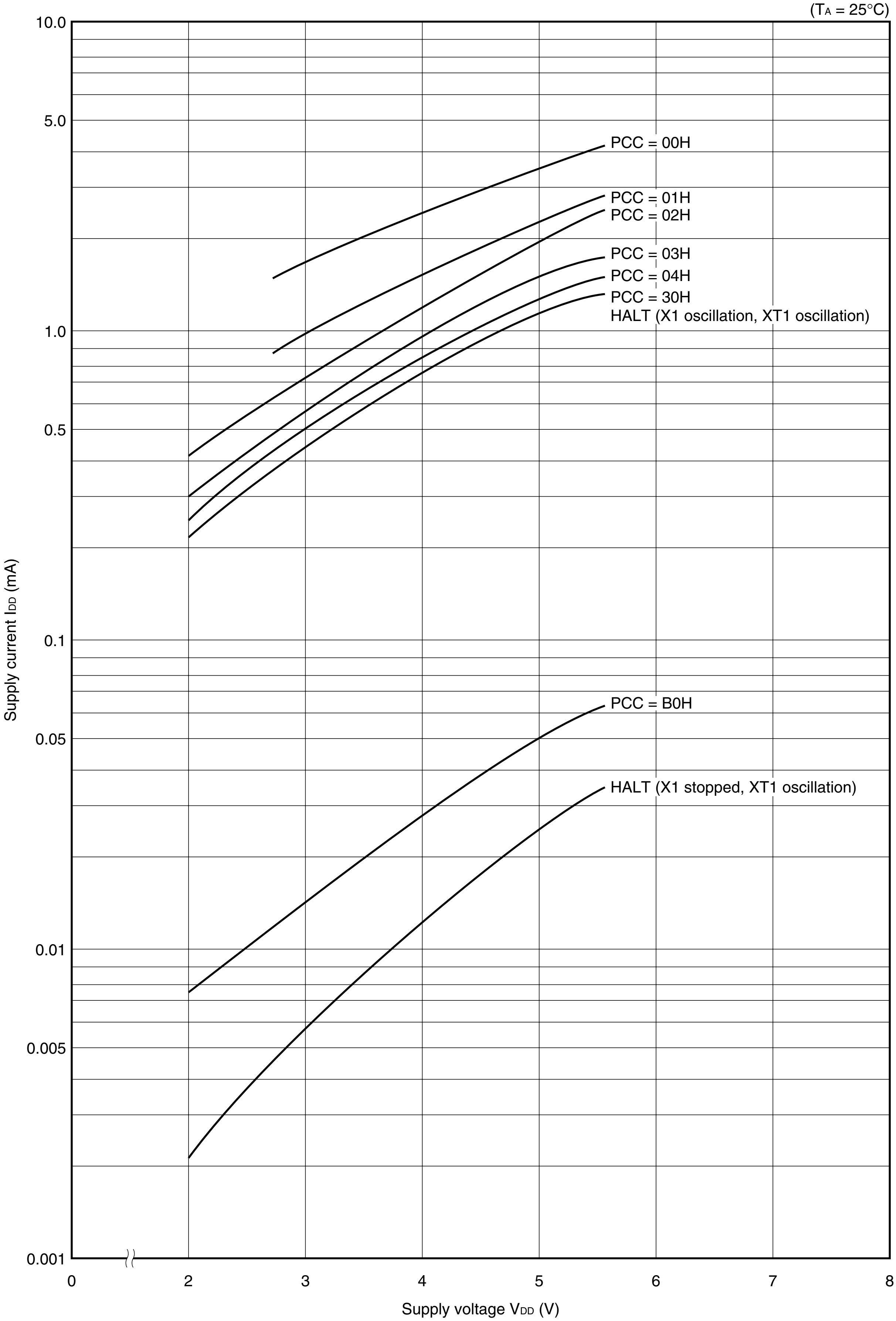


Characteristic Curves (Reference Value): μPD780306, 780306Y, 780308, 780308Y only

I_{DD} vs V_{DD} ($f_X = f_{XX} = 5.0$ MHz)



I_{DD} vs V_{DD} (f_X = 5.0 MHz, f_{XX} = 2.5 MHz)



PROM Programming Characteristics: μPD78P0308, 78P0308Y only

DC Characteristics

(1) PROM write mode (T_A = 25 ±5°C, V_{DD} = 6.5 ±0.25 V, V_{PP} = 12.5 ±0.3 V)

Parameter	Symbol	Conditions	MIN.	TYP.	MAX.	Unit
Input voltage, high	V_{IH}		$0.7V_{DD}$		V_{DD}	V
Input voltage, low	V_{IL}		0		$0.3V_{DD}$	V
Output voltage, high	V_{OH}	I_{OH} = –1 mA	V_{DD} – 1.0			V
Output voltage, low	V_{OL}	I_{OL} = 1.6 mA			0.4	V
Input leakage current	I_{LI}	$0 \le V_{IN} \le V_{DD}$	–10		+10	μA
V_{PP} supply voltage	V_{PP}		12.2	12.5	12.8	V
V_{DD} supply voltage	V_{DD}		6.25	6.5	6.75	V
V_{PP} supply current	I_{PP}	$\overline{PGM} = V_{IL}$			50	mA
V_{DD} supply current	I_{DD}				50	mA

(2) PROM read mode (T_A = 25 ±5°C, V_{DD} = 5.0 ±0.5 V, V_{PP} = V_{DD} ±0.6 V)

Parameter	Symbol	Conditions	MIN.	TYP.	MAX.	Unit
Input voltage, high	V_{IH}		$0.7V_{DD}$		V_{DD}	V
Input voltage, low	V_{IL}		0		$0.3V_{DD}$	V
Output voltage, high	V_{OH1}	I_{OH} = –1 mA	V_{DD} – 1.0			V
	V_{OH2}	I_{OH} = –100 μA	V_{DD} – 0.5			V
Output voltage, low	V_{OL}	I_{OL} = 1.6 mA			0.4	V
Input leakage current	I_{LI}	$0 \le V_{IN} \le V_{DD}$	–10		+10	μA
Output leakage current	I_{LO}	$0 \le V_{OUT} \le V_{DD}$, $\overline{OE} = V_{IH}$	–10		+10	μA
V_{PP} supply voltage	V_{PP}		V_{DD} – 0.6	V_{DD}	V_{DD} + 0.6	V
V_{DD} supply voltage	V_{DD}		4.5	5.0	5.5	V
V_{PP} supply current	I_{PP}	$V_{PP} = V_{DD}$			100	μA
V_{DD} supply current	I_{DD}	$\overline{CE} = V_{IL}$, $V_{IN} = V_{IH}$			50	mA

AC Characteristics

(1) PROM write mode

(a) Page program mode (T_A = 25 ±5°C, V_{DD} = 6.5 ±0.25 V, V_{PP} = 12.5 ±0.3 V)

Parameter	Symbol	Conditions	MIN.	TYP.	MAX.	Unit
Address setup time (to $\overline{\text{OE}}$↓)	t_{AS}		2			μs
$\overline{\text{OE}}$ setup time	t_{OES}		2			μs
$\overline{\text{CE}}$ setup time (to $\overline{\text{OE}}$↓)	t_{CES}		2			μs
Input data setup time (to $\overline{\text{OE}}$↓)	t_{DS}		2			μs
Address hold time (from $\overline{\text{OE}}$↑)	t_{AH}		2			μs
	t_{AHL}		2			μs
	t_{AHV}		0			μs
Input data hold time (from $\overline{\text{OE}}$↑)	t_{DH}		2			μs
Data output float delay time from $\overline{\text{OE}}$↑	t_{DF}		0		250	ns
V_{PP} setup time (to $\overline{\text{OE}}$↓)	t_{VPS}		1.0			ms
V_{DD} setup time (to $\overline{\text{OE}}$↓)	t_{VDS}		1.0		250	ms
Program pulse width	t_{PW}		0.095		0.105	ms
Valid data delay time from $\overline{\text{OE}}$↓	t_{OE}				1	μs
$\overline{\text{OE}}$ pulse width during data latching	t_{LW}		1			μs
$\overline{\text{PGM}}$ setup time	t_{PGMS}		2			μs
$\overline{\text{CE}}$ hold time	t_{CEH}		2			μs
$\overline{\text{OE}}$ hold time	t_{OEH}		2			μs

(b) Byte program mode (T_A = 25 ±5°C, V_{DD} = 6.5 ±0.25 V, V_{PP} = 12.5 ±0.3 V)

Parameter	Symbol	Conditions	MIN.	TYP.	MAX.	Unit
Address setup time (to $\overline{\text{PGM}}$↓)	t_{AS}		2			μs
$\overline{\text{OE}}$ setup time	t_{OES}		2			μs
$\overline{\text{CE}}$ setup time (to $\overline{\text{PGM}}$↓)	t_{CES}		2			μs
Input data setup time (to $\overline{\text{PGM}}$↓)	t_{DS}		2			μs
Address hold time (from $\overline{\text{OE}}$↑)	t_{AH}		2			μs
Input data hold time (from $\overline{\text{PGM}}$↑)	t_{DH}		2			μs
Data output float delay time from $\overline{\text{OE}}$↑	t_{DF}		0		250	ns
V_{PP} setup time (to $\overline{\text{PGM}}$↓)	t_{VPS}		1.0			ms
V_{DD} setup time (to $\overline{\text{PGM}}$↓)	t_{VDS}		1.0			ms
Program pulse width	t_{PW}		0.095		0.105	ms
Valid data delay time from $\overline{\text{OE}}$↓	t_{OE}				1	μs
$\overline{\text{OE}}$ hold time	t_{OEH}		2			μs

