User's Manual

NEC

μPD78014, 78014Y SUBSERIES

8-BIT SINGLE-CHIP MICROCONTROLLERS

μPD78011B	μPD78011BY
μPD78012B	μPD78012BY
μPD78013	μPD78013Y
μPD78014	μPD78014Y
μPD78P014	μPD78P014Y
μPD78011B (A)	
μPD78012B (A)	
μPD78013 (A)	
μPD78014 (A)	

Document No. U10085EJ7V0UM00 (7th edition)
Date Published October 1997 N

[MEMO]

NOTES FOR CMOS DEVICES

① **PRECAUTION AGAINST ESD FOR SEMICONDUCTORS**

Note:

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 once, when it has occurred. Environmental control must be adequate. When it is dry, humidifier should be used. It is recommended to avoid using insulators that easily build 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 bench and floor should be grounded. The operator should be grounded using wrist strap. Semiconductor devices must not be touched with bare hands. Similar precautions need to be taken for PW boards with semiconductor devices on it.

② **HANDLING OF UNUSED INPUT PINS FOR CMOS**

Note:

No connection for CMOS device inputs can be cause of malfunction. If no connection is provided to the input pins, it is possible that an internal input level may be generated due to noise, etc., hence causing malfunction. CMOS devices behave differently than Bipolar or NMOS devices. Input levels of CMOS devices must be fixed high or low by using a pull-up or pull-down circuitry. Each unused pin should be connected to V_{DD} or GND with a resistor, if it is considered to have a possibility of being an output pin. All handling related to the unused pins must be judged device by device and related specifications governing the devices.

③ **STATUS BEFORE INITIALIZATION OF MOS DEVICES**

Note:

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

FIP, IEBus, and QTOP are trademarks of NEC Corporation.
Microsoft and Windows are either registered trademarks or trademarks of Microsoft Corporation in the United States and/or other countries.
IBM DOS, PC/AT, and PC DOS are trademarks of International Business Machines Corporation.
HP9000 Series 300, HP9000 Series 700 and HP-UX are trademarks of Hewlett-Packard Company.
SPARCstation is a trademark of SPARC International, Inc.
SunOS is a trademark of Sun Microsystems, Inc.
Ethernet is a trademark of XEROX Corporation.
NEWS, NEWS-OS are trademarks of SONY Corporation.
OSF/Motif is a trademark of Open Softwear Foundation, Inc.
TRON is an abbreviation of The Real-time Operating system Nucleus.
ITRON is an abbreviation of Industrial TRON.

The export of these products from Japan is regulated by the Japanese government. The export of some or all of these products may be prohibited without governmental license. To export or re-export some or all of these products from a country other than Japan may also be prohibited without a license from that country. Please call an NEC sales representative.

License not needed : μPD78P014DW, 78P014YDW
The customer must judge the need for license:
μPD78011BCW-×××, 78011BGC-×××-AB8,
μPD78011BCW(A)-×××, 78011BGC(A)-×××-AB8,
μPD78011BYCW-×××, 78011BYGC-×××-AB8,
μPD78012BCW-×××, 78012BGC-×××-AB8,
μPD78012BCW(A)-×××, 78012BGC(A)-×××-AB8,
μPD78012BYCW-×××, 78012BYGC-×××-AB8,
μPD78013CW-×××, 78013GC-×××-AB8,
μPD78013CW(A)-×××, 78013GC(A)-×××-AB8,
μPD78013YCW-×××, 78013YGC-×××-AB8,
μPD78014CW-×××, 78014GC-×××-AB8,
μPD78014CW(A)-×××, 78014GC(A)-×××-AB8,
μPD78014YCW-×××, 78014YGC-×××-AB8,
μPD78P014CW, 78P014GC-AB8,
μPD78P014YCW, 78P014YGC-AB8

Caution Purchase of NEC I^2C components conveys a license under the Philips I^2C Patent Rights to use these components in an I^2C system, provided that the system conforms to the I^2C Standard Specification as defined by Philips.

The application circuits and their parameters are for reference only and are not intended for use in actual design-ins.

While NEC Corporation has been making continuous effort to enhance the reliability of its semiconductor devices, the possibility of defects cannot be eliminated entirely. To minimize risks of damage or injury to persons or property arising from a defect in an NEC semiconductor device, customers must incorporate sufficient safety measures in its design, such as redundancy, fire-containment, and anti-failure features.

NEC devices are classified into the following three quality grades:

"Standard", "Special", and "Specific". The Specific quality grade applies only to devices developed based on a customer designated "quality assurance program" for a specific application. The recommended applications of a device depend on its quality grade, as indicated below. Customers must check the quality grade of each device 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: Aircrafts, aerospace equipment, submersible repeaters, nuclear reactor control systems, life support systems or medical equipment for life support, etc.

The quality grade of NEC devices is "Standard" unless otherwise specified in NEC's Data Sheets or Data Books.

If customers intend to use NEC devices for applications other than those specified for Standard quality grade, they should contact an NEC sales representative in advance.

Anti-radioactive design is not implemented in this product.

M7 96.5

Regional Information

Some information contained in this document may vary from country to country. Before using any NEC product in your application, please contact the NEC office in your country to obtain a list of authorized representatives and distributors. They will verify:

- Device availability
- Ordering information
- Product release schedule
- Availability of related technical literature
- Development environment specifications (for example, specifications for third-party tools and components, host computers, power plugs, AC supply voltages, and so forth)
- Network requirements

In addition, trademarks, registered trademarks, export restrictions, and other legal issues may also vary from country to country.

NEC Electronics Inc. (U.S.)
Santa Clara, California
Tel: 408-588-6000
800-366-9782
Fax: 408-588-6130
800-729-9288

NEC Electronics (Germany) GmbH
Duesseldorf, Germany
Tel: 0211-65 03 02
Fax: 0211-65 03 490

NEC Electronics (UK) Ltd.
Milton Keynes, UK
Tel: 01908-691-133
Fax: 01908-670-290

NEC Electronics Italiana s.r.1.
Milano, Italy
Tel: 02-66 75 41
Fax: 02-66 75 42 99

NEC Electronics (Germany) GmbH
Benelux Office
Eindhoven, The Netherlands
Tel: 040-2445845
Fax: 040-2444580

NEC Electronics (France) S.A.
Velizy-Villacoublay, France
Tel: 01-30-67 58 00
Fax: 01-30-67 58 99

NEC Electronics (France) S.A.
Spain Office
Madrid, Spain
Tel: 01-504-2787
Fax: 01-504-2860

NEC Electronics (Germany) GmbH
Scandinavia Office
Taeby, Sweden
Tel: 08-63 80 820
Fax: 08-63 80 388

NEC Electronics Hong Kong Ltd.
Hong Kong
Tel: 2886-9318
Fax: 2886-9022/9044

NEC Electronics Hong Kong Ltd.
Seoul Branch
Seoul, Korea
Tel: 02-528-0303
Fax: 02-528-4411

NEC Electronics Singapore Pte. Ltd.
United Square, Singapore 1130
Tel: 253-8311
Fax: 250-3583

NEC Electronics Taiwan Ltd.
Taipei, Taiwan
Tel: 02-719-2377
Fax: 02-719-5951

NEC do Brasil S.A.
Cumbica-Guarulhos-SP, Brasil
Tel: 011-6465-6810
Fax: 011-6465-6829

J97. 8

Main Revisions in This Edition

Pages	Description
p. 43, 44, 56, 57	The following subseries were added in sections 1.6 and 2.6, "78K/0 Series Expansion." μPD78075B, 78075BY, 780018, 780018Y, 780058, 780058Y, 780034, 780034Y, 780024, 780024Y, 78014H, 780964, 780924, 780228, 78044H, 78044F, 78098B, 780973 Subseries
p. 137 through 141	The illustrations were modified in Figures 6-6 and 6-8, "P20, P21, P23 to P26 Block Diagrams", Figures 6-7 and 6-9, "P22 and P27 Block Diagrams", and Figure 6-10, "P30 to P37 Block Diagrams".
p. 215, 219	Figures 9-10 and 9-13, "Square Wave Output Operation Timings" were added.
p. 264, 321	Cautions were added in sections 15.1 and 16.1, "Serial Interface Channel 0 Configuration".
p. 271, 330	Cautions were added in sections 15.3 and 16.3, "Serial Interface Channel 0 Control Register (2) Serial operating mode register 0 (CSIM0)".
p. 287, 310, 348, 371	Cautions were added in sections 15.4.3 and 16.4.3, "(2) (a) Bus release signal (REL), (b) Command signal (CMD), (11) Cautions on SBI mode".
p. 421	(3) MSB/LSB switching as the start bit was added in section 17.4.2, "3-wire serial I/O mode operation".
p. 439	(3) (d) Busy control option, (e) Busy & strobe control option, and (f) Bit slippage detection function in section 17.4.3 of the former edition were changed to (4) Synchronization control and the description was improved.
p. 495	Caution was added in Table 22-1, "Differences between μPD78P014, 78P014Y, and Mask ROM Version".
p. 521	APPENDIX A DIFFERENCES BETWEEN μPD78014, 78014H, AND 78018F SUBSERIES was added.
p. 528, 529	Windows compatible 5-inch FD products was erased in APPENDIX B DEVELOPMENT TOOLS.
p. 527, 529	The following products were changed from "Under development" to "Developed". IE-78000-R-A ID78K0

The mark ★ shows major revised points.

[MEMO]

PREFACE

Readers This manual has been prepared for user engineers who want to understand the functions of the μPD78014, 78014Y Subseries and design and develop its application systems and programs.

Target subseries are as follows.

- μPD78014 Subseries : μPD78011B, 78012B, 78013, 78014, 78P014
  μPD78011B(A), 78012B(A), 78013(A), 78014(A)
- μPD78014Y Subseries: μPD78011BY, 78012BY, 78013Y, 78014Y, 78P014Y

Caution Of the above members, the μPD78P014DW, 78P014YDW should be used only for experiment or function evaluation, because they are not intended for use in equipment that will be mass-produced and do not have enough reliability.

Purpose This manual is intended to help users understand the functions described following the Organization below.

Organization The μPD78014, 78014Y Subseries manual is separated into two parts: this manual and Instructions (common to the 78K/0 Series)

μPD78014, 78014Y SUBSERIES USER'S MANUAL (This manual)	78K/0 SERIES USER'S MANUAL – Instructions –
• Pin functions • Internal block functions • Interrupts • Miscellaneous on-chip peripheral functions	• CPU functions • Instruction set • Explanation of each instruction

How to Read This Manual

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

☐ When using this manual as the one for the μPD78011B(A), 78012B(A), 78013(A), 78014 (A):
→ The μPD78011, 78012B, 78013, 78014 and the μPD78011B(A), 78012B(A), 78013(A), 78014(A) are different only in quality grade. For products (A), regard the product name as follows.
μPD78011B → μPD78011B(A)   μPD78012B → μPD78012B(A)
μPD78013 → μPD78013(A)   μPD78014 → μPD78014(A)

☐ To understand the functions in general:
→ Read this manual in the order of the contents.

☐ To interpret the register format:
→ For the circled bit number, the bit name is defined as a reserved word in RA78K/0, and in CC78K/0, already defined in the header file named sfrbit.h.

- ☐ To confirm the details of the register whose register name is known:
  → Refer to **APPENDIX D REGISTER INDEX**.
- ☐ For the details of the μPD78014, 78014Y Subseries instruction function:
  → Refer to the **78K/0 SERIES USER'S MANUAL, Instructions.** (IEU1372).
- ☐ For the electrical specifications of the μPD78014, 78014Y Subseries:
  → Refer to Data Sheet.
- ☐ For the application examples of μPD78014, 78014Y Subseries functions:
  → Refer to Application Note.

Caution **The use examples in the manual apply to the general electric devices of the standard quality grade. To use the use examples in the manual for applications requiring the special quality grade, examine the quality grade of the actually used parts and circuits.**

Chapter composition This manual describes points for which functions of μPD78014 and μPD78014Y Subseries are not same in different chapters. The chapters explaining the subseries are shown in the table below.

If you use one of the subseries, you should read the chapters with √ marks.

Chapter	μPD78014 Subseries	μPD78014Y Subseries
Chapter 1 Outline (μPD78014 Subseries)	√	—
Chapter 2 Outline (μPD78014Y Subseries)	—	√
Chapter 3 Pin Function (μPD78014 Subseries)	√	—
Chapter 4 Pin Function (μPD78014Y 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 Control Circuit	√	√
Chapter 13 Buzzer Output Control Circuit	√	√
Chapter 14 A/D Converter	√	√
Chapter 15 Serial Interface Channel 0 (μPD78014 Subseries)	√	—
Chapter 16 Serial Interface Channel 0 (μPD78014Y Subseries)	—	√
Chapter 17 Serial Interface Channel 1	√	√
Chapter 18 Interrupt Functions and Test Function	√	√
Chapter 19 External Device Expansion Function	√	√
Chapter 20 Standby Function	√	√
Chapter 21 Reset Function	√	√
Chapter 22 μPD78P014, 78P014Y	√	√
Chapter 23 Instruction Set	√	√

Differences between μPD78014 Subseries and μPD78014Y Subseries

The μPD78014 Subseries and μPD78014Y Subseries differ in some of the serial interface channel 0 modes as shown below.

Mode of Serial Interface Channel 0	μPD78014 Subseries	μPD78014Y Subseries
3-wire serial I/O mode	√	√
2-wire serial I/O mode	√	√
SBI (serial bus interface) mode	√	√
I^2C (Inter IC) bus mode	—	√

√ : available

— : not available

Legend

Data representation weight : High digits on the left and low digits on the right

Active low representations : $\overline{\times\times\times}$ (top bar over pin or signal name)

Note : Description of “Note” in the text

Caution : Information requiring particular attention

Remark : Additionally explanatory material

Numeral representations : Binary .... ×××× or ××××B

Decimal .... ××××

Hexadecimal ..... ××××H

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 Number	
		Japanese Version	English Version
μPD78014, 78014Y Subseries User’s Manual		U10085J	This manual
μPD78011B, 78012B, 78013, 78014 Data Sheet		IC-8201	IC-3179
μPD78P014 Data Sheet		IC-8111	IC-3098
μPD78011B(A), 78012B(A), 78013(A), 78014(A) Data Sheet		IC-8874	IC-3411
μPD78011BY, 78012BY, 78013Y, 78014Y Data Sheet		IC-8573	IC-3405
μPD78P014Y Data Sheet		IC-8572	IC-3180
μPD78014, 78014Y Series Special Function Register Table		IEM-5527	—
78K/0 Series User’s Manual — Instruction		U12326J	IEU-1372
78K/0 Series Instruction Set		U10903J	—
78K/0 Series Instruction Table		U10904J	—
μPD78014 Series Application Note		IEA-744	IEA-1301
78K/0 Series Application Note	Basic (I)	IEA-715	IEA-1288
	Floating Point Operation Program	IEA-718	IEA-1289

• **Document related to development tools (User's Manual)**

Document name		Document Number	
		Japanese Version	English Version
RA78K Series Assembler Package	Operation	EEU-809	EEU-1399
	Language	EEU-815	EEU-1404
RA78K Series Structured Assembler Preprocessor		U12323J	EEU-1402
★ RA78K0 Assembler Package	Operation	U11802J	U11802E
★	Assembly Language	U11801J	U11801E
★	Structured Assembly Language	U11789J	U11789E
CC78K Series C Compiler	Operation	EEU-656	EEU-1280
	Language	EEU-655	EEU-1284
★ CC78K0 C Compiler	Operation	U11517J	U11517E
★	Language	U11518J	U11518E
CC78K/0 C Compiler Application Note	Programming know-how	EEA-618	EEA-1208
CC78K Series Library Source File		U12322J	—
PG-1500 PROM Programmer		U11940J	EEU-1335
PG-1500 Controller PC-9800 Series (MS-DOS™ based)		EEU-704	EEU-1291
PG-1500 Controller IBM PC Series (PC DOS™) based)		EEU-5008	U10540E
IE-78000-R		EEU-810	U11376E
IE-78000-R-A		U11376J	U10057E
IE-78000-R-BK		EEU-867	EEU-1427
IE-78014-R-EM		EEU-805	EEU-1400
IE-78014-R-EM-A		EEU-962	EEU-1487
EP-78240		EEU-986	U10332E
SM78K0 System Simulator Windows™	Reference	U10181J	U10181E
SM78 Series System Simulator	External Part User Open Interface Specification	U10092J	U10092E
ID79K0 Integrated Debugger EWS based	Reference	U11151J	—
★ ID78K0 Integrated Debugger PC based	Reference	U11539J	U11539E
★ ID78K0 Integrated Debugger Windows based	Guides	U11649J	U11649E
SD78K/0 Screen Debugger	Basic	EEU-852	—
PC-9800 Series (MS-DOS) based	Reference	U10952J	—
SD78K/0 Screen Debugger	Basic	EEU-5024	U10539E
IBM PC/AT™ (PC DOS) based	Reference	U11279J	U11279E

• **Documents related to embedded software (User's Manual)**

Document Name		Document Number	
		Japanese Version	English Version
78K/0 Series Real-Time OS	Basic	U11537J	U11537E
	Install	U11536J	U11536E
78K/0 Series OS MX78K0	Basic	U12257J	—
Fuzzy Knowledge Data Creation Tool		EEU-829	EEU-1438
78K/0, 78K/II, 87AD Series Fuzzy Inference Development Support System (Translator)		EEU-862	EEU-1444
78K/0 Series Fuzzy Inference Development Support System (Fuzzy Inference Module)		EEU-858	EEU-1441
78K/0 Series Fuzzy Inference Development Support System (Fuzzy Inference Debugger)		EEU-921	EEU-1458

- **Other related documents**

Document Name	Document Number	
	Japanese Version	English Version
IC Package Manual	C10943X	
Semiconductor Device Mounting Technology Manual	C10535J	C10535E
Quality Grade of NEC Semiconductor Devices	C11531J	C11531E
Reliability and Quality Control of NEC Semiconductor Devices	C10983J	C10983E
Electrostatic Discharge (ESD) Test	MEM-539	—
Guide to Quality Assurance of Semiconductor Devices	C11893J	MEI-1202
Guide to Microcontroller-Related Products - Other Manufacturers	U11416J	—

Caution The contents of the above documents are subject to change without notice. Be sure to use the latest edition for designing.

CONTENTS

[MEMO]

LIST OF FIGURES (1/7)

Figure No.	Title, Page

LIST OF FIGURES (2/7)

Figure No. Title, Page

LIST OF FIGURES (3/7)

Figure No. Title, Page

LIST OF FIGURES (4/7)

Figure No. Title, Page

Figure No. Title, Page

LIST OF FIGURES (6/7)

Figure No. Title, Page

LIST OF FIGURES (7/7)

Figure No. Title, Page

[MEMO]

LIST OF TABLES (1/3)

Table No. Title, Page

LIST OF TABLES (2/3)

LIST OF TABLES (3/3)

[MEMO]

CHAPTER 1 OUTLINE (μPD78014 Subseries)

1.1 Features

- On-chip large-capacity ROM and RAM

Item / Part Number	Program Memory (ROM)	Data Memory	
		Internal high-speed RAM	Buffer RAM
μPD78011B	8 Kbytes	512 bytes	32 bytes
μPD78012B	16 Kbytes		
μPD78013	24 Kbytes	1024 bytes	
μPD78014	32 Kbytes		
μPD78P014	32 Kbytes[Note 1]	1024 bytes[Note 2]	

Notes 1. 8, 16, 24, or 32 Kbytes can be selected with the memory size switching register (IMS).
2. 512 or 1024 bytes can be selected with IMS.

- External memory expanded space: 64 Kbytes
- Minimum instruction execution time changeable from high speed (0.4 μs: @ 10.0 MHz with main system clock) to ultra-low speed (122 μs: @ 32.768 KHz with subsystem clock)
- Instruction set suitable for system control
  - Bit manipulation can be enabled in all the address space
  - Multiplication/division instruction

- 53 I/O ports (N-ch open-drain: 4)
- 8-bit resolution A/D converter: 8 channels
  - Low-voltage operation (AV_{DD} = 2.7 to 6.0 V: operable at the same voltage range as CPU)
- Serial interface: 2 channels
  - 3-wire serial I/O, SBI, 2-wire serial I/O mode: 1 channel
  - 3-wire serial I/O mode (Automatic transmit/receive function): 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
- 14 vectored interrupt sources
- 2 test inputs
- 2 types of on-chip clock oscillator (main system clock and subsystem clock)
- Power supply voltage: 2.7 to 6.0 V

1.2 Application Fields

For the μPD78011B, 78012B, 78013, 78014, and 78P014
Telephone, VCR, audio system, camera, home electric appliances, etc.

For the μPD78011B(A), 78012B(A), 78013(A), and 78014(A)
Automobile electronic equipment, gas detection breaker, safety equipment, etc.

1.3 Ordering Information

Part Number	Package	Internal ROM
μPD78011BCW-×××	64-pin plastic shrink DIP (750 mils)	Mask ROM
μPD78011BGC-×××-AB8	64-pin plastic QFP (14 × 14 mm)	Mask ROM
μPD78012BCW-×××	64-pin plastic shrink DIP (750 mils)	Mask ROM
μPD78012BGC-×××-AB8	64-pin plastic QFP (14 × 14 mm)	Mask ROM
μPD78013CW-×××	64-pin plastic shrink DIP (750 mils)	Mask ROM
μPD78013GC-×××-AB8	64-pin plastic QFP (14 × 14 mm)	Mask ROM
μPD78014CW-×××	64-pin plastic shrink DIP (750 mils)	Mask ROM
μPD78014GC-×××-AB8	64-pin plastic QFP (14 × 14 mm)	Mask ROM
μPD78P014CW	64-pin plastic shrink DIP (750 mils)	One-time PROM
μPD78P014DW	64-pin ceramic shrink DIP with window (750 mils)	EPROM
μPD78P014GC-AB8	64-pin plastic QFP (14 × 14 mm)	One-time PROM
μPD78011BCW(A)-×××	64-pin plastic shrink DIP (750 mils)	Mask ROM
μPD78011BGC-×××-AB8	64-pin plastic QFP (14 × 14 mm)	Mask ROM
μPD78012BCW(A)-×××	64-pin plastic shrink DIP (750 mils)	Mask ROM
μPD78012BGC(A)-×××-AB8	64-pin plastic QFP (14 × 14 mm)	Mask ROM
μPD78013CW(A)-×××	64-pin plastic shrink DIP (750 mils)	Mask ROM
μPD78013GC(A)-×××-AB8	64-pin plastic QFP (14 × 14 mm)	Mask ROM
μPD78014CW(A)-×××	64-pin plastic shrink DIP (750 mils)	Mask ROM
μPD78014GC(A)-×××-AB8	64-pin plastic QFP (14 × 14 mm)	Mask ROM

Remark ××× is ROM code suffix.

1.4 Quality Grade

Part Number	Package	Quality Grade
μPD78011BCW-×××	64-pin plastic shrink DIP (750 mils)	Standard
μPD78011BGC-×××-AB8	64-pin plastic QFP (14 × 14 mm)	Standard
μPD78012BCW-×××	64-pin plastic shrink DIP (750 mils)	Standard
μPD78012BGC-×××-AB8	64-pin plastic QFP (14 × 14 mm)	Standard
μPD78013CW-×××	64-pin plastic shrink DIP (750 mils)	Standard
μPD78013GC-×××-AB8	64-pin plastic QFP (14 × 14 mm)	Standard
μPD78014CW-×××	64-pin plastic shrink DIP (750 mils)	Standard
μPD78014GC-×××-AB8	64-pin plastic QFP (14 × 14 mm)	Standard
μPD78P014CW	64-pin plastic shrink DIP (750 mils)	Standard
★ μPD78P014DW	64-pin ceramic shrink DIP with window (750 mils)	Not applicable (for function evaluation only)
μPD78P014GC-AB8	64-pin plastic QFP (14 × 14 mm)	Standard
μPD78011BCW(A)-×××	64-pin plastic shrink DIP (750 mils)	Special
μPD78011BGC-×××-AB8	64-pin plastic QFP (14 × 14 mm)	Special
μPD78012BCW(A)-×××	64-pin plastic shrink DIP (750 mils)	Special
μPD78012BGC(A)-×××-AB8	64-pin plastic QFP (14 × 14 mm)	Special
μPD78013CW(A)-×××	64-pin plastic shrink DIP (750 mils)	Special
μPD78013GC(A)-×××-AB8	64-pin plastic QFP (14 × 14 mm)	Special
μPD78014CW(A)-×××	64-pin plastic shrink DIP (750 mils)	Special
μPD78014GC(A)-×××-AB8	64-pin plastic QFP (14 × 14 mm)	Special

Caution Of the above members, the μPD78P014DW should be used only for experiment or function evaluation, because it is not intended for use in equipment that will be mass-produced and require high reliability.

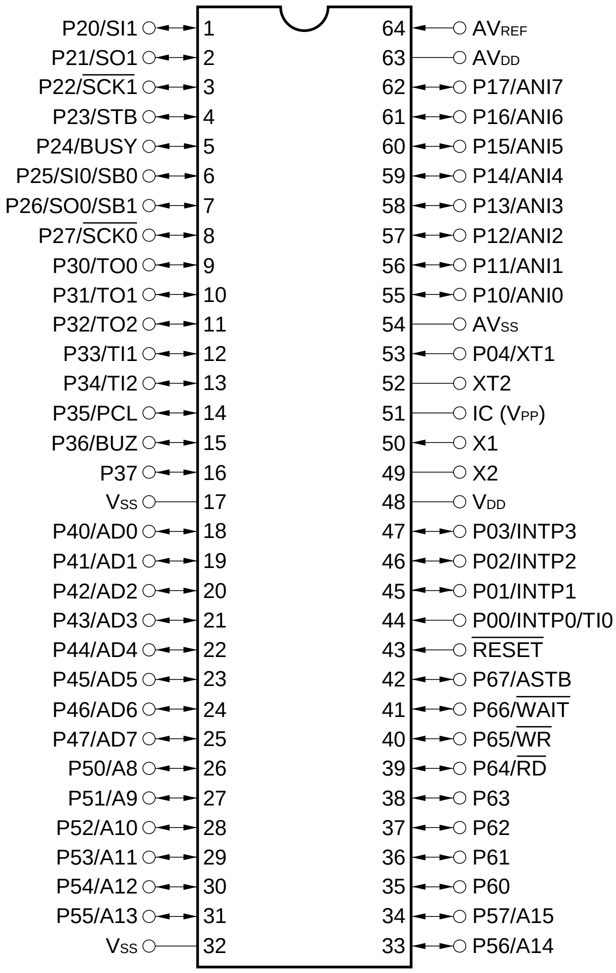
Remark ××× is the ROM code suffix.

Please refer to the **Quality grade on NEC Semiconductor Devices** (C11531E) published by NEC Corporation to know the specification of quality grade on the devices and its recommended applications.

1.5 Pin Configurations (Top View)

(1) Normal operating mode

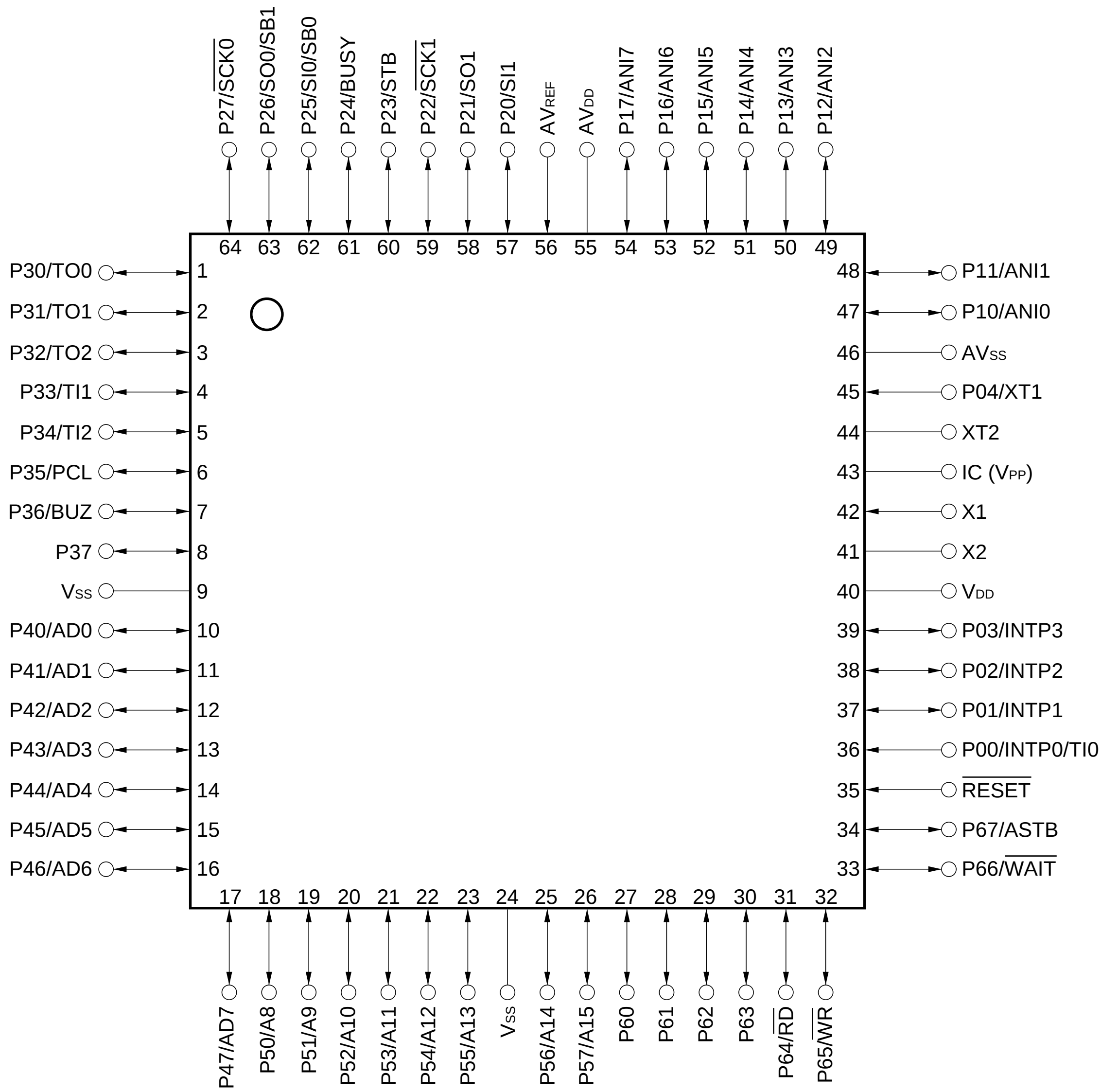
- **64-pin plastic shrink DIP (750 mils)**
  μPD78011BCW-×××, 78012BCW-×××
  μPD78013CW-×××, 78014CW-×××, 78P014CW
  μPD78011BCW(A)-×××, 78012BCW(A)-×××
  μPD78013CW(A)-×××, 78014CW(A)-×××
- **64-pin ceramic shrink DIP with window (750 mils)**
  μPD78P014DW



Cautions 1. Connect IC (Internally Connected) pin directly to V_{SS}.
2. Connect AV_{DD} pin to V_{DD}.
3. Connect AV_{SS} pin to V_{SS}.

Remark Pin connection in parentheses is intended for the μPD78P014.

- **64-pin plastic QFP (14 × 14 mm)**
  μPD78011BGC-×××-AB8, 78012BGC-×××-AB8
  μPD78013GC-×××-AB8, 78014GC-×××-AB8, 78P014GC-AB8
  μPD78011BGC(A)-×××-AB8, 78012BGC(A)-×××-AB8
  μPD78013GC(A)-×××-AB8, 78014GC(A)-×××-AB8



Cautions 1. Connect IC (Internally Connected) pin directly to V_{SS}.

2. Connect AV_{DD} pin to V_{DD}.

3. Connect AV_{SS} pin to V_{SS}.

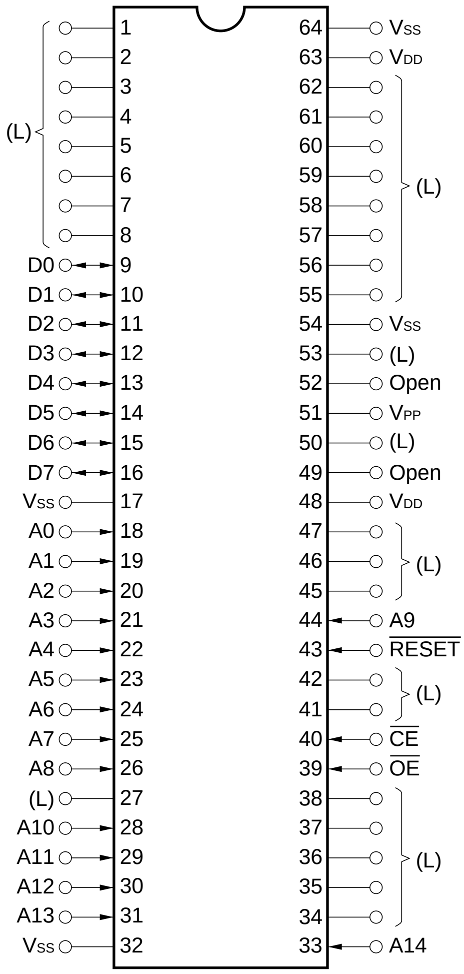
Remark Pin connection in parentheses is intended for the μPD78P014.

A8 to A15	:	Address Bus
AD0 to AD7	:	Address/Data Bus
ANI0 to ANI7	:	Analog Input
ASTB	:	Address Strobe
AV_{DD}	:	Analog Power Supply
AV_{REF}	:	Analog Reference Voltage
AV_{SS}	:	Analog Ground
BUSY	:	Busy
BUZ	:	Buzzer Clock
IC	:	Internally Connected
INTP0 to INTP3	:	Interrupt from Peripherals
P00 to P04	:	Port0
P10 to P17	:	Port1
P20 to P27	:	Port2
P30 to P37	:	Port3
P40 to P47	:	Port4
P50 to P57	:	Port5
P60 to P67	:	Port6
PCL	:	Programmable Clock
$\overline{RD}$	:	Read Strobe
$\overline{RESET}$	:	Reset
SB0, SB1	:	Serial Bus
$\overline{SCK0}$, $\overline{SCK1}$	:	Serial Clock
SI0, SI1	:	Serial Input
SO0, SO1	:	Serial Output
STB	:	Strobe
TI0 to TI2	:	Timer Input
TO0 to TO2	:	Timer Output
V_{DD}	:	Power Supply
V_{PP}	:	Programming Power Supply
V_{SS}	:	Ground
$\overline{WAIT}$	:	Wait
$\overline{WR}$	:	Write Strobe
X1, X2	:	Crystal (Main System Clock)
XT1, XT2	:	Crystal (Subsystem Clock)

Remark V_{PP} is intended for the μPD78P014. For Mask ROM versions, IC is applied.

(2) PROM programming mode

- **64-pin plastic shrink DIP (750 mils)**
  μPD78P014CW
- **64-pin ceramic shrink DIP with window (750 mils)**
  μPD78P014DW



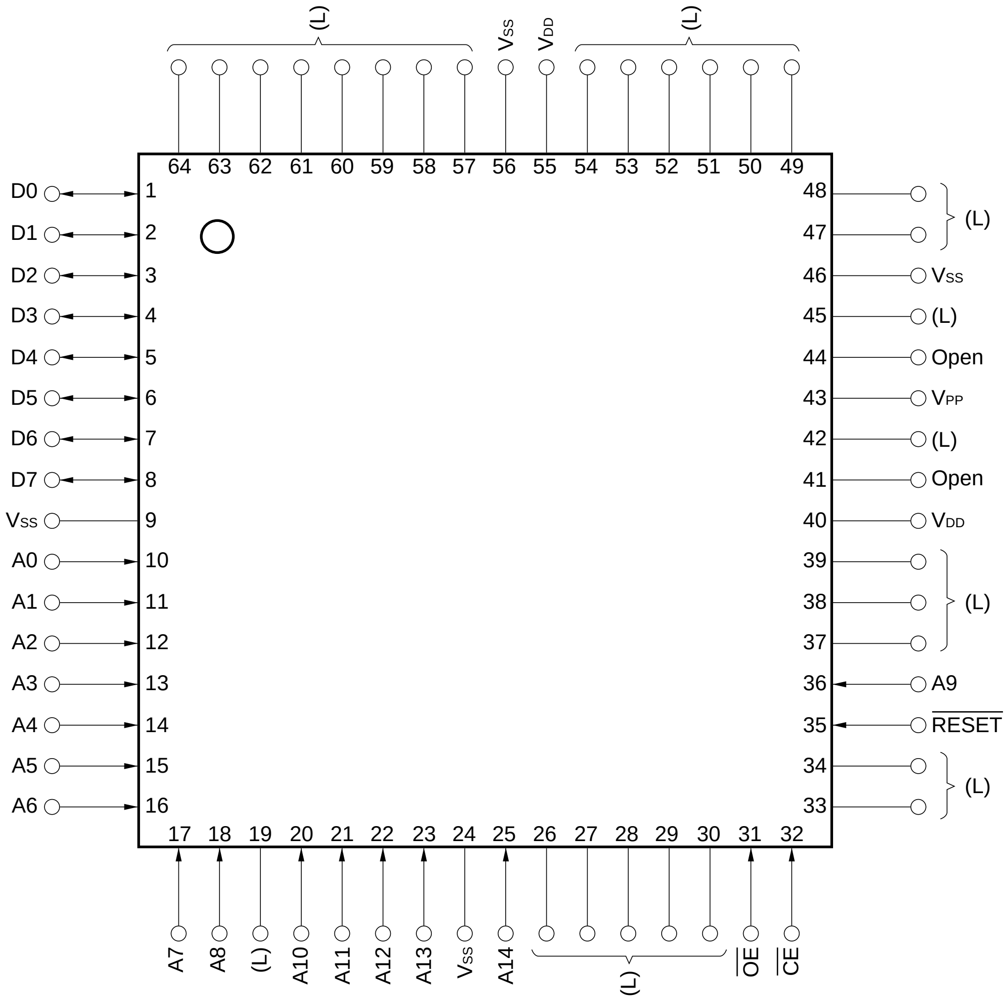
Cautions 1. (L) : Connect individually to V_{SS} via a pull-down resistor.

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

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

4. Open : No connection required.

- **64-pin plastic QFP (14 × 14 mm)**
  μPD78P014GC-AB8

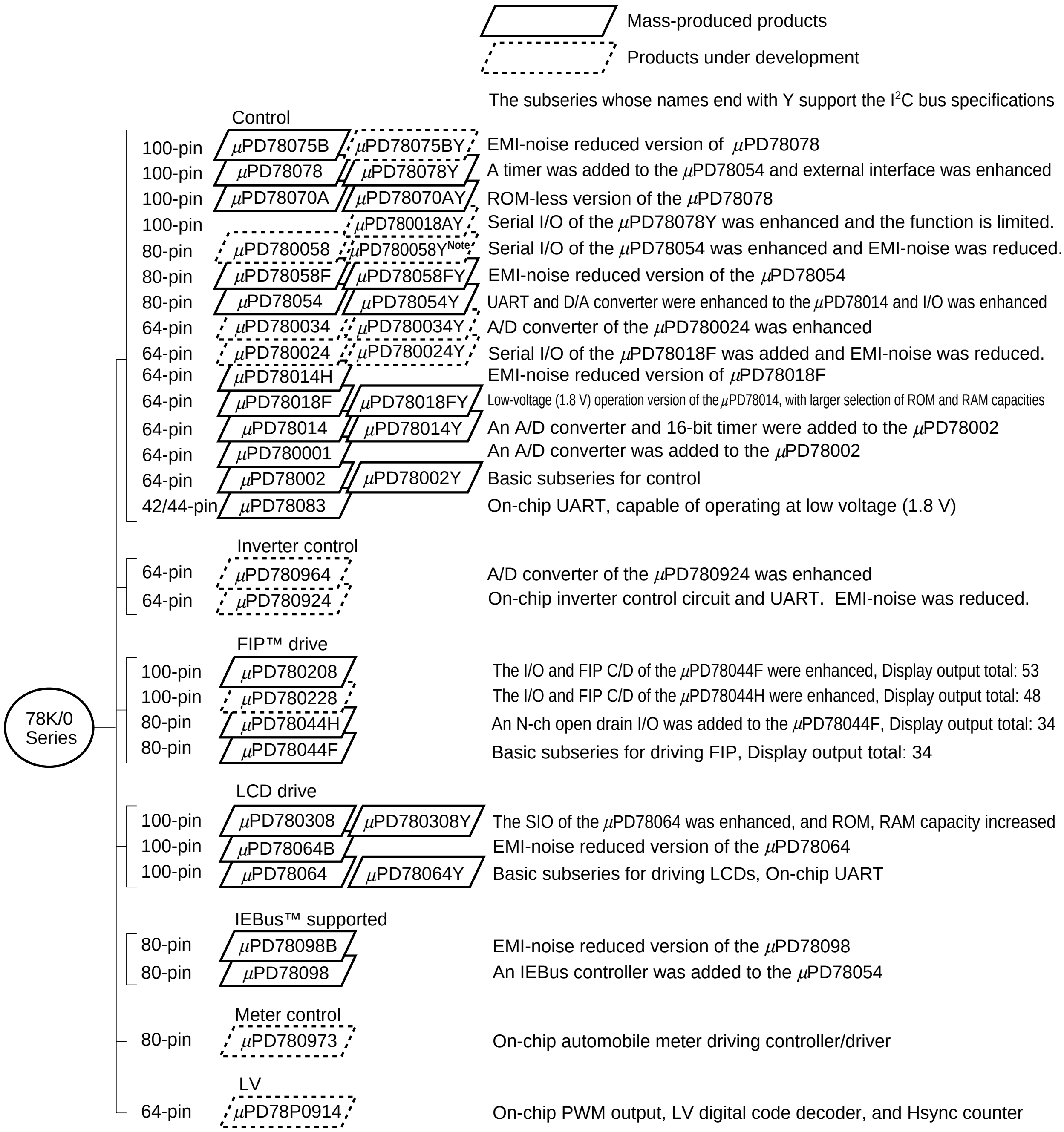


Cautions 1. (L) : Connect individually to V_{SS} via a pull-down resistor.
2. V_{SS} : Connect to the ground.
3. RESET : Set to the low level.
4. Open : No connection required.

A0 to A14	:	Address Bus	RESET	:	Reset
CE	:	Chip Enable	V_{DD}	:	Power Supply
D0 to D7	:	Data Bus	V_{PP}	:	Programming Power Supply
OE	:	Output Enable	V_{SS}	:	Ground

★ ## 1.6 78K/0 Series Expansion

The following shows the 78K/0 Series products development. Subseries names are shown inside frames.



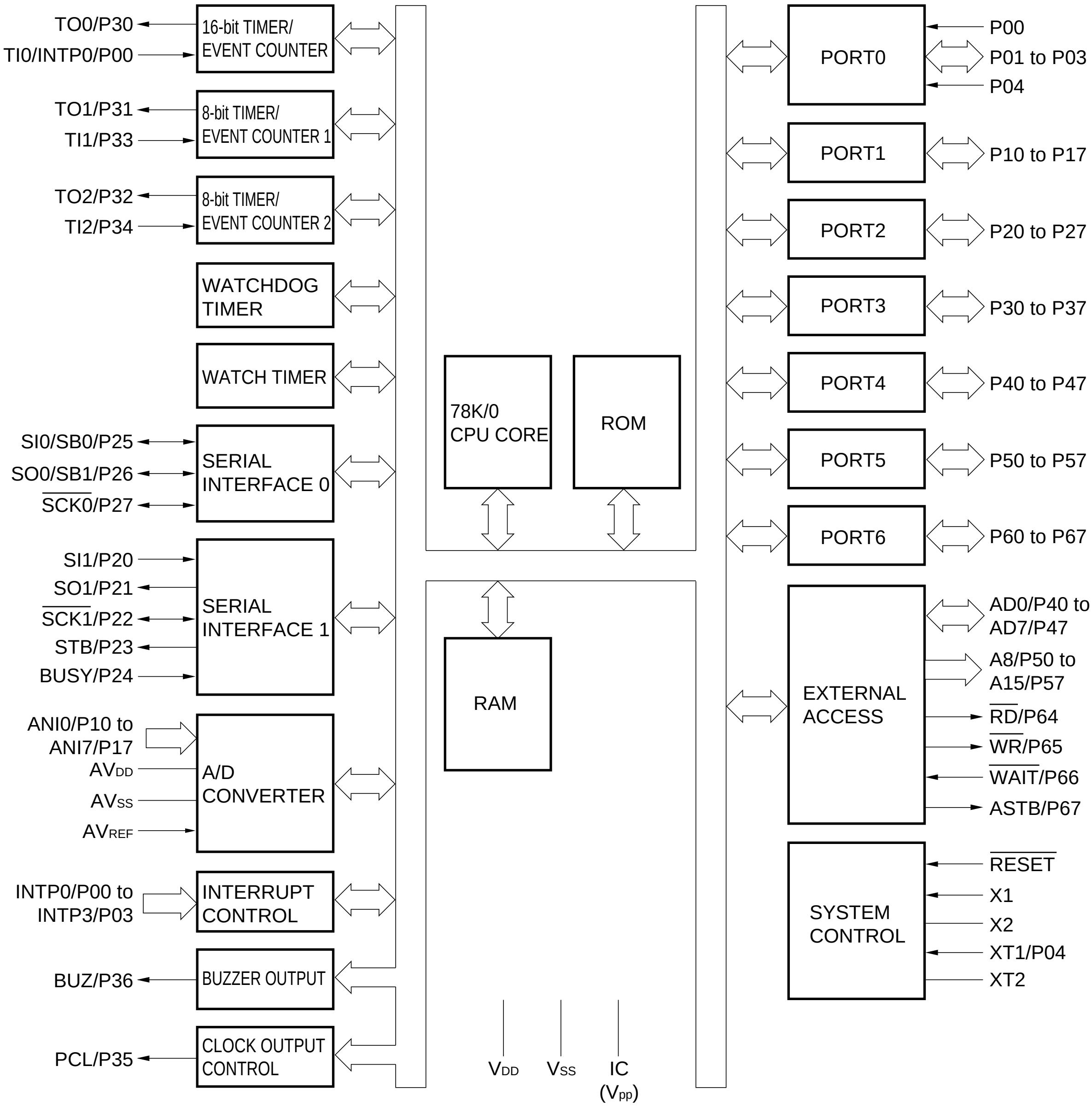
Note Under planning

The following table shows the differences among subseries functions.

Part Number		ROM Capacity	Timer 8-bit	Timer 16-bit	Timer Watch	Timer WDT	8-bit A/D	10-bit A/D	8-bit D/A	Serial Interface	I/O	V_{DD} MIN. Value	External Expansion
Control	μPD78075B	32K to 40K	4ch	1ch	1ch	1ch	8ch	—	2ch	3ch (UART: 1ch)	88	1.8 V	○
	μPD78078	48K to 60K											
	μPD78070A	—									61	2.7 V	
	μPD780058	24K to 60K	2ch						2ch	3ch (time division UART: 1ch)	68	1.8 V	
	μPD78058F	48K to 60K								3ch (UART: 1ch)	69	2.7 V	
	μPD78054	16K to 60K										2.0 V	
	μPD780034	8K to 32K					—	8ch	—	3ch (UART: 1ch, time division 3-wire: 1ch)	51	1.8 V	
	μPD780024						8ch	—					
	μPD78014H									2ch	53	1.8 V	
	μPD78018F	8K to 60K											
	μPD78014	8K to 32K										2.7 V	
	μPD780001	8K		—	—					1ch	39		—
	μPD78002	8K to 16K			1ch		—				53		○
	μPD78083				—		8ch			1ch (UART: 1ch)	33	1.8 V	—
Inverter control	μPD780964	8K to 32K	3ch	**Note**	—	1ch	—	8ch	—	2ch (UART: 2ch)	47	2.7 V	○
	μPD780924						8ch	—					
FIP drive	μPD780208	32K to 60K	2ch	1ch	1ch	1ch	8ch	—	—	2ch	74	2.7 V	—
	μPD780228	48K to 60K	3ch	—	—					1ch	72	4.5 V	
	μPD78044H	32K to 48K	2ch	1ch	1ch						68	2.7 V	
	μPD78044F	16K to 40K								2ch			
LCD drive	μPD780308B	48K to 60K	2ch	1ch	1ch	1ch	8ch	—	—	3ch (time division UART: 1ch)	57	2.0 V	—
	μPD78064B	32K								2ch (UART: 1ch)			
	μPD78064	16K to 32K											
IEBus supported	μPD78098B	40K to 60K	2ch	1ch	1ch	1ch	8ch	—	2ch	3ch (UART: 1ch)	69	2.7 V	○
	μPD78098	32K to 60K											
Meter control	μPD780973	24K to 32K	3ch	1ch	1ch	1ch	5ch	—	—	2ch (UART: 1ch)	56	4.5 V	—
LV	μPD78P0914	32K	6ch	—	—	1ch	8ch	—	—	2ch	54	4.5 V	○

Note 10-bit timer: 1 channel

1.7 Block Diagram



Remarks 1. The internal ROM and RAM capacities depend on the product.

2. Pin connection in parentheses is intended for the μPD78P014.

1.8 Outline of Function

Item		μPD78011B	μPD78012B	μPD78013	μPD78014	μPD78P014
Internal memory	ROM	Mask ROM				One-time PROM/EPROM
		8 Kbytes	16 Kbytes	24 Kbytes	32 Kbytes	32 Kbytes[Note 1]
	High-speed RAM	512 bytes		1024 bytes		1024 bytes[Note 2]
	Buffer RAM	32 bytes				
Memory space		64 Kbytes				
General registers		8 bits × 8 registers × 4 banks				
Minimum instruction execution time	When main system clock selected	0.4 μs/0.8 μs/1.6 μs/3.2 μs/6.4 μs (@ 10.0 MHz)				
	When subsystem clock selected	122 μs (@ 32.768 kHz)				
Instruction set		• 16-bit operation • Multiply/divide (8 bits × 8 bits, 16 bits ÷ 8 bits) • Bit manipulation (set, reset, test, and Boolean operation) • BCD adjust, and other related operations				
I/O ports		Total : 53 I/O port pins • CMOS input : 2 inputs • CMOS I/O : 47 inputs/outputs (on-chip pull-up resistor can be turned on/off by software.) • N-ch open-drain I/O : 4 inputs/outputs (15-V withstand, on-chip pull-up resistor with mask options in mask ROM versions only)				
A/D converter		• 8-bit resolution × 8 channels • Low-voltage operation: AV_{DD} = 2.7 to 6.0 V				
Serial interface		• 3-wire serial I/O, SBI, 2-wire serial I/O mode selectable: 1 channel • 3-wire serial I/O mode (Maximum 32-byte on-chip automatic transmit/receive function): 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				

Notes 1. 8, 16, 24, or 32 Kbytes can be selected by memory size switching register (IMS).

2. 512 or 1024 bytes can be selected by IMS.

Item \ Part Number		μPD78011B	μPD78012B	μPD78013	μPD78014	μPD78P014
Timer output		3 outputs: (14-bit PWM generation possible from one output)				
Clock output		39.1 kHz, 78.1 kHz, 156 kHz, 313 kHz, 625 kHz, 1.25 MHz (@ 10.0 MHz with main system clock) 32.768 kHz (@ 32.768 kHz with subsystem clock)				
Buzzer output		2.4 kHz, 4.9 kHz, 9.8 kHz (@ 10.0 MHz with main system clock)				
Vectored interrupt sources	Maskable	Internal : 8, external : 4				
	Non-maskable	Internal : 1				
	Software	1				
Test input		Internal : 1, external : 1				
Power supply voltage		V_{DD} = 2.7 to 6.0 V				
Operating ambient temperature		T_A = –40 to +85°C				
Package		• 64-pin plastic shrink DIP (750 mils) • 64-pin plastic QFP (14 × 14 mm) • 64-pin ceramic shrink DIP with window (750 mils): μPD78P014 only				

1.9 Differences among μPD78011B, 78012B, 78013, 78014 and μPD78011B(A), 78012B(A), 78013(A), 78014(A)

Table 1-1. Differences among μPD78011B, 78012B, 78013, 78014 and μPD78011B(A), 78012B(A), 78013(A), 78014(A)

Item \ Part Number	μPD78011B, 78012B, 78013, 78014	μPD78011B(A), 78012B(A), 78013(A), 78014(A)
Quality grade	Standard	Special

1.10 Mask Options

The mask ROM versions (μPD78011B, μPD78012B, μPD78013, μPD78014) have the mask options. By specifying the mask options when ordering, the pull-up resistors and pull-down resistors listed in Table 1-2 can be incorporated. When these resistors are necessary, the number of external components and mounting space can be saved by utilizing the mask options.

Mask options provided in the μPD78014 Subseries are shown in Table 1-2.

Table 1-2. Mask Options in Mask ROM Versions

Pin Name	Mask Option
P60 to P63	Pull-down resistors can be incorporated bit-wise.

CHAPTER 2 OUTLINE (μPD78014Y Subseries)

2.1 Features

- On-chip large-capacity ROM and RAM

Item / Part Number	Program Memory (ROM)	Data Memory	
		Internal high-speed RAM	Buffer RAM
μPD78011BY	8 Kbytes	512 bytes	32 bytes
μPD78012BY	16 Kbytes		
μPD78013Y	24 Kbytes	1024 bytes	
μPD78014Y	32 Kbytes		
μPD78P014Y	32 Kbytes**Note 1**	1024 bytes**Note 2**	

Notes 1. 8, 16, 24, or 32 Kbytes can be selected by memory size switching register (IMS).

2. 512 or 1024 bytes can be selected by IMS.

- External memory expanded space: 64 Kbytes
- Minimum instruction execution time changeable from high speed (0.4 μs: @ 10 MHz with main system clock) to ultra-low speed (122 μs: @ 32.768 kHz with subsystem clock)
- Instruction set suitable for system control
  - Bit manipulation enable in all the address space
  - Multiplication/division instruction
- 53 I/O ports (N-ch open-drain: 4)
- 8-bit resolution A/D converter: 8 channels
  - Low-voltage operation (AV_{DD} = 2.7 to 6.0 V: operable at the same supply voltage range as CPU)
- Serial interface: 2 channels
  - 3-wire serial I/O, SBI, 2-wire serial I/O, I^2C bus mode: 1 channel
  - 3-wire serial I/O mode (Automatic transmit/receive function): 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
- 14 vectored interrupt sources
- 2 test inputs
- 2 types of on-chip clock oscillator (main system clock and subsystem clock)
- Power supply voltage: V_{DD} = 2.7 to 6.0 V

2.2 Application Fields

Telephone, VCR, audio system, camera, home electric appliances, etc.

2.3 Ordering Information

Part Number	Package	Internal ROM
μPD78011BYCW-×××	64-pin plastic shrink DIP (750 mils)	Mask ROM
μPD78011BYGC-×××-AB8	64-pin plastic QFP (14 × 14 mm)	Mask ROM
μPD78012BCW-×××	64-pin plastic shrink DIP (750 mils)	Mask ROM
μPD78012BYGC-×××-AB8	64-pin plastic QFP (14 × 14 mm)	Mask ROM
μPD78013YCW-×××	64-pin plastic shrink DIP (750 mils)	Mask ROM
μPD78013YGC-×××-AB8	64-pin plastic QFP (14 × 14 mm)	Mask ROM
μPD78014YCW-×××	64-pin plastic shrink DIP (750 mils)	Mask ROM
μPD78014YGC-×××-AB8	64-pin plastic QFP (14 × 14 mm)	Mask ROM
μPD78P014YCW	64-pin plastic shrink DIP (750 mils)	One-time PROM
μPD78P014YDW	64-pin ceramic shrink DIP with window (750 mils)	EPROM
μPD78P014YGC-AB8	64-pin plastic QFP (14 × 14 mm)	One-time PROM

Remark ××× is ROM code suffix.

2.4 Quality Grade

Part Number	Package	Quality Grade
μPD78011BYCW-×××	64-pin plastic shrink DIP (750 mils)	Standard
μPD78011BYGC-×××-AB8	64-pin plastic QFP (14 × 14 mm)	Standard
μPD78012BYCW-×××	64-pin plastic shrink DIP (750 mils)	Standard
μPD78012BYGC-×××-AB8	64-pin plastic QFP (14 × 14 mm)	Standard
μPD78013YCW-×××	64-pin plastic shrink DIP (750 mils)	Standard
μPD78013YGC-×××-AB8	64-pin plastic QFP (14 × 14 mm)	Standard
μPD78014YCW-×××	64-pin plastic shrink DIP (750 mils)	Standard
μPD78014YGC-×××-AB8	64-pin plastic QFP (14 × 14 mm)	Standard
μPD78P014YCW	64-pin plastic shrink DIP (750 mils)	Standard
★ μPD78P014YDW	64-pin ceramic shrink DIP with window (750 mils)	Not applicable (for function evaluation only)
μPD78P014YGC-AB8	64-pin plastic QFP (14 × 14 mm)	Standard

Caution Of the above members, the μPD78P014YDW should be used only for experiment or function evaluation, because it is not intended for use in equipment that will be mass-produced and require high reliability.

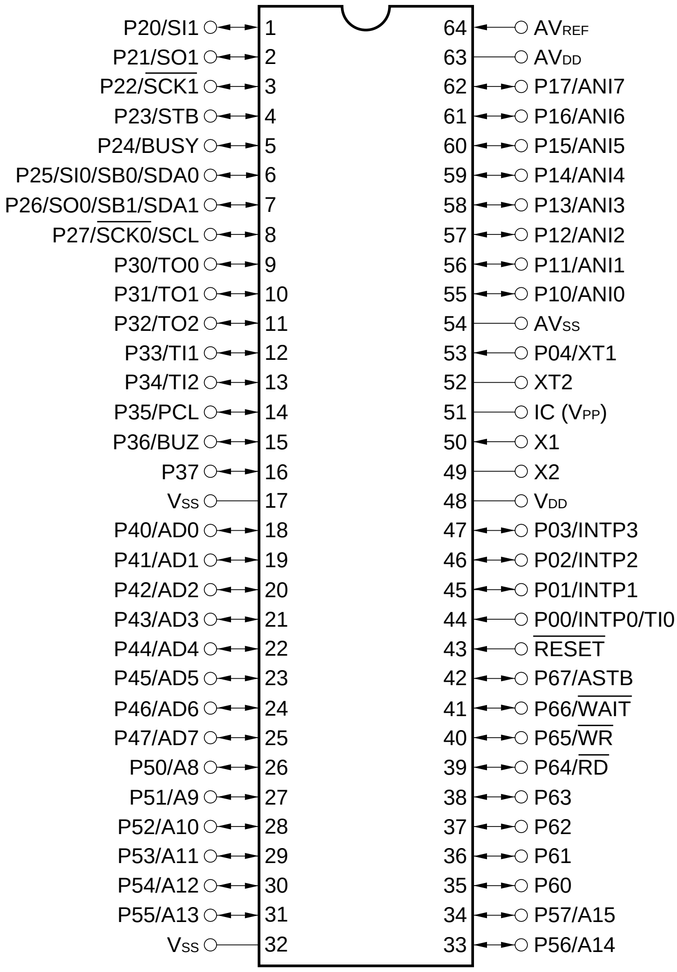
Remark ××× is the ROM code suffix.

Please refer to the **Quality grade on NEC Semiconductor Devices** (**C11531E**) published by NEC Corporation to know the specification of quality grade on the devices and its recommended applications.

2.5 Pin Configurations (Top View)

(1) Normal operating mode

- **64-pin plastic shrink DIP (750 mils)**
  μPD78011BYCW-×××, 78012BYCW-×××
  μPD78013YCW-×××, 78014YCW-×××, 78P014YCW
- **64-pin ceramic shrink DIP with window (750 mils)**
  μPD78P014YDW



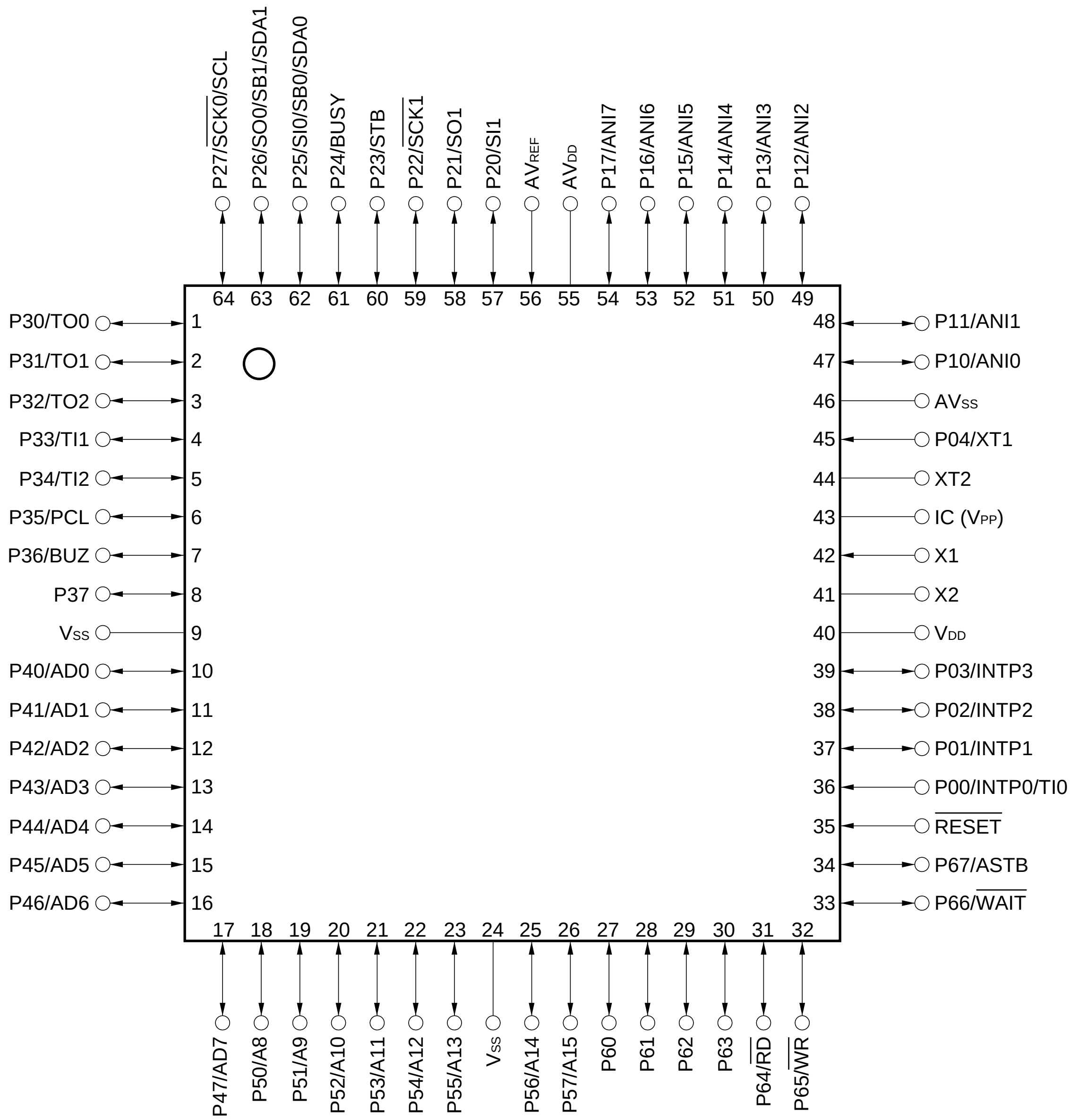
Cautions 1. Connect IC (Internally Connected) pin directly to V_{SS}.

2. Connect AV_{DD} pin to V_{DD}.

3. Connect AV_{SS} pin to V_{SS}.

Remark Pin connection in parentheses is intended for the μPD78P014Y.

- **64-pin plastic QFP (14 × 14 mm)**
  μPD78011BYGC-×××-AB8, 78012BYGC-×××-AB8
  μPD78013YGC-×××-AB8, 78014YGC-×××-AB8, 78P014YGC-AB8



Cautions 1. Connect IC (Internally Connected) pin directly to V_{SS}.

2. Connect AV_{DD} pin to V_{DD}.

3. Connect AV_{SS} pin to V_{SS}.

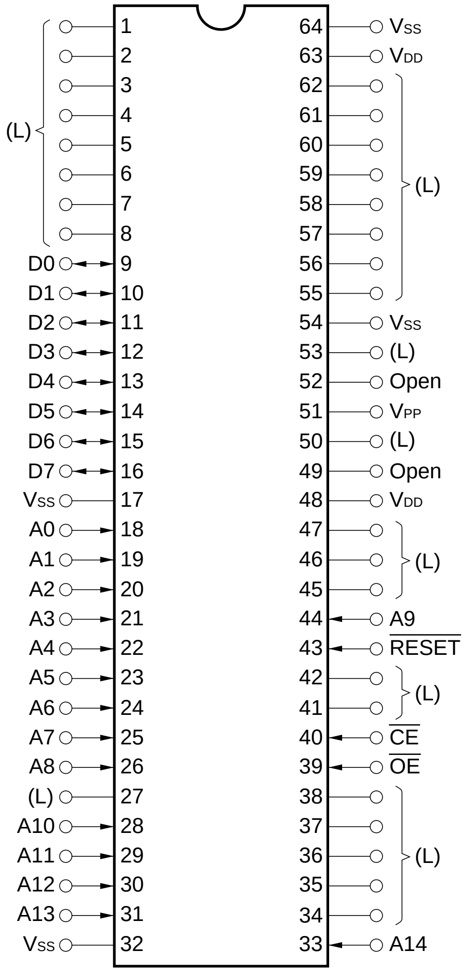
Remark Pin connection in parentheses is intended for the μPD78P014Y.

A8 to A15 : Address Bus
AD0 to AD7 : Address/Data Bus
ANI0 to ANI7 : Analog Input
ASTB : Address Strobe
AV_{DD} : Analog Power Supply
AV_{REF} : Analog Reference Voltage
AV_{SS} : Analog Ground
BUSY : Busy
BUZ : Buzzer Clock
IC : Internally Connected
INTP0 to INTP3 : Interrupt from Peripherals
P00 to P04 : Port0
P10 to P17 : Port1
P20 to P27 : Port2
P30 to P37 : Port3
P40 to P47 : Port4
P50 to P57 : Port5
P60 to P67 : Port6
PCL : Programmable Clock
$\overline{RD}$: Read Strobe
$\overline{RESET}$: Reset
SB0, SB1 : Serial Bus
$\overline{SCK0}$, $\overline{SCK1}$: Serial Clock
SCL : Serial Clock
SDA0, SDA1 : Serial Data
SI0, SI1 : Serial Input
SO0, SO1 : Serial Output
STB : Strobe
TI0 to TI2 : Timer Input
TO0 to TO2 : Timer Output
V_{DD} : Power Supply
V_{PP} : Programming Power Supply
V_{SS} : Ground
$\overline{WAIT}$: Wait
$\overline{WR}$: Write Strobe
X1, X2 : Crystal (Main System Clock)
XT1, XT2 : Crystal (Subsystem Clock)

Remark V_{PP} is intended for the μPD78P014Y. For Mask ROM versions, IC is applied.

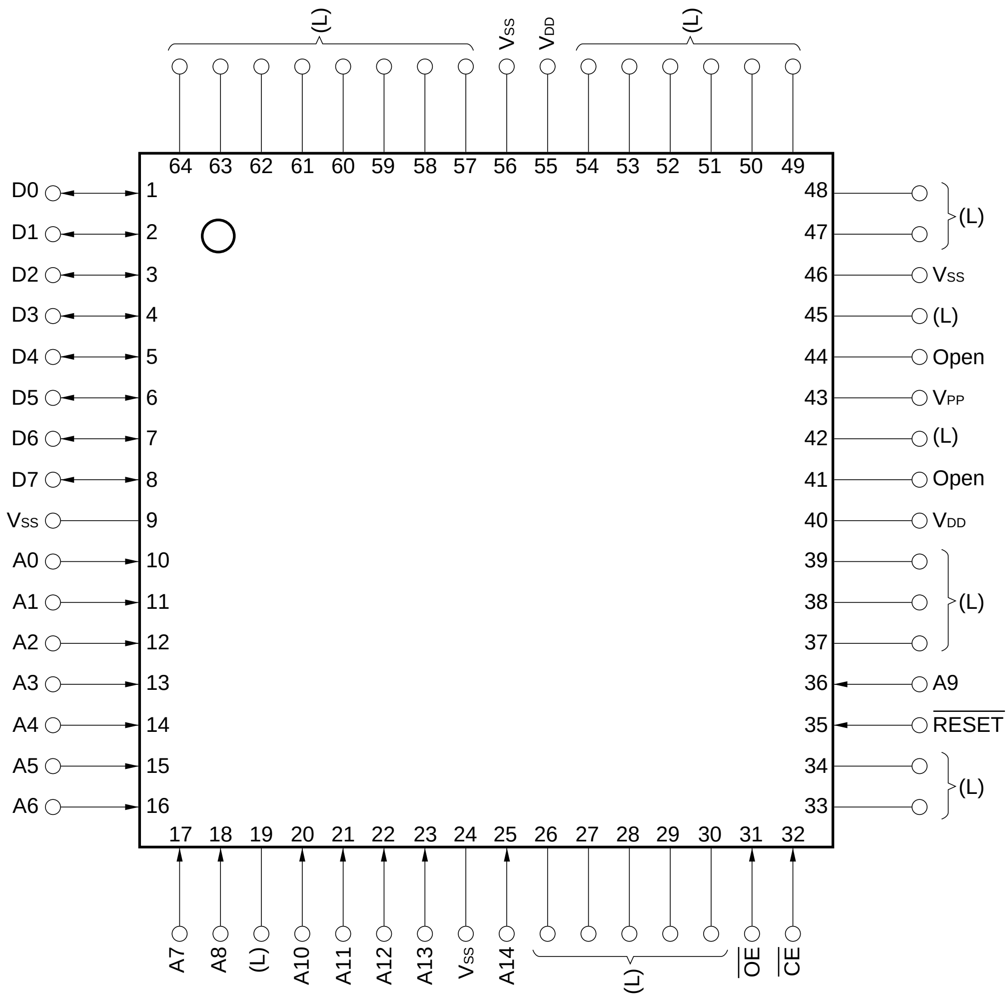
(2) PROM programming mode

- **64-pin plastic shrink DIP (750 mils)**
  μPD78P014YCW
- **64-pin ceramic shrink DIP with window (750 mils)**
  μPD78P014YDW



Cautions 1. (L) : Connect individually 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 : No connection required.

- **64-pin plastic QFP (14 × 14 mm)**
  μPD78P014YGC-AB8



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

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

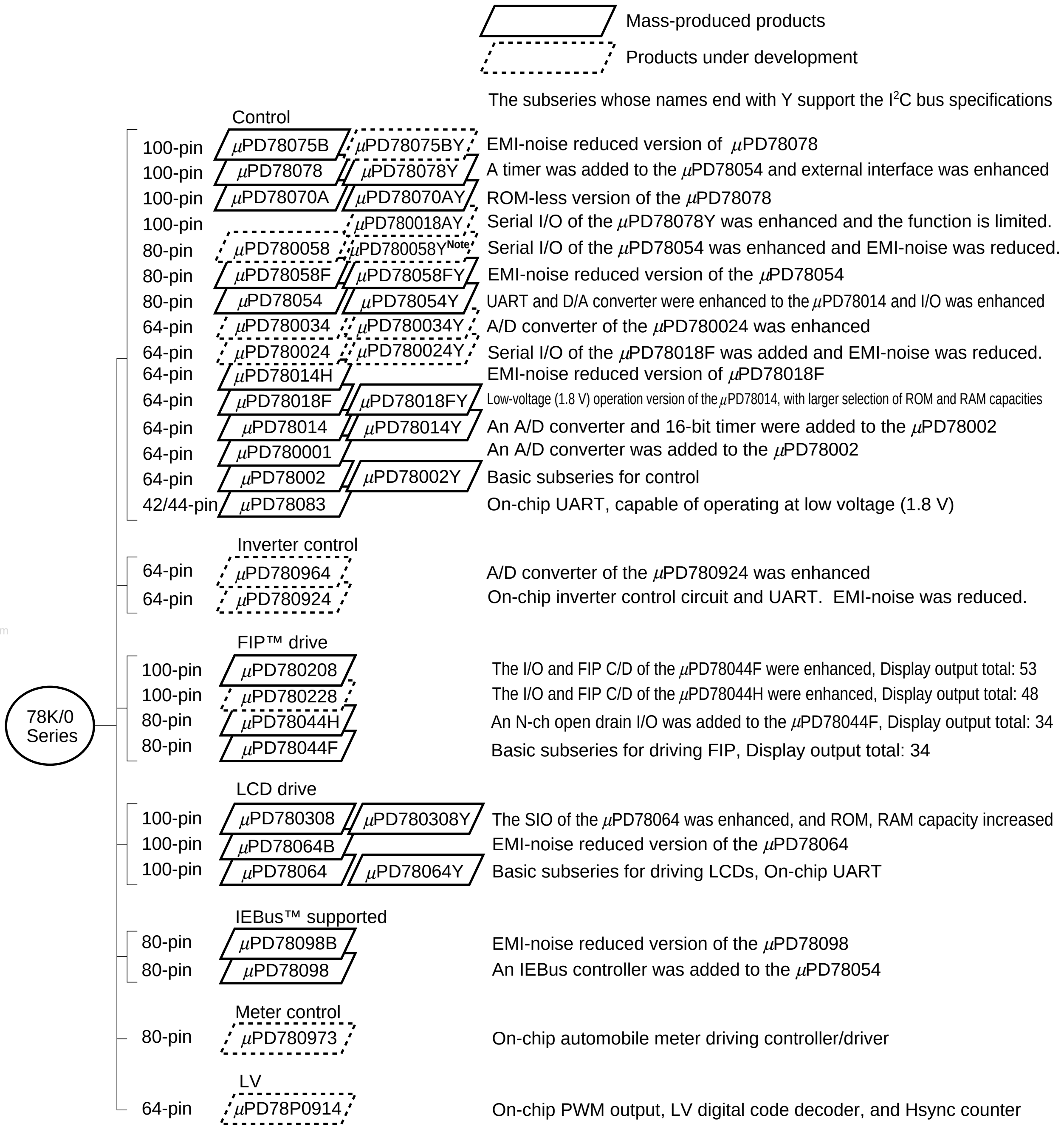
3. $\overline{RESET}$: Set to low level.

4. Open : No connection required.

A0 to A14	:	Address Bus	$\overline{RESET}$	:	Reset
$\overline{CE}$	:	Chip Enable	V_{DD}	:	Power Supply
D0 to D7	:	Data Bus	V_{PP}	:	Programming Power Supply
$\overline{OE}$	:	Output Enable	V_{SS}	:	Ground

★ 2.6 78K/0 Series Expansion

The following shows the 78K/0 Series products development. Subseries names are shown inside frames.



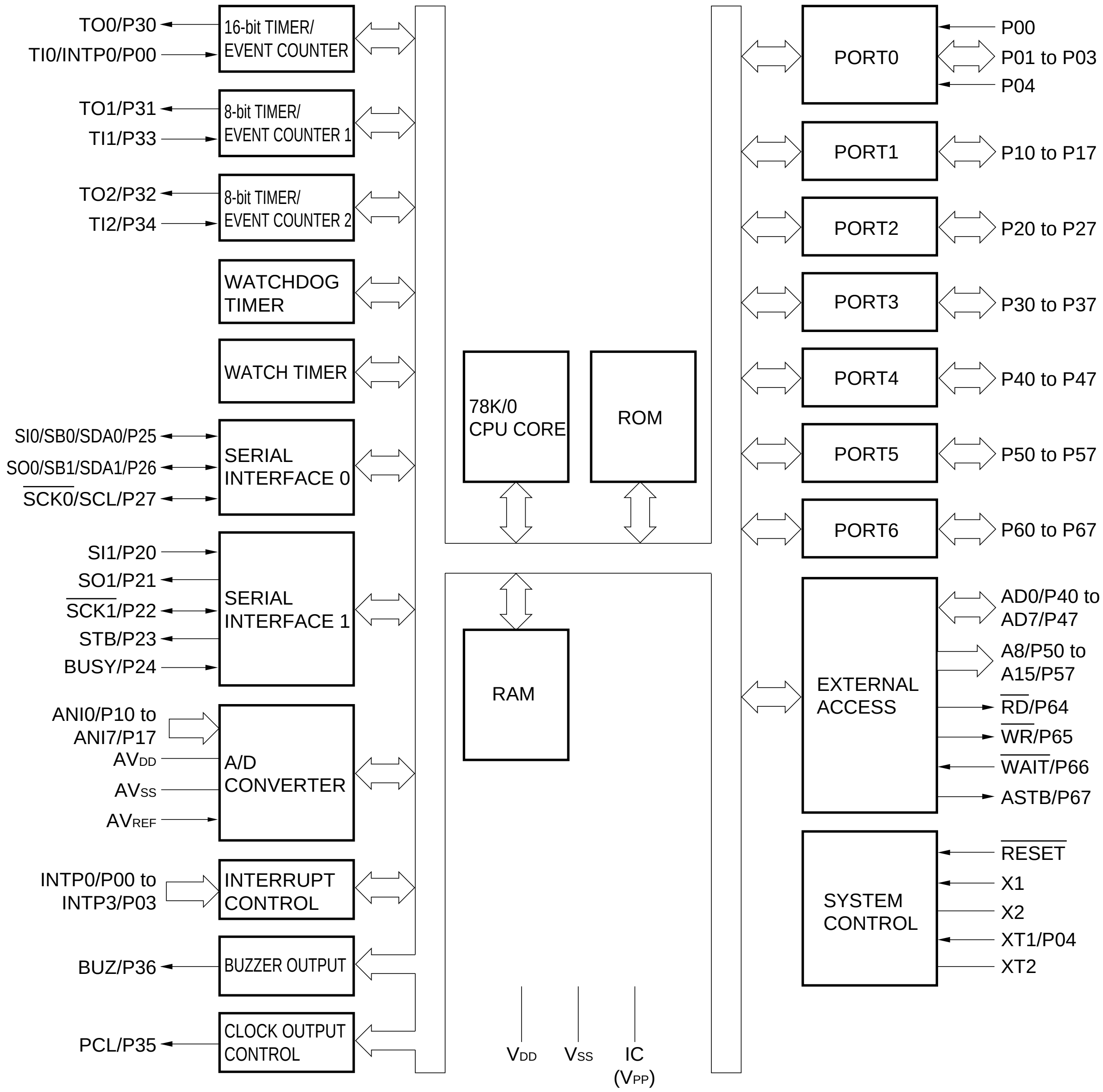
Note Under planning

The following table shows the differences among Y subseries functions.

Part number \ Function		ROM Capacity	Configuration of Serial Interface	I/O	V_{DD} MIN. Value
Control	μPD78075BY	32K to 40K	3-wire/2-wire/I^{2}C : 1ch	88	1.8 V
	μPD78078Y	48K to 60K	3-wire with automatic transmit/receive function : 1ch		
	μPD78070AY	—	3-wire/UART : 1ch	61	2.7 V
	μPD780018AY	48K to 60K	3-wire with automatic transmit/receive function : 1ch Time division 3-wire : 1ch I^{2}C bus (supports multi-master) : 1ch	88	
	μPD780058Y	24K to 60K	3-wire/2-wire/I^{2}C : 1ch 3-wire with automatic transmit/receive function : 1ch 3-wire/time division UART : 1ch	68	1.8 V
	μPD78058FY	48K to 60K	3-wire/2-wire/I^{2}C : 1ch 3-wire with automatic transmit/receive function : 1ch 3-wire/UART : 1ch	69	2.7 V
	μPD78054Y	16K to 60K			2.0 V
	μPD780034Y	8K to 32K	UART : 1ch 3-wire : 1ch I^{2}C bus (supports multi-master) : 1ch	51	1.8 V
	μPD780024Y				
	μPD78018FY	8K to 60K	3-wire/2-wire/I^{2}C : 1ch 3-wire with automatic transmit/receive function : 1ch	53	
	μPD78014Y	8K to 32K	3-wire/2-wire/SBI/I^{2}C : 1ch 3-wire with automatic transmit/receive function : 1ch		2.7 V
	μPD78002Y	8K to 16K	3-wire/2-wire/SBI/I^{2}C : 1ch		
LCD drive	μPD780308Y	48K to 60K	3-wire/2-wire/I^{2}C : 1ch 3-wire/time division UART : 1ch 3-wire : 1ch	57	2.0 V
	μPD78064Y	16K to 32K	3-wire/2-wire/I^{2}C : 1ch 3-wire/UART : 1ch		

Remark The functions except serial interface are common with subseries without Y.

2.7 Block Diagram



Remarks 1. The internal ROM and RAM capacities depend on the product.

2. Pin connection in parentheses is intended for the μPD78P014Y.

2.8 Outline of Function

Item / Part Number		μPD78011BY	μPD78012BY	μPD78013Y	μPD78014Y	μPD78P014Y
Internal memory	ROM	Mask ROM				One-time PROM/EPROM
		8 Kbytes	16 Kbytes	24 Kbytes	32 Kbytes	32 Kbytes[Note 1]
	High-speed RAM	512 bytes		1024 bytes		1024 bytes[Note 2]
	Buffer RAM	32 bytes				
Memory space		64 Kbytes				
General registers		8 bits × 8 registers × 4 banks				
Minimum instruction execution time	When main system clock selected	0.4 μs/0.8 μs/1.6 μs/3.2 μs/6.4 μs (@ 10.0 MHz)				
	When subsystem clock selected	122 μs (@ 32.768 kHz)				
Instruction set		• 16-bit operation • Multiply/divide (8 bits × 8 bits, 16 bits ÷ 8 bits) • Bit manipulation (set, reset, test, and Boolean operation) • BCD adjust, and other related operations				
I/O ports		Total : 53 I/O ports • CMOS input : 2 inputs • CMOS I/O : 47 inputs/outputs (on-chip pull-up resistor can be turn on/off by software.) • N-ch open-drain I/O : 4 inputs/outputs (15-V withstand, on-chip pull-up resistor with mask options in mask ROM versions only)				
A/D converter		• 8-bit resolution × 8 channels • Low-voltage operation: AV_{DD} = 2.7 to 6.0 V				
Serial interface		• 3-wire serial I/O, SBI, 2-wire serial I/O, I^2C bus mode selectable: 1 channel • 3-wire serial I/O mode (Maximum 32-byte on-chip automatic transmit/receive function): 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				

Notes 1. 8, 16, 24, or 32 Kbytes can be selected by memory size switching register (IMS).