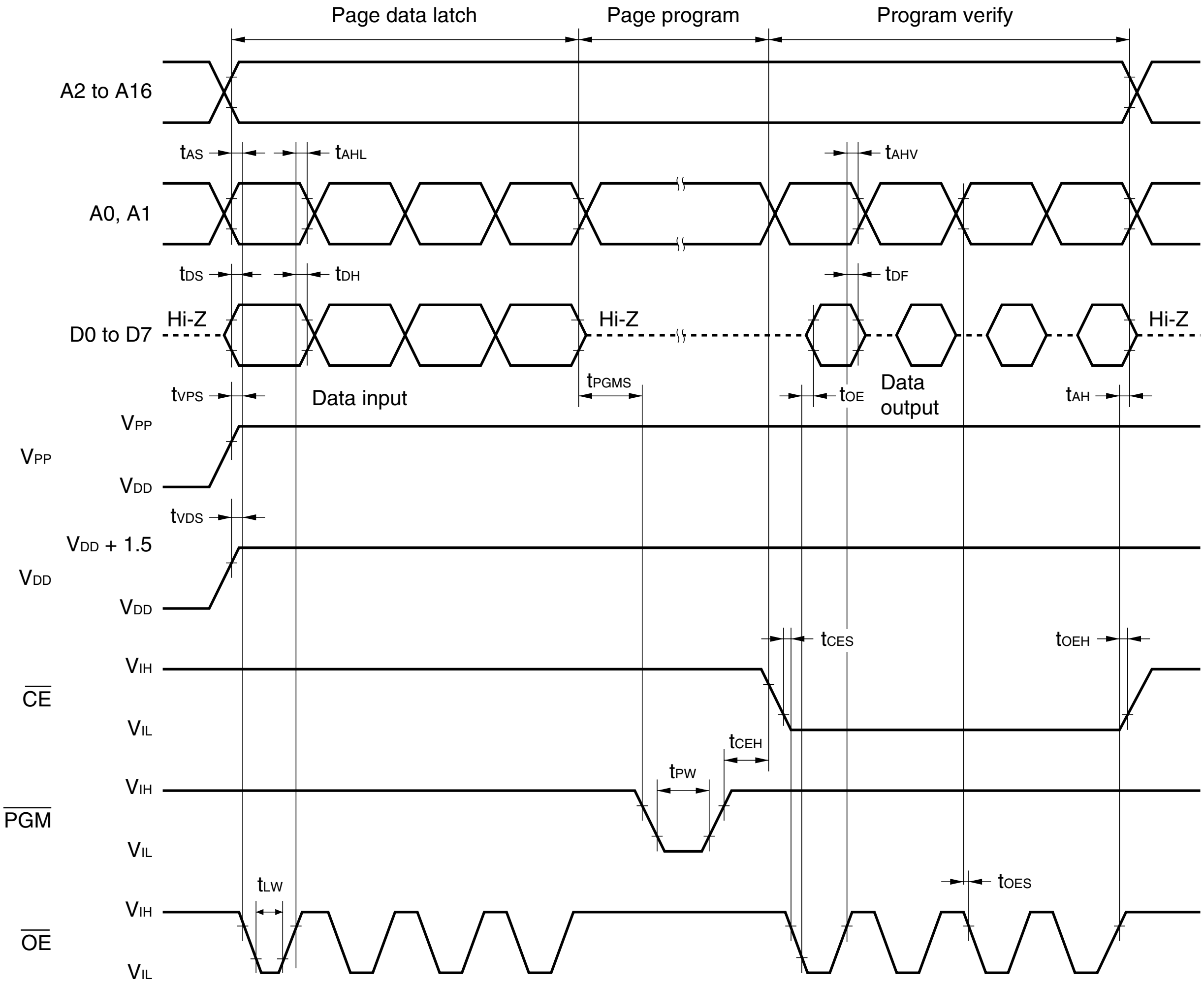
(2) PROM read mode (T_A = 25 ±5°C, V_{DD} = 5.0 ±0.5 V, V_{PP} = V_{DD} ±0.6 V)

Parameter	Symbol	Conditions	MIN.	TYP.	MAX.	Unit
Data output delay time from address	t_{ACC}	$\overline{CE}$ = $\overline{OE}$ = V_{IL}			800	ns
Data output delay time from $\overline{CE}$↓	t_{CE}	$\overline{OE}$ = V_{IL}			800	ns
Data output delay time from $\overline{OE}$↓	t_{OE}	$\overline{CE}$ = V_{IL}			200	ns
Data output float delay time from $\overline{OE}$↑	t_{DF}	$\overline{CE}$ = V_{IL}	0		60	ns
Data hold time from address	t_{OH}	$\overline{CE}$ = $\overline{OE}$ = V_{IL}	0			ns

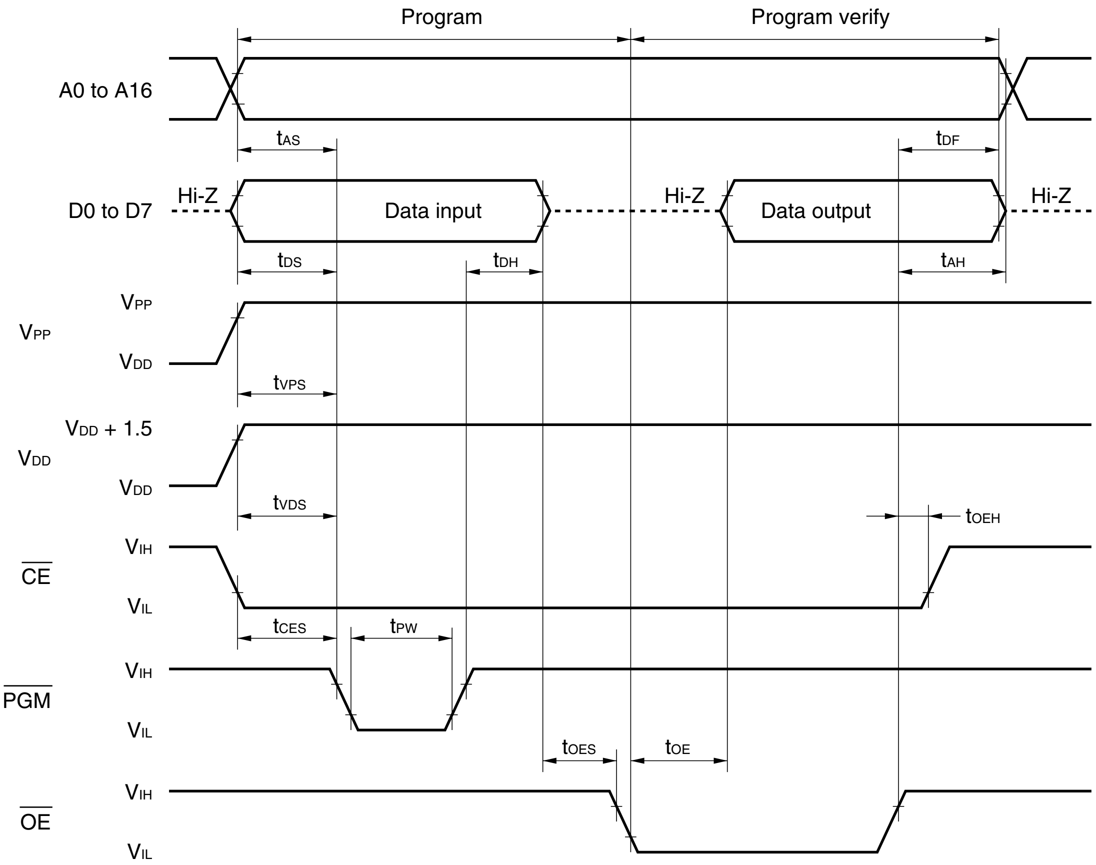
(3) PROM programming mode setting (T_A = 25°C, V_{SS} = 0 V)

Parameter	Symbol	Conditions	MIN.	TYP.	MAX.	Unit
PROM programming mode setup time	t_{SMA}		10			μs

PROM Write Mode Timing (Page Program Mode)



PROM Write Mode Timing (Byte Program Mode)

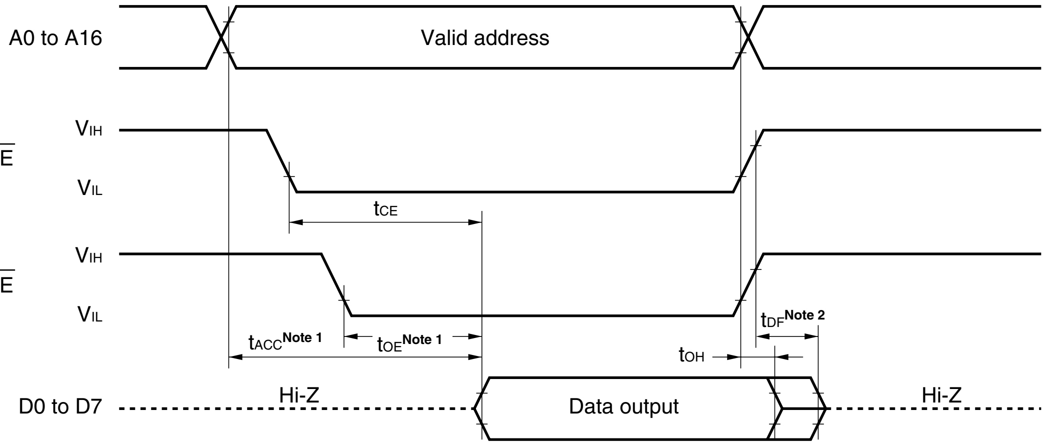


Cautions 1. V_{DD} should be applied before V_{PP}, and cut after V_{PP}.

2. V_{PP} should not exceed +13.5 V, including overshoot.

3. Disconnection during application of 12.5 V to V_{PP} may have an adverse effect on reliability.

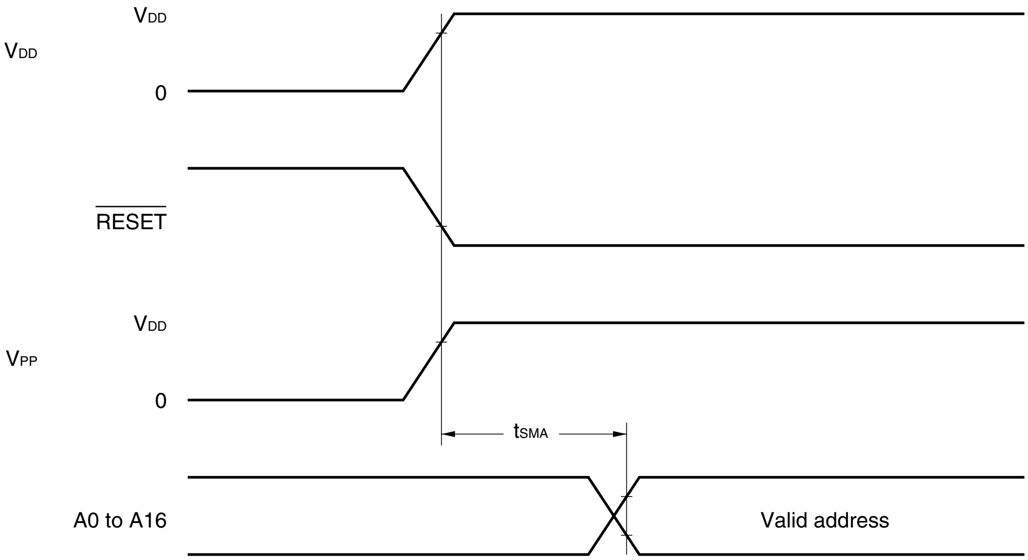
PROM Read Mode Timing



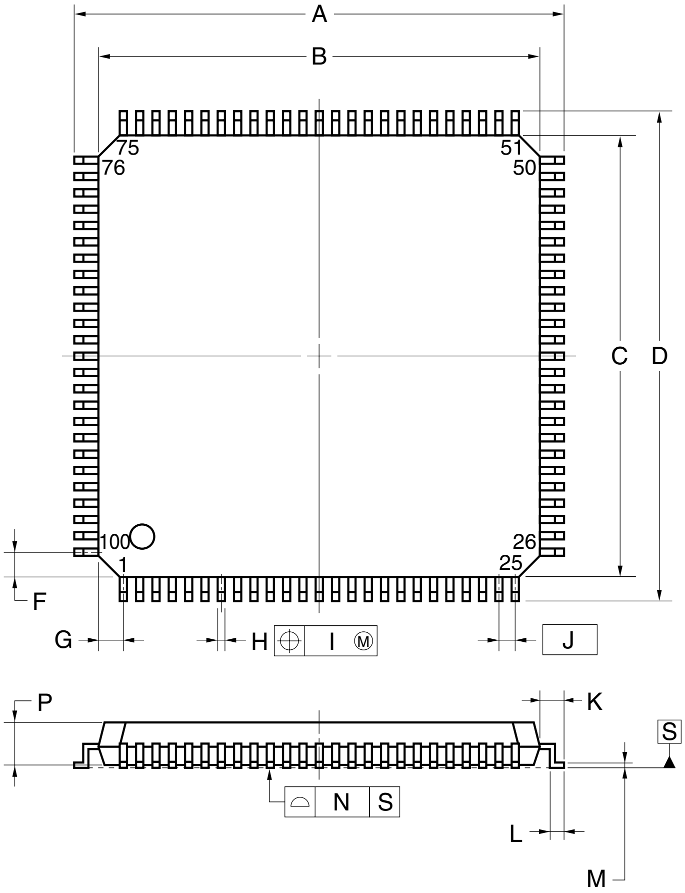
Notes 1. If you want to read within the t_{ACC} range, make the $\overline{OE}$ input delay time from the fall of $\overline{CE}$ the maximum of $t_{ACC} - t_{OE}$.

2. t_{DF} is the time from when either $\overline{OE}$ or $\overline{CE}$ first reaches V_{IH}.

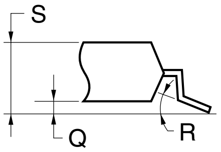
PROM Programming Mode Setting Timing



100-PIN PLASTIC LQFP (FINE PITCH) (14x14)



detail of lead end



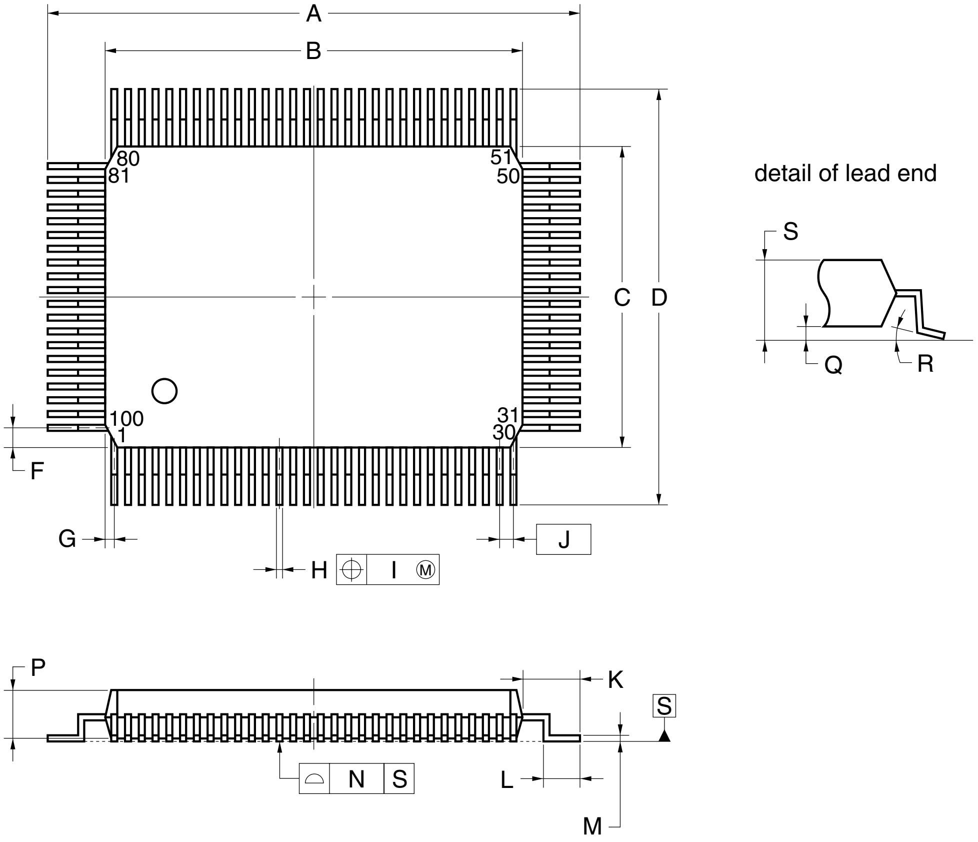
NOTE

Each lead centerline is located within 0.08 mm of its true position (T.P.) at maximum material condition.

ITEM	MILLIMETERS
A	16.00±0.20
B	14.00±0.20
C	14.00±0.20
D	16.00±0.20
F	1.00
G	1.00
H	$0.22^{+0.05}_{-0.04}$
I	0.08
J	0.50 (T.P.)
K	1.00±0.20
L	0.50±0.20
M	$0.17^{+0.03}_{-0.07}$
N	0.08
P	1.40±0.05
Q	0.10±0.05
R	$3°^{+7°}_{-3°}$
S	1.60 MAX.

S100GC-50-8EU, 8EA-2

100-PIN PLASTIC QFP (14x20)



NOTE

Each lead centerline is located within 0.15 mm of its true position (T.P.) at maximum material condition.

ITEM	MILLIMETERS
A	23.6±0.4
B	20.0±0.2
C	14.0±0.2
D	17.6±0.4
F	0.8
G	0.6
H	0.30±0.10
I	0.15
J	0.65 (T.P.)
K	1.8±0.2
L	0.8±0.2
M	$0.15^{+0.10}_{-0.05}$
N	0.10
P	2.7±0.1
Q	0.1±0.1
R	5°±5°
S	3.0 MAX.

P100GF-65-3BA1-4

<R>

CHAPTER 27 RECOMMENDED SOLDERING CONDITIONS

This product should be soldered and mounted under the following recommended conditions.

For soldering methods and conditions other than those recommended below, contact an NEC Electronics sales representative.

For technical information, see the following website.

Semiconductor Device Mount Manual (http://www.necel.com/pkg/en/mount/index.html)

Table 27-1. Surface Mounting Type Soldering Conditions

(1) 100-pin plastic QFP (14 × 20)
μPD780306GF-×××-3BA, 780306GF(A)-×××-3BA, 780306YGF-×××-3BA,
μPD780308GF-×××-3BA, 780308GF(A)-×××-3BA, 780308YGF-×××-3BA,
μPD78P0308GF-3BA, 78P0308YGF-3BA

Soldering Method	Soldering Conditions	Recommended Condition Symbol
Infrared reflow	Package peak temperature: 235°C, Time: 30 seconds max. (at 210°C or higher), Count: Three times or less	IR35-00-3
VPS	Package peak temperature: 215°C, Time: 25 to 40 seconds (at 200°C or higher), Count: Three times or less	VP15-00-3
Wave soldering	Solder bath temperature: 260°C max., Time: 10 seconds max., Count: Once, Preheating temperature: 120°C max. (package surface temperature)	WS60-00-1
Partial heating	Pin temperature: 350°C max., Time: 3 seconds max. (per pin row)	–

(2) 100-pin plastic LQFP (fine pitch) (14 × 14)
μPD780306GC-×××-8EU, 780306YGC-×××-8EU,
μPD780308GC-×××-8EU, 780308YGC-×××-8EU

Soldering Method	Soldering Conditions	Recommended Condition Symbol
Infrared reflow	Package peak temperature: 235°C, Time: 30 seconds max. (at 210°C or higher), Count: Twice or less	IR35-00-2
VPS	Package peak temperature: 215°C, Time: 25 to 40 seconds (at 200°C or higher), Count: Twice or less	VP15-00-2
Partial heating	Pin temperature: 350°C max., Time: 3 seconds max. (per pin row)	–

Caution Do not use different soldering methods together (except for partial heating).

(3) 100-pin plastic LQFP (fine pitch) (14 × 14)

μPD78P0308GC-8EU, 78P0308YGC-8EU

Soldering Method	Soldering Conditions	Recommended Condition Symbol
Infrared reflow	Package peak temperature: 235°C, Time: 30 seconds max. (at 210°C or higher), Count: Twice or less, Exposure limit: 7 days[Note] (after that, prebake at 125°C for 10 to 72 hours)	IR35-107-2
VPS	Package peak temperature: 215°C, Time: 25 to 40 seconds (at 200°C or higher), Count: Twice or less, Exposure limit: 7 days[Note] (after that, prebake at 125°C for 10 to 72 hours)	VP15-107-2
Partial heating	Pin temperature: 350°C max., Time: 3 seconds max. (per pin row)	–

(4) 100-pin plastic QFP (14 × 20)

μPD780306GF-×××-3BA-A, 780306YGF-×××-3BA-A,
μPD780308GF-×××-3BA-A, 780308YGF-×××-3BA-A,
μPD78P0308GF-3BA-A, 78P0308YGF-3BA-A

Soldering Method	Soldering Conditions	Recommended Condition Symbol
Infrared reflow	Package peak temperature: 260°C, Time: 60 seconds max. (at 220°C or higher), Count: Three times or less, Exposure limit: 3 days[Note] (after that, prebake at 125°C for 20 to 72 hours)	IR60-203-3
Wave soldering	For details, contact an NEC Electronics sales representative.	–
Partial heating	Pin temperature: 350°C max., Time: 3 seconds max. (per pin row)	–

(5) 100-pin plastic LQFP (fine pitch) (14 × 14)

μPD780306GC-×××-8EU-A, 780306YGC-×××-8EU-A,
μPD780308GC-×××-8EU-A, 780308YGC-×××-8EU-A,
μPD78P0308GC-8EU-A, 78P0308YGC-8EU-A

Soldering Method	Soldering Conditions	Recommended Condition Symbol
Infrared reflow	Package peak temperature: 260°C, Time: 60 seconds max. (at 220°C or higher), Count: Three times or less, Exposure limit: 7 days[Note] (after that, prebake at 125°C for 20 to 72 hours)	IR60-207-3
Partial heating	Pin temperature: 350°C max., Time: 3 seconds max. (per pin row)	–

Note After opening the dry pack, store it at 25°C or less and 65% RH or less for the allowable storage period.

Caution Do not use different soldering methods together (except for partial heating).

Remark Products that have the part numbers suffixed by “-A” are lead-free products.

<R>

APPENDIX A DEVELOPMENT TOOLS

The following development tools are available for the development of systems which employ the μPD780308, 780308Y Subseries.

Figure A-1 shows the configuration of the development tools.