2. 512 or 1024 bytes can be selected by IMS.

Item \ Part Number		μPD78011BY	μPD78012BY	μPD78013Y	μPD78014Y	μPD78P014Y
Timer output		3 outputs: (14-bit PWM generation possible from one output)				
Clock output		39.1 kHz, 78.1 kHz, 156 kHz, 313 kHz, 625 kHz, 1.25 MHz (@ 10.0 MHz with main system clock) 32.768 kHz (@ 32.768 kHz with subsystem clock)				
Buzzer output		2.4 kHz, 4.9 kHz, 9.8 kHz (@ 10.0 MHz with main system clock)				
Vectored interrupt sources	Maskable	Internal : 8, external : 4				
	Non-maskable	Internal : 1				
	Software	1				
Test input		Internal : 1, external : 1				
Power supply voltage		V_{DD} = 2.7 to 6.0 V				
Operating ambient temperature		T_A = –40 to +85°C				
Package		• 64-pin plastic shrink DIP (750 mil) • 64-pin plastic QFP (14 × 14 mm) • 64-pin ceramic shrink DIP with window (750 mils): μPD78P014Y only				

2.9 Mask Options

The mask ROM versions (μPD78011BY, μPD78012BY, μPD78013Y, μPD78014Y) have mask options. By specifying the mask options when ordering, the pull-up resistors and pull-down resistors listed in Table 2-1 can be incorporated. When these resistors are necessary, the number of external components and mounting space can be saved by utilizing the mask options.

Mask options provided in the μPD78014Y subseries are shown in Table 2-1.

Table 2-1. Mask Options in Mask ROM Versions

Pin Name	Mask Option
P60 to P63	Pull-up resistors can be incorporated bit-wise.

[MEMO]

CHAPTER 3 PIN FUNCTION (μPD78014 Subseries)

3.1 Pin Function List

3.1.1 Normal operating mode pins

(1) Port pins (1/2)

Pin Name	Input/ Output	Function		After Reset	Alternate Function
P00	Input	Port 0 5-bit input/output port	Input only	Input	INTP0/TI0
P01	Input/ Output		Input/output specifiable bit-wise. If used as an input port, an on-chip pull-up resistor can be connected by software.	Input	INTP1
P02					INTP2
P03					INTP3
P04[Note 1]	Input		Input only	Input	XT1
P10 to P17	Input/ Output	Port 1 8-bit input/output port. Input/output specifiable bit-wise. If used as an input port, an on-chip pull-up resistor can be connected by software[Note 2].		Input	ANI0 to ANI7
P20	Input/ Output	Port 2 8-bit input/output port. Input/output specifiable bit-wise. If used as an input port, an on-chip pull-up resistor can be connected by software.		Input	SI1
P21					SO1
P22					$\overline{\text{SCK1}}$
P23					STB
P24					BUSY
P25					SI0/SB0
P26					SO0/SB1
P27					$\overline{\text{SCK0}}$
P30	Input/ Output	Port 3 8-bit input/output port. LED can be driven directly. Input/output specifiable bit-wise. If used as an input port, an on-chip pull-up resistor can be connected by software.		Input	TO0
P31					TO1
P32					TO2
P33					TI1
P34					TI2
P35					PCL
P36					BUZ
P37					—

Notes 1. When the P04/XT1 pin is used as an input port, set the bit 6 (FRC) of the processor clock control register (PCC) to 1 (do not use the on-chip feedback resistor of the subsystem clock oscillator).

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

(1) Port pins (2/2)

<table>
<tr><th>Pin Name</th><th>Input/
Output</th><th colspan="2">Function</th><th>After
Reset</th><th>Alternate
Function</th></tr>
<tr><td>P40 to P47</td><td>Input/
Output</td><td colspan="2">Port 4
8-bit input/output port.
Input/output specifiable in 8-bit wise.
When used as an input port, an on-chip pull-up resistor can be connected by software.
Test input flag (KRIF) is set to 1 by falling edge detection.</td><td>Input</td><td>AD0 to AD7</td></tr>
<tr><td>P50 to P57</td><td>Input/
Output</td><td colspan="2">Port 5
8-bit input/output port.
Input/output specifiable in 8-bit wise.
LED can be driven directly.
When used as an input port, an on-chip pull-up resistor can be connected by software.</td><td>Input</td><td>A8 to A15</td></tr>
<tr><td>P60</td><td rowspan="8">Input/
Output</td><td rowspan="8">Port 6
8-bit input/output port.
Input/output specifiable bit-wise.</td><td rowspan="4">N-ch open drain input/output port.
On-chip pull-up resistor can be specified by mask option.
LED can be driven directly.</td><td rowspan="8">Input</td><td rowspan="4">—</td></tr>
<tr><td>P61</td></tr>
<tr><td>P62</td></tr>
<tr><td>P63</td></tr>
<tr><td>P64</td><td rowspan="4">When used as an input port, an on-chip pull-up resistor can be connected by software.</td><td>$\overline{RD}$</td></tr>
<tr><td>P65</td><td>$\overline{WR}$</td></tr>
<tr><td>P66</td><td>$\overline{WAIT}$</td></tr>
<tr><td>P67</td><td>ASTB</td></tr>
</table>

(2) Non-Port Pins (1/2)

Pin Name	Input/ Output	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/TI0
INTP1				P01
INTP2				P02
INTP3		External interrupt request input with falling edge detection	Input	P03
SI0	Input	Serial interface serial data input	Input	P25/SB0
SI1				P20
SO0	Output	Serial interface serial data output	Input	P26/SB1
SO1				P21
SB0	Input/ Output	Serial interface serial data input/output	Input	P25/SI0
SB1				P26/SO0
$\overline{SCK0}$	Input/ Output	Serial interface serial clock input/output	Input	P27
$\overline{SCK1}$				P22
STB	Output	Serial interface automatic transmit/receive strobe output	Input	P23
BUSY	Input	Serial interface automatic transmit/receive busy input	Input	P24
TI0	Input	External count clock input to 16-bit timer (TM0)	Input	P00/INTP0
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
AD0 to AD7	Input/ Output	Low-order address/data bus at external memory expansion.	Input	P40 to P47
A8 to A15	Output	High-order address bus at external memory expansion.	Input	P50 to P57
$\overline{RD}$	Output	External memory read operation strobe signal output.	Input	P64
$\overline{WR}$		External memory write operation strobe signal output.		P65
$\overline{WAIT}$	Input	Wait insertion at external memory access.	Input	P66
ASTB	Output	Strobe output which latches the address data output for ports 4 or 5 to access external memory.	Input	P67

(2) Non-Port Pins (2/2)

Pin Name	Input/ Output	Function	After Reset	Alternate Function
ANI0 to ANI7	Input	A/D converter analog input.	Input	P10 to P17
AV_{REF}	Input	A/D converter reference voltage input.	—	—
AV_{DD}	—	A/D converter analog power supply. Connect to V_{DD}.	—	—
AV_{SS}	—	A/D converter ground potential. Connect to V_{SS}.	—	—
$\overline{RESET}$	Input	System reset input	—	—
X1	Input	Crystal connection for main system clock oscillation	—	—
X2	—		—	—
XT1	Input	Crystal connection for subsystem clock oscillation	Input	P04
XT2	—		—	—
V_{DD}	—	Positive power supply	—	—
V_{PP}	—	High-voltage application for program write/verify. Connect directly to V_{SS} in normal operating mode.	—	—
V_{SS}	—	Ground potential	—	—
IC	—	Internally connected. Connect directly to V_{SS}.	—	—

3.1.2 PROM programming mode pins (μPD78P014 only)

Pin Name	Input/ Output	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 A14	Input	Address bus
D0 to D7	Input/ output	Data bus
$\overline{CE}$	Input	PROM enable input/program pulse input
$\overline{OE}$	Input	Read strobe input to PROM
V_{DD}	—	Positive power supply
V_{SS}	—	Ground potential

3.2 Description of Pin Functions

3.2.1 P00 to P04 (Port 0)

These are 5-bit input/output ports. Besides serving as input/output 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 bit-wise.

(1) Port mode

P00 and P04 function as input-only ports and P01 to P03 function as input/output ports.

P01 to P03 can be specified for input or output ports bit-wise with a port mode register 0 (PM0). When they are used as input ports, a pull-up resistor can be connected to them with an on-chip pull-up resistor option register (PUO).

(2) Control mode

In this 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 INTP3

INTP0 to INTP2 are external interrupt request input pins which can specify valid edges (rising edge, falling edge, and both rising and falling edges). INTP0 becomes a 16-bit timer/event counter capture trigger signal input pin with a valid edge input. INTP3 becomes a falling edge detection external interrupt request input pin.

(b) TI0

TI0 is a pin for external count clock input to 16-bit timer/event counter.

(c) XT1

Crystal connection pin for subsystem clock oscillation

3.2.2 P10 to P17 (Port 1)

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

The following operating modes can be specified bit-wise.

(1) Port mode

These ports function as 8-bit input/output ports. They can be specified bit-wise as input or output ports with port mode register 1 (PM1). When used as an input port, an on-chip pull-up resistor can be connected to these ports with a pull-up resistor option register (PUO).

(2) Control mode

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

3.2.3 P20 to P27 (Port 2)

These are 8-bit input/output ports. Besides serving as input/output ports, they function as data input/output to/from the serial interface, clock input/output, automatic transmit/receive busy input, and strobe output functions.

The following operating modes can be specified bit-wise.

(1) Port mode

These ports function as 8-bit input/output ports. They can be specified bit-wise as input or output ports with a port mode register 2 (PM2). When they are used as input ports, an on-chip pull-up resistor can be connected to them with a pull-up resistor option register (PUO).

(2) Control mode

These ports function as serial interface data input/output, clock input/output, automatic transmit/receive busy input, and strobe output.

(a) SI0, SI1, SO0, SO1

Serial interface serial data input/output pins

(b) $\overline{\text{SCK0}}$ and $\overline{\text{SCK1}}$

Serial interface serial clock input/output pins

(c) SB0 and SB1

NEC standard serial bus interface input/output pins

(d) BUSY

Serial interface automatic transmit/receive busy input pins

(e) STB

Serial interface automatic transmit/receive strobe output pins

Caution When these ports are used as pins of serial interface, set the input/output and output latches depending on their functions. For the setting method, refer to Figure 15-5 Serial Operating Mode Register 0 Format and Figure 17-3 Serial Operating Mode Register 1 Format.

3.2.4 P30 to P37 (Port 3)

These are 8-bit input/output ports. Beside serving as input/output ports, they function as timer input/output, clock output and buzzer output.

The following operating modes can be specified bit-wise.

(1) Port mode

These ports function as 8-bit input/output ports. They can be specified bit-wise as input or output ports with port mode register 3 (PM3). When they are used as input ports, an on-chip pull-up resistor can be connected with a pull-up resistor option register (PUO).

(2) Control mode

These ports function as timer input/output, 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 P40 to P47 (Port 4)

These are 8-bit input/output ports. Besides serving as input/output ports, they function as address/data bus.
Test input flag (KRIF) is set to 1 by falling edge detection.
The following operating modes can be specified in 8-bit units.

(1) Port mode

These ports function as 8-bit input/output ports. They can be specified in 8-bit units as input or output ports with a memory expansion mode register (MM). When they are used as input ports, a pull-up resistor can be connected to them with an on-chip pull-up resistor option register (PUO).

(2) Control mode

These ports function as low-order address/data bus pins in external memory expansion mode. When they are used as address/data bus, an on-chip pull-up resistor is automatically unused.

3.2.6 P50 to P57 (Port 5)

These are 8-bit input/output ports. Besides serving as input/output ports, they function as address/data bus.
They can drive LEDs directly.
The following operating modes can be specified bit-wise.

(1) Port mode

These ports function as 8-bit input/output ports. They can be specified bit-wise as input or output ports with port mode register 5 (PM5). When they are used as input ports, an on-chip pull-up resistor can be connected to them with a pull-up resistor option register (PUO).

(2) Control mode

These ports function as high-order address bus pins in external memory expansion mode. When they are used as address/data bus, the on-chip pull-up resistor is automatically disabled.

3.2.7 P60 to P67 (Port 6)

These are 8-bit output dedicated ports. Besides serving as input/output port, they have control functions in external memory expansion mode.

P60 to P63 can drive LEDs directly.

The following operating modes can be specified bit-wise.

(1) Port mode

These ports function as 8-bit input/output ports. They can be specified bit-wise for input or output ports by port mode register 6 (PM6).

P60 to P63 are N-ch open-drain outputs. Mask ROM version can contain pull-up resistors with the mask option.

When P64 to P67 are used as input ports, an on-chip pull-up resistor can be connected with a pull-up resistor option resistor (PUO).

(2) Control mode

These ports function as control signal output pins ($\overline{RD}$, $\overline{WR}$, $\overline{WAIT}$, ASTB) in external memory expansion mode.

When a pin is used as control signal output, the on-chip pull-up resistor is automatically disabled.

Caution When external wait is not used in external memory expansion mode, P66 can be used as an input/output port.

3.2.8 AV_{REF}

A/D converter reference voltage input pin.

When the A/D converter is not used, connect it to V_{SS}.

3.2.9 AV_{DD}

Analog power supply pin of A/D converter.

Always use the same voltage as that of the V_{DD} pin even when A/D converter is not used.

3.2.10 AV_{SS}

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

3.2.11 $\overline{RESET}$

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

3.2.12 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.

3.2.13 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.

3.2.14 V_{DD}

Positive power supply pin

3.2.15 V_{SS}

Ground potential pin

3.2.16 V_{PP} (μPD78P014 only)

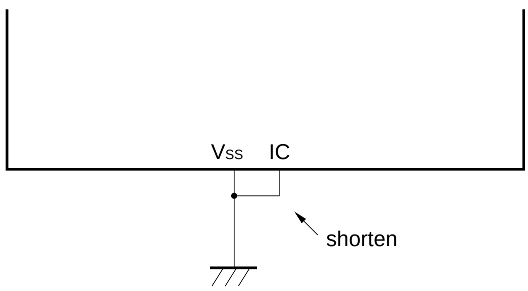
High-voltage apply pin for PROM programming mode setting and program write/verify. Connect directly to V_{SS} in normal operating mode.

3.2.17 IC (Mask ROM version only)

The IC (Internally Connected) pin sets a test mode in which the μPD78011B, 78012B, 78013 and 78014 are tested before shipment. In normal operation mode, connect the IC pin directly to V_{SS} with as short a wiring length as possible.

If there is a potential difference between the IC and V_{SS} pins because the wiring length between the IC and V_{SS} pins is too long, or external noise is superimposed on the IC pin, your program may not run correctly.

- **Directly connect the IC pin to the V_{SS}.**



3.3 Input/Output Circuit and Recommended Connection of Unused Pins

Table 3-1 shows the input/output circuit types of pins and the recommended conditions for unused pins.
Refer to **Figure 3-1** for the configuration of the input/output circuit of each type.

Table 3-1. Pin Input/Output Circuit Types (1/2)

Pin Name	Input/Output Circuit Type	Input/Output	Recommended Connection for Unused Pins
P00/INTP0/TI0	2	Input	Connect to V_{SS}.
P01/INTP1	8-A	Input/output	Independently connect to V_{SS} via a resistor.
P02/INTP2			
P03/INTP3			
P04/XT1	16	Input	Connect to V_{DD} or V_{SS}.
P10/ANI0 to P17/ANI7	11	Input/output	Independently connect to V_{DD} or V_{SS} via a resistor.
P20/SI1	8-A		
P21/SO1	5-A		
P22/$\overline{SCK1}$	8-A		
P23/STB	5-A		
P24/BUSY	8-A		
P25/SI0/SB0	10-A		
P26/SO0/SB1			
P27/$\overline{SCK0}$			
P30/TO0	5-A		
P31/TO1			
P32/TO2			
P33/TI1	8-A		
P34/TI2			
P35/PCL	5-A		
P36/BUZ			
P37			
P40/AD0 to P47/AD7	5-E		Independently connect to V_{DD} via a resistor.
P50/A8 to P57/A15	5-A		Independently connect to V_{DD} or V_{SS} via a resistor.
P60 to P63 (Mask ROM Version)	13-B		Independently connect to V_{DD} via a resistor.
P60 to P63 (PROM Version)	13		
P64/$\overline{RD}$	5-A		Independently connect to V_{DD} or V_{SS} via a resistor.
P65/$\overline{WR}$			
P66/$\overline{WAIT}$			
P67/ASTB			

Table 3-1. Pin Input/Output Circuit Types (2/2)

Pin Name	Input/Output Circuit Type	Input/Output	Recommended Connection for Unused Pins
$\overline{RESET}$	2	Input	—
XT2	16	—	Leave open
AV_{REF}	—		Connect to V_{SS}.
AV_{DD}			Connect to V_{DD}.
AV_{SS}			Connect to V_{SS}.
IC (Mask ROM Version)			Connect directly to V_{SS}.
V_{PP} (PROM Version)			

Figure 3-1. Pin Input/Output Circuit List (1/2)

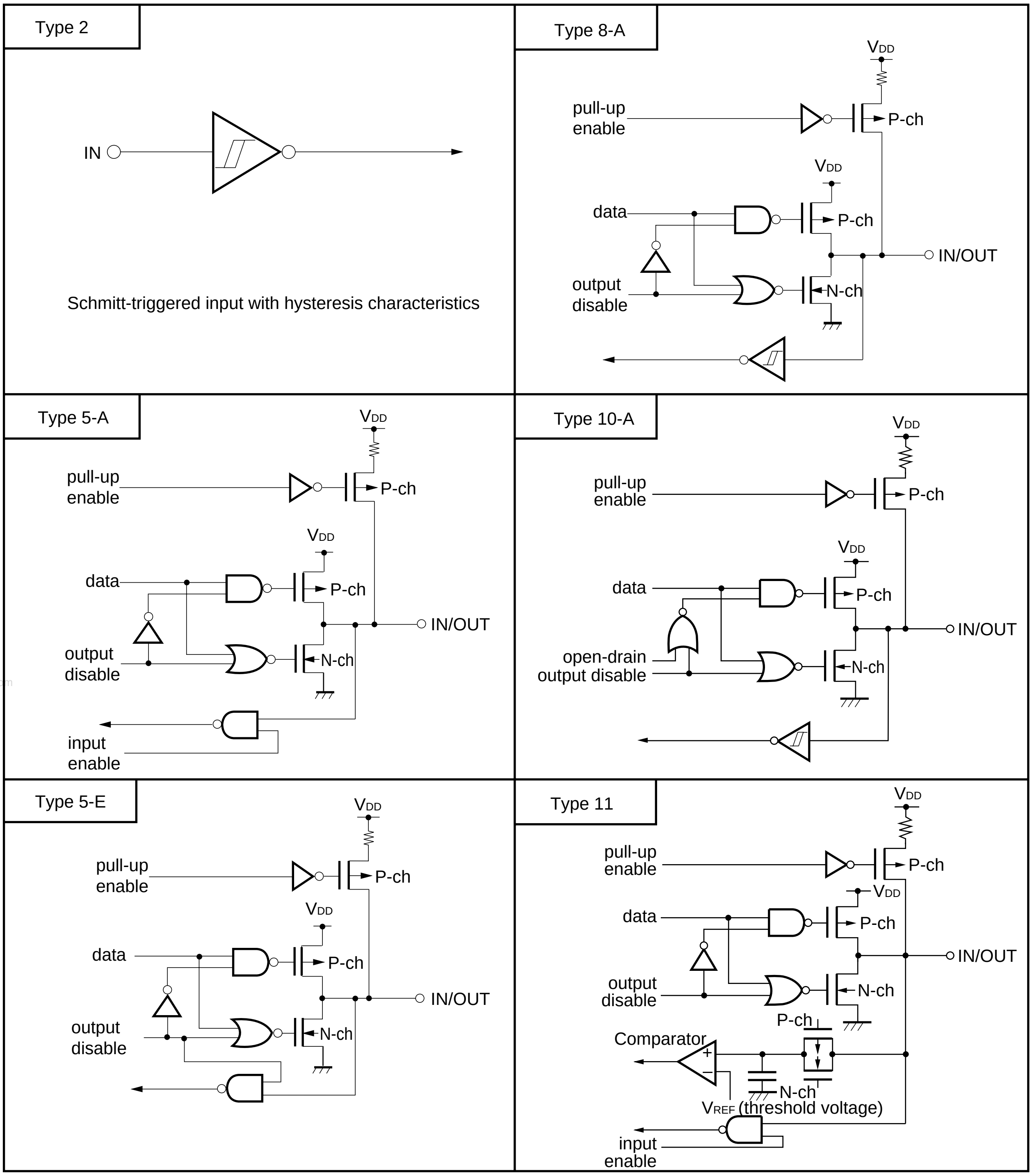
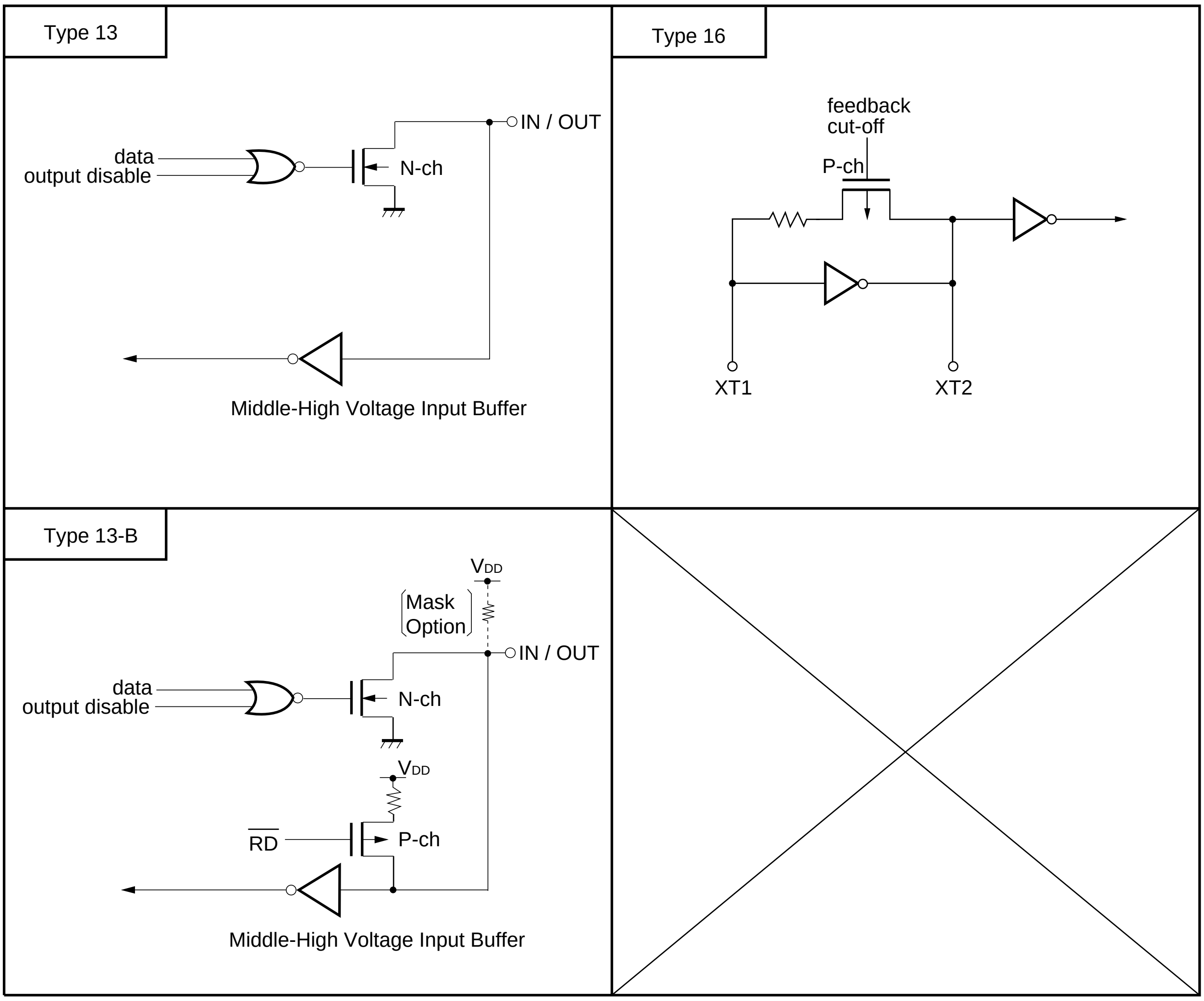


Figure 3-1. Pin Input/Output Circuit List (2/2)



[MEMO]

CHAPTER 4   PIN FUNCTION (μPD78014Y Subseries)

4.1 Pin Function List

4.1.1 Normal operating mode pins

(1) Port pins (1/2)

Pin Name	Input/ Output	Function		After Reset	Alternate Function
P00	Input	Port 0 5-bit input/output port	Input only	Input	INTP0/TI0
P01	Input/ Output		Input/output specifiable bit-wise. If used as an input port, an on-chip pull-up resistor can be connected by software.	Input	INTP1
P02					INTP2
P03					INTP3
P04[Note 1]	Input		Input only	Input	XT1
P10 to P17	Input/ Output	Port 1 8-bit input/output port. Input/output specifiable bit-wise. If used as an input port, an on-chip pull-up resistor can be connected by software[Note 2].		Input	ANI0 to ANI7
P20	Input/ Output	Port 2 8-bit input/output port. Input/output specifiable bit-wise. If used as an input port, an on-chip pull-up resistor can be connected by software.		Input	SI1
P21					SO1
P22					$\overline{SCK1}$
P23					STB
P24					BUSY
P25					SI0/SB0/SDA0
P26					SO0/SB1/SDA1
P27					$\overline{SCK0}$/SC1
P30	Input/ Output	Port 3 8-bit input/output port. Input/output specifiable bit-wise. If used as an input port, an on-chip pull-up resistor can be connected by software.		Input	TO0
P31					TO1
P32					TO2
P33					TI1
P34					TI2
P35					PCL
P36					BUZ
P37					—

Notes 1. When the P04/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 on-chip feedback resistor of the subsystem clock oscillator).

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

(1) Port pins (2/2)

Pin Name	Input/ Output	Function		After Reset	Alternate Function
P40 to P47	Input/ Output	Port 4 8-bit input/output port. Input/output specifiable in 8-bit units. When used as an input port, an on-chip pull-up resistor can be connected by software. Test input flag (KRIF) is set to 1 by falling edge detection.		Input	AD0 to AD7
P50 to P57	Input/ Output	Port 5 8-bit input/output port. LED can be driven directly. Input/output specifiable bit-wise. When used as an input port, an on-chip pull-up resistor can be connected by software.		Input	A8 to A15
P60	Input/ Output	Port 6 8-bit input/output port. Input/output specifiable bit-wise.	N-ch open drain input/output port. On-chip pull-up resistor can be specified by mask option. LED can be driven directly.	Input	—
P61					
P62					
P63					
P64			When used as an input port, an on-chip pull-up resistor can be connected by software.		$\overline{RD}$
P65					$\overline{WR}$
P66					$\overline{WAIT}$
P67					ASTB

(2) Non-Port Pins (1/2)

Pin Name	Input/ Output	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/TI0
INTP1				P01
INTP2				P02
INTP3		External interrupt request input with falling edge detection.	Input	P03
SI0	Input	Serial interface serial data input	Input	P25/SB0/SDA0
SI1				P20
SO0	Output	Serial interface serial data output	Input	P26/SB1/SDA1
SO1				P21
SB0	Input/ Output	Serial interface serial data input/output	Input	P25/SI0/SDA0
SB1				P26/SO0/SDA1
SDA0				P25/SI0/SB0
SDA1				P26/SO0/SB1
$\overline{\text{SCK0}}$	Input/ Output	Serial interface serial clock input/output	Input	P27/SCL
$\overline{\text{SCK1}}$				P22
SCL				P27/$\overline{\text{SCK0}}$
STB	Output	Serial interface automatic transmit/receive strobe output	Input	P23
BUSY	Input	Serial interface automatic transmit/receive busy input	Input	P24
TI0	Input	External count clock input to 16-bit timer (TM0)	Input	P00/INTP0
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
AD0 to AD7	Input/ Output	Low-order address/data bus at external memory expansion.	Input	P40 to P47
A8 to A15	Output	High-order address bus at external memory expansion.	Input	P50 to P57
$\overline{\text{RD}}$	Output	External memory read operation strobe signal output.	Input	P64
$\overline{\text{WR}}$		External memory write operation strobe signal output.		P65
$\overline{\text{WAIT}}$	Input	Wait insertion at external memory access.	Input	P66
ASTB	Output	Strobe output which latches the address data output for ports 4 or 5 to access external memory.	Input	P67

(2) Non-Port Pins (2/2)

Pin Name	Input/ Output	Function	After Reset	Alternate Function
ANI0 to ANI7	Input	A/D converter analog input.	Input	P10 to P17
AV_{REF}	Input	A/D converter reference voltage input.	—	—
AV_{DD}	—	A/D converter analog power supply. Connect to V_{DD}.	—	—
AV_{SS}	—	A/D converter ground potential. Connect to V_{SS}.	—	—
$\overline{\text{RESET}}$	Input	System reset input	—	—
X1	Input	Crystal connection for main system clock oscillation	—	—
X2	—		—	—
XT1	Input	Crystal connection for subsystem clock oscillation	Input	P04
XT2	—		—	—
V_{DD}	—	Positive power supply	—	—
V_{PP}	—	High-voltage application for program write/verify. Connect directly to V_{SS} in normal operating mode.	—	—
V_{SS}	—	Ground potential	—	—
IC	—	Internally connected. Directly connect to V_{SS}.	—	—

4.1.2 PROM programming mode pins (μPD78P014Y only)

Pin Name	Input/ Output	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 A14	Input	Address bus
D0 to D7	Input/ output	Data bus
$\overline{\text{CE}}$	Input	PROM enable input/program pulse input
$\overline{\text{OE}}$	Input	Read strobe input to PROM
V_{DD}	—	Positive power supply
V_{SS}	—	Ground potential

4.2 Description of Pin Functions

4.2.1 P00 to P04 (Port 0)

These are 5-bit input/output ports. Besides serving as input/output 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 bit-wise.

(1) Port mode

P00 and P04 function as input-only ports and P01 to P03 function as input/output ports.

P01 to P03 can be specified for input or output ports bit-wise with port mode register 0 (PM0). When they are used as input ports, a pull-up resistor can be connected to them with an on-chip pull-up resistor option register (PUO).

(2) Control mode

In this 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 INTP3

INTP0 to INTP2 are external interrupt request input pins which can specify valid edges (rising edge, falling edge, and both rising and falling edges). INTP0 become a 16-bit timer/event counter capture trigger signal input pin with a valid edge input. INTP3 become a falling edge detection external interrupt request input pin.

(b) TI0

TI0 is a pin for external count clock input to 16-bit timer/event counter.

(c) XT1

Crystal connection pin for subsystem clock oscillation

4.2.2 P10 to P17 (Port 1)

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

The following operating modes can be specified bit-wise.

(1) Port mode

These ports function as 8-bit input/output ports. They can be specified bit-wise as input or output ports with port mode register 1 (PM1). When used as input ports, an on-chip pull-up resistor can be connected to these ports with a pull-up resistor option register (PUO).

(2) Control mode

These ports function as A/D converter analog input pins (ANI0 to ANI7). If the pins are specified as analog input, the on-chip pull-up resistor is automatically disabled.

4.2.3 P20 to P27 (Port 2)

These are 8-bit input/output ports. Besides serving as input/output ports, they function as data input/output to/from the serial interface, clock input/output, automatic transmit/receive busy input, and strobe output.

The following operating modes can be specified bit-wise.

(1) Port mode

These ports function as 8-bit input/output ports. They can be specified bit-wise as input or output ports with port mode register 2 (PM2). When they are used as input ports, an on-chip pull-up resistor can be connected to them with a pull-up resistor option register (PUO).

(2) Control mode

These ports function as serial interface data input/output, clock input/output, automatic transmit/receive busy input, and strobe output.

(a) SI0, SI1, SO0, SO1, SDA0, SDA1

Serial interface serial data input/output pins

(b) $\overline{\text{SCK0}}$, $\overline{\text{SCK1}}$, SCL

Serial interface serial clock input/output pins

(c) SB0 and SB1

NEC standard serial bus interface input/output pins

(d) BUSY

Serial interface automatic transmit/receive busy input pins

(e) STB

Serial interface automatic transmit/receive strobe output pins

Caution When these ports are used as pins of serial interface, set the input/output and output latches depending on their functions. For the setting method, refer to Figure 16-6 Serial Operating Mode Register 0 Format and Figure 17-3 Serial Operating Mode Register 1 Format.

4.2.4 P30 to P37 (Port 3)

These are 8-bit input/output ports. Beside serving as input/output ports, they function as timer input/output, clock output and buzzer output.

The following operating modes can be specified bit-wise.

(1) Port mode

These ports function as 8-bit input/output ports. They can be specified bit-wise as input or output ports with port mode register 3 (PM3). When they are used as input ports, an on-chip pull-up resistor can be connected with a pull-up resistor option register (PUO).

(2) Control mode

These ports function as timer input/output, 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

4.2.5 P40 to P47 (Port 4)

These are 8-bit input/output ports. Besides serving as input/output ports, they function as address/data bus.
The test input flag (KRIF) is set to 1 by falling edge detection.
The following operating modes can be specified in 8-bit units.

(1) Port mode

These ports function as 8-bit input/output ports. They can be specified in 8-bit units as input or output ports with a memory expansion mode register (MM). When they are used as input ports, a pull-up resistor can be connected to them with an on-chip pull-up resistor option register (PUO).

(2) Control mode

These ports function as low-order address/data bus pins in external memory expansion mode. When they are used as address/data bus, the on-chip pull-up resistor is automatically disabled.

4.2.6 P50 to P57 (Port 5)

These are 8-bit input/output ports. Besides serving as input/output ports, they function as address/data bus.
They can drive LEDs directly.
The following operating modes can be specified bit-wise.

(1) Port mode

These ports function as 8-bit input/output ports. They can be specified bit-wise as an input or output ports with port mode register 5 (PM5). When they are used as input ports, an on-chip pull-up resistor can be connected to them with a pull-up resistor option register (PUO).

(2) Control mode

These ports function as high-order address/data bus pins (A8 to A15) in external memory expansion mode. When they are used as address bus, the on-chip pull-up resistor is automatically disabled.

4.2.7 P60 to P67 (Port 6)

These are 8-bit input/output ports. Besides serving as an input/output port, they have control functions in external memory expansion mode.

P60 to P63 can drive LEDs directly.

The following operating modes can be specified bit-wise.

(1) Port mode

These ports function as 8-bit input/output ports. They can be specified bit-wise as input or output ports by port mode register 6 (PM6).

P60 to P63 are N-ch open-drain outputs. Mask ROM versions can contain pull-up resistors with the mask option.

When P64 to P67 are used as input ports, an on-chip pull-up resistor can be connected with a pull-up resistor option resistor (PUO).

(2) Control mode

These ports function as control signal output pins ($\overline{RD}$, $\overline{WR}$, $\overline{WAIT}$, ASTB) in external memory expansion mode.

When a pin is used as control signal output, the on-chip pull-up resistor is automatically disabled.

Caution When external wait is not used in external memory expansion mode, P66 can be used as an input/ output port.

4.2.8 AV_{REF}

A/D converter reference voltage input pin.

When the A/D converter is not used, connect it to the V_{SS}.

4.2.9 AV_{DD}

Analog power supply pin of A/D converter.

Always use the same voltage as that of the V_{DD} pin even when A/D converter is not used.

4.2.10 AV_{SS}

This is a ground voltage pin of A/D converter.

Always use the same voltage as that of the V_{SS} pin even when A/D converter is not used.

4.2.11 $\overline{RESET}$

This is a low level active system reset input pin.

4.2.12 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.

4.2.13 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.

4.2.14 V_{DD}

Positive power supply pin

4.2.15 V_{SS}

Ground potential pin

4.2.16 V_{PP} (μPD78P014Y only)

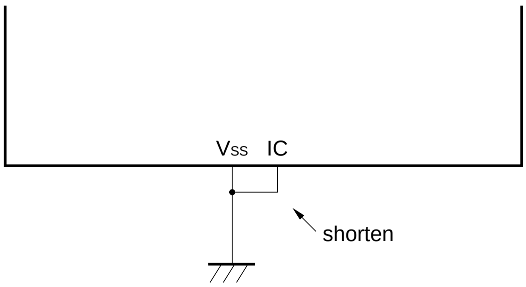
High-voltage apply pin for PROM programming mode setting and program write/verify. Connect directly to V_{SS} in normal operating mode.

4.2.17 IC (Mask ROM versions only)

The IC (Internally Connected) pin sets a test mode in which the μPD78011BY, 78012BY, 78013Y and 78014Y are tested before shipment. In normal operation mode, connect the IC pin directly to V_{SS} with as short a wiring length as possible.

If there is a potential difference between the IC and V_{SS} pins because the wiring length between the IC and V_{SS} pin is too long, or external noise is superimposed on the IC pin, your program may not run correctly.

- **Directly connect the IC pin to the V_{SS}.**



4.3 Input/Output Circuit and Recommended Connection of Unused Pins

Table 4-1 shows the input/output circuit types of pins and the recommended conditions for unused pins. Refer to **Figure 4-1** for the configuration of the input/output circuit of each type.