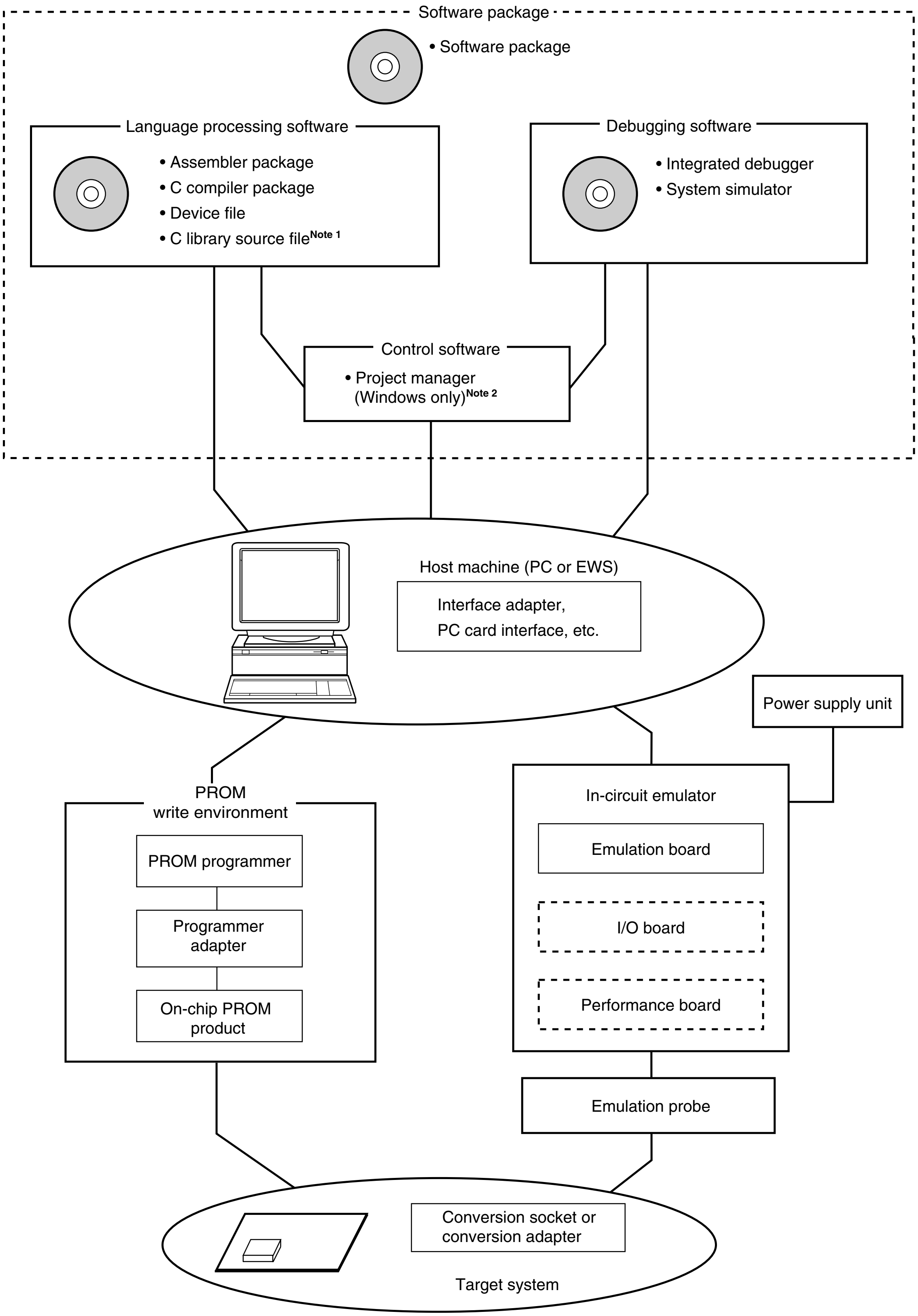
- **Support for PC98-NX series**

  Unless otherwise specified, products supported by IBM PC/AT™ compatible machines can be used for PC98-NX series computers. When using PC98-NX series computers, refer to the description for IBM PC/AT compatible machines.

- **Windows**

  Unless otherwise specified, "Windows" means the following OSs.
  - Windows 98
  - Windows 2000
  - Windows NT™ Ver. 4.0
  - Windows XP

Figure A-1. Configuration of Development Tools



Notes 1. The C library source file is not included in the software package.

2. The project manager PM plus is included in the assembler package. PM plus is only used for Windows.

A.1 Software Package

SP78K0 Software package	This package contains various software tools for 78K/0 Series development. The following tools are included. RA78K0, CC78K0, ID78K0-NS, SM78K0, and various device files
	Part Number: μS××××SP78K0

Remark ×××× in the part number differs depending on the OS used.

μS××××SP78K0

××××	Host Machine	OS	Supply Medium
AB17	PC-9800 series, IBM PC/AT and compatibles	Windows (Japanese version)	CD-ROM
BB17		Windows (English version)	

A.2 Language Processing Software

RA78K0 Assembler package	This assembler converts programs written in mnemonics into object codes executable with a microcontroller. Further, this assembler is provided with functions capable of automatically creating symbol tables and branch instruction optimization. This assembler should be used in combination with a device file (DF78064) (sold separately). **<Caution when using RA78K0 in PC environment>** This assembler package is a DOS-based application. It can also be used in Windows, however, by using PM plus (included in assembler package) in Windows.
	Part number: μS××××RA78K0
CC78K0 C compiler package	This compiler converts programs written in C language into object codes executable with a microcontroller. This compiler should be used in combination with an assembler package and device file (both sold separately). **<Caution when using CC78K0 in PC environment>** This C compiler package is a DOS-based application. It can also be used in Windows, however, by using PM plus (included in assembler package) in Windows.
	Part number: μS××××CC78K0
DF78064[Note 1] Device file	This file contains information peculiar to the device. This device file should be used in combination with tools (RA78K0, CC78K0, SM78K0, ID78K0-NS, and ID78K0) (sold separately). The corresponding OS and host machine differ depending on the tool used.
	Part number: μS××××DF78064[Note 2]
CC78K0-L[Note 3] C library source file	This is a source file of functions configuring the object library included in the C compiler package. This file is required to match the object library included in C compiler package to the user's specifications. It does not depend on the operating environment because it is a source file.
	Part number: μS××××CC78K0-L

Notes **1.** The DF78064 can be used in common with the RA78K0, CC78K0, SM78K0, ID78K0-NS, and ID78K0.

2. The DF78064 is for the μPD780308, 780308Y, 78064, and 78064Y Subseries.

3. CC78K0-L is not included in the software package (SP78K0).

Remark ×××× in the part number differs depending on the host machine and OS used.

μS××××RA78K0
μS××××CC78K0
μS××××CC78K0-L

××××	Host Machine	OS	Supply Medium
AB17	PC-9800 series,	Windows (Japanese version)	CD-ROM
BB17	IBM PC/AT and compatibles	Windows (English version)	
3P17	HP9000 series 700TM	HP-UXTM (Rel. 10.10)	
3K17	SPARCstationTM	SunOSTM (Rel. 4.1.4), SolarisTM (Rel. 2.5.1)	

μS××××DF78064

××××	Host Machine	OS	Supply Medium
AB13	PC-9800 series,	Windows (Japanese version)	3.5-inch 2HD FD
BB13	IBM PC/AT and compatibles	Windows (English version)	

A.3 Control Software

PM plus Project manager	This is control software designed to enable efficient user program development in the Windows environment. All operations used in development of a user program, such as starting the editor, building, and starting the debugger, can be performed from PM plus. **<Caution>** PM plus is included in the assembler package (RA78K0). It can only be used in Windows.

A.4 PROM Programming Tools

A.4.1 Hardware

PG-1500[Note] PROM programmer	This PROM programmer allows users to encode the PROM in single-chip microcontrollers stand-alone or using a host machine. This requires connection of the accompanying board and separately-sold PROM programmer adapter to the PROM programmer. Besides internal PROMs, general discrete PROM devices whose capacities range from 256 Kb to 4 Mb can be programmed.
PA-78P0308GC PROM programmer adapter	This PROM programmer adapter is for the μPD78P0308 and 78P0308Y, and should be connected to the PG-1500. This adapter is for a 100-pin plastic LQFP (GC-8EU type).
PA-78P0308GF PROM programmer adapter	This PROM programmer adapter is for the μPD78P0308 and 78P0308Y, and should be connected to the PG-1500. This adapter is for a 100-pin plastic QFP (GF-3BA type).

Note Production discontinued

A.4.2 Software

PG-1500 controller[Note]	This software allows users to control the PG-1500 from a host machine which is connected to the PG-1500 via serial/parallel interface cable(s).
	Part Number: μS××××PG1500

Note Production discontinued

Remark ×××× in the part number differs depending on the host machine and OS used.

μS××××PG1500

××××	Host Machine	OS	Supply Medium
5A13	PC-9800 series	MS-DOS (Ver. 3.30 to Ver. 6.2[Note 1])	3.5-inch 2HD
7B13	IBM PC/AT and compatibles	**Note 2**	3.5-inch 2HD

Notes 1. Although a task swap function is incorporated in MS-DOS Ver. 5.0 or later, this function cannot be used with the above software.

2. The following OSs for IBM PCs are supported (Ver. 5.0 or later has a task swap function, but this function cannot be used with the above software).

OS	Version
PC DOS	Ver.5.02 to Ver.6.3 J6.1/V to J6.3/V (Only the English version is supported.)
MS-DOS	Ver.5.0 to Ver.6.22 5.0/V to 6.2/V (Only the English version is supported.)
IBM DOS™	J5.02/V (Only the English version is supported.)

A.5 Debugging Tools (Hardware)

A.5.1 When using in-circuit emulator IE-78K0-NS, IE-78K0-NS-A

IE-78K0-NS In-circuit emulator		The in-circuit emulator serves to debug hardware and software when developing application systems using a 78K/0 Series product. It can be used with an integrated debugger (ID78K0-NS). This emulator should be used in combination with a power supply unit, emulation probe, and interface adapter, which is required to connect this emulator to the host machine.
IE-78K0-NS-PA Performance board		This board is used for extending the IE-78K0-NS functions. With the addition of this board, the addition of a coverage function, enhancement of tracer and timer functions, and other such debugging function enhancements are possible.
IE-78K0-NS-A In-circuit emulator		In-circuit emulator that combines the IE-78K0-NS and IE-78K0-NS-PA
IE-70000-MC-PS-B Power supply unit		This adapter is used for supplying power from a 100 to 240 V AC outlet.
IE-70000-CD-IF-A PC card interface		This is the PC card and interface cable required when using a notebook-type computer as the IE-78K0-NS host machine (PCMCIA socket compatible).
IE-70000-PC-IF-C Interface adapter		This adapter is required when using an IBM PC/AT compatible computer as the IE-78K0-NS host machine (ISA bus compatible).
IE-70000-PCI-IF-A Interface adapter		This adapter is required when using a PC with a PCI bus as the IE-78K0-NS host machine.
IE-780308-NS-EM1 Emulation board		This board emulates the operations of the peripheral hardware peculiar to a device. It should be used in combination with an in-circuit emulator.
NP-100GC NP-H100GC-TQ Emulation probe		This probe is used to connect the in-circuit emulator to the target system and is designed for a 100-pin plastic LQFP (GC-8EU type). It should be used in combination with the TGC-100SDW.
	TGC-100SDW Conversion adapter	This conversion socket connects the NP-100GC or NP-H100GC-TQ to the target system board designed to mount a 100-pin plastic LQFP (GC-8EU type).
NP-100GF-TQ NP-H100GF-TQ Emulation probe		This probe is used to connect the in-circuit emulator to the target system and is designed for a 100-pin plastic QFP (GF-3BA type). It should be used in combination with the TGF-100RBP.
	TGF-100RBP Conversion adapter	This conversion socket connects the NP-100GF-TQ or NP-H100GF-TQ to the target system board designed to mount a 100-pin plastic QFP (GF-3BA type).
NP-100GF Emulation probe		This probe is used to connect the in-circuit emulator to the target system and is designed for a 100-pin plastic QFP (GF-3BA type).
	EV-9200GF-100 Conversion socket (See **Figures A-3** and **A-4**)	This conversion socket connects the NP-100GF to the target system board designed to mount a 100-pin plastic QFP (GF-3BA type).

Remarks 1. NP-100GC, NP-100GF, NP-100GF-TQ, NP-H100GC-TQ, and NP-H100GF-TQ are products of Naito Densei Machida Mfg. Co., Ltd.
Contact: Naito Densei Machida Mfg. Co., Ltd. +81-45-475-4191

2. TGC-100SDW and TGF-100RBP are products of TOKYO ELETECH CORPORATION.
Inquiry: Daimaru Kogyo, Ltd. Phone: Tokyo Electronics Dept. +81-3-3820-7112
Osaka Electronics 2nd Dept. +81-6-6244-6672

3. EV-9200GF-100 is sold in a set of five units.

4. TGC-100SDW and TGF-100RBP are sold in single units.

A.5.2 When using in-circuit emulator IE-78001-R-A[Note]

IE-78001-R-A[Note] In-circuit emulator		This is an in-circuit emulator for debugging the hardware and software when an application system using the 78K/0 Series is developed. It can be used with an integrated debugger (ID78K0). This emulator is used with an emulation probe and interface adapter for connecting a host machine.
IE-70000-98-IF-C Interface adapter		This adapter is necessary when a PC-9800 series PC (except notebook type) is used as the host machine for the IE-78001-R-A (C bus compatible).
IE-70000-PC-IF-C Interface adapter		This adapter is necessary when an IBM PC/AT or compatible machine is used as the host machine for the IE-78001-R-A (ISA bus compatible).
IE-780308-R-EM[Note] Emulation board		This board is used with an in-circuit emulator to emulate device-specific peripheral hardware.
EP-78064GC-R Emulation probe		This probe is for a 100-pin plastic LQFP (GC-8EU type) and connects an in-circuit emulator and the target system.
	TGC-100SDW Conversion adapter (See **Figure A-2**)	This conversion adapter connects the EP-78064GC-R to the target system board designed to mount a 100-pin plastic LQFP (GC-8EU type).
EP-78064GF-R Emulation probe		This probe is for a 100-pin plastic QFP (GF-3BA type) and connects an in-circuit emulator and the target system.
	EV-9200GF-100 Conversion socket (See **Figures A-3** and **A-4**)	This conversion socket connects the EP-78064GF-R to the target system board designed to mount a 100-pin plastic QFP (GF-3BA type).

Note Production discontinued

Remarks 1. TGC-100SDW is a product of TOKYO ELETECH CORPORATION.
Inquiry: Daimaru Kogyo, Ltd. Phone: Tokyo Electronics Dept. +81-3-3820-7112
Osaka Electronics 2nd Dept. +81-6-6244-6672

2. TGC-100SDW is sold in single units.

3. EV-9200GF-100 is sold in a set of five units.

A.6 Debugging Tools (Software)

SM78K0 System simulator	This is a system simulator for the 78K/0 Series. The SM78K0 is Windows-based software. It is used to perform debugging at the C source level or assembler level while simulating the operation of the target system on a host machine. Use of the SM78K0 allows the execution of application logical testing and performance testing on an independent basis from hardware development, thereby providing higher development efficiency and software quality. The SM78K0 should be used in combination with a device file (DF78064) (sold separately).
	Part Number: μS××××SM78K0
ID78K0-NS Integrated debugger (supporting in-circuit emulators IE-78K0-NS and IE-78K0-NS-A)	This debugger supports the in-circuit emulators for the 78K/0 Series. The ID78K0-NS is Windows-based software. It has improved C-compatible debugging functions and can display the results of tracing with the source program using an integrating window function that associates the source program, disassemble display, and memory display with the trace result. It should be used in combination with a device file (sold separately).
ID78K0 Integrated debugger (supporting in-circuit emulator IE-78001-R-A)	Part Number: μS××××ID78K0-NS μS××××ID78K0

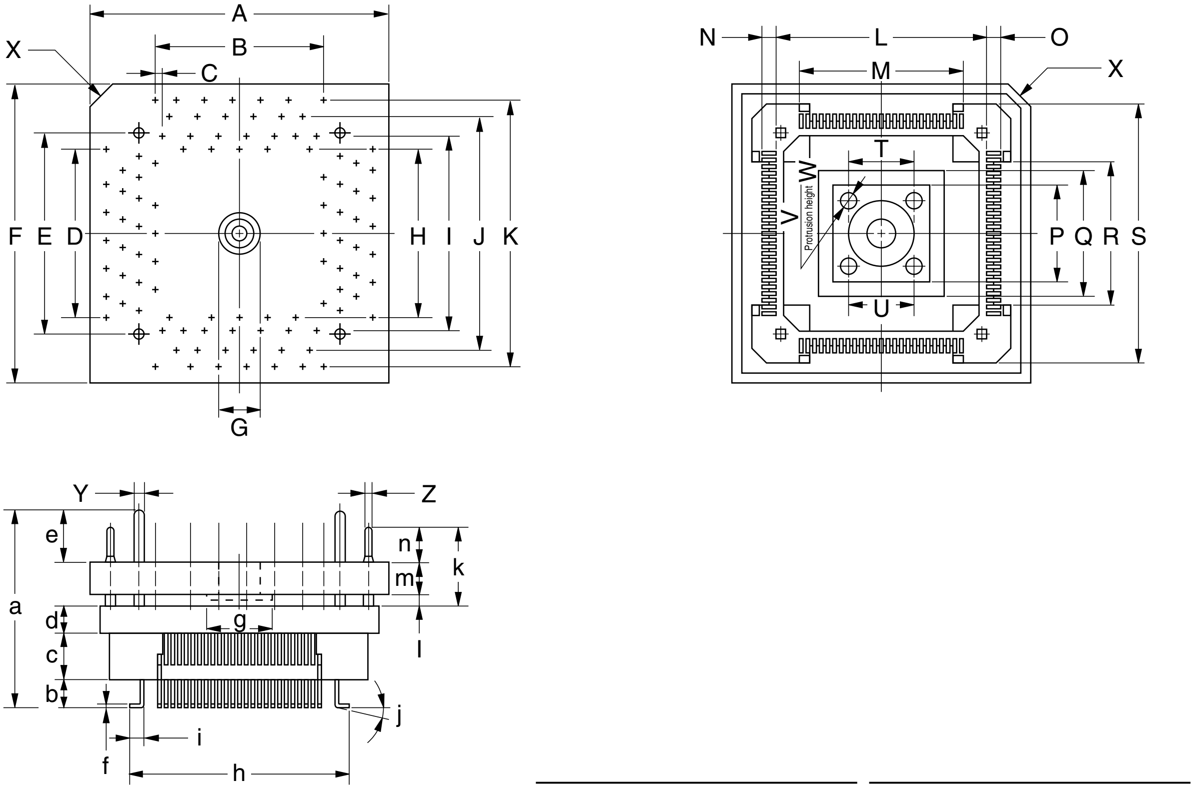
Remark ×××× in the part number differs depending on the host machine and OS used.

μS××××SM78K0
μS××××ID78K0-NS
μS××××ID78K0

××××	Host Machine	OS	Supply Medium
AB17	PC-9800 series, IBM PC/AT and compatibles	Windows (Japanese version)	CD-ROM
BB17		Windows (English version)	

A.7 Drawing for Conversion Adapter (TGC-100SDW[Note])

Figure A-2. TGC-100SDW[Note] Drawing (For Reference Only)



ITEM	MILLIMETERS	INCHES
A	21.55	0.848
B	0.5x24=12	0.020x0.945=0.472
C	0.5	0.020
D	0.5x24=12	0.020x0.945=0.472
E	15.0	0.591
F	21.55	0.848
G	ϕ3.55	ϕ0.140
H	10.9	0.429
I	13.3	0.524
J	15.7	0.618
K	18.1	0.713
L	13.75	0.541
M	0.5x24=12.0	0.020x0.945=0.472
N	1.125±0.3	0.044±0.012
O	1.125±0.2	0.044±0.008
P	7.5	0.295
Q	10.0	0.394
R	11.3	0.445
S	18.1	0.713
T	ϕ5.0	ϕ0.197
U	5.0	0.197
V	4-ϕ1.3	4-ϕ0.051
W	1.8	0.071
X	C 2.0	C 0.079
Y	ϕ0.9	ϕ0.035
Z	ϕ0.3	ϕ0.012

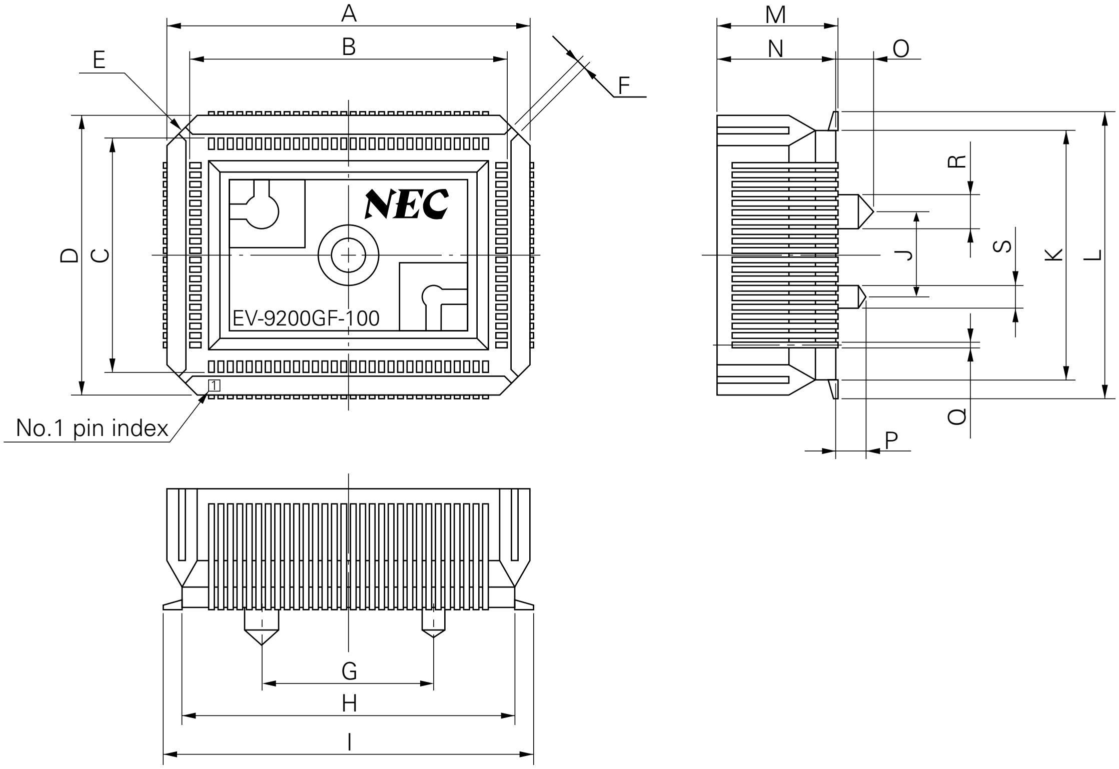
ITEM	MILLIMETERS	INCHES
a	14.45	0.569
b	1.85±0.25	0.073±0.010
c	3.5	0.138
d	2.0	0.079
e	3.9	0.154
f	0.25	0.010
g	ϕ4.5	ϕ0.177
h	16.0	0.630
i	1.125±0.3	0.044±0.012
j	0~5°	0.000~0.197°
k	5.9	0.232
l	0.8	0.031
m	2.4	0.094
n	2.7	0.106

TGC-100SDW-G1E

Note Product of TOKYO ELETECH CORPORATION.