Table 4-1. Pin Input/Output Circuit Types (1/2)

Pin Name	Input/Output	Input/Output Circuit Type	Recommended Connection for Unused Pins
P00/INTP0/TI0	2	Input	Connect to V_{SS}.
P01/INTP1	8-A	Input/output	Independently connect to V_{SS} via a resistor.
P02/INTP2			
P03/INTP3			
P04/XT1	16	Input	Connect to V_{DD} or V_{SS}.
P10/ANI0 to P17/ANI7	11	Input/output	Independently connect to V_{DD} or V_{SS} via a resistor.
P20/SI1	8-A		
P21/SO1	5-A		
P22/$\overline{\text{SCK1}}$	8-A		
P23/STB	5-A		
P24/BUSY	8-A		
P25/SI0/SB0/SDA0	10-A		
P26/SO0/SB1/SDA1			
P27/$\overline{\text{SCK0}}$/SCL			
P30/TO0	5-A		
P31/TO1			
P32/TO2			
P33/TI1	8-A		
P34/TI2			
P35/PCL	5-A		
P36/BUZ			
P37			
P40/AD0 to P47/AD7	5-E		Independently connect to V_{DD} via a resistor.
P50/A8 to P57/A15	5-A		Independently connect to V_{DD} or V_{SS} via a resistor.
P60 to P63 (Mask ROM Version)	13-B		Independently connect to V_{DD} via a resistor.
P60 to P63 (PROM Version)	13		
P64/$\overline{\text{RD}}$	5-A		Independently connect to V_{DD} or V_{SS} via a resistor.
P65/$\overline{\text{WR}}$			
P66/$\overline{\text{WAIT}}$			
P67/ASTB			

Table 4-1. Pin Input/Output Circuit Types (2/2)

Pin Name	Input/Output	Input/Output Circuit Type	Recommended Connection for Unused Pins
$\overline{\text{RESET}}$	2	Input	—
XT2	16	—	Leave open
AV_{REF}	—		Connect to V_{SS}.
AV_{DD}			Connect to V_{DD}.
AV_{SS}			Connect to V_{SS}.
IC (Mask ROM Version)			Connect directly to V_{SS}.
V_{PP} (PROM Version)			

Figure 4-1. Pin Input/Output Circuit List (1/2)

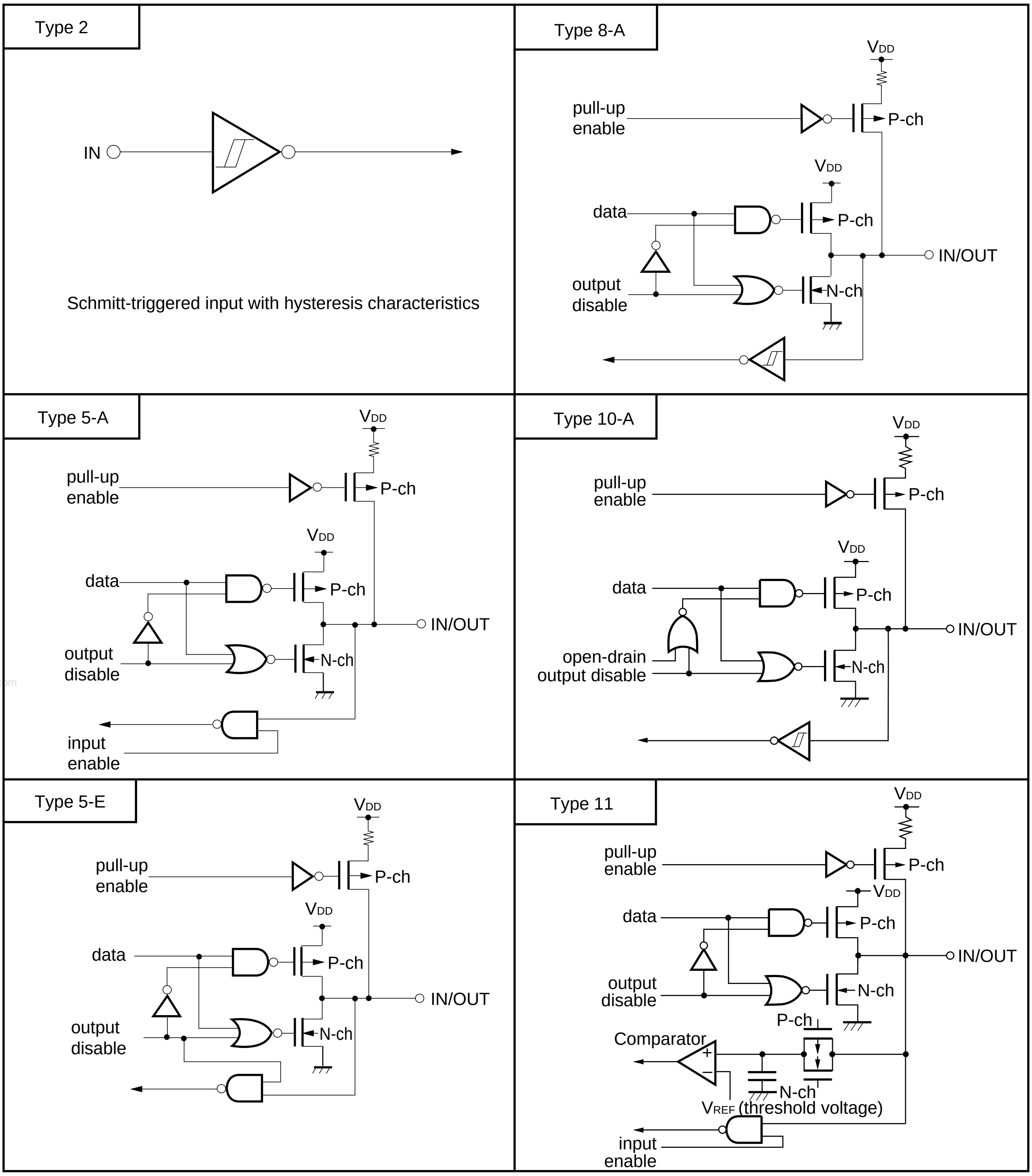
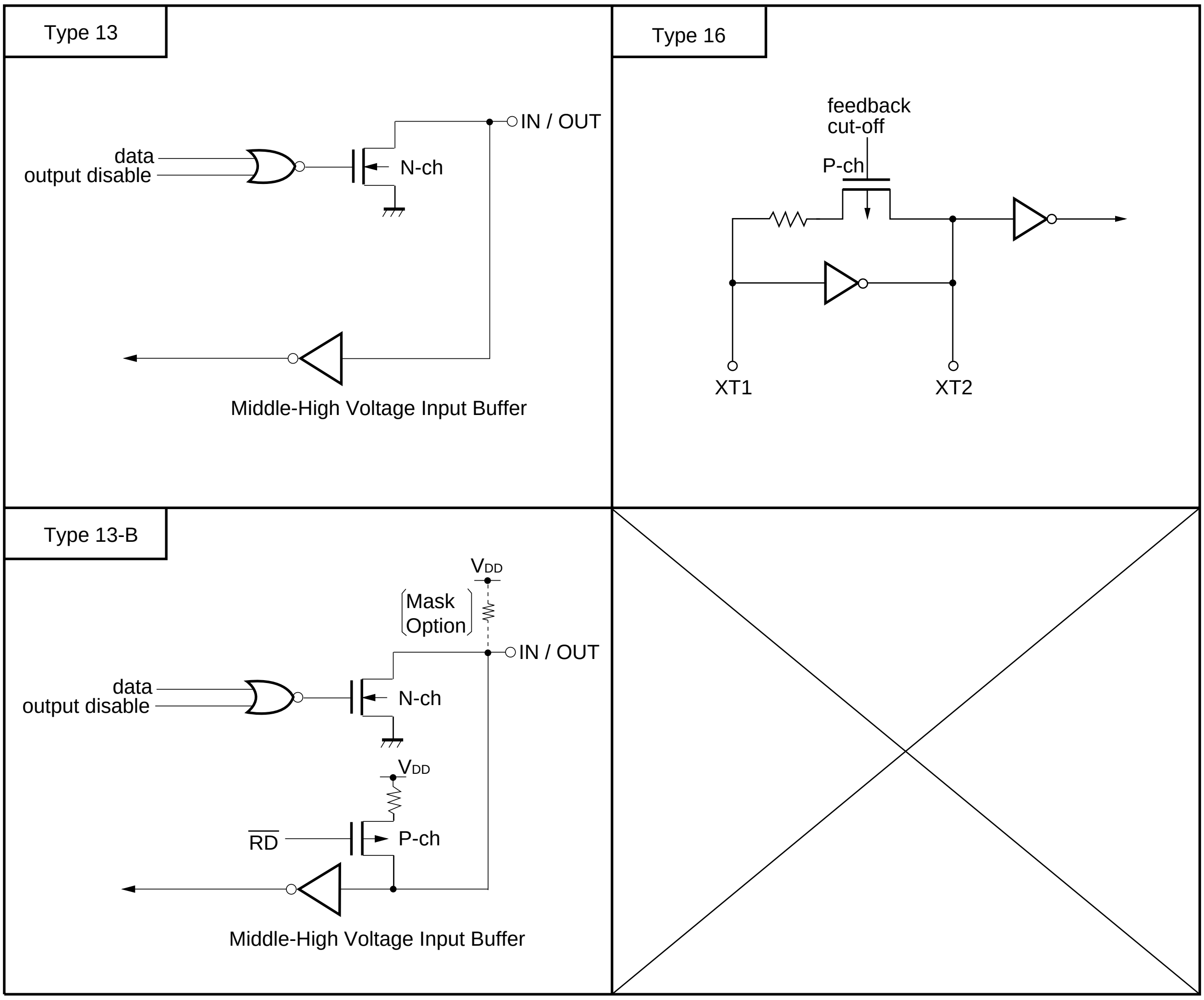


Figure 4-1. Pin Input/Output Circuit List (2/2)



[MEMO]

CHAPTER 5 CPU ARCHITECTURE

5.1 Memory Spaces

The μPD78014 and 78014Y Subseries can each access a memory space of 64 Kbytes. Figures 5-1 to 5-5 show memory maps.

Figure 5-1. Memory Map (μPD78011B, 78011BY)

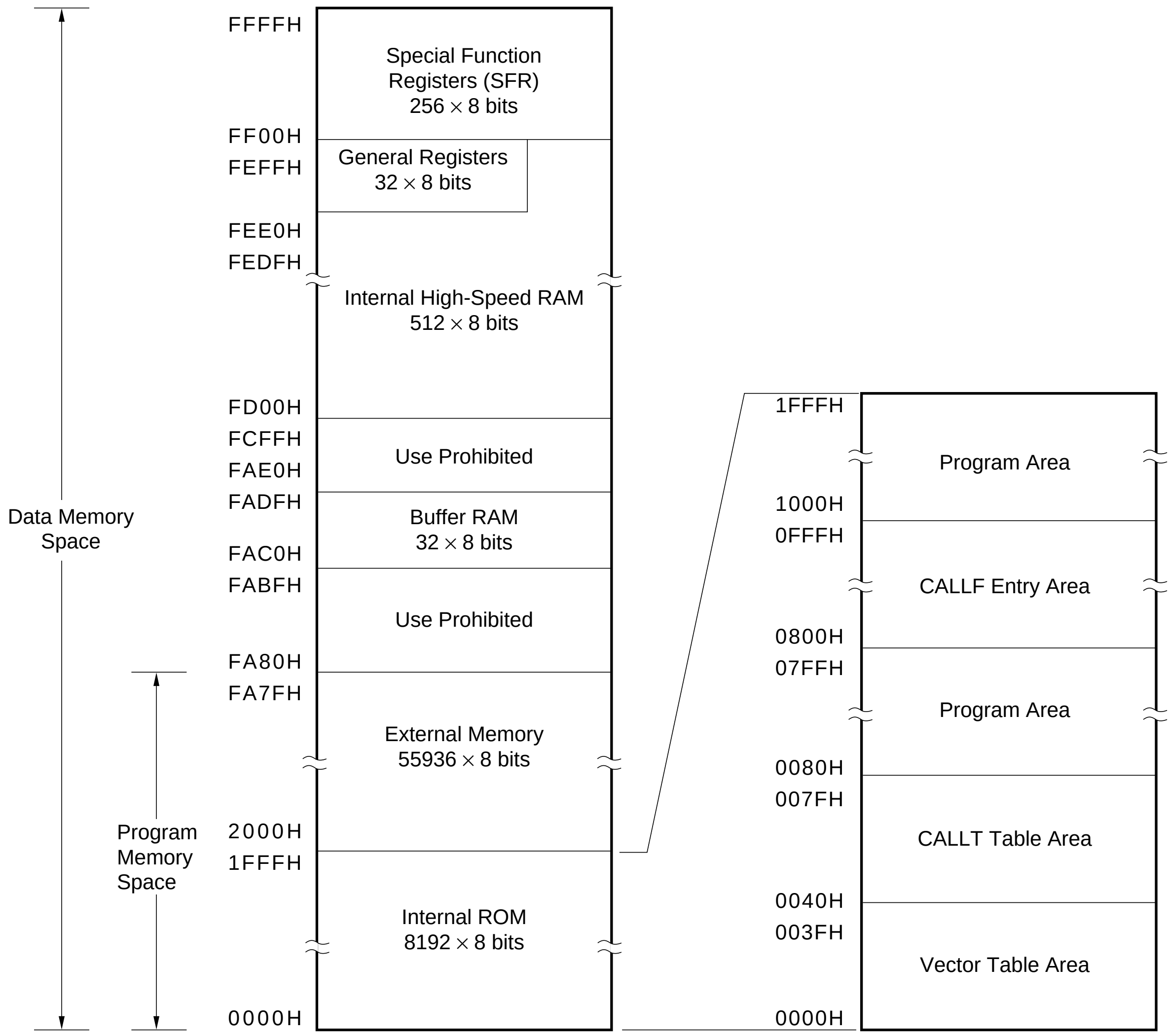


Figure 5-2. Memory Map (μPD78012B, 78012BY)

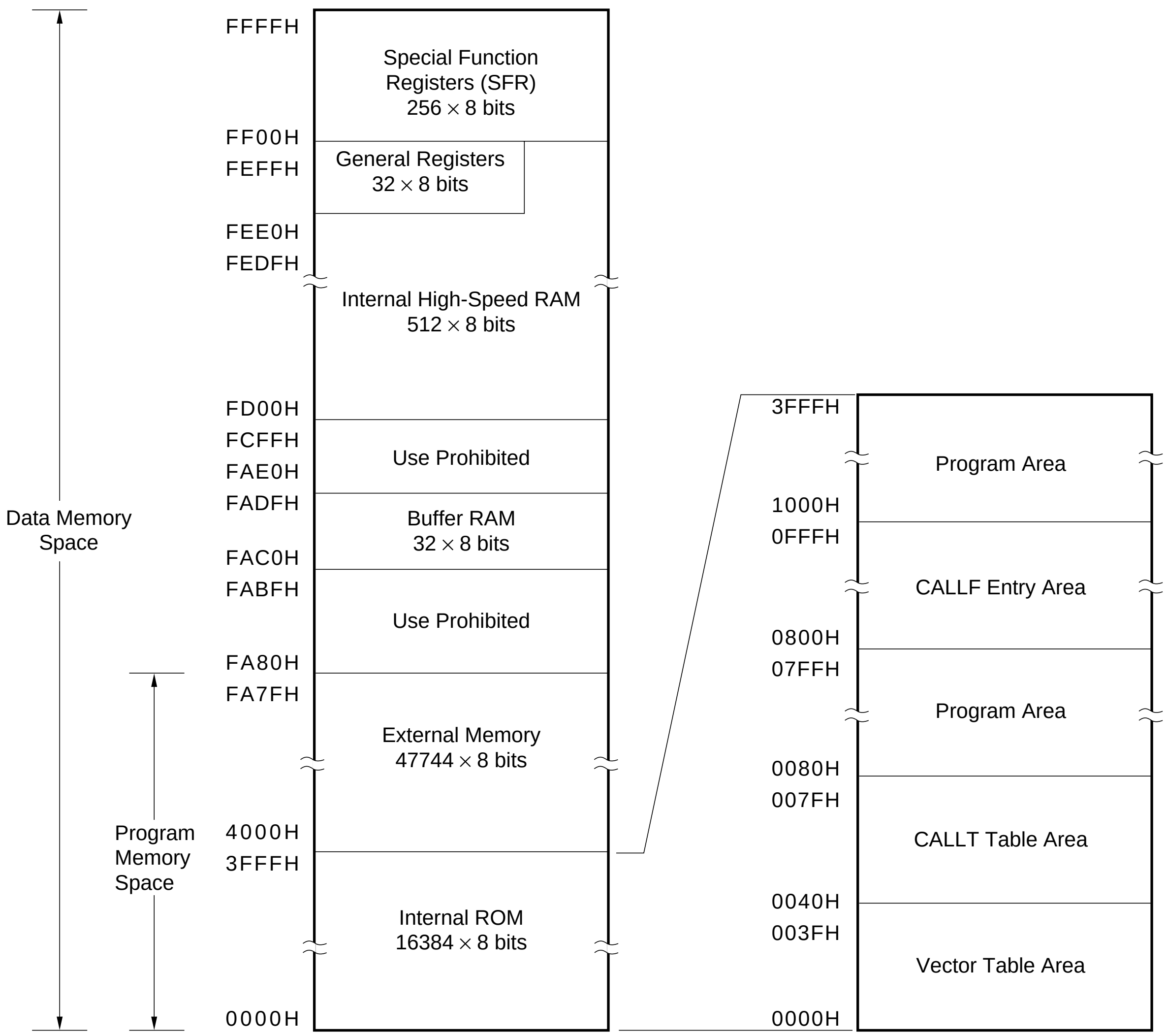


Figure 5-3. Memory Map (μPD78013, 78013Y)

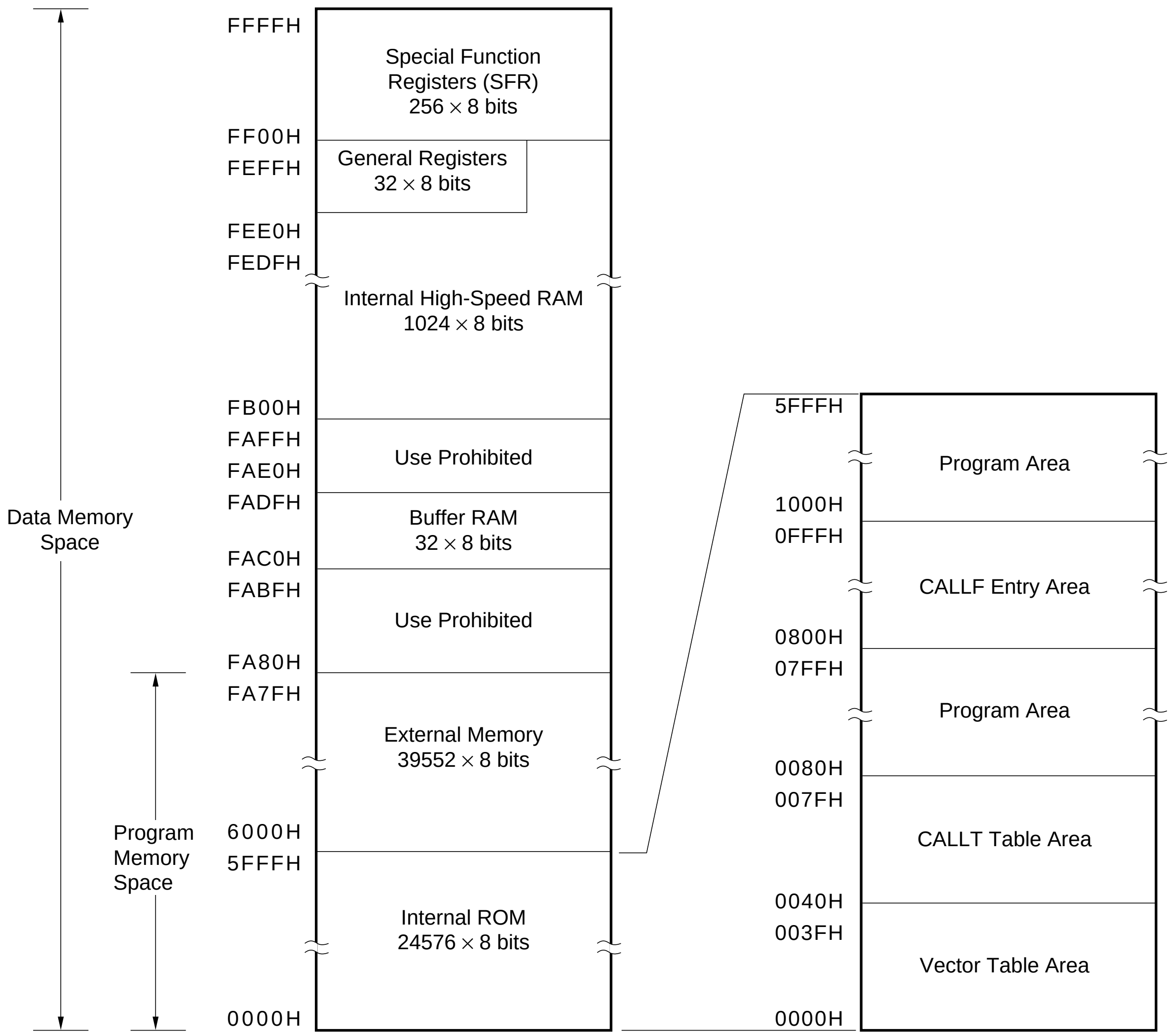


Figure 5-4. Memory Map (μPD78014, 78014Y)

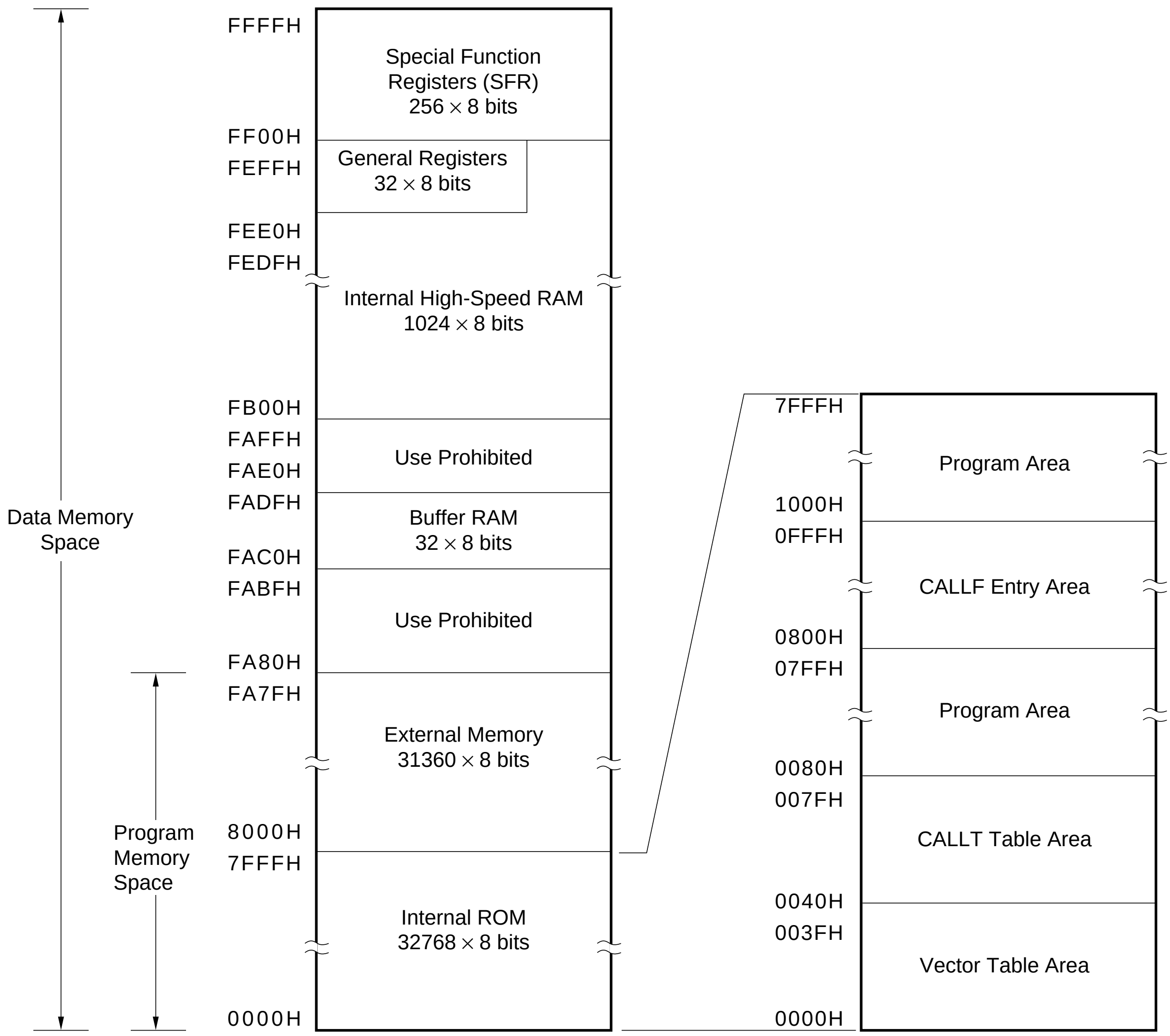
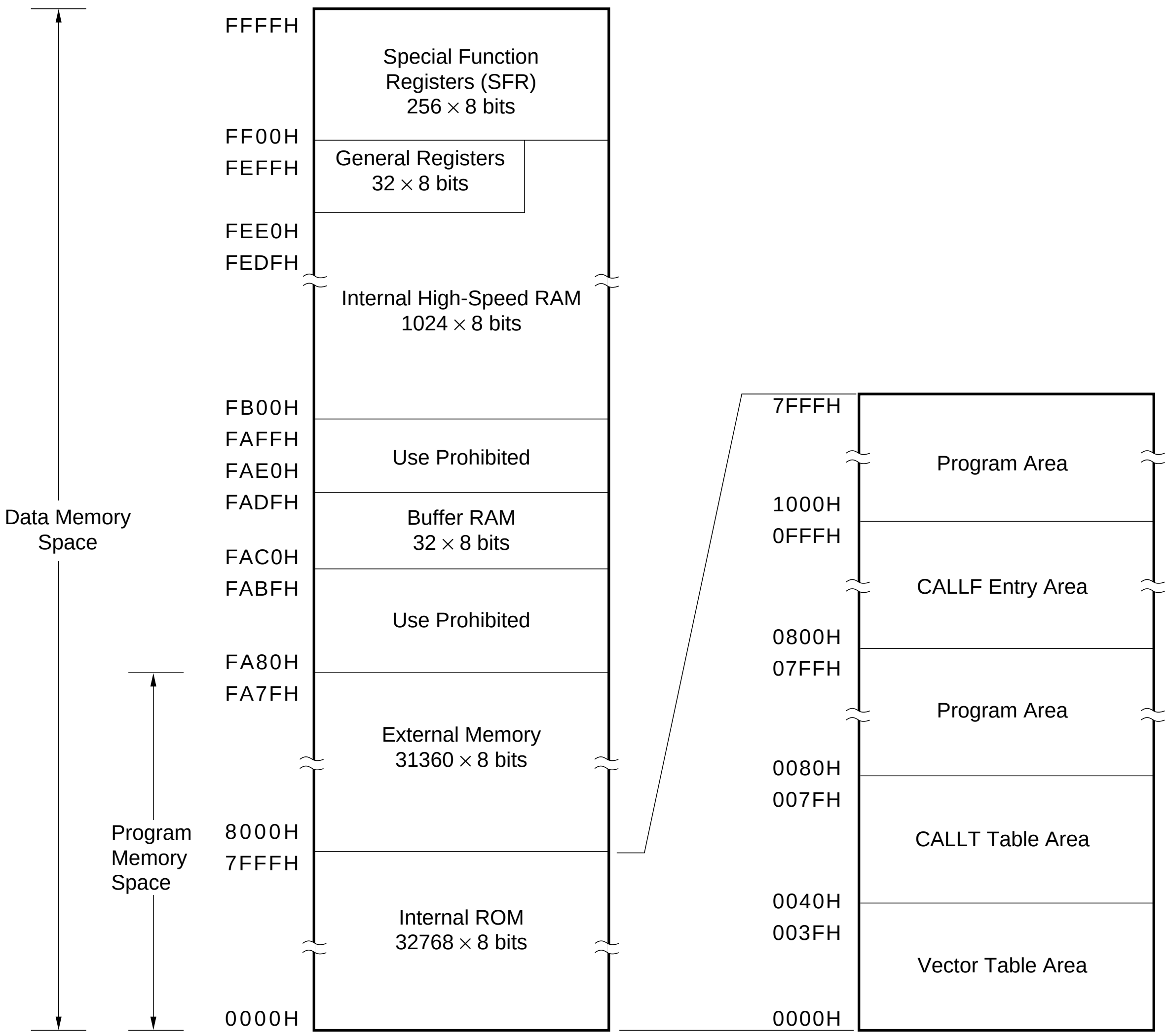


Figure 5-5. Memory Map (μPD78P014, 78P014Y)



5.1.1 Internal program memory space

Internal program memory store programs and table data. Normally, they are addressed with a program counter (PC).

The μPD78014 and 78014Y Subseries contain internal ROM (or PROM) in each product having the capacities shown below.

Table 5-1. Internal ROM Capacity

Part Number	Internal ROM	
	Configuration	Capacity
μPD78011B, 78011BY	Mask ROM	8192 × 8 bits
μPD78012B, 78012BY		16384 × 8 bits
μPD78013, 78013Y		24576 × 8 bits
μPD78014, 78014Y		32768 × 8 bits
μPD78P014, 78P014Y	PROM	

The following areas are allocated in the internal program memory space.

(1) Vector table area

The 64-byte area 0000H to 003FH is reserved as 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, the low-order 8 bits are stored at even addresses and the high-order 8 bits are stored at odd addresses.

Table 5-2. Vector Table

Vector Table Address	Interrupt Source	Vector Table Address	Interrupt Source
0000H	$\overline{\text{RESET}}$ input	0010H	INTCSI1
0004H	INTWDT	0012H	INTTM3
0006H	INTP0	0014H	INTTM0
0008H	INTP1	0016H	INTTM1
000AH	INTP2	0018H	INTTM2
000CH	INTP3	001AH	INTAD
000EH	INTCSI0	003EH	BRK Instruction

(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 μPD78014 and 78014Y Subseries incorporate the following RAMs.

(1) Internal high-speed RAM

The μPD78014 and 78014Y Subseries incorporate the following capacity of internal high-speed RAM in each product.

Table 5-3. Internal High-Speed RAM Capacities

Part Number	Internal High-speed RAM Capacity
μPD78011B, 78011BY	512 × 8 bits
μPD78012B, 78012BY	
μPD78013, 78013Y	1024 × 8 bits
μPD78014, 78014Y	
μPD78P014, 78P014Y	

4 banks of general 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 memory area.

(2) Buffer RAM

Buffer RAM is allocated to the 32-byte area from FAC0H to FADFH. Buffer RAM is used for storing transmit/receive data of serial interface channel 1 (3-wire serial I/O mode with automatic transmit/receive function). When not used in the 3-wire serial I/O mode with automatic transmit/receive function, buffer RAM can also be used as normal 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 5-5. Special Function Register List** of **5.2.3 Special function register (SFR)**).

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

5.1.4 External memory space

External memory space that can be accessed by setting a memory expansion mode register (MM). Program and table data can be stored, and peripheral devices can be allocated.

5.2 Processor Registers

The μPD78014 and 78014Y Subseries incorporate the following processor registers.

5.2.1 Control registers

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

(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 5-6. 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 5-7. Program Status Word Configuration

	7							0
PSW	IE	Z	RBS1	AC	RBS0	0	ISP	CY

(a) Interrupt enable flag (IE)

This flag controls interrupt request acknowledge operations of CPU.

When IE = 0, the IE is set to interrupt disabled (DI) status. All interrupt requests except non-maskable interrupt are disabled.

When IE = 1, the IE is set to interrupt enabled (EI) status and interrupt request acknowledgement is controlled with an inservice 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 request 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 interrupts.

When ISP = 0, acknowledgment of the vectored interrupt request specified to low-order priority with the priority specify flag registers (PR0L and PR0H) (refer to **18.3 (3) Priority specify flag registers (PR0L, PR0H)**) is disabled. Whether an actual interrupt request is acknowledged or not 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 can be set as the stack area. Internal high-speed RAM of each product is as follows.

μPD78011B, 78011BY, 78012B, 78012BY: FD00H to FEFFH

μPD78013, 78013Y, 78014, 78014Y, 78P014, 78P014Y: FB00H to FEFFH

Figure 5-8. 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 5-9 and 5-10.

Caution Since SP contents will be undefined by $\overline{\text{RESET}}$ input, be sure to initialize the SP before instruction execution.

Figure 5-9. Data to be Saved to Stack Memory

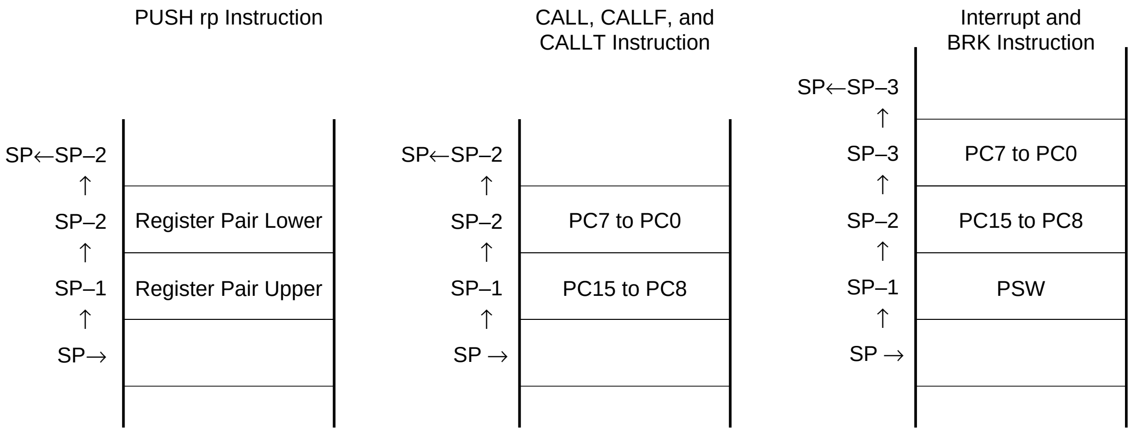
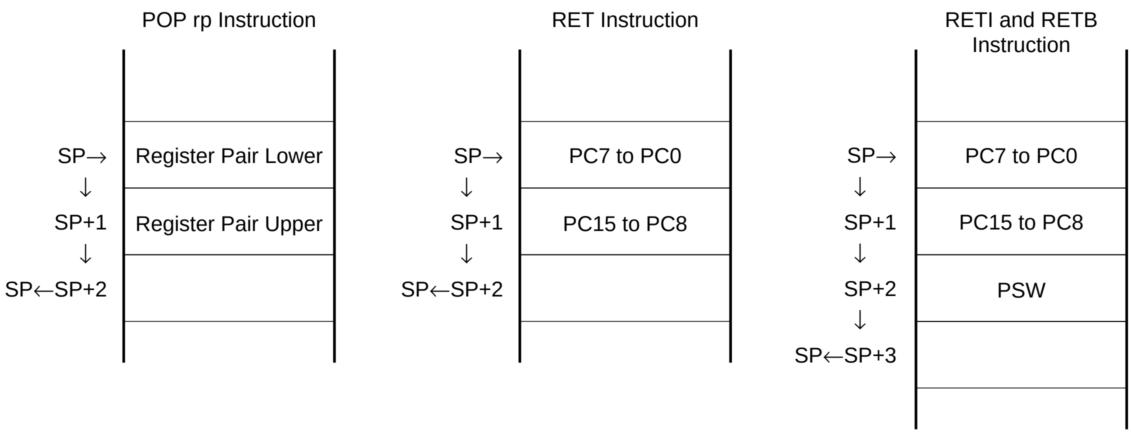


Figure 5-10. Data to be Reset from Stack Memory



5.2.2 General registers

A general 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).

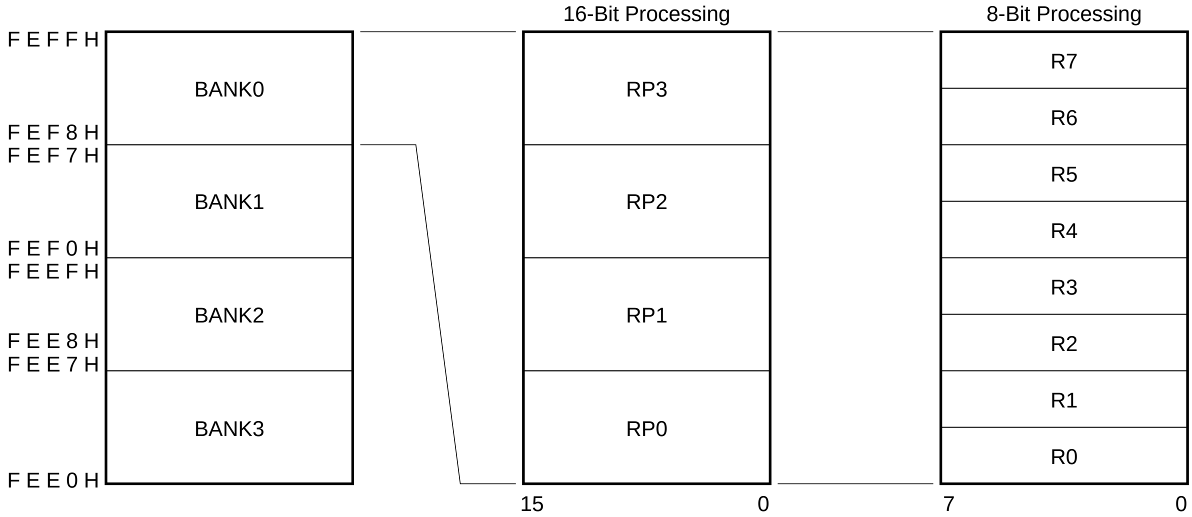
Register banks to be used for instruction execution are set with the CPU control instruction (SEL RBn). Because of the 4-register bank configuration, an efficient program can be created by switching between a register for normal processing and a register for interrupt request for each bank.

Table 5-4. Absolute Address Corresponding to General Registers

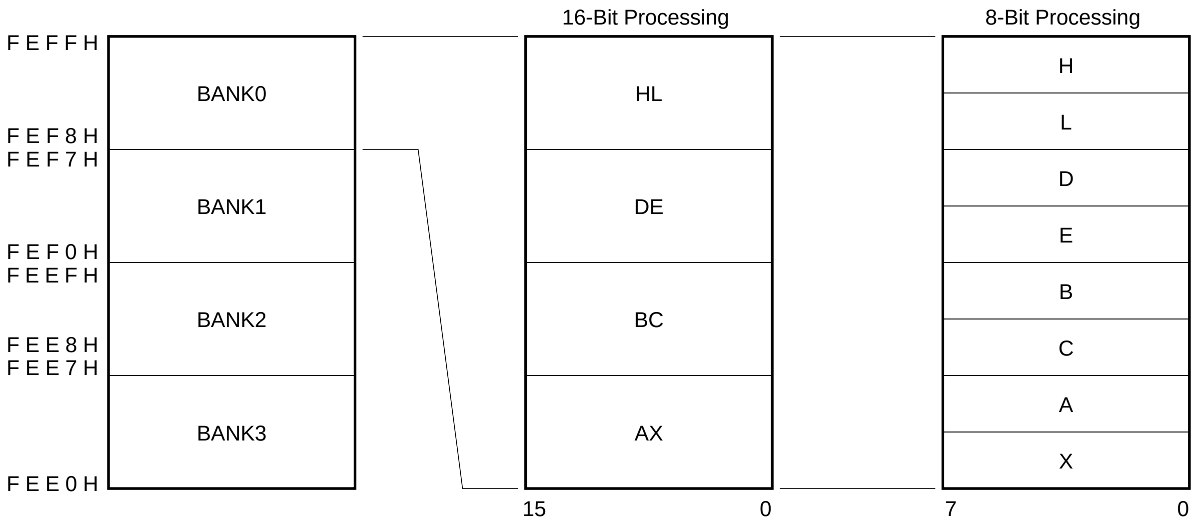
Bank Name	Register		Absolute Address	Bank Name	Register		Absolute Address
	Function Name	Absolute Name			Function Name	Absolute Name	
BANK0	H	R7	FEFFH	BANK2	H	R7	FEEFH
	L	R6	FEFEH		L	R6	FEEEH
	D	R5	FEFDH		D	R5	FEEDH
	E	R4	FEFCH		E	R4	FEECH
	B	R3	FEFBH		B	R3	FEEBH
	C	R2	FEFAH		C	R2	FEEAH
	A	R1	FEF9H		A	R1	FEE9H
	X	R0	FEF8H		X	R0	FEE8H
BANK1	H	R7	FEF7H	BANK3	H	R7	FEE7H
	L	R6	FEF6H		L	R6	FEE6H
	D	R5	FEF5H		D	R5	FEE5H
	E	R4	FEF4H		E	R4	FEE4H
	B	R3	FEF3H		B	R3	FEE3H
	C	R2	FEF2H		C	R2	FEE2H
	A	R1	FEF1H		A	R1	FEE1H
	X	R0	FEF0H		X	R0	FEE0H

Figure 5-11. General Register Configuration

(a) Absolute Name



(b) Function Name



5.2.3 Special function register (SFR)

Unlike a general 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 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**
  Describes 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**
  Describes 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**
  Describes the symbol reserved with assembler for the 16-bit manipulation instruction operand (sfrp). When addressing an address, describe an even address.

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

- **Symbols**
  A symbol indicates an address of the special function register. Symbols are reserved words in RA78K/0 and have been defined by a header file sfrbt.h in CC78K/0. They can be used as instruction operands when RA78K/0, ID78K0, or SD78K/0 is used.
- **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**
  The register can be manipulated in bit units (1, 8, and 16) marked with "○".
  The register cannot be manipulated in bit units marked with "—".
- **When reset**
  Indicates each register status upon $\overline{\text{RESET}}$ input.

Table 5-5. Special Function Register List (1/2)

Address	Special Function Register (SFR) Name	Symbol	R/W	Manipulatable Bit Unit 1 Bit	Manipulatable Bit Unit 8 Bits	Manipulatable Bit Unit 16 Bits	When Reset
FF00H	Port 0	P0	R/W	○	○	—	00H
FF01H	Port 1	P1		○	○	—	
FF02H	Port 2	P2		○	○	—	
FF03H	Port 3	P3		○	○	—	
FF04H	Port 4	P4		○	○	—	Undefined
FF05H	Port 5	P5		○	○	—	
FF06H	Port 6	P6		○	○	—	
FF10H FF11H	16-bit compare register	CR00		—	—	○	
FF12H FF13H	16-bit capture register	CR01	R	—	—	○	
FF14H FF15H	16-bit timer register	TM0		—	—	○	0000H
FF16H	8-bit compare register	CR10	R/W	—	○	—	Undefined
FF17H	8-bit compare register	CR20		—	○	—	
FF18H	8-bit timer register 1	TMS TM1	R	—	○	○	00H
FF19H	8-bit timer register 2	TM2		—	○		
FF1AH	Serial I/O shift register 0	SIO0	R/W	—	○	—	Undefined
FF1BH	Serial I/O shift register 1	SIO1		—	○	—	
FF1FH	A/D conversion result register	ADCR	R	—	○	—	
FF20H	Port mode register 0	PM0	R/W	○	○	—	1FH
FF21H	Port mode register 1	PM1		○	○	—	FFH
FF22H	Port mode register 2	PM2		○	○	—	
FF23H	Port mode register 3	PM3		○	○	—	
FF25H	Port mode register 5	PM5		○	○	—	
FF26H	Port mode register 6	PM6		○	○	—	
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
FF47H	Sampling clock select register	SCS		—	○	—	00H
FF48H	16-bit timer mode control register	TMC0		○	○	—	
FF49H	8-bit timer mode control register	TMC1		○	○	—	
FF4AH	Watch timer mode control register	TMC2		○	○	—	
FF4EH	16-bit timer output control register	TOC0		○	○	—	
FF4FH	8-bit timer output control register	TOC1		○	○	—	

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

Address	Special Function Register (SFR) Name	Symbol		R/W	Manipulatable Bit Unit			When Reset
					1 Bit	8 Bits	16 Bits	
FF60H	Serial operating mode register 0	CSIM0		R/W	○	○	—	00H
FF61H	Serial bus interface control register	SBIC			○	○	—	
FF62H	Slave address register	SVA			—	○	—	Undefined
FF63H	Interrupt timing specify register	SINT			○	○	—	00H
FF68H	Serial operating mode register 1	CSIM1			○	○	—	
FF69H	Automatic data transmit/receive control register	ADTC			○	○	—	
FF6AH	Automatic data transmit/receive address pointer	ADTP			—	○	—	
FF80H	A/D converter mode register	ADM			○	○	—	01H
FF84H	A/D converter input select register	ADIS			—	○	—	00H
FFD0H to FFDFH	External access area[Note 1]				○	○	—	Undefined
FFE0H	Interrupt request flag register 0L	IF0	IF0L		○	○	○	00H
FFE1H	Interrupt request flag register 0H		IF0H		○	○		
FFE4H	Interrupt mask flag register 0L	MK0	MK0L		○	○	○	FFH
FFE5H	Interrupt mask flag register 0H		MK0H		○	○		
FFE8H	Priority specify flag register 0L	PR0	PR0L		○	○	○	
FFE9H	Priority specify flag register 0H		PR0H		○	○		
FFECH	External interrupt mode register	INTM0			—	○	—	00H
FFF0H	Internal memory size switching register	IMS		W	—	○	—	**Note 2**
FFF6H	Key return mode register	KRM		R/W	○	○	—	02H
FFF7H	Pull-up resistor option register	PUO			○	○	—	00H
FFF8H	Memory expansion mode register	MM			○	○	—	10H
FFF9H	Watchdog timer mode register	WDTM			○	○	—	00H
FFFAH	Oscillation stabilization time select register	OSTS			—	○	—	04H
FFFBH	Processor clock control register	PCC			○	○	—	

Notes 1. The external access area cannot be accessed in SFR addressing. Access the area with the direct addressing.