A.8 Drawing and Footprint for Conversion Socket (EV-9200GF-100)

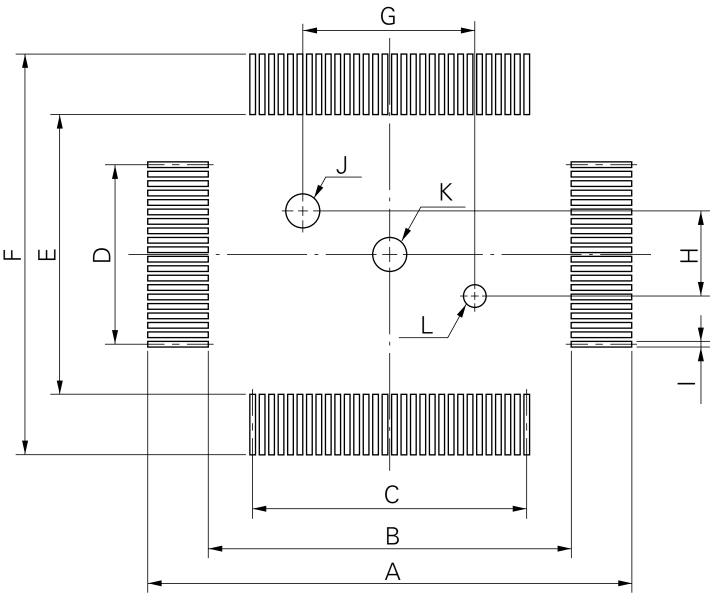
Figure A-3. EV-9200GF-100 Drawing (For Reference Only)



EV-9200GF-100-G0

ITEM	MILLIMETERS	INCHES
A	24.6	0.969
B	21	0.827
C	15	0.591
D	18.6	0.732
E	4-C 2	4-C 0.079
F	0.8	0.031
G	12.0	0.472
H	22.6	0.89
I	25.3	0.996
J	6.0	0.236
K	16.6	0.654
L	19.3	0.76
M	8.2	0.323
N	8.0	0.315
O	2.5	0.098
P	2.0	0.079
Q	0.35	0.014
R	ϕ2.3	ϕ0.091
S	ϕ1.5	ϕ0.059

Figure A-4. EV-9200GF-100 Footprint (For Reference Only)



EV-9200GF-100-P1

ITEM	MILLIMETERS	INCHES
A	26.3	1.035
B	21.6	0.85
C	0.65±0.02 × 29=18.85±0.05	$0.026^{+0.001}_{-0.002}$ × 1.142=$0.742^{+0.002}_{-0.002}$
D	0.65±0.02 × 19=12.35±0.05	$0.026^{+0.001}_{-0.002}$ × 0.748=$0.486^{+0.003}_{-0.002}$
E	15.6	0.614
F	20.3	0.799
G	12±0.05	$0.472^{+0.003}_{-0.002}$
H	6±0.05	$0.236^{+0.003}_{-0.002}$
I	0.35±0.02	$0.014^{+0.001}_{-0.001}$
J	ϕ2.36±0.03	$\phi 0.093^{+0.001}_{-0.002}$
K	ϕ2.3	ϕ0.091
L	ϕ1.57±0.03	$\phi 0.062^{+0.001}_{-0.002}$

Caution The dimensions of the mount pad for EV-9200 and that for target device (QFP) may be different in some parts. For the recommended mount pad dimensions for QFP, refer to the “Semiconductor Device Mount Manual” website (http://www.necel.com/pkg/en/mount/index.html).

A.9 Notes on Target System Design

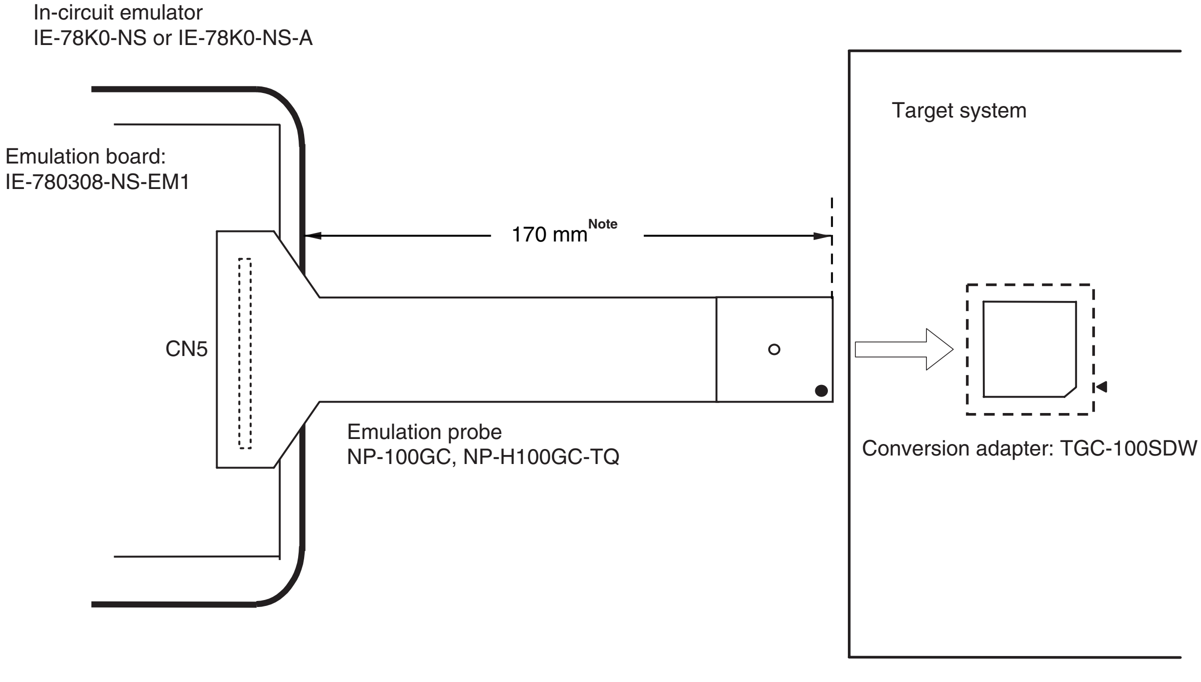
The following shows a diagram of the connection conditions between the emulation probe and conversion adapter. Design your system making allowances for conditions such as the shape of parts mounted on the target system, as shown below.

Among the products described in this appendix, NP-100GC, NP-H100GC-TQ, NP-100GF-TQ, and NP-H100GF-TQ are products of Naito Densei Machida Mfg. Co., Ltd., and TGC-100SDW and TGF-100RBP are products of TOKYO ELETECH CORPORATION.

Table A-1. Distance Between IE System and Conversion Adapter

Emulation Probe	Conversion Adapter	Distance Between IE System and Conversion Adapter
NP-100GC	TGC-100SDW	170 mm
NP-H100GC-TQ		370 mm
NP-100GF-TQ	TGF-100RBP	170 mm
NP-H100GF-TQ		370 mm

Figure A-5. Distance Between IE System and Conversion Adapter (When Using 100GC)



Note Distance when using NP-100GC. This is 370 mm when using NP-H100GC-TQ.

Figure A-6. Connection Conditions of Target System (When Using NP-100GC)

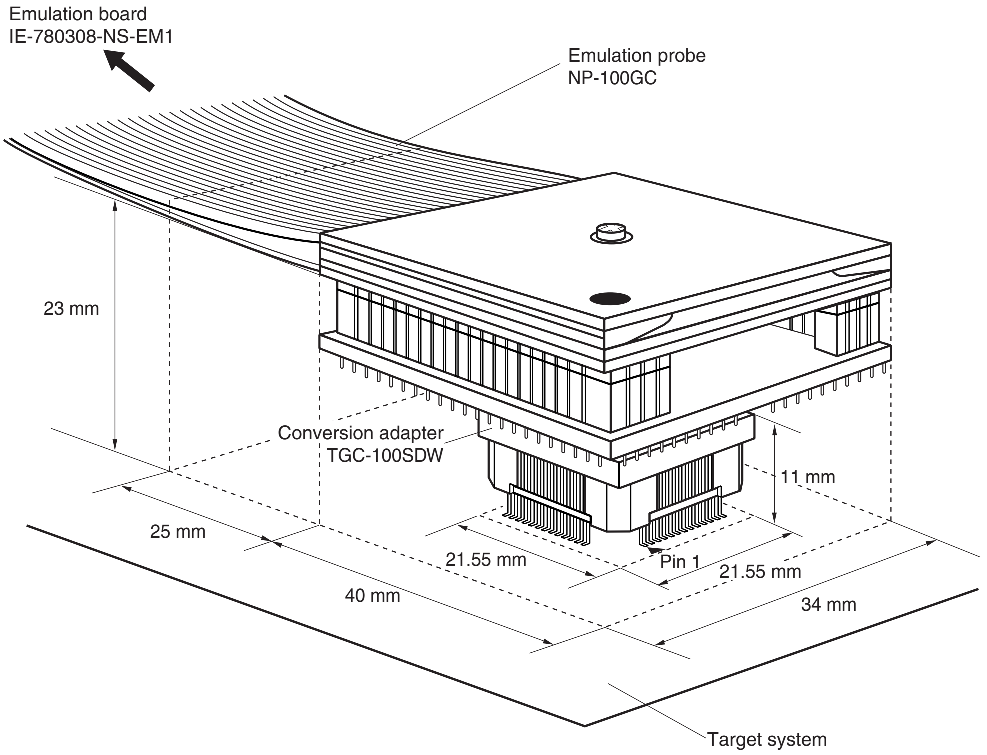


Figure A-7. Connection Conditions of Target System (When Using NP-H100GC-TQ)