2. The value when reset depends on products.
μPD78011B, 78011BY: 42H, μPD78012B, 78012BY: 44H, μPD78013, 78013Y: C6H,
μPD78014, 78014Y, 78P014, 78P014Y: C8H
If using the mask ROM version, do not set any value other than that when reset to IMS.

5.3 Instruction Address Addressing

An instruction address is determined by program counter (PC) contents. The PC contents are 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, however, the branch destination information is set to the PC and branched by the following addressing (For details of instructions, refer to **78K/0 Series User's Manual, Instructions** (**U12326E**).

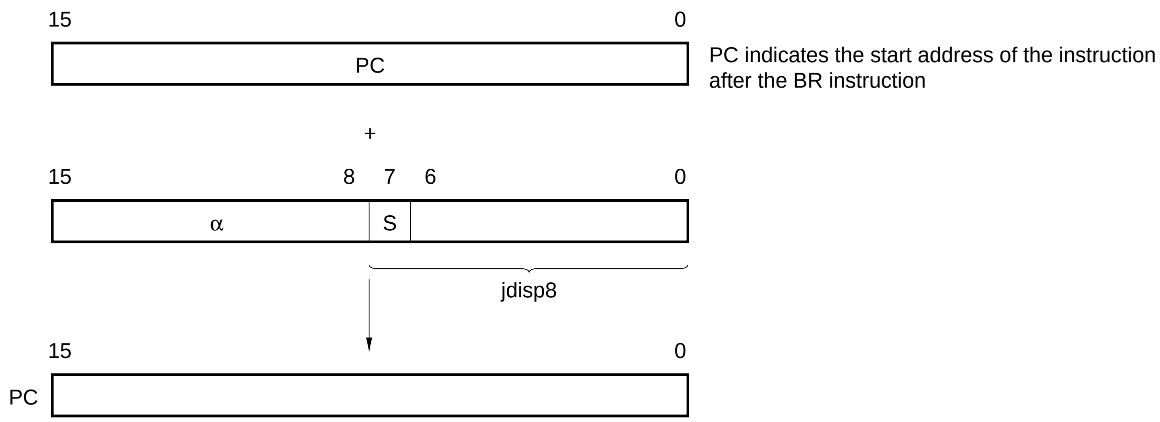
5.3.1 Relative addressing

[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, the range of branch in relative addressing is between –128 and +127 of the start address of the following instruction.

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

[Illustration]



When S = 0, all bits of α are 0.

When S = 1, all bits of α are 1.

5.3.2 Immediate addressing

[Function]

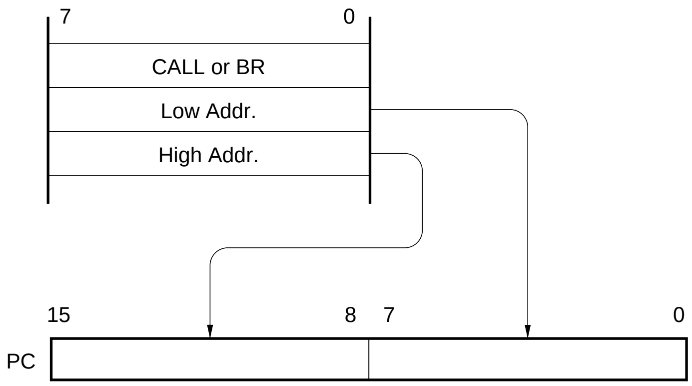
Immediate data in the instruction word is transferred to the program counter (PC) and branched.

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

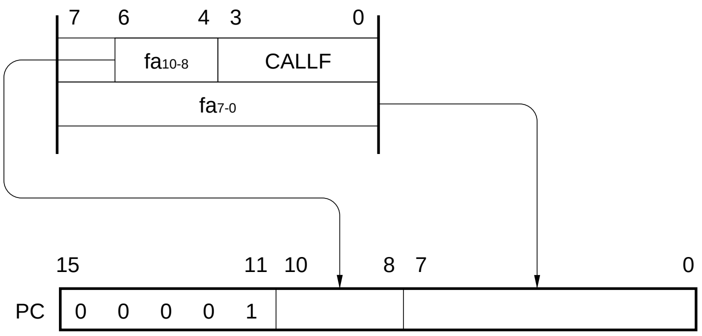
CALL ! addr16 and BR ! addr16 instructions can branch to all the memory spaces. CALLF ! addr11 instruction branches to the area from 0800H to 0FFFH.

[Illustration]

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



In the case of CALLF ! addr11 instruction



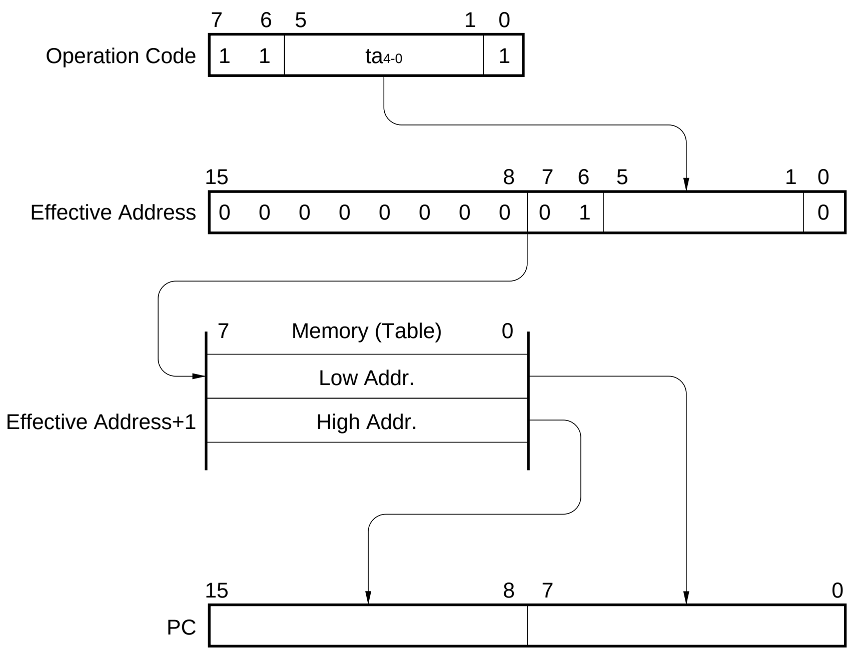
5.3.3 Table indirect addressing

[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 can refer to the address stored in the memory table 40H to 7FH and branch to all the memory spaces.

[Illustration]



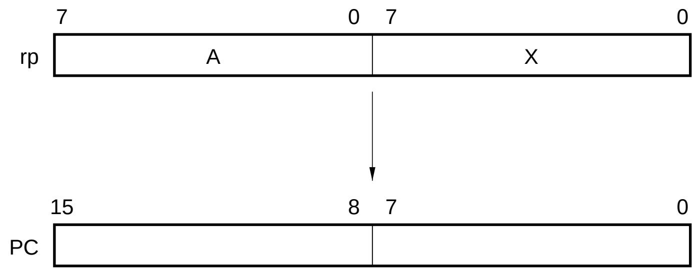
5.3.4 Register addressing

[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

5.4.1 Data memory addressing

Addressing is a method to specify the instruction address to be executed next and the register and memory address to be manipulated when instructions are executed. The instruction address to be executed next is addressed by the program counter (PC) (for details, refer to **5.3 Instruction Address Addressing**).

For the addressing of the memory to be manipulated when instructions are executed, the μPD78014 and 78014Y Subseries are provided with several addressing modes which take account of optimum manipulability. In particular, specific types of addressing can be used which match the functions of the special function registers (SFRs), general registers, etc. Data memory addressing is shown in Figures 5-12 to 5-16.

Figure 5-12. Data Memory Addressing (μPD78011B, 78011BY)

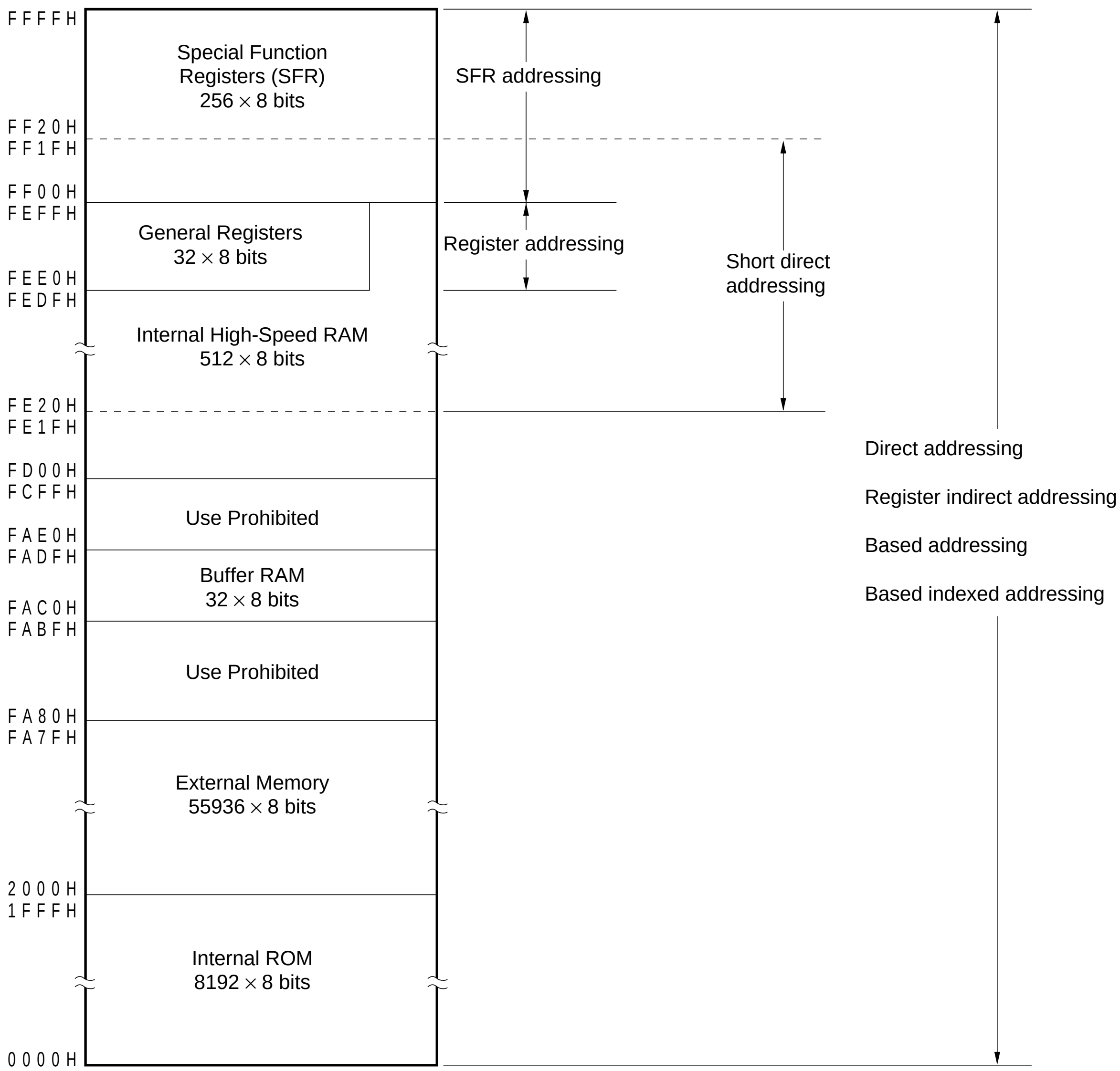


Figure 5-13. Data Memory Addressing (μPD78012B, 78012BY)

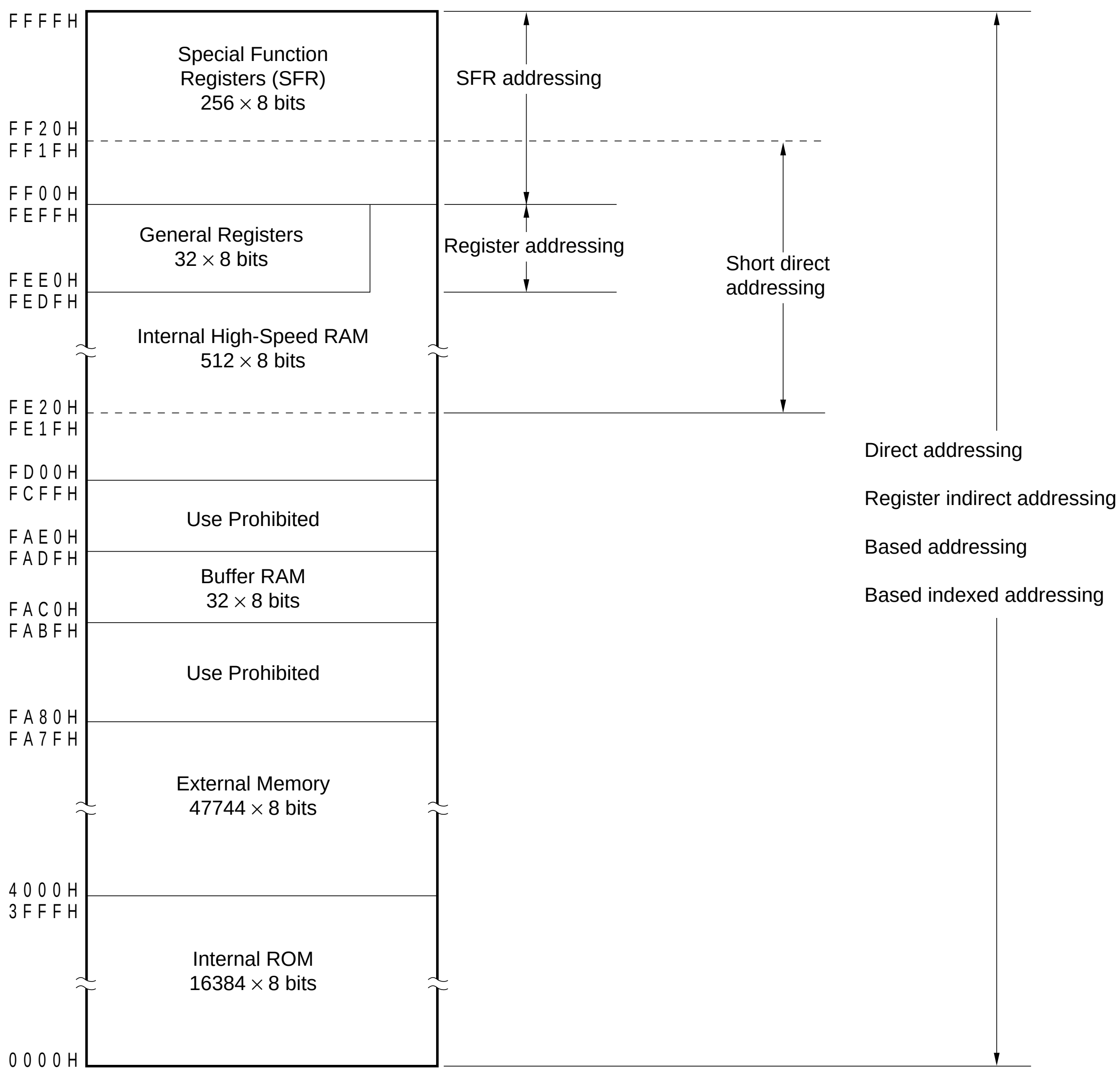


Figure 5-14. Data Memory Addressing (μPD78013, 78013Y)

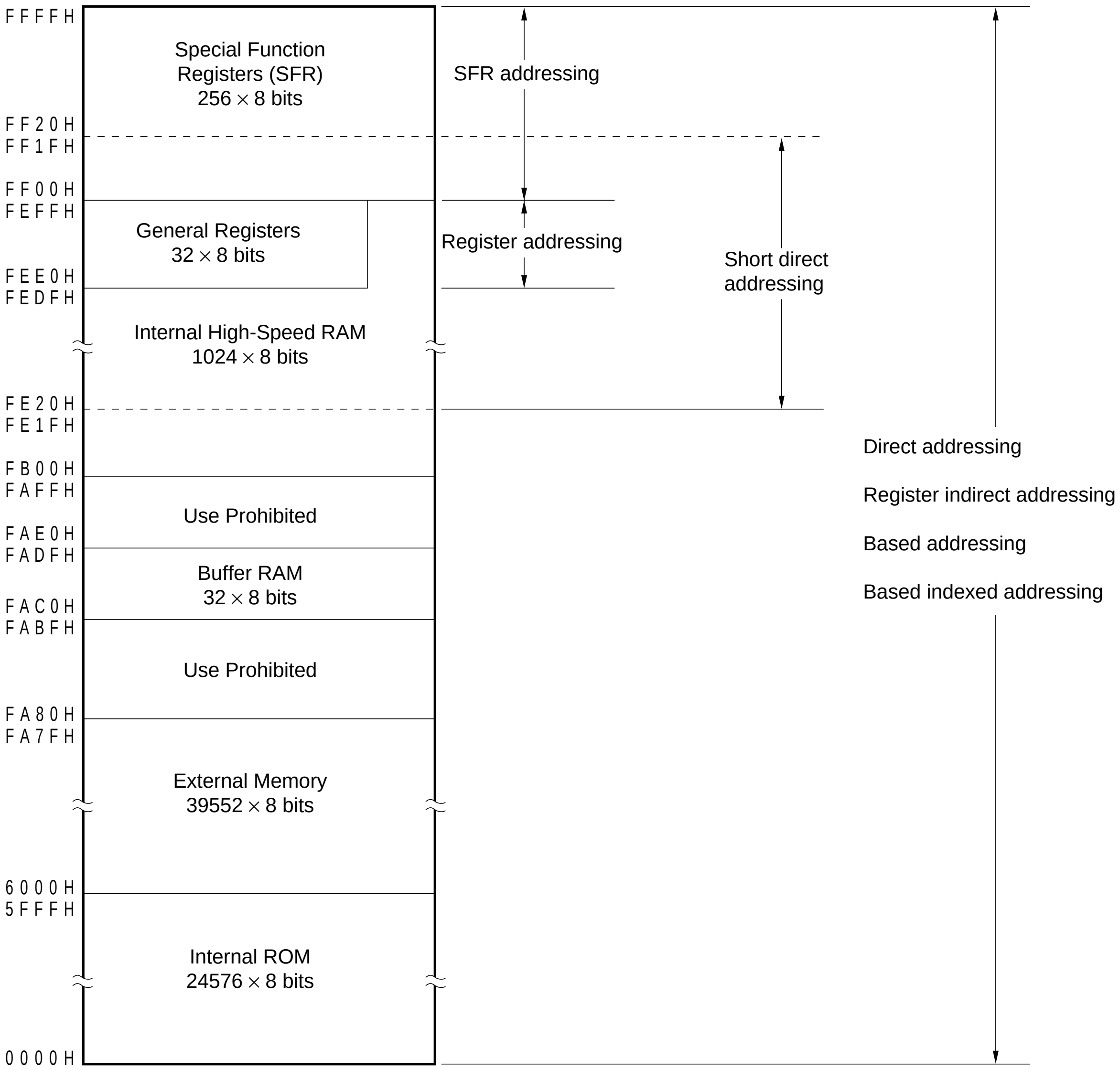


Figure 5-15. Data Memory Addressing (μPD78014, 78014Y)

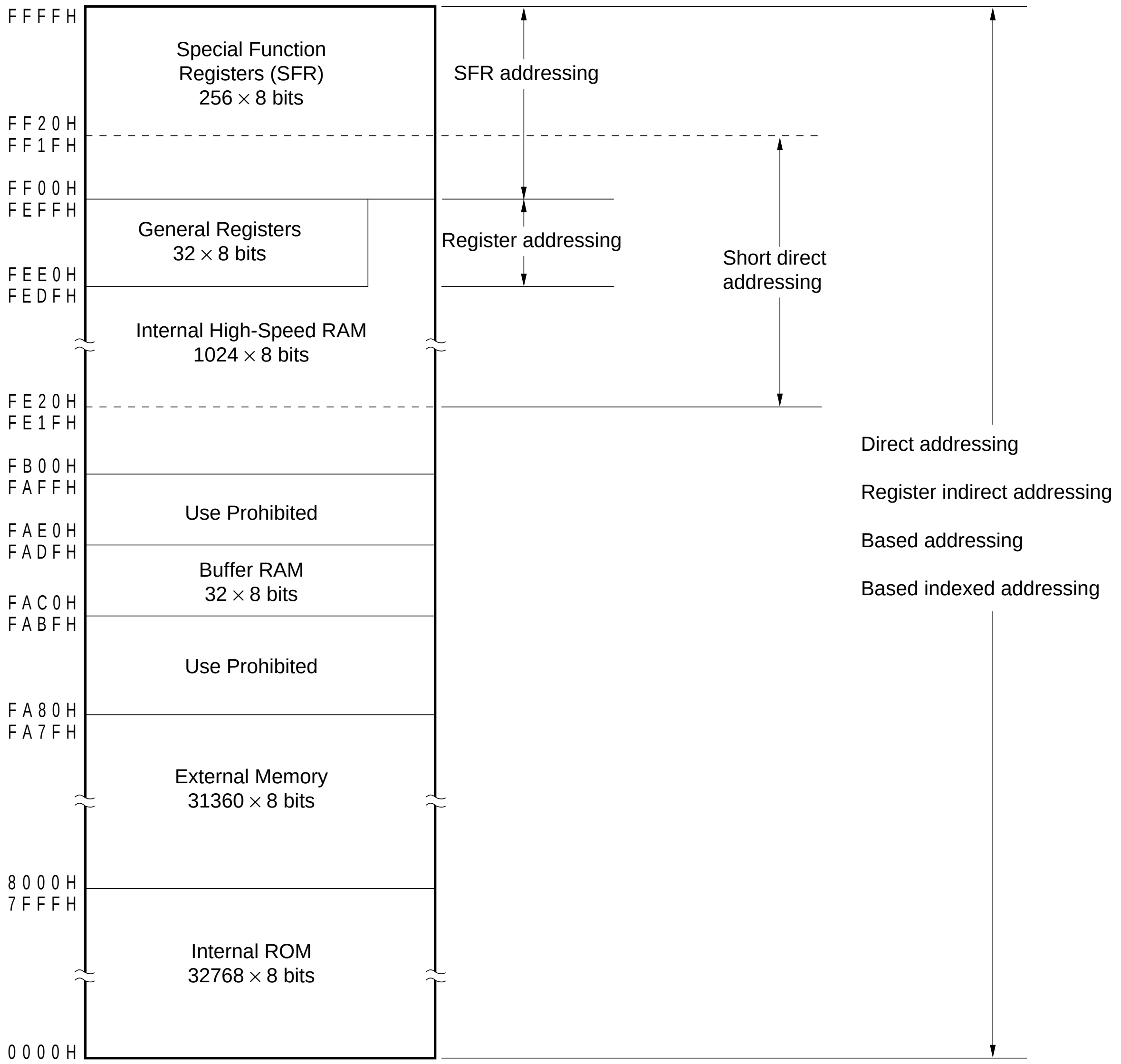
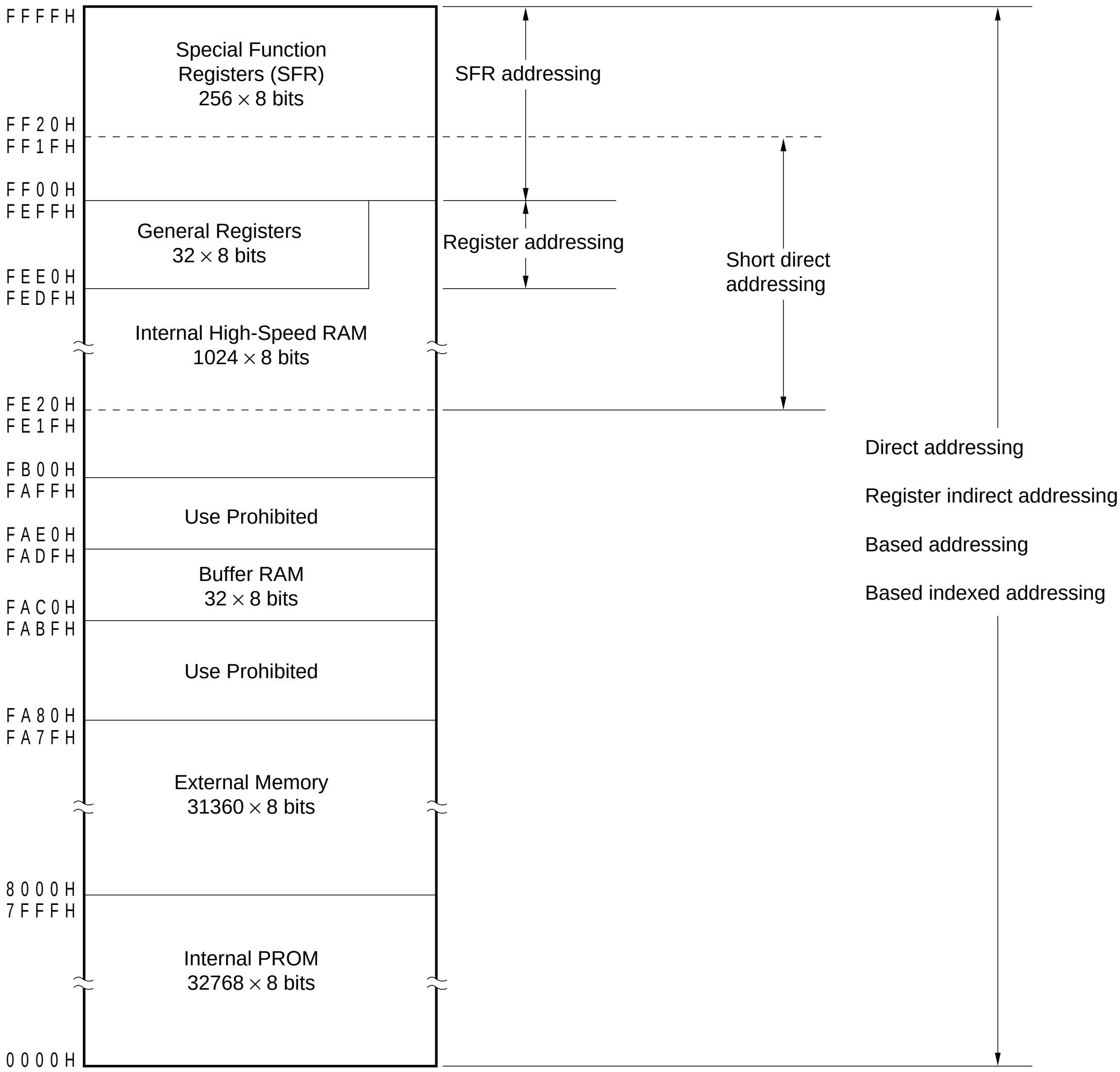


Figure 5-16. Data Memory Addressing (μPD78P014, 78P014Y)



5.4.2 Implied addressing

[Function]

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

Of the μPD78014 and 78014Y 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 subject to decimal adjustment
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 x 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.

5.4.3 Register addressing

[Function]

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

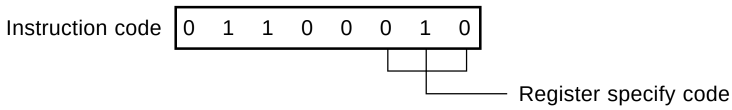
[Operand format]

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

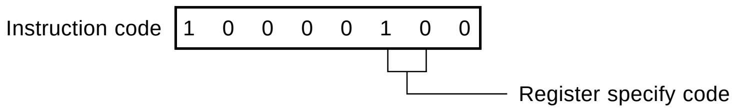
'r' and 'rp' can be described with function names (X, A, C, B, E, D, L, H, AX, BC, DE and HL) as well as absolute names (R0 to R7 and RP0 to RP3).

[Description example]

MOV A, C; when selecting C register as r



INCW DE; when selecting DE register pair as rp



5.4.4 Direct addressing

[Function]

The memory indicated by immediate data in an instruction word is directly addressed.

[Operand format]

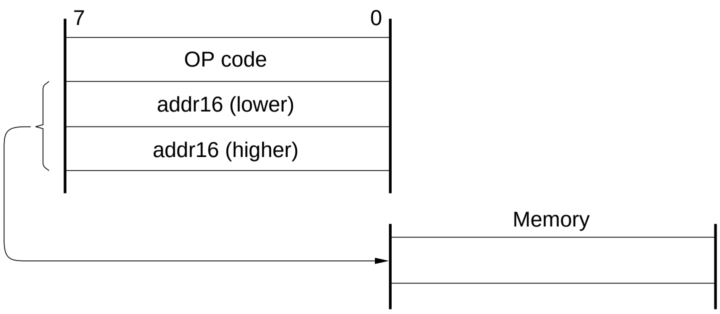
Identifier	Description
addr16	Label or 16-bit immediate data

[Description example]

MOV A, ! 0FE00H; when setting ! addr16 to FE00H

Instruction 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	FEH

★ **[Illustration]**



5.4.5 Short direct addressing

[Function]

The memory to be manipulated in the fixed space is directly addressed with 8-bit data in an instruction word.

The fixed space where this addressing is applied to is the 256-byte space 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) where short direct addressing is applied is a part of all SFR areas. In this area, ports which are frequently accessed in a program and a compare register of the timer/event counter and a capture register of the timer/event counter are mapped 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 **[Illustration]** on next page.

[Operand format]

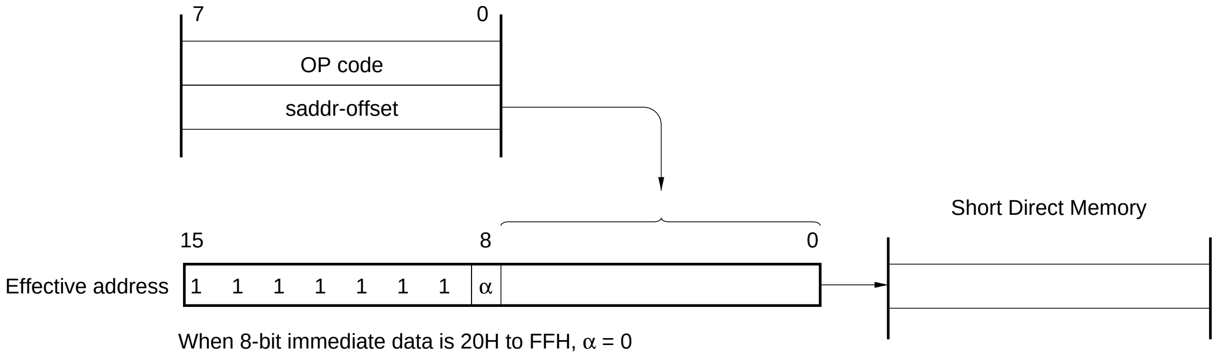
Identifier	Description
saddr	Label or FE20H to FF1FH immediate data
saddrp	Label or FE20H to FF1FH immediate data (even address only)

[Description example]

MOV 0FE30H, #50H; when setting saddr to FE30H and immediate data to 50H

Instruction 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

5.4.6 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 also 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)

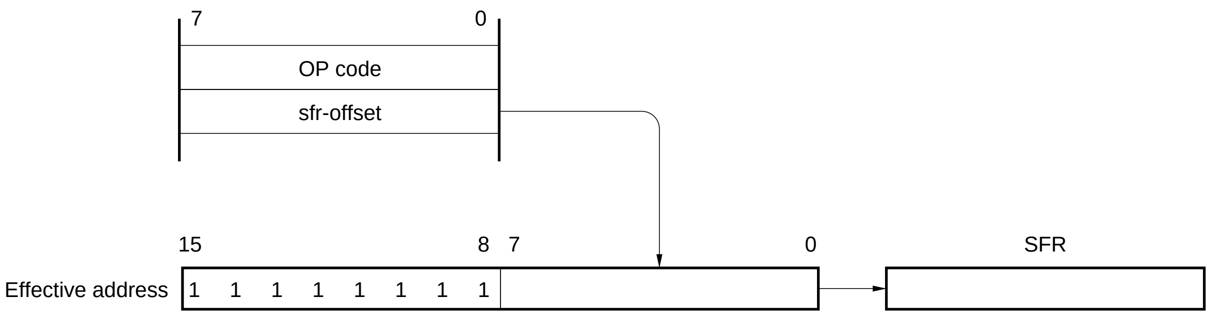
[Description example]

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

Instruction code | 1 1 1 1 0 1 1 0 | OP code

| 0 0 1 0 0 0 0 0 | 20H (sfr-offset)

[Illustration]



5.4.7 Register indirect addressing

[Function]

The memory is addressed with the contents of the register pair specified as an operand. The register pair to be accessed is specified with the register bank select flag (RBS0 and RBS1) and the register pair specify code in the instruction code. This addressing can be carried out for all the memory spaces.

[Operand format]

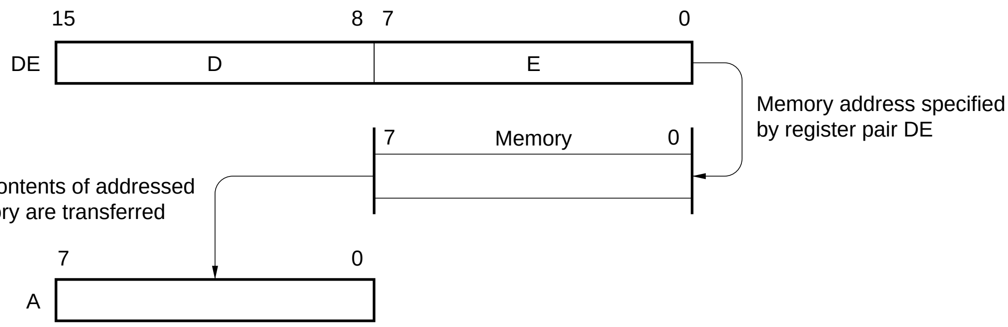
Identifier	Description
—	[DE], [HL]

[Description example]

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

Instruction code	1	0	0	0	0	1	0	1

[Illustration]



5.4.8 Based addressing

[Function]

8-bit immediate data is added to the contents of the base register, that is, the HL register pair, and the sum is used to address the memory. The HL register pair to be accessed is in the register bank specified with the register bank select flags (RBS0 and RBS1). The offset data as a positive number is expanded to 16 bits to be added. 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

Instruction code	1	0	1	0	1	1	1	0
	0	0	0	1	0	0	0	0

5.4.9 Based indexed addressing

[Function]

The B or C register contents specified in an instruction are added to the contents of the base register, that is, the HL register pair, and the sum is used to address the memory. The HL, B, and C registers to be accessed are registers in the register bank specified with the register bank select flag (RBS0 and RBS1). The contents of the B or C register as a positive number are expanded to 16 bits to be added. 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]

Instruction code | 1 0 1 0 1 0 1 1 |

5.4.10 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

Instruction code | 1 0 1 1 0 1 0 1 |

CHAPTER 6   PORT FUNCTIONS

6.1 Port Functions

The μPD78014 and 78014Y Subseries each incorporate two input ports and fifty-one input/output ports. Figure 6-1 shows the port types. 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 input/output pins.

Figure 6-1. Port Types

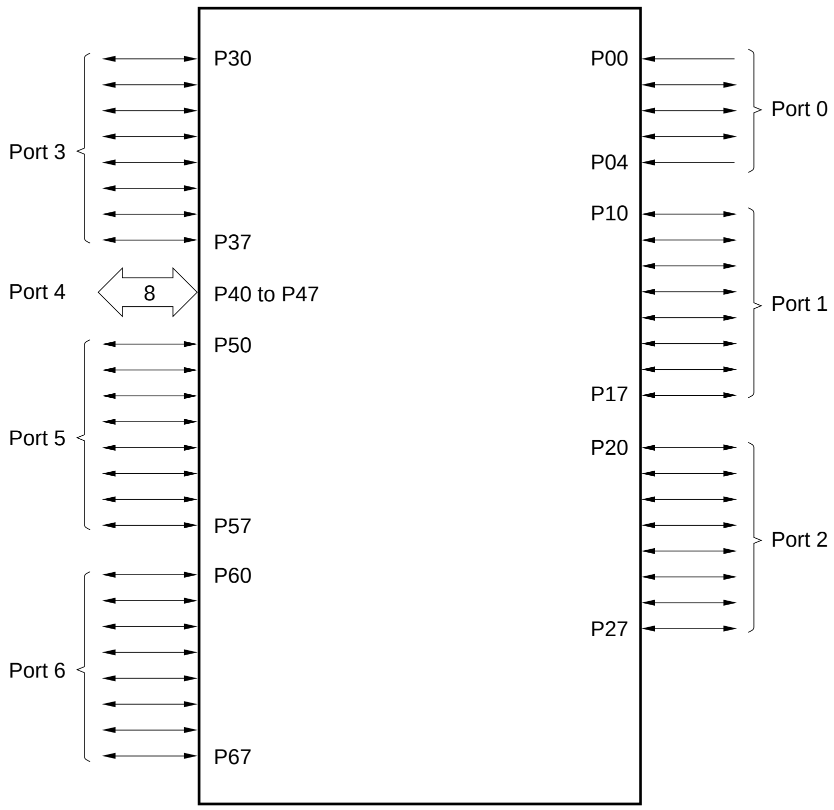


Table 6-1. Port Functions (μPD78014 Subseries) (1/2)

Pin Name	Function		Alternate Function
P00	Port 0. 5-bit input/output port.	Input only.	INTP0/TI0
P01		Input/output specifiable bit-wise. If used as an input port, on-chip pull-up resistor can be enabled by software.	INTP1
P02			INTP2
P03			INTP3
P04		Input only.	XT1
P10 to P17	Port 1. 8-bit input/output port. Input/output specifiable bit-wise. If used as an input port, on-chip pull-up resistor is enabled by software.		ANI0 to ANI7
P20	Port 2. 8-bit input/output port. Input/output specifiable bit-wise. If used as an input port, on-chip pull-up resistor is enabled by software.		SI1
P21			SO1
P22			$\overline{\text{SCK1}}$
P23			STB
P24			BUSY
P25			SI0/SB0
P26			SO0/SB1
P27			$\overline{\text{SCK0}}$
P30	Port 3. 8-bit input/output port. Input/output specifiable bit-wise. If used as an input port, on-chip pull-up resistor is enabled by software.		TO0
P31			TO1
P32			TO2
P33			TI1
P34			TI2
P35			PCL
P36			BUZ
P37			—
P40 to P47	Port 4. 8-bit input/output port. Input/output can be specified in 8-bit units. When used as an input port, on-chip pull-up resistor is enabled by software. Test input flag (KRIF) is set to 1 by falling edge detection.		AD0 to AD7
P50 to P57	Port 5. 8-bit input/output port. LED can be driven directly. Input/output specifiable bit-wise. When used as an input port, on-chip pull-up resistor is enabled by software.		A8 to A15

Table 6-1. Port Functions (μPD78014 Subseries) (2/2)

Pin Name	Function		Alternate Function
P60	Port 6. 8-bit input/output port. Input/output specifiable bit-wise.	N-ch open-drain input/output port. On-chip pull-up resistor can be specified by mask option only for mask ROM versions. LED can be driven directly.	—
P61			
P62			
P63			
P64		When used as an input port, on-chip pull-up resistor is enabled by software.	$\overline{\text{RD}}$
P65			$\overline{\text{WR}}$
P66			$\overline{\text{WAIT}}$
P67			ASTB

Table 6-2. Port Functions (μPD78014Y Subseries) (1/2)

Pin Name	Function		Alternate Function
P00	Port 0.	Input only.	INTP0/TI0
P01	5-bit input/output port.	Input/output specifiable bit-wise. If used as an input port, on-chip pull-up resistor can be enabled by software.	INTP1
P02			INTP2
P03			INTP3
P04		Input only.	XT1
P10 to P17	Port 1. 8-bit input/output port. Input/output specifiable bit-wise. If used as an input port, on-chip pull-up resistor is enabled by software.		ANI0 to ANI7
P20	Port 2. 8-bit input/output port. Input/output specifiable bit-wise. If used as an input port, on-chip pull-up resistor is enabled by software.		SI1
P21			SO1
P22			$\overline{\text{SCK1}}$
P23			STB
P24			BUSY
P25			SI0/SB0/SDA0
P26			SO0/SB1/SDA1
P27			$\overline{\text{SCK0}}$/SCL
P30	Port 3. 8-bit input/output port. Input/output specifiable bit-wise. If used as an input port, on-chip pull-up resistor is enabled by software.		TO0
P31			TO1
P32			TO2
P33			TI1
P34			TI2
P35			PCL
P36			BUZ
P37			—
P40 to P47	Port 4. 8-bit input/output port. Input/output can be specified in 8-bit units. When used as an input port, on-chip pull-up resistor is enabled by software. Test input flag (KRIF) is set to 1 by falling edge detection.		AD0 to AD7
P50 to P57	Port 5. 8-bit input/output port. LED can be driven directly. Input/output specifiable bit-wise. When used as an input port, on-chip pull-up resistor is enabled by software.		A8 to A15

Table 6-2. Port Functions (μPD78014Y Subseries) (2/2)

Pin Name	Function		Alternate Function
P60	Port 6. 8-bit input/output port. Input/output specifiable bit-wise.	N-ch open-drain input/output port. On-chip pull-up resistor can be specified by mask option only for mask ROM versions. LED can be driven directly.	—
P61			
P62			
P63			
P64		When used as an input port, on-chip pull-up resistor is enabled by software.	$\overline{RD}$
P65			$\overline{WR}$
P66			$\overline{WAIT}$
P67			ASTB

6.2 Port Block Diagram

A port consists of the following hardware.