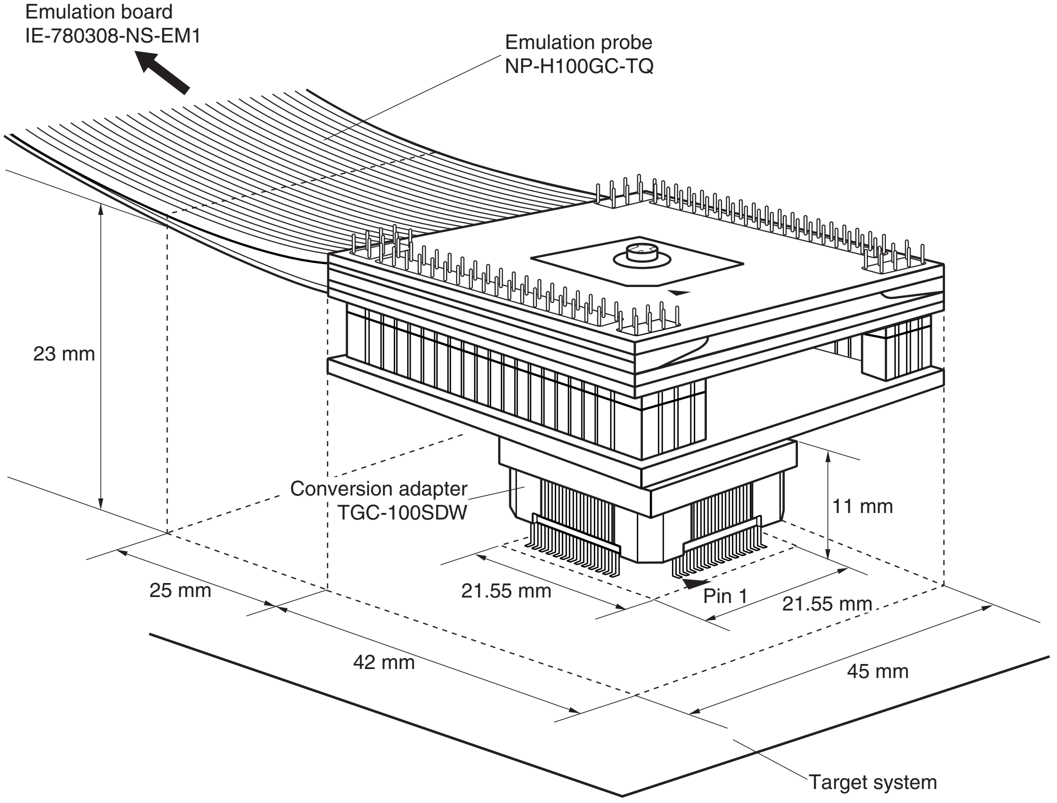
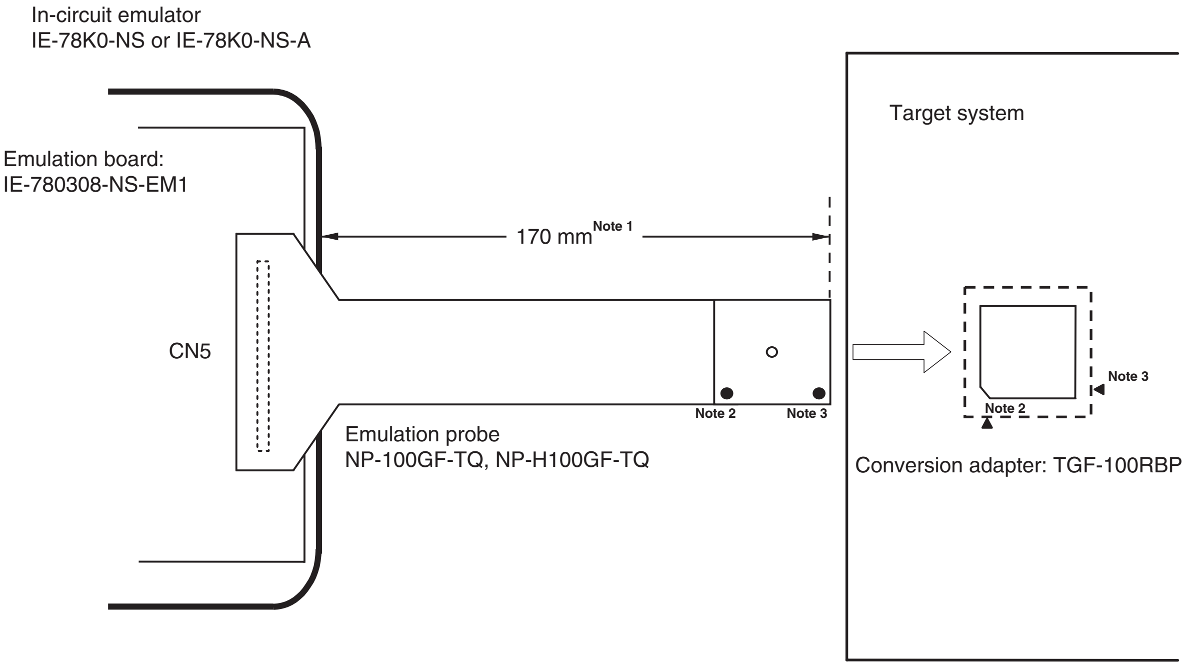


Figure A-8. Distance Between IE System and Conversion Adapter (When Using 100GF)



Notes 1. Distance when using NP-100GF-TQ. This is 370 mm when using NP-H100GF-TQ.

2. This is the position of pin 1 when using NP-100GF-TQ.

3. This is the position of pin 1 when using NP-H100GF-TQ.

Figure A-9. Connection Conditions of Target System (When Using NP-100GF-TQ)

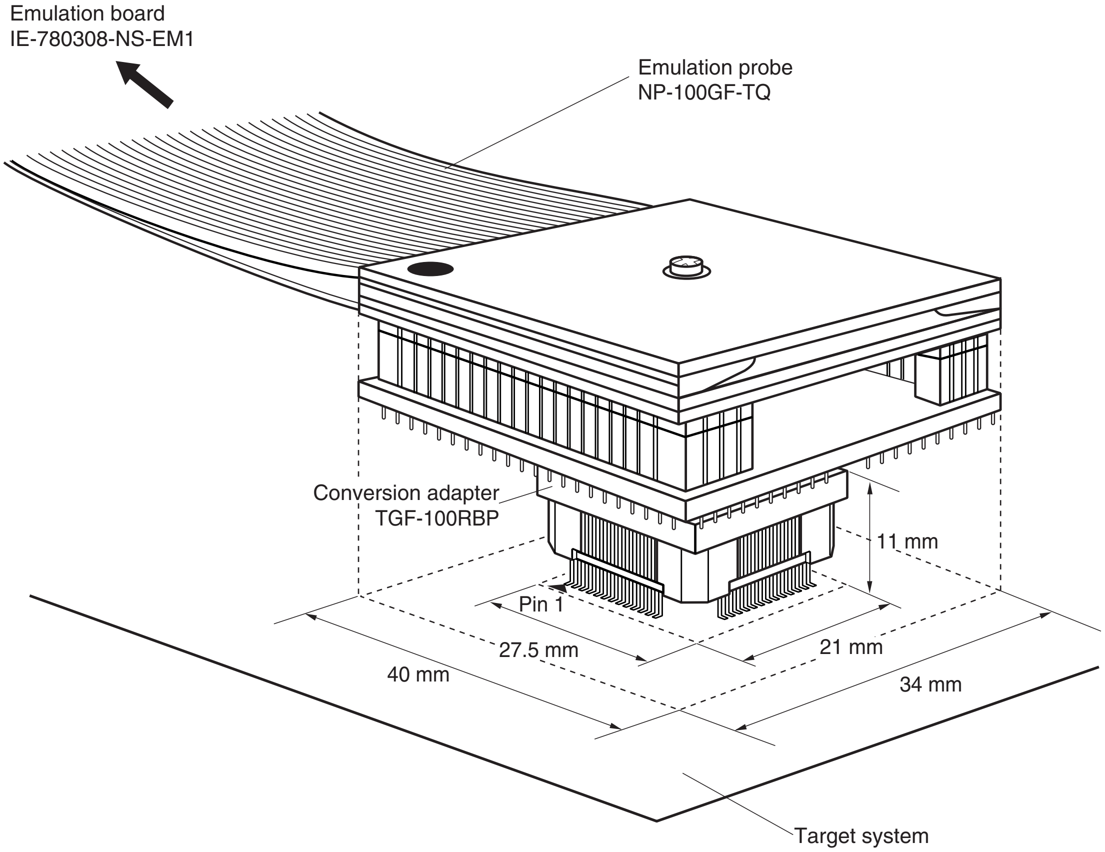
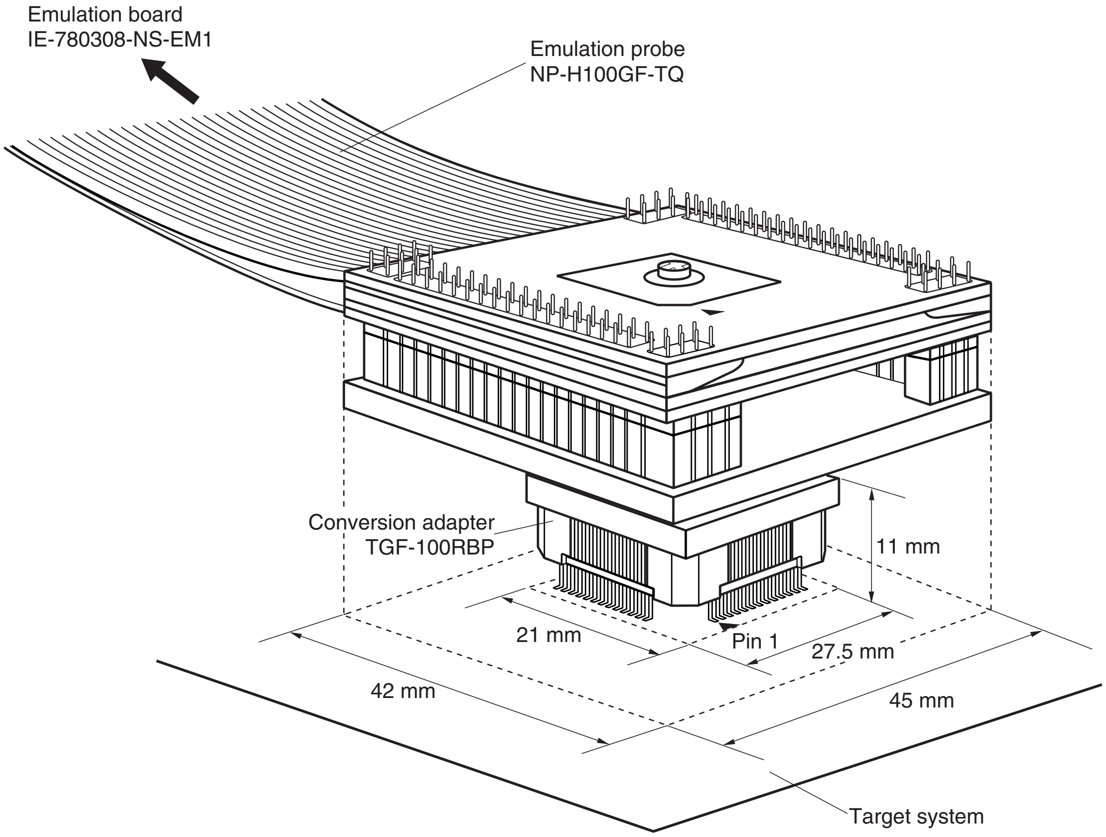