Table 6-3. Port Block Diagram

Item	Configuration	
Control register	Port mode register	(PMm: m = 0, 1, 2, 3, 5, 6)
	Pull-up resistor option register	(PUO)
	Memory expansion mode register	(MM)Note
	Key return mode register	(KRM)
Port	Total: 53 ports (2 inputs, 51 inputs/outputs)	
Pull-up resistor	• Mask ROM versions	
	Total: 51 (software control: 47, mask option control: 4)	
	• μPD78P014, 78P014Y Total: 47	

Note Memory expansion mode registers specify input/output of Port 4.

6.2.1 Port 0

Port 0 is a 5-bit input/output port with output latch. P01 to P03 pins can be set to the input mode/output mode bit-wise with the port mode register 0 (PM0). P00 and P04 pins are input-only ports. When P01 to P03 pins are used as input ports, a pull-up resistor can be connected to them in 3-bit units with an on-chip pull-up resistor option register (PUO).

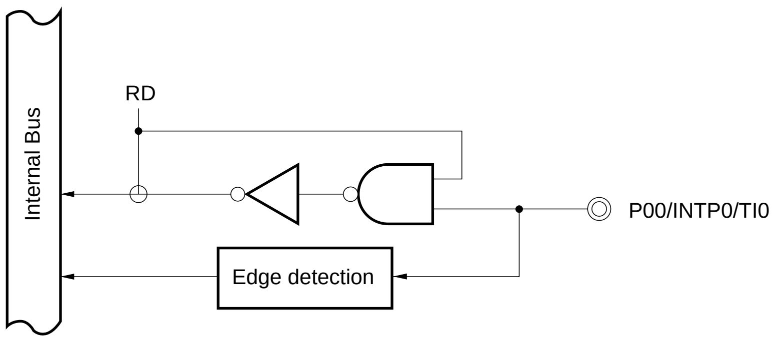
Alternate functions include external interrupt request input, external count clock input to the timer and crystal connection for subsystem clock oscillation.

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

Figures 6-2 to 6-4 show block diagrams of port 0.

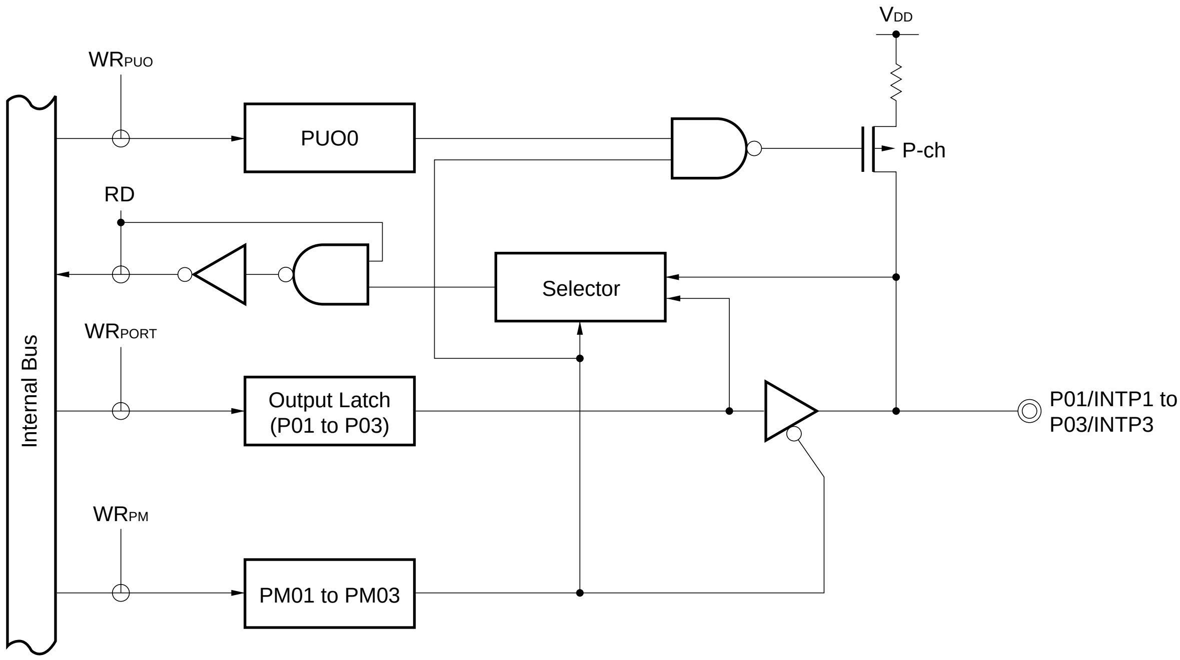
Caution Because port 0 is also used for external interrupt request input, when the port function output mode is specified and the output level is changed, the interrupt request flag is set. Thus, when the output mode is used, set the interrupt mask flag to 1.

Figure 6-2. P00 Block Diagram



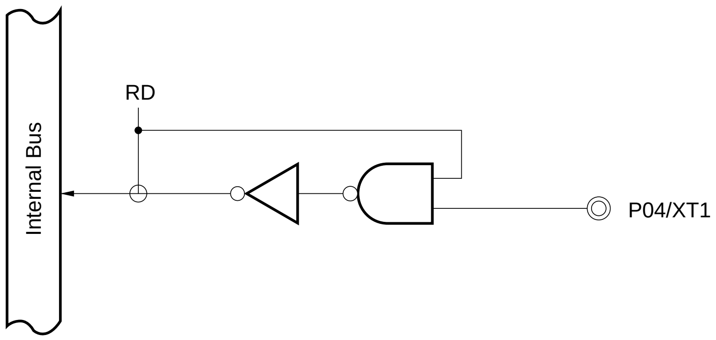
RD : Port 0 read signal

Figure 6-3. P01 to P03 Block Diagrams



PUO : Pull-up resistor option register
PM : Port mode register
RD : Port 0 read signal
WR : Port 0 write signal

Figure 6-4. P04 Block Diagram



RD : Port 0 read signal

6.2.2 Port 1

Port 1 is an 8-bit input/output port with output latch. P10 to P17 pins can be set to the input mode/output mode bit-wise with a port mode register 1 (PM1). When P10 to P17 pins are used as input ports, an on-chip pull-up resistor can be connected to them in 8-bit units with a pull-up resistor option register (PUO).

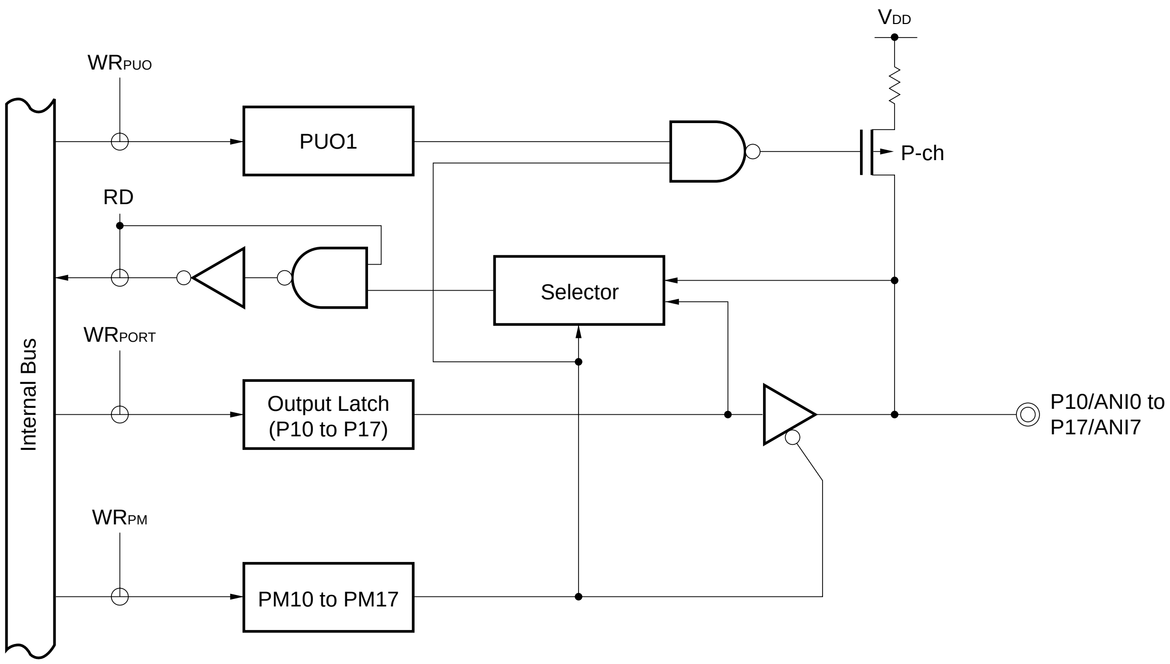
Alternate functions include an A/D converter analog input.

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

Figure 6-5 shows a block diagram of port 1.

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

Figure 6-5. P10 to P17 Block Diagrams



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 (μPD78014 Subseries)

Port 2 is an 8-bit input/output port with output latch. P20 to P27 pins can be set to the input mode/output mode bit-wise with the port mode register 2 (PM2). When P20 to P27 pins are used as input ports, a pull-up resistor can be connected to them in 8-bit units with an on-chip pull-up resistor option register (PUO).

Alternate functions include serial interface data input/output, clock input/output, automatic transmit/receive busy input, and strobe output.

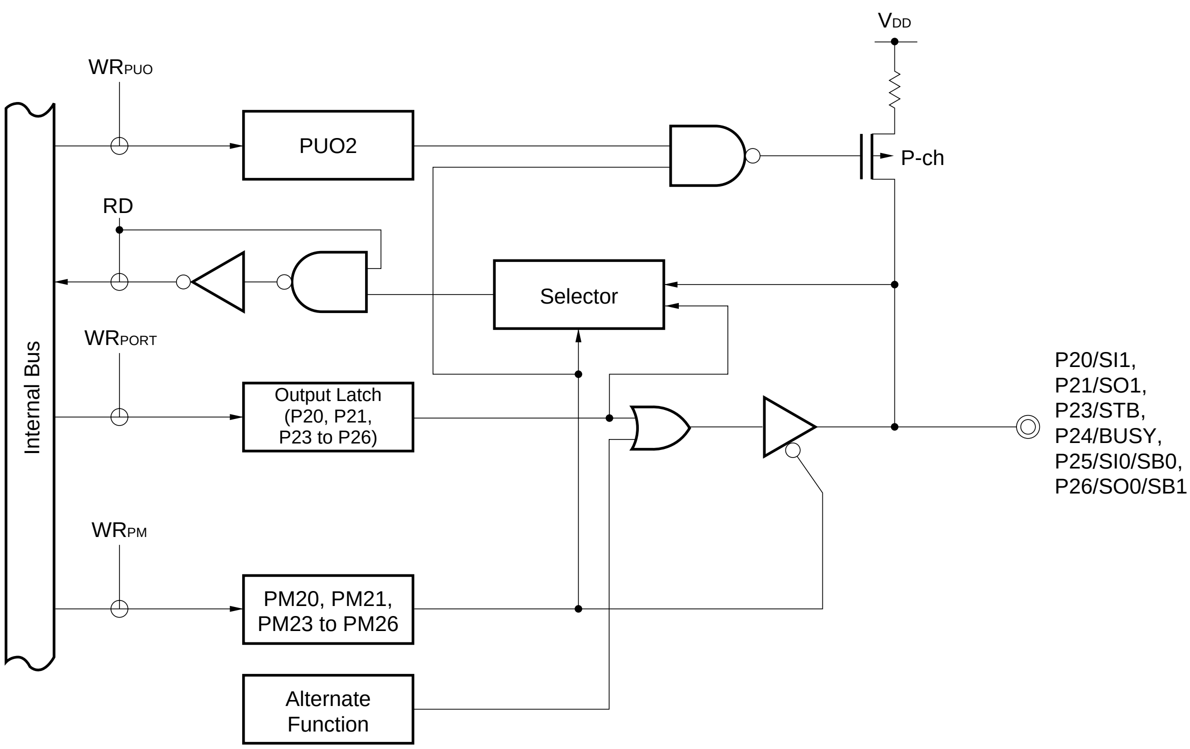
$\overline{\text{RESET}}$ input sets port 2 to input mode.

Figures 6-6 and 6-7 show a block diagram of port 2.

Cautions 1. If used as alternate function pin, set the input/output latch according to the functions. Refer to Figure 15-5 Serial Operating Mode Register 0 Format and Figure 17-3 Serial Operating Mode Register 1 Format for setting.

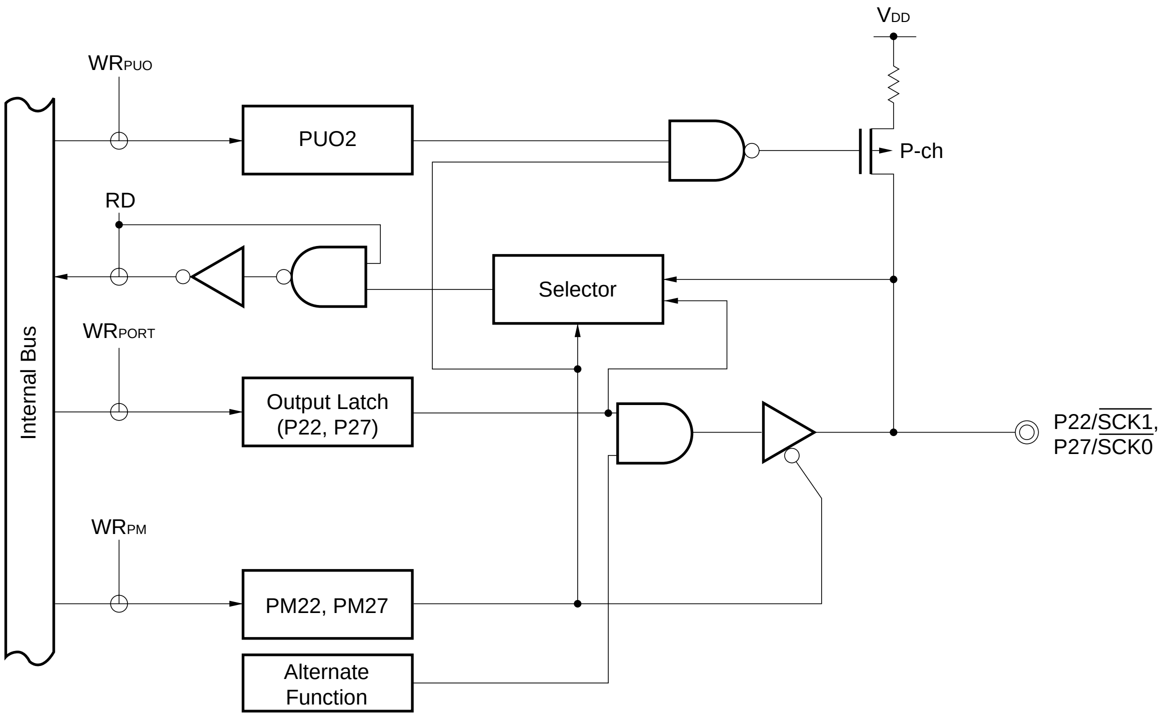
2. When the status of pins is read in the SBI mode, set PM2n of the PM2 to 1 (n = 5 or 6) (refer to 15.4.3 SBI mode operation (10) Distinction method of slave busy state).

★ **Figure 6-6. P20, P21, P23 to P26 Block Diagrams (μPD78014 Subseries)**



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

★ **Figure 6-7. P22 and P27 Block Diagrams (μPD78014 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 (μPD78014Y Subseries)

Port 2 is an 8-bit input/output port with output latch. P20 to P27 pins can be set to the input mode/output mode bit-wise with the port mode register 2 (PM2). When P20 to P27 pins are used as input ports, an on-chip pull-up resistor can be connected to them in 8-bit units with a pull-up resistor option register (PUO).

Alternate functions include serial interface data input/output, clock input/output, automatic transmit/receive busy input, and strobe output.

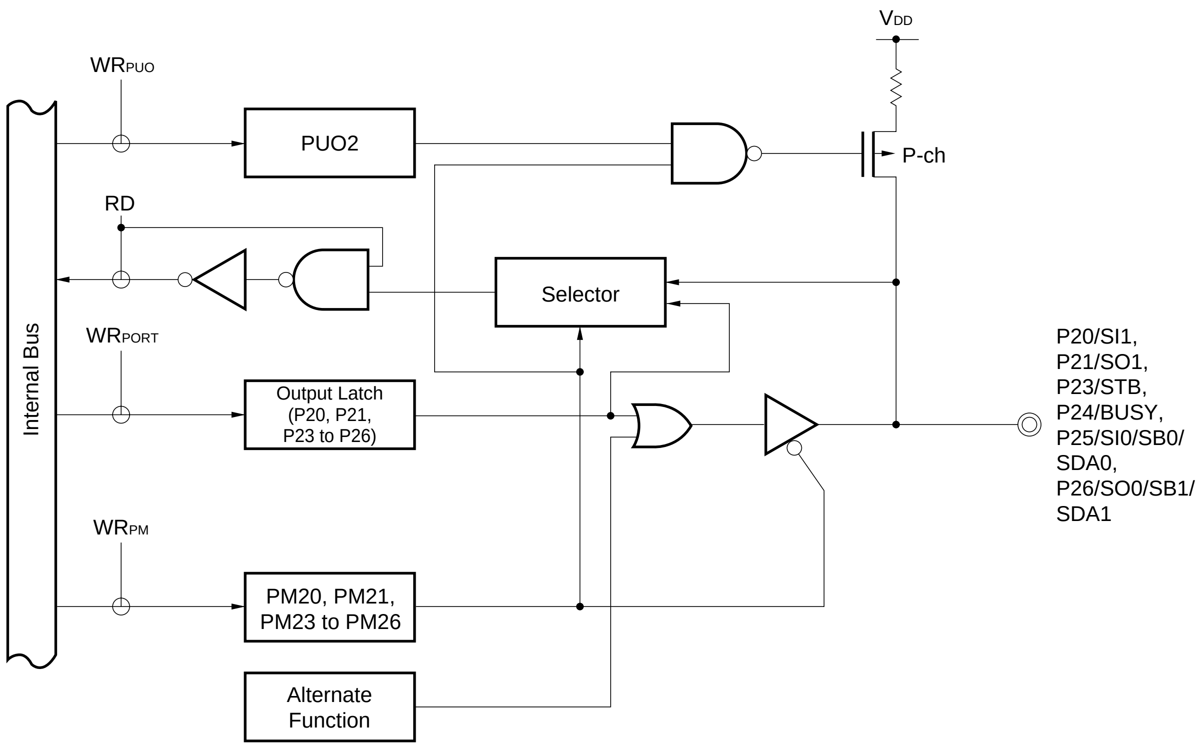
$\overline{\text{RESET}}$ input sets port 2 to input mode.

Figures 6-8 and 6-9 show a block diagram of port 2.

Cautions 1. If used as alternate function pin, set the input/output latch according to the functions. Refer to Figure 16-6 Serial Operating Mode Register 0 Format and Figure 17-3 Serial Operating Mode Register 1 Format for setting.

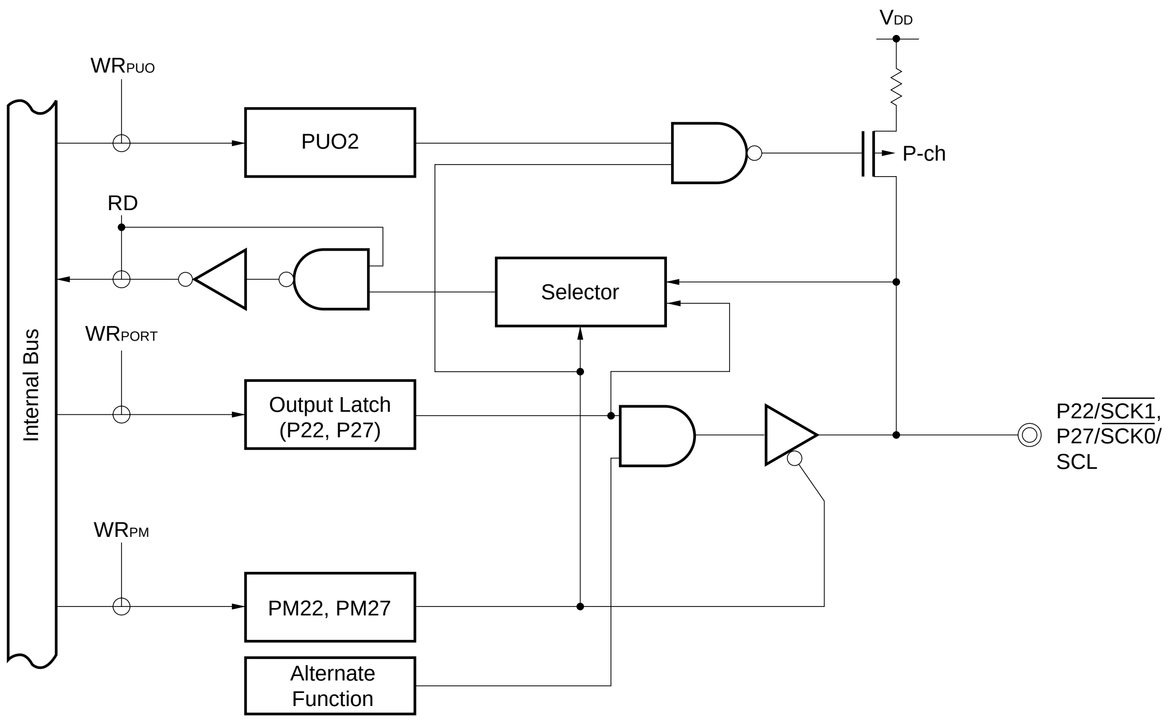
2. When the status of pins is read in the SBI mode, set PM2n of the PM2 to 1 (n = 5 or 6) (refer to 16.4.3 SBI mode operation (10) Distinction method of slave busy state).

★ **Figure 6-8. P20, P21, P23 to P26 Block Diagrams (μPD78014Y Subseries)**



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

★ **Figure 6-9. P22 and P27 Block Diagrams (μPD78014Y 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

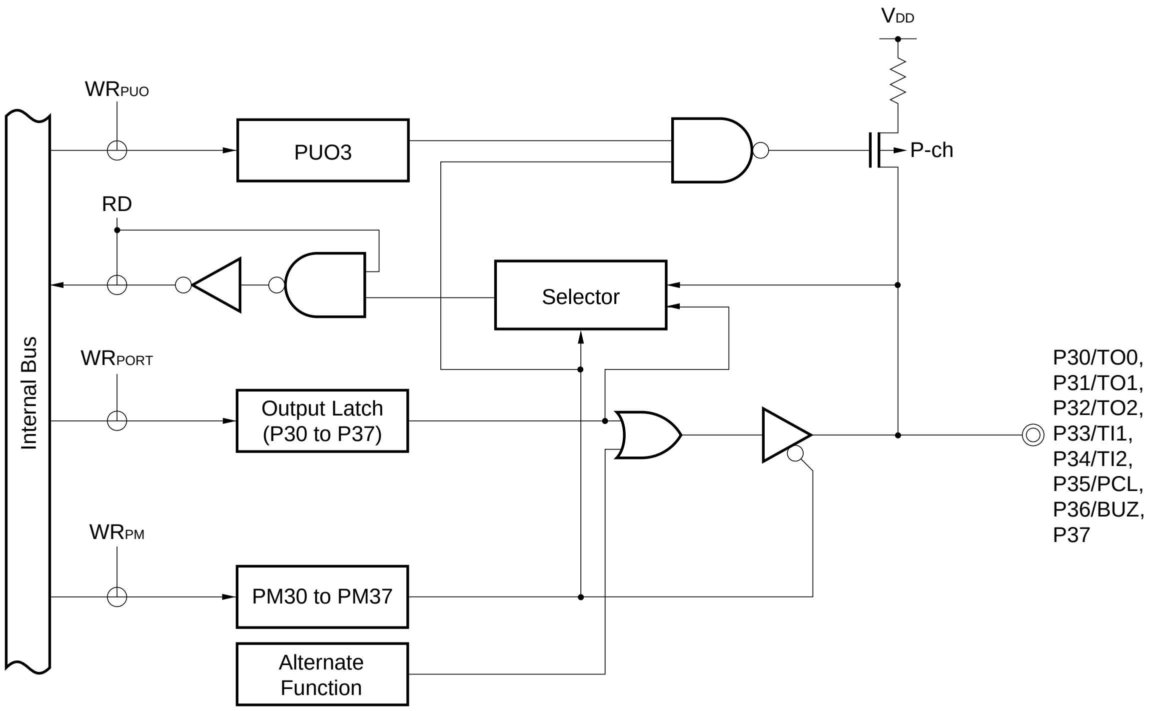
Port 3 is an 8-bit input/output port with output latch. P30 to P37 pins can be set to the input mode/output mode bit-wise with the port mode register (PM3). When P30 to P37 pins are used as input ports, an on-chip pull-up resistor can be connected to them in 8-bit units with a pull-up resistor option register (PUO).

Alternate functions include timer input/output, clock output and buzzer output.

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

Figure 6-10 shows a block diagram of port 3.

★ **Figure 6-10. P30 to P37 Block Diagrams**



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

6.2.6 Port 4

Port 4 is an 8-bit input/output port with output latch. P40 to P47 pins can be set to the input mode/output mode in 8-bit units with the memory expansion mode register (MM). When P40 to P47 pins are used as input ports, a pull-up resistor can be connected to them in 8-bit units with an on-chip pull-up resistor option register (PUO).

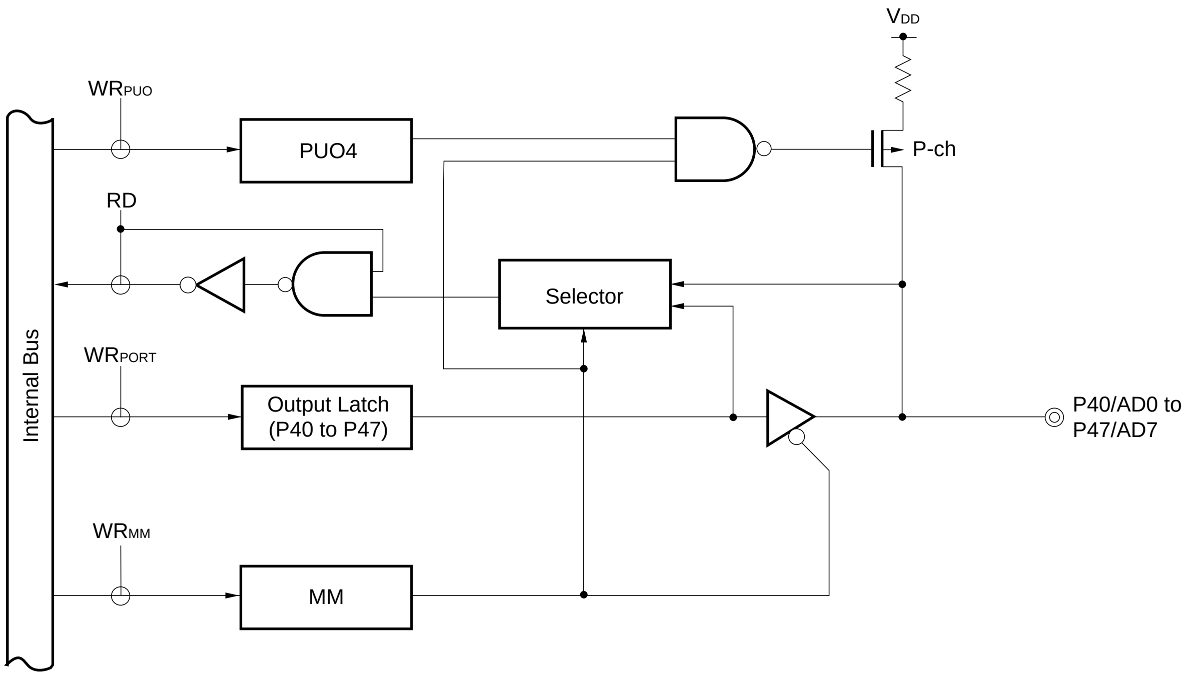
Test input flag (KRIF) is set to 1 by falling edge detection.

Alternate functions include address/data bus in external memory expansion mode.

$\overline{\text{RESET}}$ input sets port 4 to input mode.

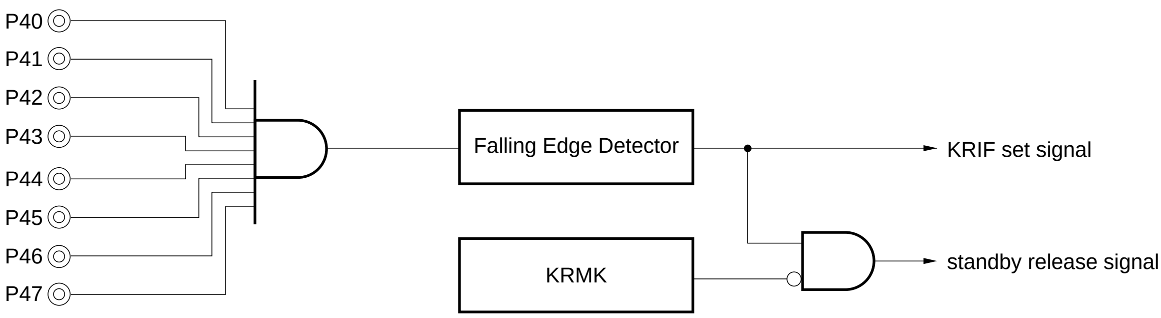
Figure 6-11 shows a block diagram of port 4. Figure 6-12 shows a block diagram of the falling edge detector.

Figure 6-11. P40 to P47 Block Diagrams



PUO : Pull-up resistor option register
MM : Memory expansion mode register
RD : Port 4 read signal
WR : Port 4 write signal

Figure 6-12. Block Diagram of Falling Edge Detector



KRIF : Test input flag
KRMK : Test mask flag

6.2.7 Port 5

Port 5 is an 8-bit input/output port with output latch. P50 to P57 pins can be set to the input mode/output mode bit-wise with the port mode register 5 (PM5). When P50 to P57 pins are used as input ports, an on-chip pull-up resistor can be connected to them in 8-bit units with a pull-up resistor option register (PUO).

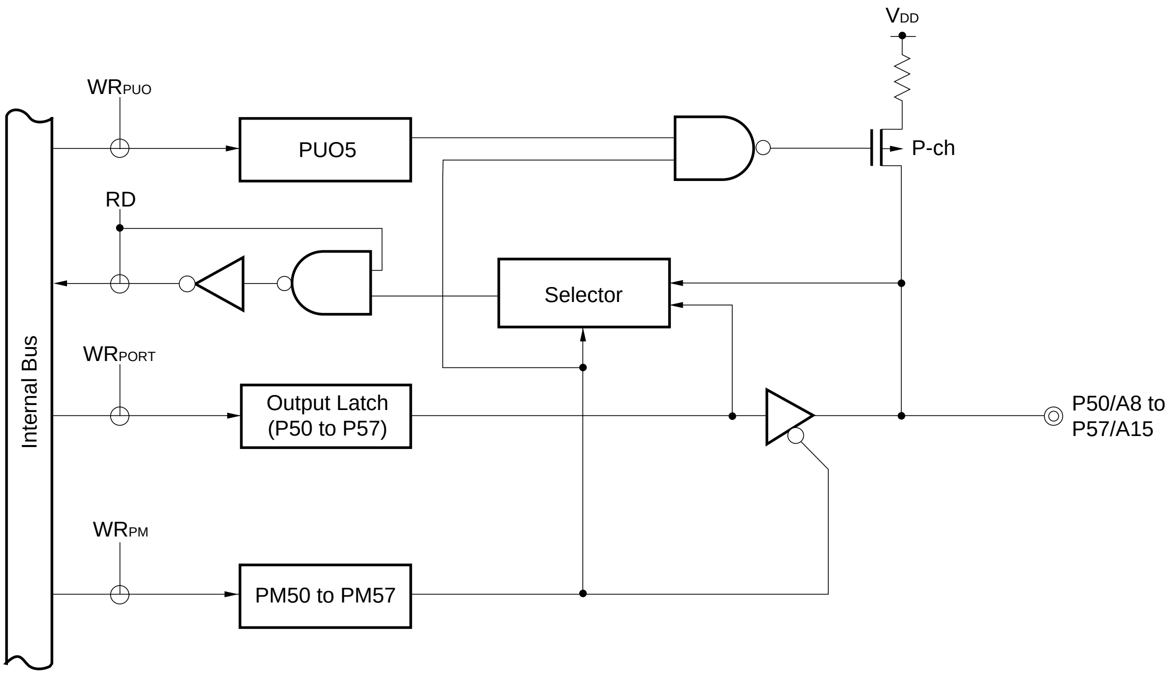
Port 5 can drive LEDs directly.

Alternate functions include address bus in external memory expansion mode.

$\overline{\text{RESET}}$ input sets port 5 to input mode.

Figure 6-13 shows a block diagram of port 5.

Figure 6-13. P50 to P57 Block Diagrams



PUO : Pull-up resistor option register
PM : Port mode register
RD : Port 5 read signal
WR : Port 5 write signal

6.2.8 Port 6

Port 6 is an 8-bit input/output port with output latches. P60 to P67 pins can be set to either input mode or output mode in 1-bit units with port mode register 6 (PM6).

This port has the following functions related to the pull-up resistor. These functions deffer depending on the upper 4 bits or the lower 4 bits of the port, and mask ROM versions or PROM versions.

Table 6-4. Pull-Up Resistors for Port 6

	Upper 4 bits (P64 to P67 pins)	Lower 4 bits (P60 to P63 pins)
Mask version	The on-chip pull-up resistor can be connected to the P64 to P67 pins in 4-bit units with PU06.	The pull-up resistor can be connected bit-wise by mask option.
PROM version		The pull-up resistor is not contained.

PUO6: Bit 6 of the pull-up resistor option register

P60 to P63 pins can drive LEDs directly.

The alternate function of the P60 to P63 pins is control signal output in external memory expansion mode.

$\overline{\text{RESET}}$ input sets port 6 to input mode.

Tables 6-14 and 6-15 show the block diagrams of port 6.

Cautions 1. When external wait is not used in external memory expansion mode, P66 can be used as an input/output port.

2. The value of low-level input leakage current at the P60 to P63 pins depends on the following conditions.

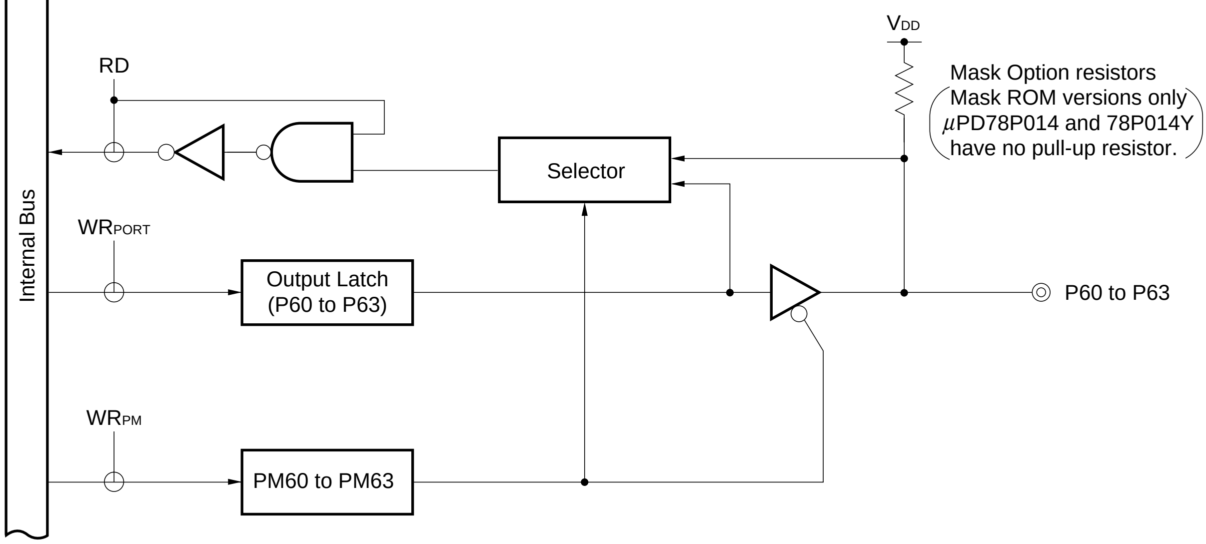
[Mask ROM version]

- **When the pull-up resistor is contained: always –3 μA (max.)**
- **When the pull-up resistor is not contained:**
  - **Within 3 clocks after a read is executed to Port 6 (P6) or Port mode register 6 (PM6) (when no wait cycles are inserted): –200 μA (max.)**
  - **In other cases: –3 μA (max.)**

[PROM version]

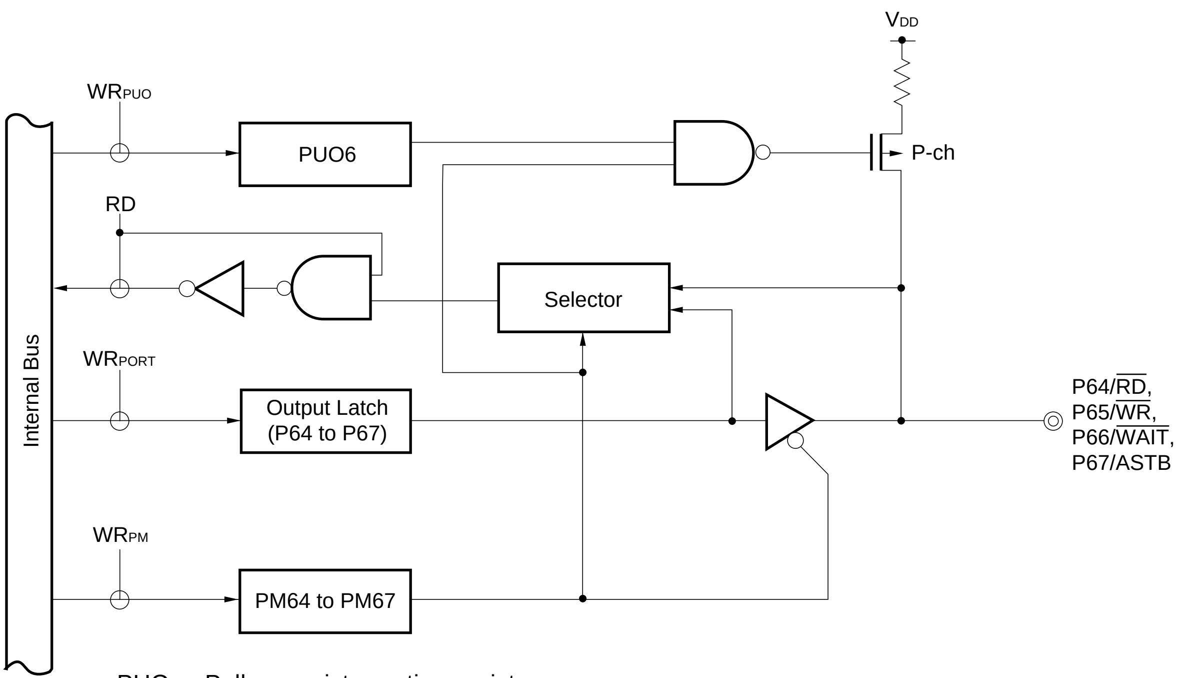
- **Within 3 clocks after a read is executed to Port 6 (P6) or Port mode register 6 (PM6) (when no wait cycles are inserted): –200 μA (max.)**
- **In other cases: –3 μA (max.)**

Figure 6-14. P60 to P63 Block Diagrams



PM : Port mode register
RD : Port 6 read signal
WR : Port 6 write signal

Figure 6-15. P64 to P67 Block Diagrams



PUO : Pull-up resistor option register
PM : Port mode register
RD : Port 6 read signal
WR : Port 6 write signal

6.3 Port Function Control Registers

The following four types of registers control the ports.

- Port mode registers (PM0, PM1, PM2, PM3, PM5, PM6)
- Pull-up resistor option register (PUO)
- Memory expansion mode register (MM)
- Key return mode register (KRM)

(1) Port mode registers (PM0, PM1, PM2, PM3, PM5, PM6)

These registers are used to set port input/output bit-wise.

PM0, PM1, PM2, PM3, PM5, and PM6 are independently set with a 1-bit or 8-bit memory manipulation instruction.

$\overline{\text{RESET}}$ input sets PM0 to 1FH and other registers to FFH.

When a port pin is used as its alternate function pin, set the port mode register and the output latch according to Table 6-5.

Cautions 1. P00 and P04 pins are input-only pins.

2. Input/output of P40 to P47 pins are specifiable with a memory expansion mode register (MM).

3. As port 0 is also used 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. When the output mode is used, therefore, the interrupt mask flag should be set to 1 beforehand.

Table 6-5. Port Mode Register and Output Latch Setting when Alternate Function is Used

Pin Name	Alternate Function		PM××	P××	Pin Name	Alternate Function		PM××	P××
	Function Name	Input/ Output				Function Name	Input/ Output		
P00	INTP0	Input	1 (defined)	None	P40 to P47	AD0 to AD7	Input/ Output	×[Note 2]	
	TI0	Input	1 (defined)	None					
P01 to P03	INTP1 to INTP3	Input	1	×	P50 to P57	A8 to A15	Output	×[Note 2]	
P04[Note 1]	XT1	Input	1 (defined)	None	P64	$\overline{RD}$	Output	×[Note 2]	
P10 to P17[Note 1]	ANI0 to ANI7	Input	1	×	P65	$\overline{WR}$	Output	×[Note 2]	
P30 to P32	TO0 to TO2	Output	0	0	P66	$\overline{WAIT}$	Input	×[Note 2]	
P33, P34	TI1, TI2	Input	1	×	P67	ASTB	Output	×[Note 2]	
P35	PCL	Output	0	0					
P36	BUZ	Output	0	0					

Notes 1. Read data will be undefined if the read instruction is executed for the port when used as alternate function pin.

2. When pins P40 to P47, P50 to P57 and P64 to P67 are used as alternate function pins, set the functions with a memory expansion mode register (MM).

Cautions 1. When external wait is not used in memory expansion mode, P66 pin can be used as an input/ output port.

2. When Port 2 is used as serial interface pin, input/output and the output latch should be set according to functions. For setting, refer to Figure 15-5 Serial Operating Mode Register 0 Format, Figure 16-6 Serial Operating Mode Register 0 Format, and Figure 17-3 Serial Operating Mode Register 1 Format.

Remark × : don't care (no setting required)

PM×× : Port mode register

P×× : Port output latch

Figure 6-16. Port Mode Register Format

Symbol	7	6	5	4	3	2	1	0	Address	When Reset	R/W
PM0	0	0	0	1	PM03	PM02	PM01	1	FF20H	1FH	R/W
PM1	PM17	PM16	PM15	PM14	PM13	PM12	PM11	PM10	FF21H	FFH	R/W
PM2	PM27	PM26	PM25	PM24	PM23	PM22	PM21	PM20	FF22H	FFH	R/W
PM3	PM37	PM36	PM35	PM34	PM33	PM32	PM31	PM30	FF23H	FFH	R/W
PM5	PM57	PM56	PM55	PM54	PM53	PM52	PM51	PM50	FF25H	FFH	R/W
PM6	PM67	PM66	PM65	PM64	PM63	PM62	PM61	PM60	FF26H	FFH	R/W

PMmn	Pmn Pin Input/Output Mode Select (m = 0, 1, 2, 3, 5, 6: n = 0 to 7)
0	Output mode (output buffer ON)
1	Input mode (output buffer OFF)

(2) Pull-up resistor option register (PUO)

This register is used to set whether to use an on-chip pull-up resistor at each port or not. An on-chip pull-up resistor is internally used only for the bits that are set to the input mode at a port where pull-up resistor use has been specified with PUO. No on-chip 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 PUO setting.

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

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

Cautions 1. P00 and P04 pins do not incorporate a pull-up resistor.

2. When port 1, port 4, port 5, or P64 to P67 is used as an alternate function pin, an on-chip pull-up resistor cannot be connected even if 1 is set in PUOm (m = 1, 4 to 6).

3. For P60 to P63 pins, only mask ROM versions can contain pull-up resistors by mask option.

Figure 6-17. Pull-Up Resistor Option Register Format

Symbol	7	<6>	<5>	<4>	<3>	<2>	<1>	<0>	Address	When Reset	R/W
PUO	0	PUO6	PUO5	PUO4	PUO3	PUO2	PUO1	PUO0	FFF7H	00H	R/W

PUOm	Pm On-chip Pull-up Resistor (m = 0, 1, 2, 3, 4, 5, 6)
0	On-chip pull-up resistor not used
1	On-chip pull-up resistor used

(3) Memory expansion mode register (MM)

The registers are used to set port 4 input/output.

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

$\overline{\text{RESET}}$ input sets MM to 10H.

Figure 6-18. Memory Expansion Mode Register Format

Symbol	7	6	5	4	3	2	1	0	Address	When Reset	R/W
MM	0	0	PW1	PW0	0	MM2	MM1	MM0	FFF8H	10H	R/W

<table>
<tr><th rowspan="2">MM2</th><th rowspan="2">MM1</th><th rowspan="2">MM0</th><th colspan="2" rowspan="2">Single-chip/Memory Expansion Mode Selection</th><th colspan="6">P40 to P47, P50 to P57, P64 to P67 Pins Condition</th></tr>
<tr><th colspan="2">P40 to P47</th><th>P50 to P53</th><th>P54, P55</th><th>P56, P57</th><th>P64 to P67</th></tr>
<tr><td>0</td><td>0</td><td>0</td><td colspan="2" rowspan="2">Single-chip mode</td><td rowspan="2">Port mode</td><td>Input</td><td colspan="4" rowspan="2">Port mode</td></tr>
<tr><td>0</td><td>0</td><td>1</td><td>Output</td></tr>
<tr><td>0</td><td>1</td><td>1</td><td rowspan="4">Memory expansion mode</td><td>256 bytes mode</td><td colspan="2" rowspan="4">AD0 to AD7</td><td colspan="3">Port mode</td><td rowspan="4">P64 = $\overline{\text{RD}}$
P65 = $\overline{\text{WR}}$
P66 = $\overline{\text{WAIT}}$
P67 = ASTB</td></tr>
<tr><td>1</td><td>0</td><td>0</td><td>4 Kbytes mode</td><td rowspan="3">A8 to A11</td><td colspan="2">Port mode</td></tr>
<tr><td>1</td><td>0</td><td>1</td><td>16 Kbytes mode</td><td rowspan="2">A12, A13</td><td>Port mode</td></tr>
<tr><td>1</td><td>1</td><td>1</td><td>Full-address mode[Note]</td><td>A14, A15</td></tr>
<tr><td colspan="3">Other than above</td><td colspan="8">Setting prohibited</td></tr>
</table>

PW1	PW0	Wait Control
0	0	No wait
0	1	With wait (1-wait state insertion)
1	0	Setting prohibited
1	1	Wait control with an external wait pin

Note Full-address mode is the mode that can execute external expansion to all the areas other than internal ROM, RAM, SFR areas and the reserved areas in 64 K address space.

Remarks 1. P60 to P63 pins enter the port mode regardless of the single-chip mode or memory expansion mode.

2. Besides it is used to set port 4 input/output, MM has functions to set the number of waits and external expansion area.

(4) Key return mode register (KRM)

The registers are used to set standby mode release enable/disable with the key return signal (falling edge detection of port 4).

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

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

Figure 6-19. Key Return Mode Register Format

Symbol	7	6	5	4	3	2	<1>	<0>	Address	When Reset	R/W
KRM	0	0	0	0	0	0	KRMK	KRIF	FFF6H	02H	R/W

KRIF	Key Return Signal Detection Flag
0	Undetected
1	Detected (falling edge detection of port 4)

KRMK	Standby Mode Control with Key Return Signal
0	Standby mode release enable
1	Standby mode release disable

Caution When falling edge detection is used in port 4, be sure to clear KRIF to 0 (KRIF cannot be 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 input/output 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 also undefined in addition to the manipulated bit.

6.4.2 Reading from input/output 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 input/output 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 also undefined in addition to the manipulated bit.

6.5 Mask Options

Mask ROM versions can contain a pull-up resistor in P60 to P63 pins bit-wise with the mask option.
The μPD78P014 and 78P014Y have no mask option and do not contain a pull-up resistor for P60 to P63 pins.

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
Oscillates at frequencies of 1.0 to 10.0 MHz. Oscillation can be stopped by executing the STOP instruction or setting the processor clock control register (PPC).

(2) Subsystem clock oscillator
Oscillates at a frequency of 32.768 kHz. Oscillation cannot be stopped. If the subsystem clock oscillator is not used, the on-chip feedback resistor can be set to disable by the processor clock control register (PCC). This enables to decrease power consumption in the STOP mode.

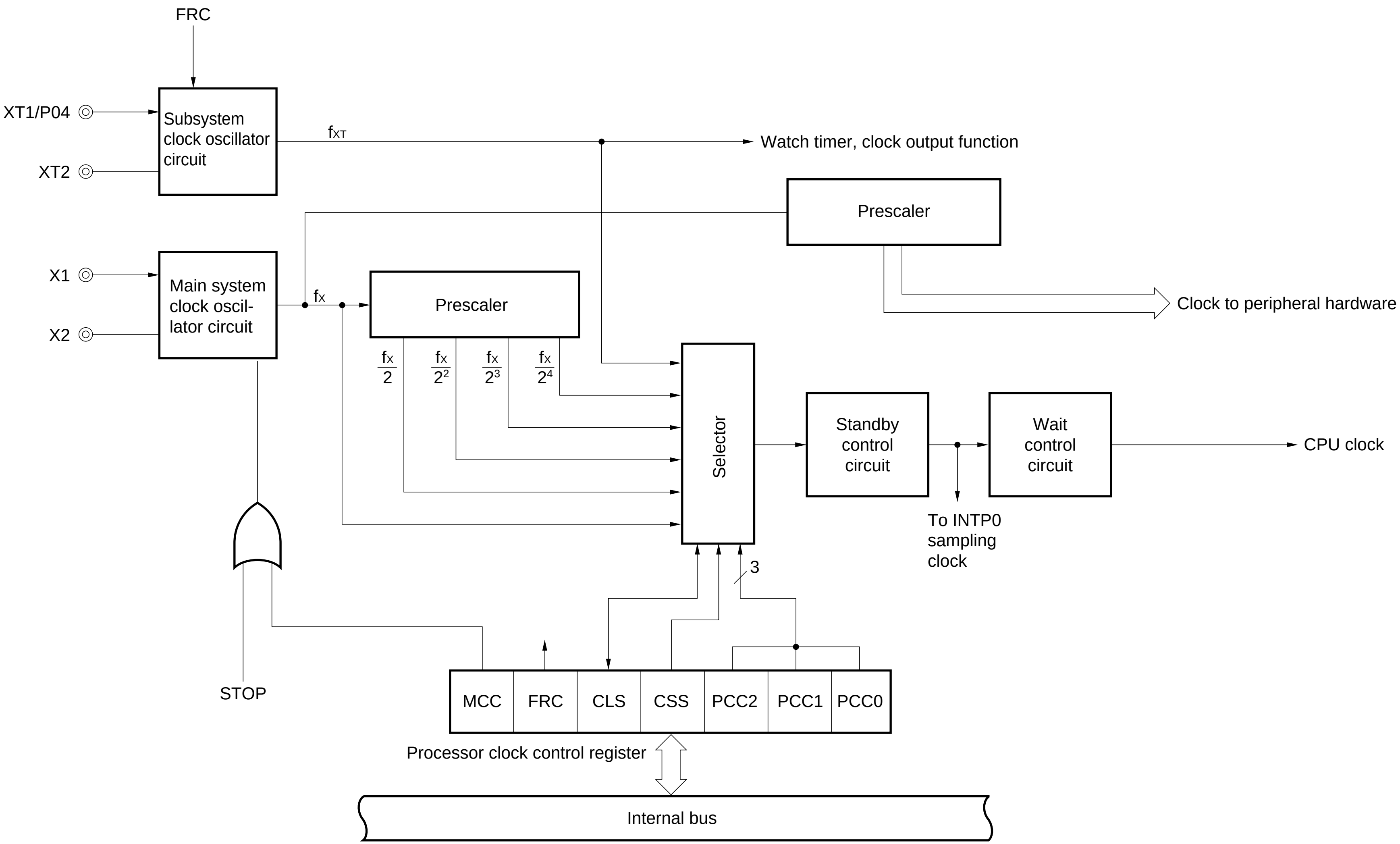
7.2 Clock Generator Configuration

The clock generator consists of the following hardware.

Table 7-1. Clock Generator Configuration

Item	Configuration
Control register	Processor clock control register (PCC)
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 processor clock control register (PCC). The PCC sets CPU clock selection, the ratio of division, main system clock oscillator operation/stop and subsystem clock oscillator on-chip feedback resistor enable/disable.

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. Feedback Resistor of Subsystem Clock

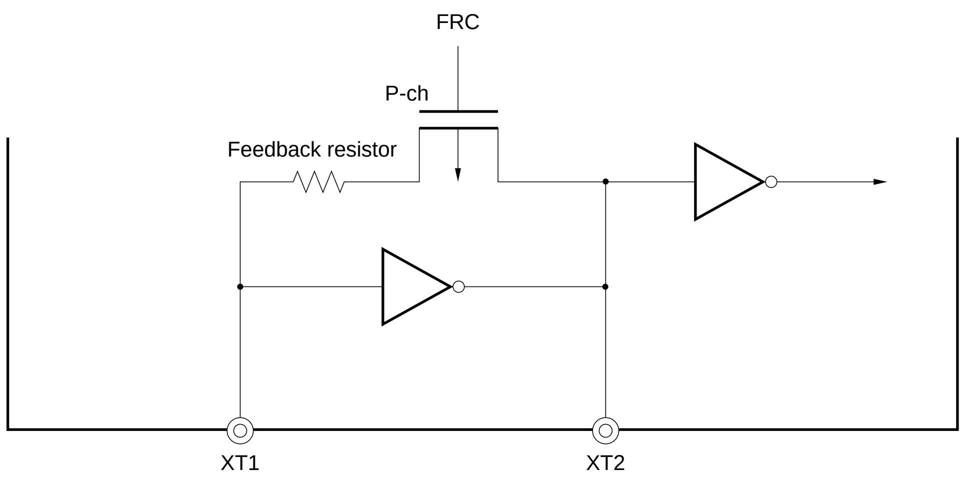


Figure 7-3. Processor Clock Control Register Format

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

R/W

CSS	PCC2	PCC1	PCC0	CPU Clock (f_{CPU}) selection
0	0	0	0	f_X
	0	0	1	$f_X/2$
	0	1	0	$f_X/2^2$
	0	1	1	$f_X/2^3$
	1	0	0	$f_X/2^4$
1	0	0	0	f_{XT}
	0	0	1	
	0	1	0	
	0	1	1	
	1	0	0	
Other than above				Setting prohibited

R

CLS	CPU Clock Status
0	Main system clock
1	Subsystem clock

R/W

FRC	Subsystem Clock Feedback Resistor Selection
0	On-chip feedback resistor used
1	On-chip feedback resistor not used

R/W

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

Notes 1. Bit 5 is Read Only.

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

The fastest instruction of the μPD78014, 78014Y Subseries is executed by the CPU clock 4 clock. The relationship between the CPU clock (f_{CPU}) and minimum instruction execution time is as shown in Table 7-2.

★ **Table 7-2. Relationship between CPU Clock and Minimum Instruction Execution Time**

CPU Clock (f_{CPU})	Minimum Instruction Execution Time: $4/f_{CPU}$
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_{XT}	122 μs

f_X = 10.0 MHz, f_{XT} = 32.768 kHz

f_X : Main system clock oscillation frequency

f_{XT} : Subsystem clock oscillation frequency

7.4 System Clock Oscillator

7.4.1 Main system clock oscillator

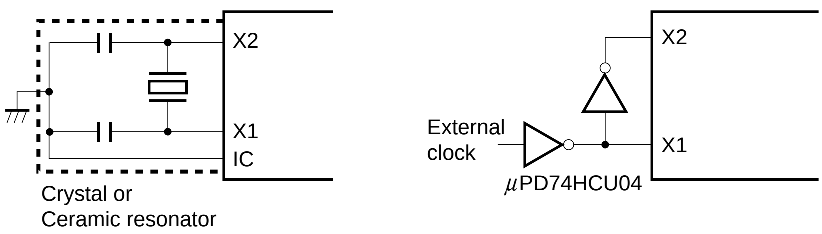
The main system clock oscillator oscillates with a crystal resonator or a ceramic resonator (standard: 10.0 MHz) connected to the X1 and X2 pins.

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

Figure 7-4 shows an external circuit of the main system clock oscillator.

Figure 7-4. External Circuit of Main System Clock Oscillator

(a) Crystal or ceramic oscillation **(b) External clock**



Caution The STOP mode cannot be set while an external clock is being input. This is because the X1 pin is short-circuited to V_{SS} in the STOP mode.

7.4.2 Subsystem clock oscillator

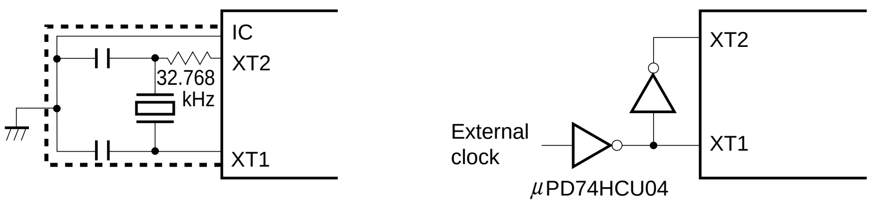
The subsystem clock oscillator oscillates with a crystal resonator (standard: 32.768 kHz) connected to the XT1 and XT2 pins.

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

Figure 7-5 shows an external circuit of the subsystem clock oscillator.

Figure 7-5. External Circuit of Subsystem Clock Oscillator

(a) Crystal oscillation **(b) External clock**



Cautions are shown on the next page.

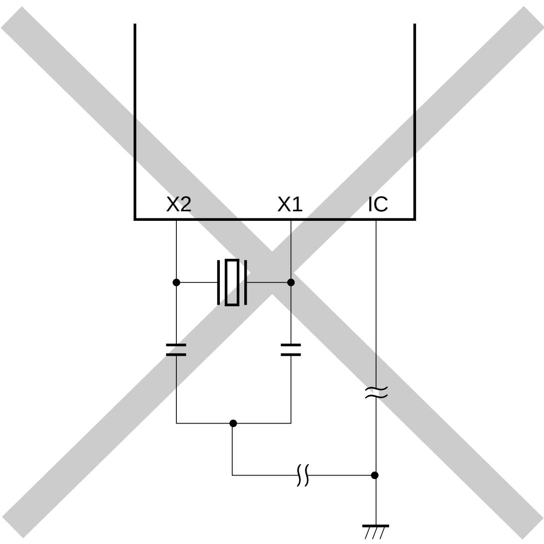
Cautions 1. When using a main system clock oscillator and a subsystem clock oscillator, wire the portion enclosed in the dotted line areas in Figures 7-4 and 7-5 as follows to avoid adverse influence on the wiring capacitance.

- **Keep the wiring length as short as possible.**
- **Do not cross the wiring over the other signal lines. Do not route the wiring in the vicinity of lines through which a high fluctuating current flows.**
- **Always keep the ground point of the capacitor of the oscillator circuit at the same potential as V_{SS}. Do not connect the power source pattern through which a high current flows.**
- **Do not extract signals from the oscillator.**

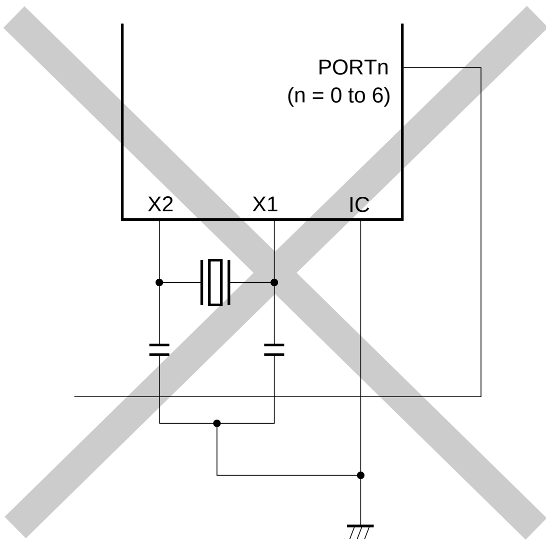
Take special note of the fact that the subsystem clock oscillator is a circuit with low-level amplification so that current consumption is maintained at low levels.
Figure 7-6 shows examples of resonator having bad connection.

Figure 7-6. Examples of Resonator with Bad Connection (1/2)

(a) Long wiring of connected circuit



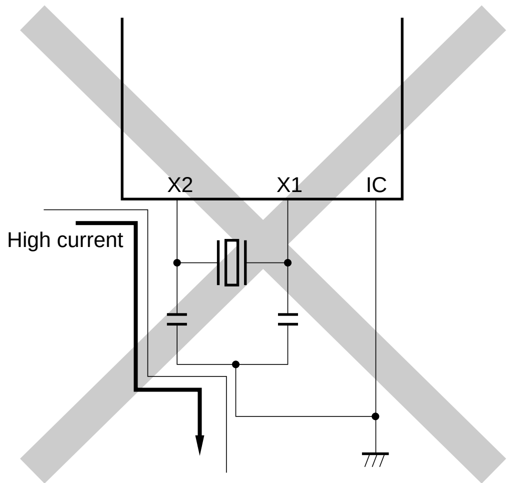
(b) Crossed signal lines



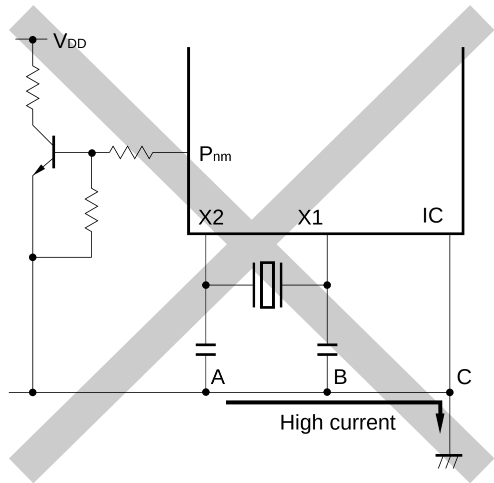
Remark When using a subsystem clock, replace X1 and X2 with XT1 and XT2, respectively. Further, insert resistors in series on the side of XT2.

Figure 7-6. Examples of Resonator with Bad Connection (2/2)

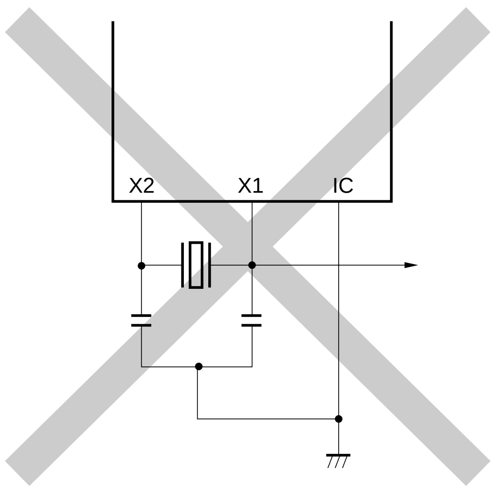
(c) High fluctuating current close to signal lines



(d) Current flowing through ground line of oscillator circuit (potentials at points A, B, and C change.)



(e) Signal extracted



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

Cautions 2. When XT2 and X1 are wired parallel, X1 crosstalk noise may affect XT2 and cause an error. To prevent that from happening, it is recommended to connect the IC pin between XT2 and X1 to V_{SS} as well as not wire XT2 and X1 in parallel.

7.4.3 Divider

The divider divides the main system clock oscillator output (f_X) 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 watch operations, connect the XT1 and XT2 pins as follows.

XT1: Connect to V_{DD} or V_{SS}
XT2: Open

In this state, however, some current may leak via the on-chip feedback resistor of the subsystem clock oscillator when the main system clock stops. To prevent that from happening, the above on-chip feedback resistor (PCC) 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_X
- Subsystem clock f_{XT}
- CPU clock f_{CPU}
- Clock to peripheral hardware

The function and operation of the clock generator circuit are determined by the processor clock control register (PCC) as follows:

(a) Upon generation of the $\overline{\text{RESET}}$ signal, the lowest speed mode of the main system clock (6.4 μs when operated at 10.0 MHz) is selected (PCC = 04H). 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 five (0.4 μs, 0.8 μs, 1.6 μs, 3.2 μs and 6.4 μs: when operated at 10.0 MHz) CPU clock stages can be selected by setting the PCC.

(c) With the main system clock selected, two standby modes, the STOP and HALT modes, are available. With the subsystem clock unused, the current consumption in STOP mode can be further decreased by disabling the subsystem clock feedback resistor with PCC bit 6 (FRC).

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

(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 watch timer and clock output functions only. Thus, the watch function and the clock output function can also be continued in the standby state. However, since all other peripheral hardware operate 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 bits 0 to 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 7-7**).

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

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

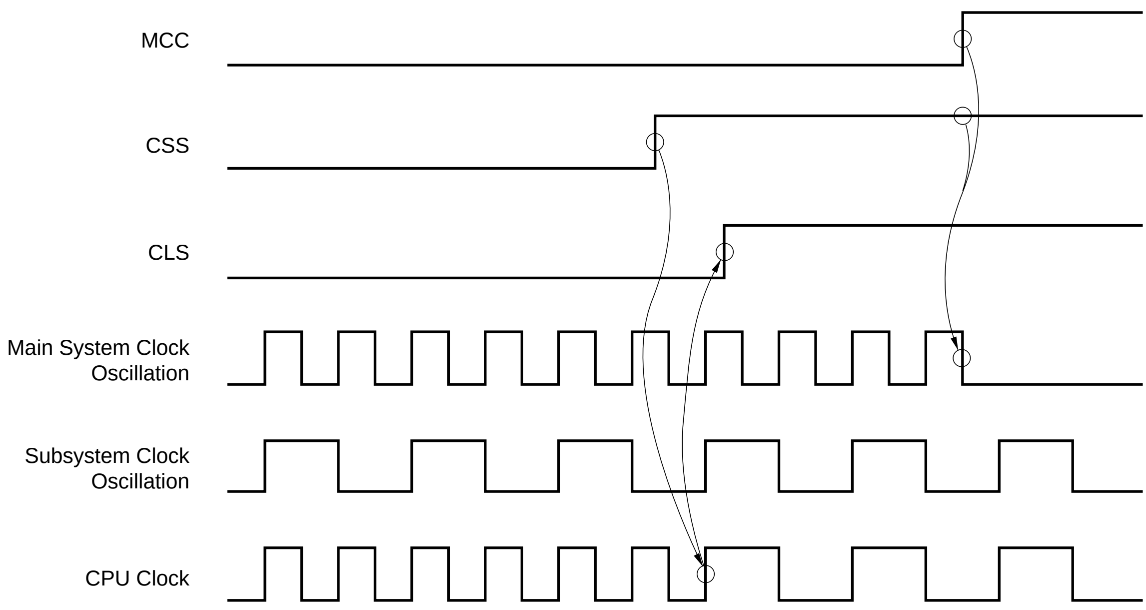
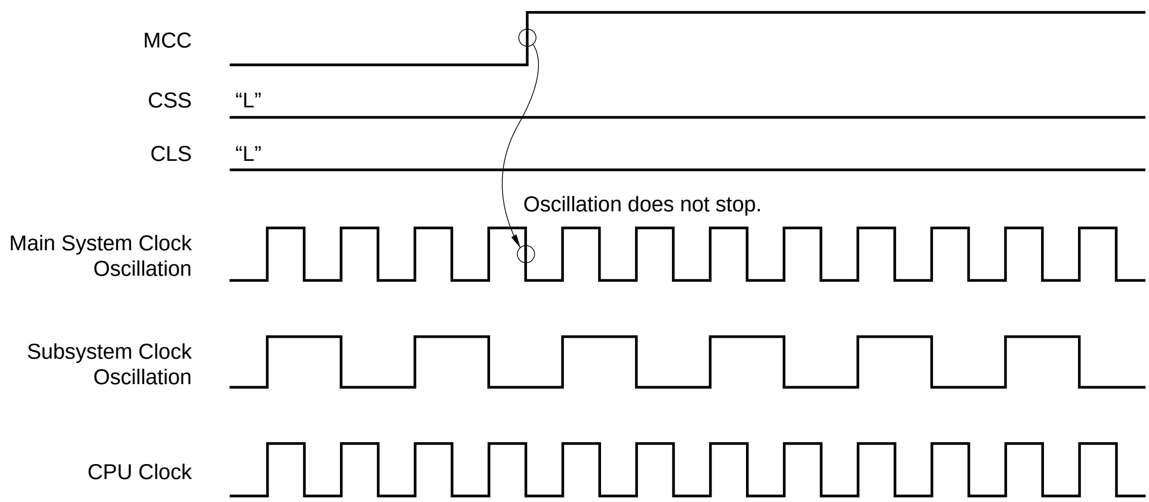
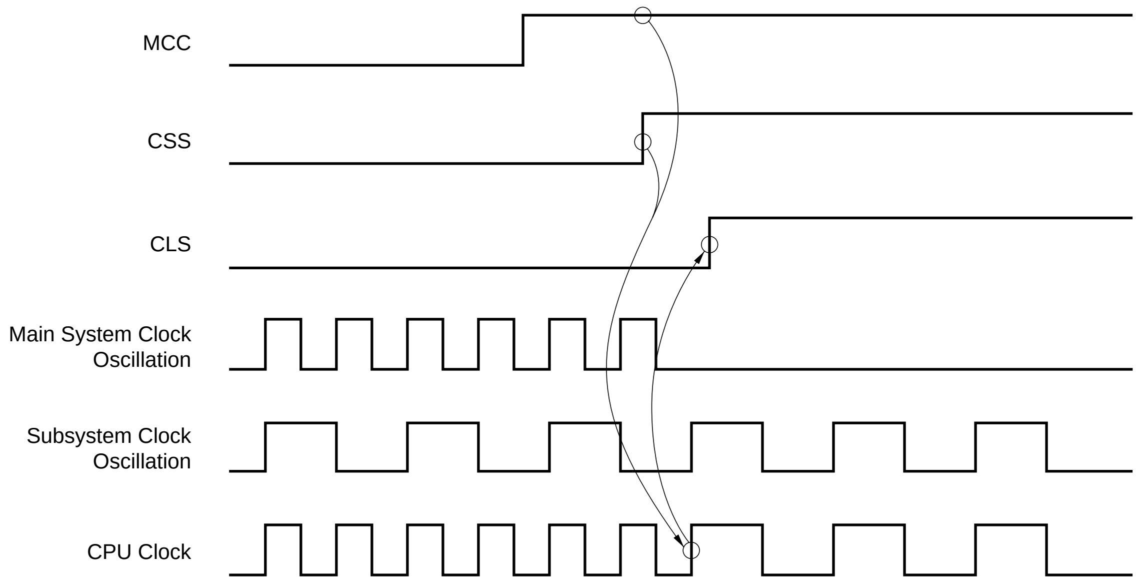


Figure 7-7. Main System Clock Stop Function (2/2)

(b) Operation when MCC is set with main system clock operation



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

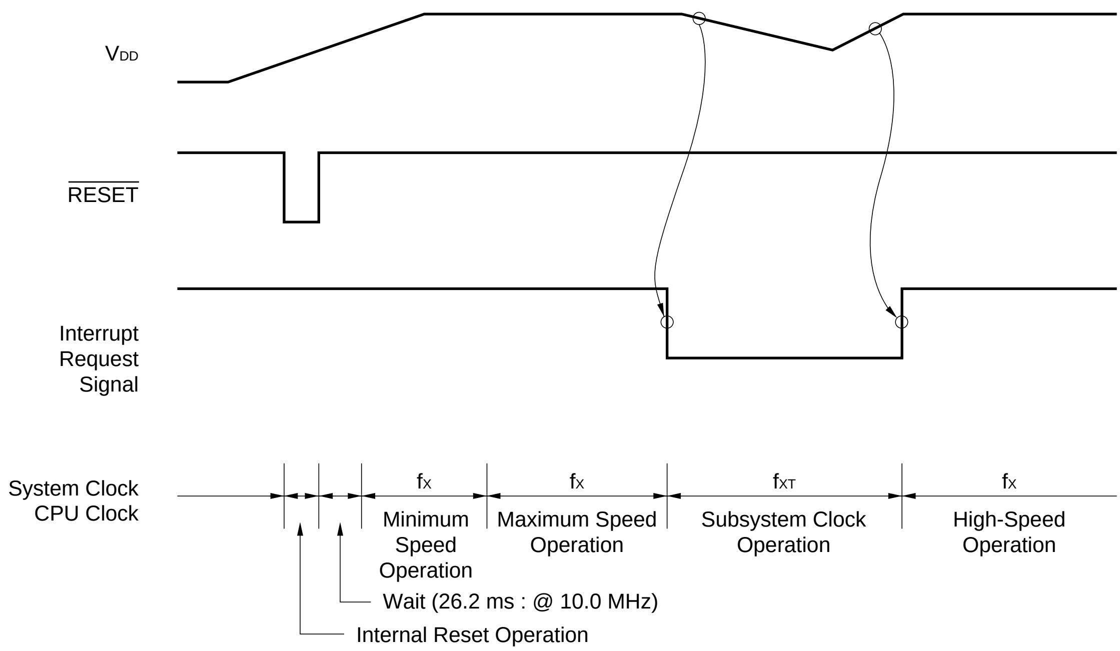
Caution **Selection of the CPU clock cycle dividing factor (PCC0 to PCC2) and switchover from the main system clock to the subsystem clock (changing CSS from 0 to 1) should not be performed simultaneously.**
Simultaneous setting is possible, however, for selection of the CPU clock cycle dividing factor (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. Values in parentheses apply to operation with f_X = 10.0 MHz or 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 7-8. 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, the main system clock starts oscillation. At this time, the oscillation stabilization time ($2^{18}/f_X$) is secured automatically.
After that, the CPU starts executing the instruction at the minimum speed of the main system clock (6.4 μs when operated at 10.0 MHz).

(2) After the lapse of a sufficient time for the V_{DD} voltage to increase to enable operation at maximum speed, the processor clock control register (PCC) is rewritten and 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 stabilization state).

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

Caution When the main system clock is stopped and the device 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.

[MEMO]

CHAPTER 8 16-BIT TIMER/EVENT COUNTER

8.1 Outline of On-chip Timer in μPD78014, 78014Y Subseries

This section describes the 16-bit timer/event counter. First an outline of the built-in timer in the μPD78014 and 78014Y Subseries and the related items are shown in the following.

(1) 16-bit timer/event counter (TM0)
The TM0 can be used for an interval timer, PWM output, pulse width measurement (infrared ray remote control receive function), external event counter or square wave output of any frequency.

(2) 8-bit timer/event counters (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 control circuit
This circuit supplies other devices with the divided main system clock and the subsystem clock (See **CHAPTER 12 CLOCK OUTPUT CONTROL CIRCUIT**).

(6) Buzzer output control circuit
This circuit outputs the buzzer frequency obtained by dividing the main system clock (See **CHAPTER 13 BUZZER OUTPUT CONTROL CIRCUIT**).

Table 8-1. Timer/Event Counter Operation

		16-Bit Timer/ Event Counter	8-Bit Timer/ Event Counter	Watch Timer	Watchdog Timer
Operation mode	Interval timer	1 channel	2 channels	1 channel[Note 1]	1 channel[Note 2]
	External event counter	○	○	—	—
Function	Timer output	○	○	—	—
	PWM output	○	—	—	—
	Pulse width measurement	○	—	—	—
	Square-wave output	○	○	—	—
	Interrupt request	○	○	○	○
	Test input	—	—	○	—

Notes 1. Watch timer can perform both watch timer and interval timer functions at the same time.
2. 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

PWM output and pulse width measurement functions at the same time.

(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
$2 \times$ TI0 input cycle	$2^{16} \times$ TI0 input cycle	TI0 input edge cycle
$2^2 \times 1/f_X$ (400 ns)	$2^{17} \times 1/f_X$ (13.1 ms)	$2 \times 1/f_X$ (200 ns)
$2^3 \times 1/f_X$ (800 ns)	$2^{18} \times 1/f_X$ (26.2 ms)	$2^2 \times 1/f_X$ (400 ns)
$2^4 \times 1/f_X$ (1.6 μs)	$2^{19} \times 1/f_X$ (52.4 ms)	$2^3 \times 1/f_X$ (800 ns)

Remarks 1. f_X: Main system clock oscillation frequency
2. Values in parentheses apply to operation with f_X = 10.0 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
2 × TI0 input cycle	2^{16} × TI0 input cycle	TI0 input edge cycle
$2^2 \times 1/f_X$ (400 ns)	$2^{17} \times 1/f_X$ (13.1 ms)	$2 \times 1/f_X$ (200 ns)
$2^3 \times 1/f_X$ (800 ns)	$2^{18} \times 1/f_X$ (26.2 ms)	$2^2 \times 1/f_X$ (400 ns)
$2^4 \times 1/f_X$ (1.6 μs)	$2^{19} \times 1/f_X$ (52.4 ms)	$2^3 \times 1/f_X$ (800 ns)

Remarks 1. f_X: Main system clock oscillation frequency
2. Values in parentheses apply to operation with f_X = 10.0 MHz

8.3 16-Bit Timer/Event Counter Configuration

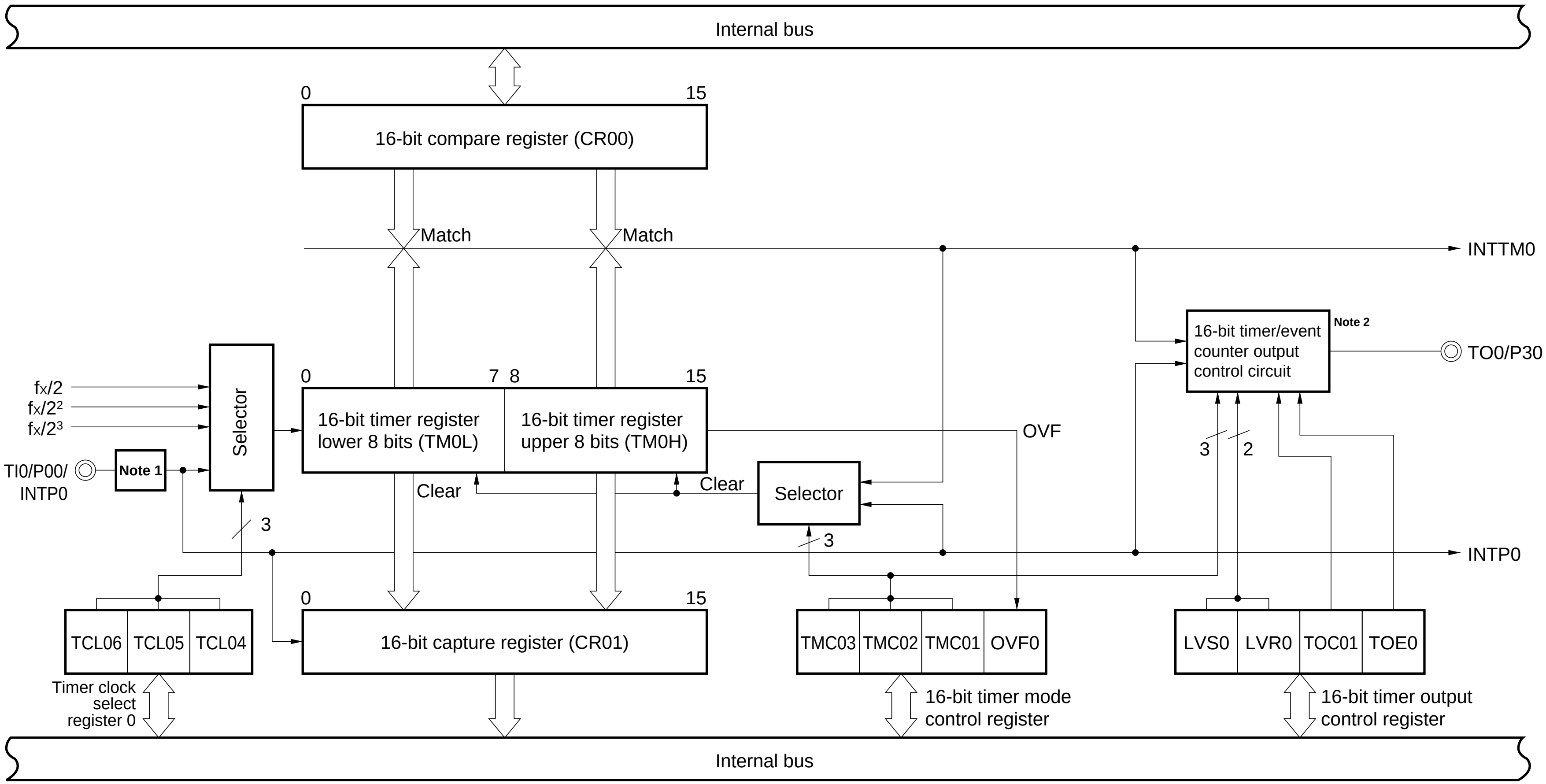
The 16-bit timer/event counter consists of the following hardware.