Figure A-10. Connection Conditions of Target System (When Using NP-H100GF-TQ)



APPENDIX B REGISTER INDEX

B.1 Register Name Index

B.2 Register Symbol Index

APPENDIX C REVISION HISTORY

<R> C.1 Major Revisions in This Edition

Page	Description
Throughout	Deletion of the following part numbers • μPD780306GC(A)-×××-8EU • μPD780308GC(A)-×××-8EU • μPD78P0308KL-T • μPD78P0308YKL-T Addition of the following part numbers (lead-free products) • μPD780306GF-×××-3BA-A • μPD780306GC-×××-8EU-A • μPD780306YGF-×××-3BA-A • μPD780306YGC-×××-8EU-A • μPD780308GF-×××-3BA-A • μPD780308GC-×××-8EU-A • μPD780308YGF-×××-3BA-A • μPD780308YGC-×××-8EU-A • μPD78P0308GF-3BA-A • μPD78P0308GC-8EU-A • μPD78P0308YGF-3BA-A • μPD78P0308YGC-8EU-A
p. 9	Modification of related documents
p. 26	Modification of **1.6 78K0 Series Lineup**
p. 38	Modification of **2.6 78K0 Series Lineup**
p. 82	Modification of **Caution** in **Figure 5-9 Stack Pointer Configuration**
p. 241	Modification of description of (5) in **14.4.1 Basic operations of A/D converter**
p. 435	Addition of **Caution 3** to **20.3 (1) Interrupt request flag registers (IF0L, IF0H, IF1L)**
p. 443	Addition of description and **Caution** to **20.4.1 Non-maskable interrupt request acknowledge operation**
p. 446	Addition of description to **20.4.2 Maskable interrupt request acknowledge operation**
p. 495	Addition of **CHAPTER 25 ELECTRICAL SPECIFICATIONS**
p. 528	Addition of **CHAPTER 26 PACKAGE DRAWINGS**
p. 530	Addition of **CHAPTER 27 RECOMMENDED SOLDERING CONDITIONS**
p. 532	Modification of **APPENDIX A DEVELOPMENT TOOLS**
p. 553	Addition of **C.1 Major Revisions in This Edition** to **APPENDIX C REVISION HISTORY**
pp. 505 to 507 in old edition	Deletion of **APPENDIX B EMBEDDED SOFTWARE** from the old edition

C.2 Revision History up to Previous Edition

Revisions up to the previous edition are shown below. The "Applied to:" column indicates the chapter in each edition to which the revision was applied.

(1/2)

Edition	Major Revision from Previous Edition	Applied to:
2nd edition	Addition of "μPD780306(A), 780308(A) ... under planning" Change of package as follows: • Deletion of 100-pin plastic QFP (GC-7EA type) • Addition of 100-pin plastic LQFP (GC-8EU type) Change of minimum supply voltage from 1.8 to 2.0 V	Throughout
	• Addition of description on following subseries to **1.6 78K/0 Series Line-up** μPD78075B, 78075BY, 780018, 780018Y, 780058, 780058Y, 78058F, 78058FY, 78054, 78054Y, 780964, 780924, 780228, 78044H, 78044F, 78098B, 780973, 78P0914	**CHAPTER 1 OUTLINE (μPD780308 Subseries)**
	• **2.5 Pin Configuration** Addition of connection diagram of 100-pin plastic LQFP (GC-8EU type)	**CHAPTER 2 OUTLINE (μPD780308Y Subseries)**
	Correction of following text in • **5.1.4 Data memory addressing** • **5.2.1 Control registers (a) Interrupt enable flag (IE)**, **(e) In-service priority flag (ISP)** • **5.3.1 Relative addressing** • **5.3.2 Immediate addressing** • **5.3.3 Table indirect addressing** • **5.4.2 Register addressing** • **5.4.6 Register indirect addressing** • **5.4.7 Based addressing** • **5.4.8 Based indexed addressing**	**CHAPTER 5 CPU ARCHITECTURE**
	• **7.3 Clock Generator Control Register** Change of **Figure 7-3 Processor Clock Control Register Format** Addition of **Table 7-2 Relation between CPU Clock and Minimum Instruction Execution Time**	**CHAPTER 7 CLOCK GENERATOR**
	• **9.4.1 8-bit timer/event counter mode** Addition of **Figure 9-10 Square Wave Output Operation Timing** • Correction of text in **9.4.2 16-bit timer/event counter mode** Addition of **Figure 9-13 Square Wave Output Operation Timing**	**CHAPTER 9 8-BIT TIMER/ EVENT COUNTERS 1 AND 2**
	• **11.2 Watchdog Timer Configuration** Change of **Figure 11-1 Watchdog Timer Block Diagram**	**CHAPTER 11 WATCHDOG TIMER**
	• **14.2 A/D Converter Configuration** Correction of **Figure 14-1 A/D Converter Block Diagram** Addition of caution on voltage	**CHAPTER 14 A/D CONVERTER**
	• **15.1 Serial Interface Channel 0 Functions** Addition of caution on operation mode	**CHAPTER 15 SERIAL INTERFACE CHANNEL 0 (μPD780308 Subseries)**
	• **15.3 Serial Interface Channel 0 Control Registers** Addition of caution on operation mode	

(2/2)

Edition	Major Revision from Previous Edition	Applied to:
2nd edition	• **16.1 Serial Interface Channel 0 Functions** Addition of caution on operation mode	**CHAPTER 16 SERIAL INTERFACE CHANNEL 0 (μPD780308Y Subseries)**
	• **16.3 Serial Interface Channel 0 Control Registers** Addition of caution on operation mode	
	• **17.4.2 Asynchronous serial interface (UART) mode** Change of **Figure 17-11 Receive Error Timing** Correction of description on **(3) UART mode cautions**	**CHAPTER 17 SERIAL INTERFACE CHANNEL 2**
	• **17.4.3 3-wire serial I/O mode** Addition of **Figure 17-14 Circuit of Switching in Transfer Bit Order**	
	• Addition of **17.4.4 Limitations of UART mode**	
	• **18.4.2 3-wire serial I/O mode** Addition of description on selecting MSB/LSB first Addition of **Figure 18-5 Circuit of Switching in Transfer Bit Order**	**CHAPTER 18 SERIAL INTERFACE CHANNEL 3**
	• **20.3 Interrupt Function Control Registers** Change of **Table 20-2 Various Flags Corresponding to Interrupt Request Sources**	**CHAPTER 20 INTERRUPT AND TEST FUNCTIONS**
	• **20.4 Interrupt Request Servicing Operations** Correction of **Figure 20-11 Non-Maskable Interrupt Request Acknowledge Timing** Correction of **Figure 20-12 Non-Maskable Interrupt Request Acknowledge Operation** Addition of description on flags to **Figure 20-13 Interrupt Request Acknowledge Processing Algorithm** • Correction of text in **20.4.4 Multiple interrupt request servicing** Correction of **Figure 20-16 Multiple Interrupt Example** • Correction of text in **20.4.5 Interrupt request reserve**	
	• **20.5 Test Functions** Correction of text in **20.5.2 Test input signal acknowledge operation**	
	• **A.1 Language Processing Software** Change of part number of device file from “DF780308” to “DF78064”	**APPENDIX A DEVELOPMENT TOOLS**
	• **A.3 Debugging Tools** **A.3.1 Hardware** Change of conversion adapter name from “EV-9500GC-100” to “TGC-100SDW” Deletion of 5-inch supply media supporting Windows	

For further information,
please contact:

NEC Electronics Corporation
1753, Shimonumabe, Nakahara-ku,
Kawasaki, Kanagawa 211-8668,
Japan
Tel: 044-435-5111
http://www.necel.com/

[America]

NEC Electronics America, Inc.
2880 Scott Blvd.
Santa Clara, CA 95050-2554, U.S.A.
Tel: 408-588-6000
800-366-9782
http://www.am.necel.com/

[Europe]

NEC Electronics (Europe) GmbH
Arcadiastrasse 10
40472 Düsseldorf, Germany
Tel: 0211-65030
http://www.eu.necel.com/

Hanover Office
Podbielskistrasse 166 B
30177 Hannover
Tel: 0 511 33 40 2-0

Munich Office
Werner-Eckert-Strasse 9
81829 München
Tel: 0 89 92 10 03-0

Stuttgart Office
Industriestrasse 3
70565 Stuttgart
Tel: 0 711 99 01 0-0

United Kingdom Branch
Cygnus House, Sunrise Parkway
Linford Wood, Milton Keynes
MK14 6NP, U.K.
Tel: 01908-691-133

Succursale Française
9, rue Paul Dautier, B.P. 52
78142 Velizy-Villacoublay Cédex
France
Tel: 01-3067-5800

Sucursal en España
Juan Esplandiu, 15
28007 Madrid, Spain
Tel: 091-504-2787

Tyskland Filial
Täby Centrum
Entrance S (7th floor)
18322 Täby, Sweden
Tel: 08 638 72 00

Filiale Italiana
Via Fabio Filzi, 25/A
20124 Milano, Italy
Tel: 02-667541

Branch The Netherlands
Steijgerweg 6
5616 HS Eindhoven
The Netherlands
Tel: 040 265 40 10

[Asia & Oceania]

NEC Electronics (China) Co., Ltd
7th Floor, Quantum Plaza, No. 27 ZhiChunLu Haidian
District, Beijing 100083, P.R.China
Tel: 010-8235-1155
http://www.cn.necel.com/

NEC Electronics Shanghai Ltd.
Room 2511-2512, Bank of China Tower,
200 Yincheng Road Central,
Pudong New Area, Shanghai P.R. China P.C:200120
Tel: 021-5888-5400
http://www.cn.necel.com/

NEC Electronics Hong Kong Ltd.
12/F., Cityplaza 4,
12 Taikoo Wan Road, Hong Kong
Tel: 2886-9318
http://www.hk.necel.com/

NEC Electronics Taiwan Ltd.
7F, No. 363 Fu Shing North Road
Taipei, Taiwan, R. O. C.
Tel: 02-8175-9600
http://www.tw.necel.com/

NEC Electronics Singapore Pte. Ltd.
238A Thomson Road,
#12-08 Novena Square,
Singapore 307684
Tel: 6253-8311
http://www.sg.necel.com/

NEC Electronics Korea Ltd.
11F., Samik Lavied'or Bldg., 720-2,
Yeoksam-Dong, Kangnam-Ku,
Seoul, 135-080, Korea
Tel: 02-558-3737
http://www.kr.necel.com/