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

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

Note Refer to **Figure 18-1 Basic Configuration of Interrupt Function**.

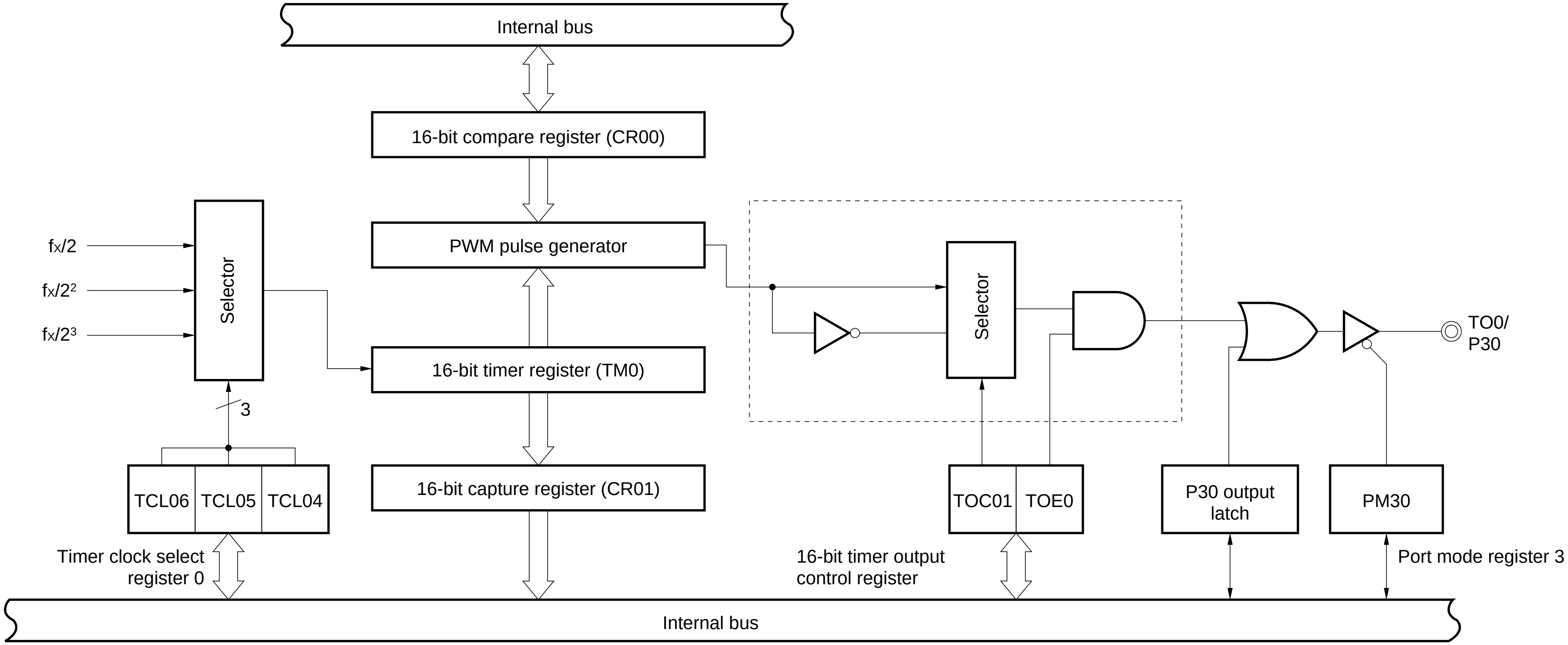
Figure 8-1. 16-Bit Timer/Event Counter (Timer Mode) Block Diagram



Notes **1.** Edge detector

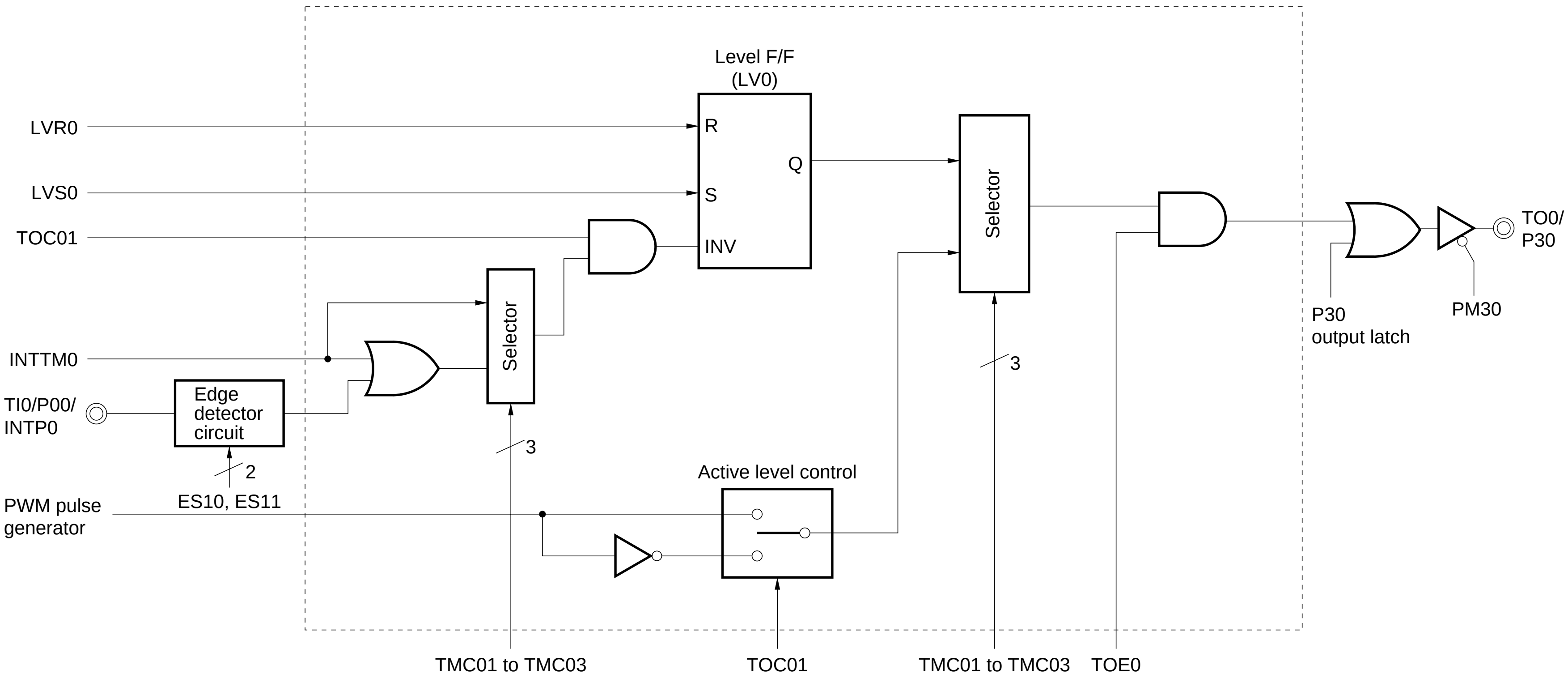
2. The configuration of the 16-bit timer/event counter output control circuit is shown in Figure 8-3.

Figure 8-2. 16-Bit Timer/Event Counter (PWM Mode) Block Diagram



Remark The circuitry enclosed by the dotted line is included in the output control circuit.

Figure 8-3. 16-Bit Timer/Event Counter Output Control Circuit Block Diagram



Remark The circuitry enclosed by the dotted line is the output control circuit.

(1) 16-bit compare register (CR00)

CR00 is a 16-bit register for which the value set in the CR00 is constantly compared with the 16-bit timer register (TM0) count value, and an interrupt request (INTTM0) is generated if they match.

It can be used as the register which holds the interval time when TM0 is set to interval timer operation, and as the register which sets the pulse width when TM0 is set PWM output operation.

CR00 is set by a 16-bit memory manipulation instruction. The value of 0001H to FFFFH can be set.

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

Cautions 1. PWM data (14 bits) must be set in the upper 14 bits of CR00. The lower 2 bits must be set to 00.

2. CR00 must be set in a value other than 0000H. Consequently, when it is used as an event counter, 1-pulse count operation is prohibited.

3. When the value of CR00 posterior to alteration is less than the value of the 16-bit timer register (TM0), TM0 keeps on counting and resumes counting from 0 after an overflow. When the value of CR00 posterior to alteration is less than the value prior to alteration, the timer must be restarted after CR00 is altered.

(2) 16-bit capture register (CR01)

CR01 is a 16-bit register capturing the content of 16-bit timer register (TM0).

Capture trigger is INTP0/P001/TI0 pin valid edge input. Setting of the INTP0 valid edge is done with the external interrupt mode register (INTM0).

CR01 is read by a 16-bit memory manipulation instruction.

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

Caution When the TI0/P00 pin's valid edge is input while CR01 is read, CR01 retains its contents without doing capture operations. However, the interrupt request flag (PIF0) is set due to the valid edge detection.

(3) 16-bit timer register (TM0)

TM0 is a 16-bit register counting count pulse.

TM0 is read by a 16-bit memory manipulation instruction.

After $\overline{\text{RESET}}$ input, the value of TM0 is 0000H.

Caution The TM0 value is read out via CR01, resulting in changing the previous CR01 contents.

8.4 16-Bit Timer/Event Counter Control Registers

The following six 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)
- 16-bit timer output control register (TOC0)
- Port mode register 3 (PM3)
- External interrupt mode register (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.
TCL0 is set with a 1-bit or 8-bit memory manipulation instruction.
$\overline{\text{RESET}}$ input sets TCL0 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 8-4. Timer Clock Select Register 0 Format

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

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

TCL06	TCL05	TCL04	16-bit Timer Register Count Clock Selection
0	0	0	TI0 (Valid edge specifiable)
0	1	0	$f_X/2$ (5.0 MHz)
0	1	1	$f_X/2^2$ (2.5 MHz)
1	0	0	$f_X/2^3$ (1.25 MHz)
Other than above			Setting prohibited

CLOE	PCL Output Control
0	Output disabled
1	Output enabled

Cautions 1. Setting of the INTP0/P00/TI0 pin valid edge is performed by external interrupt mode register (INTM0), and selection of the sampling clock frequency is performed by the sampling clock selection register (SCS).

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 TI0 has been specified as the TM0 count clock, the value should be read from TM0, not from the 16-bit capture register CR01.

4. If data other than identical data is to be rewritten to TCL0, the timer operation must be stopped first.

Remarks 1. f_X : Main system clock oscillation frequency

2. f_{XT} : Subsystem clock oscillation frequency

3. TI0 : 16-bit timer/event counter input pin

4. TM0: 16-bit timer register

5. Value in parentheses apply to operation with f_X = 10.0 MHz or f_{XT} = 32.768 kHz.

6. See **CHAPTER 12 CLOCK OUTPUT CONTROL CIRCUIT** for PCL.

(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 sets TMC0 value to 00H.

Caution The 16-bit timer register (TM0) starts operation when TMC01 to TMC03 are set to the value other than 0, 0, 0 (operation stop mode). To stop the timer operation, set TMC01 through TCM03 to 0, 0, 0.

Figure 8-5. 16-Bit Timer Mode Control Register Format

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

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

TMC03	TMC02	TMC01	Operating Mode & Clear Mode Selection	TO0 Output Timing Selection	Interrupt Request 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
0	1	0	Free running mode	Match between TM0 and CR00	
0	1	1		Match between TM0 and CR00 or TI0 valid edge	
1	0	0	Clear & start on TI0 valid edge	Match between TM0 and CR00	
1	0	1		Match between TM0 and CR00 or TI0 valid edge	
1	1	0	Clear & start on match between TM0 and CR00	Match between TM0 and CR00	
1	1	1		Match between TM0 and CR00 or TI0 valid edge	

Cautions 1. Switch the clear mode and the TO0 output timing after stopping the timer operation (by setting TMC01 through TMC03 to 0, 0, 0).

2. Set the valid edge of the INTP0/P00/TI0 pin with an external interrupt mode register (INTM0) and select the sampling clock frequency with a sampling clock select register (SCS).

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, OVF0 flag is set to 1 when the set value of CR00 is FFFFH and the value of TM0 changes from FFFFH to 0000H.

Remarks 1. TO0 : 16-bit timer/event counter output pin

2. TI0 : 16-bit timer/event counter input pin

3. TM0 : 16-bit timer register

4. CR00 : 16-bit compare register

(3) 16-bit timer output control register (TOC0)

This register controls the operation of the 16-bit timer/event counter output control circuit. It sets R-S type flip-flop (LV0) setting/resetting, the active level in PWM mode, output inversion enabling/disabling in modes other than PWM mode and data output mode.

TOC0 is set with a 1-bit or 8-bit memory manipulation instruction.

RESET input sets TOC0 value to 00H.

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

Symbol	7	6	5	4	<3>	<2>	1	<0>	Address	When Reset	R/W
TOC0	0	0	0	0	LVS0	LVR0	TOC01	TOE0	FF4EH	00H	R/W

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

TOC01	In PWM Mode	In Other Modes
	Active level selection	Timer output F/F control
0	Active high	Inversion operation disabled
1	Active low	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

Cautions 1. Timer operation must be stopped before setting TOC0.

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

(4) Port mode register 3 (PM3)

This register sets port 3 input/output bit-wise.

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	When Reset	R/W
PM3	PM37	PM36	PM35	PM34	PM33	PM32	PM31	PM30	FF23H	FFH	R/W

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

(5) External interrupt mode register (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 sets INTM0 value to 00H.

Remarks **1.** INTP0 pin is a dual function pin also used for TI0/P00.

2. INTP3 is fixed at the falling edge.

Figure 8-8. External Interrupt Mode Register Format

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

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

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

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

Caution **To specify the INTP0/P00/TI0 pin valid edge, bit 1 to bit 3 (TMC01 to TMC03) of the 16-bit timer mode control register (TMC0) must be set to 0, 0, 0, respectively after timer operaton is stopped.**

(6) 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 removed with sampling clock.

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

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

Figure 8-9. Sampling Clock Select Register Format

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

SCS1	SCS0	INTP0 Sampling Clock Selection
0	0	$f_X/2^{N+1}$
0	1	Setting prohibited
1	0	$f_X/2^6$ (156 kHz)
1	1	$f_X/2^7$ (78.1 kHz)

Caution **$f_X/2^{N+1}$ is the clock supplied to the CPU, and $f_X/2^6$ and $f_X/2^7$ are clocks supplied to peripheral hardware. $f_X/2^{N+1}$ is stopped in HALT mode.**

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

2. f_X: Main system clock oscillation frequency

3. Values in parentheses apply to operation with f_X = 10.0 MHz.

8.5 16-Bit Timer/Event Counter Operations

8.5.1 Interval timer operations

By setting bits 2 and 3 (TMC02 and TMC03) of the 16-bit timer mode control register (TMC0) to 1 and 1, they are operated as an interval timer. Interrupt requests are generated repeatedly using the count value set in 16-bit compare register (CR00) beforehand is used 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 (INTTM0) is generated.

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

For the operation after changing compare register value during timer count operation, refer to **8.6 16-Bit Timer/ Event Counter Operating Precautions (3)**.

Figure 8-10. Interval Timer Configuration Diagram

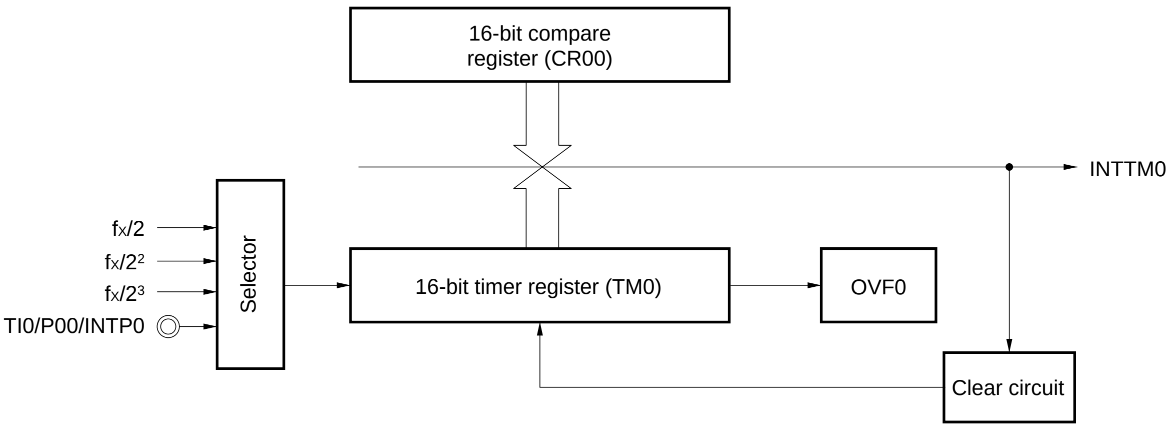
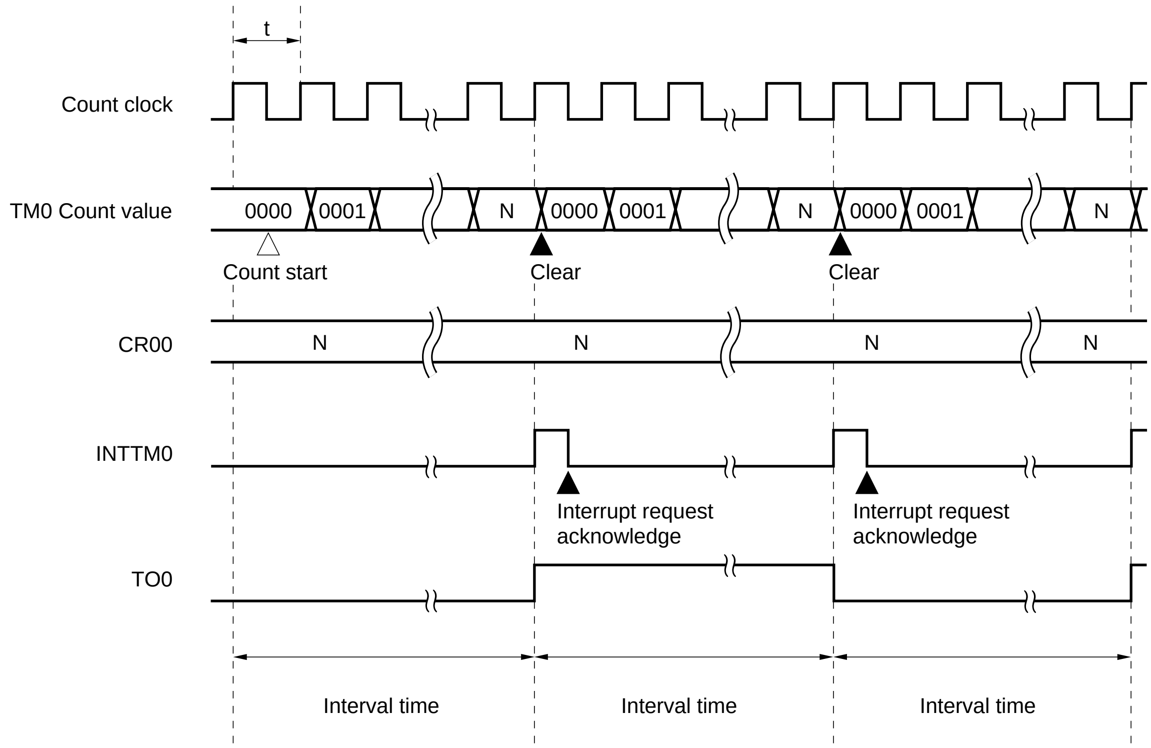


Figure 8-11. Interval Timer Operation Timings



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

Table 8-5. 16-Bit Timer/Event Counter Interval Times

TCL06	TCL05	TCL04	Minimum Interval Time	Maximum Interval Time	Resolution
0	0	0	2 × TI0 input cycle	2^{16} × TI0 input cycle	TI0 input edge cycle
0	1	0	2^2 × 1/fx (400 ns)	2^{17} × 1/fx (13.1 ms)	2 × 1/fx (200 ns)
0	1	1	2^3 × 1/fx (800 ns)	2^{18} × 1/fx (26.2 ms)	2^2 × 1/fx (400 ns)
1	0	0	2^4 × 1/fx (1.6 μs)	2^{19} × 1/fx (52.4 ms)	2^3 × 1/fx (800 ns)
Other than above			Setting prohibited		

Remarks 1. fx: Main system clock oscillation frequency

2. TCL04 to TCL06: Bits 4 to 6 of the timer clock select register 0 (TCL0)

3. Values in parentheses apply to operation with fx = 10.0 MHz

8.5.2 PWM output operations

By setting bits 1 to 3 (TMC01 to 03) of the 16-bit timer mode control register (TMC0) to 1, 0, and 0, they are operated as PWM output. Pulses with the duty rate determined by the value set in 16-bit compare register (CR00) beforehand are output from the TO0/P30 pin.

Set the active level width of the PWM pulse to the high-order 14 bits of CR00. Select the active level with bit 1 (TOC01) of the 16-bit timer output control register (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 has a combination of the basic cycle determined by $2^8/f_\Phi$ and the sub-cycle determined by $2^{14}/f_\Phi$ so that the time constant of the external LPF can be shortened. Count clock f_Φ can be selected with bits 4 to 6 (TCL04 to TCL06) of the 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 INTP0/P00/TI0 pin.

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 8-12 is as follows.

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

V_{REF}: External switching circuit reference voltage

Figure 8-12. Example of D/A Converter Configuration with PWM Output

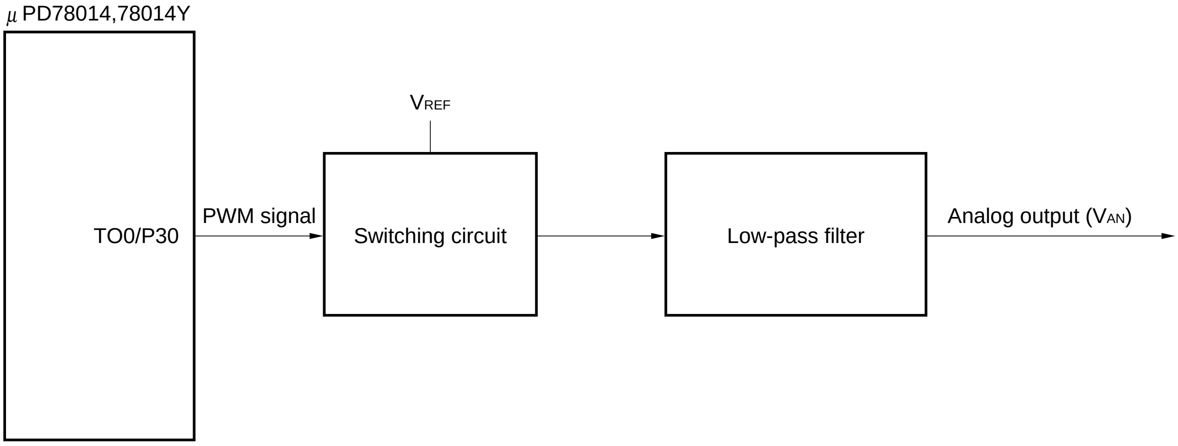
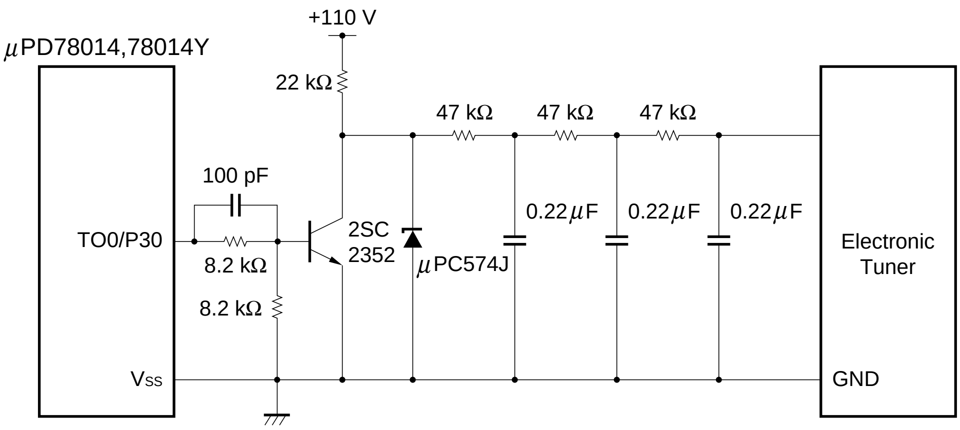


Figure 8-13 shows an example in which PWM output is converted to an analog voltage and used in a voltage synthesizer type TV tuner.

Figure 8-13. TV Tuner Application Circuit Example



8.5.3 Pulse width measurement operations

The pulse width of the signal to be input to theINTP0/P00/TI0 pin can be neasured with 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 valid edge of the signal input to the INTP0/P00/TI0 pin.

(1) Pulse width measurement with free-running

When the 16-bit timer register (TM0) is operated, the edge specified by external interrupt mode register (INTM0) is input, the value of TM0 is taken into 16-bit capture register (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 the external interrupt mode register (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 8-14. Configuration Diagram for Pulse Width Measurement by Free-Running Counter

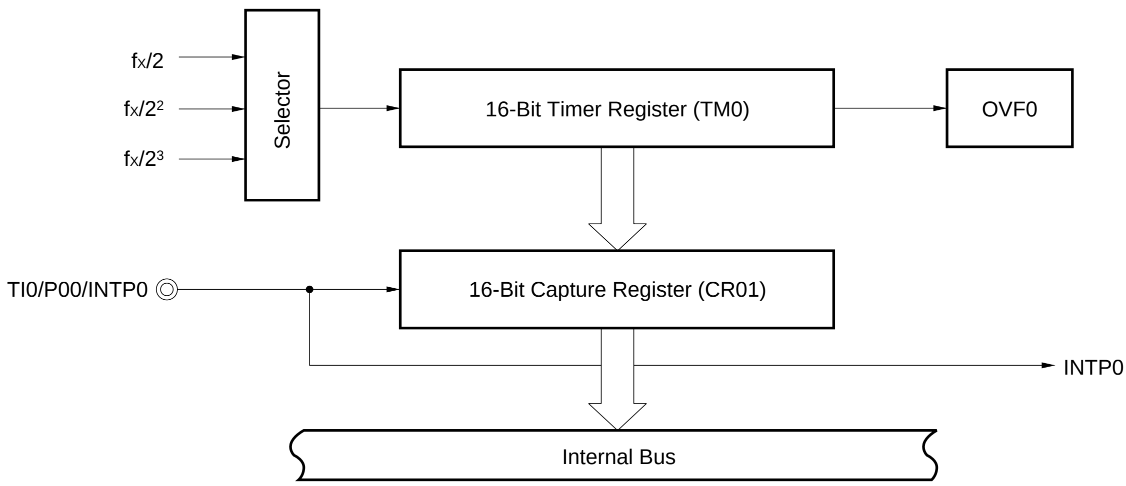
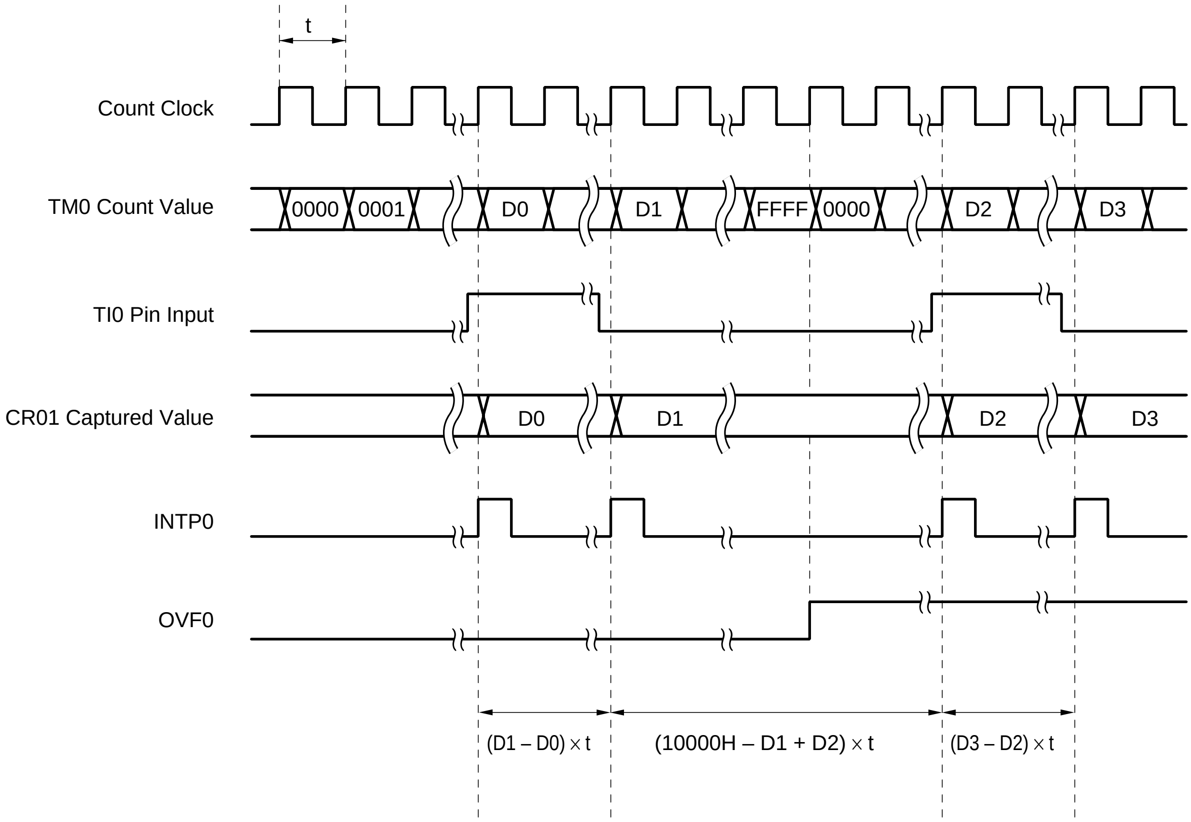


Figure 8-15. Timing of Pulse Width Measurement Operation by Free-Running Counter (with Both Edges Specified)



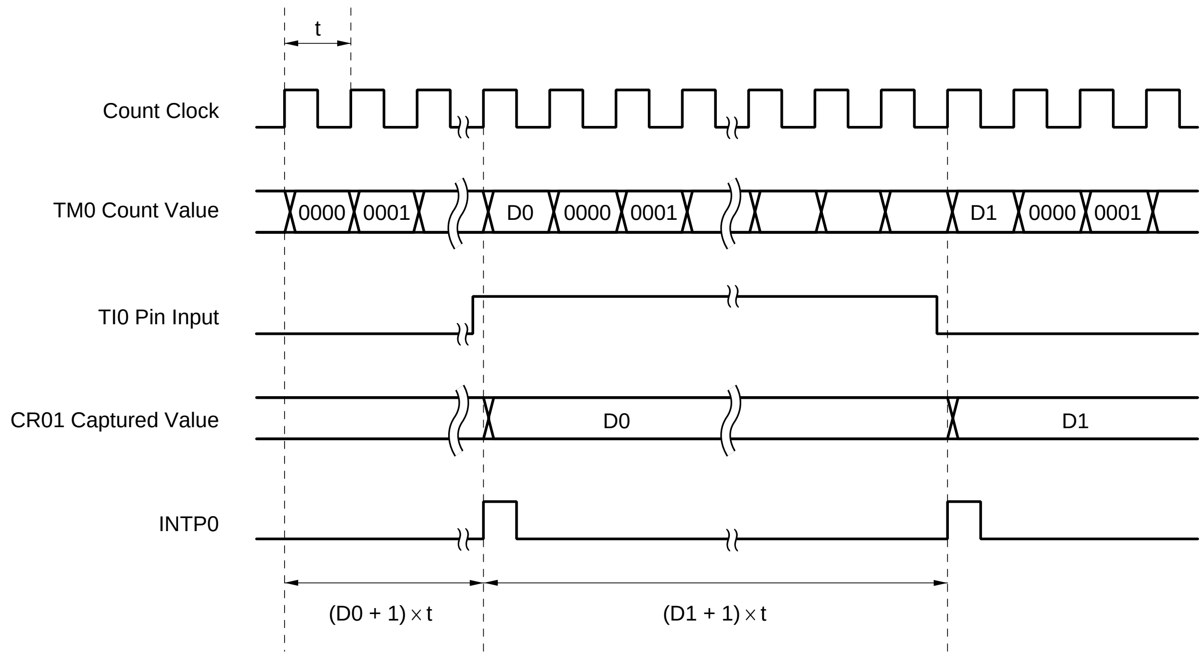
(2) Pulse width measurement by means of restart

When input of a valid edge to the INTP0/P00/TI0 pin is detected, the count value of the 16-bit timer register (TM0) is taken into the 16-bit capture register (CR01), and then the pulse width of the signal input to the INTP0/P00/TI0 pin is measured by clearing TM0 and restarting the count.

The edge specification can be selected from three types, rising, falling, and both edges by bit 2 and bit 3 (ES10 and ES11) of the external interrupt mode register (INTM0).

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 not performed before detecting valid levels twice allowing short pulse width noise to be eliminated.

Figure 8-16. Timing of Pulse Width Measurement Operation by Means of Restart (with Both Edges Specified)



8.5.4 External event counter operation

The external event counter counts the number of external clock pulses to be input to the INTP0/P00/TI0 pin with the 16-bit timer register (TM0).

TM0 is incremented each time the valid edge specified with the external interrupt mode register (INTM0) is input.

When the TM0 counted value matches the 16-bit compare register (CR00) value, TM0 is cleared to 0 and the interrupt request signal (INTTM0) is generated.

★ The 16-bit compare register (CR00) must be set to a value other than 0000H (1-pulse count operation is prohibited).

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 cycle selected with the sampling clock select register (SCS), noise with short pulse widths can be removed.

Figure 8-17. External Event Counter Configuration Diagram

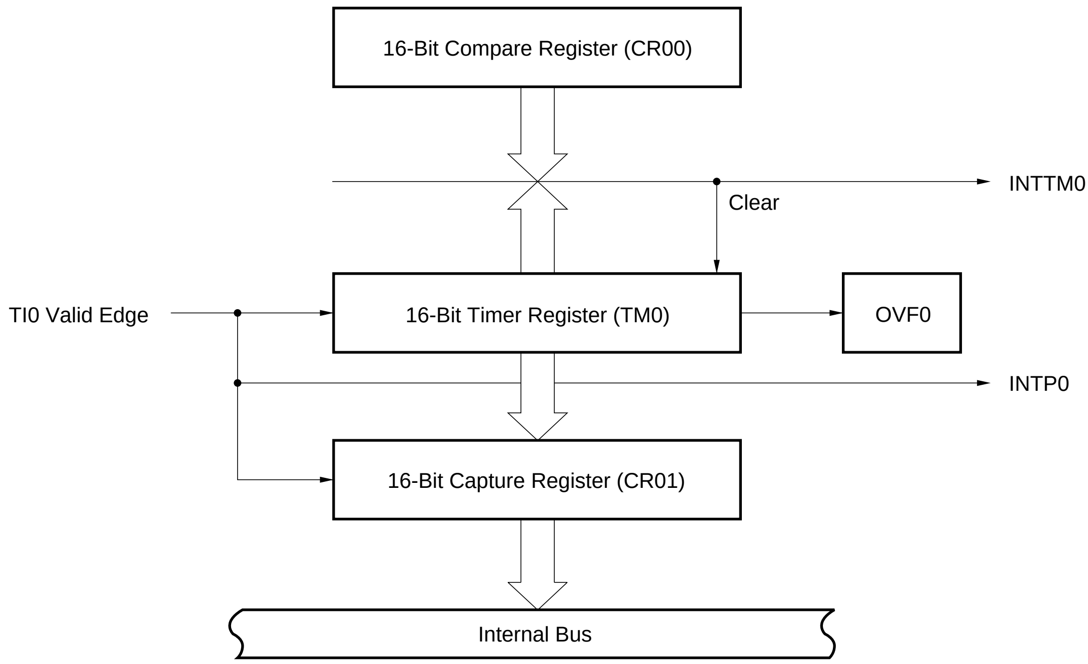
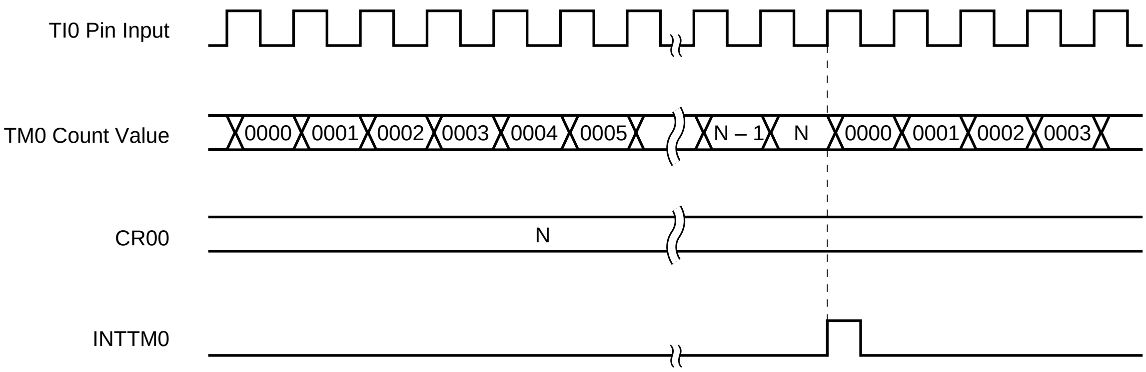


Figure 8-18. External Event Counter Operation Timings (with Rising Edge Specified)



8.5.5 Square-wave output operation

The 16-bit timer/event counter operates as a square wave with any selected frequency which is output at intervals of the count value preset to the 16-bit compare register (CR00).

The TO0/P30 pin output status is inverted 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 to 1. This enables a square wave with any selected frequency to be output.

Table 8-6. 16-Bit Timer/Event Counter Square-Wave Output Ranges

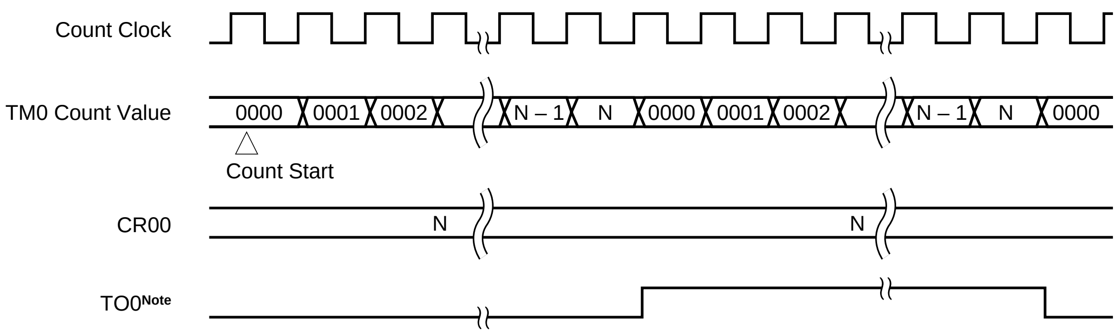
TCL06	TCL05	TCL04	Minimum Pulse Width	Maximum Pulse Width	Resolution
0	0	0	2 × TI0 input cycle	2^{16} × TI0 input cycle	TI0 input edge cycle
0	1	0	2^2 × 1/fx (400 ns)	2^{17} × 1/fx (13.1 ms)	2 × 1/fx (200 ns)
0	1	1	2^3 × 1/fx (800 ns)	2^{18} × 1/fx (26.2 ms)	2^2 × 1/fx (400 ns)
1	0	0	2^4 × 1/fx (1.6 μs)	2^{19} × 1/fx (52.4 ms)	2^3 × 1/fx (800 ns)

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

2. TCL04 to TCL06: Bit 4 to bit 6 of timer clock select register 0 (TCL0)

3. Values in parentheses apply to operation with fx = 10.0 MHz

Figure 8-19. Square-Wave Output Operation Timings



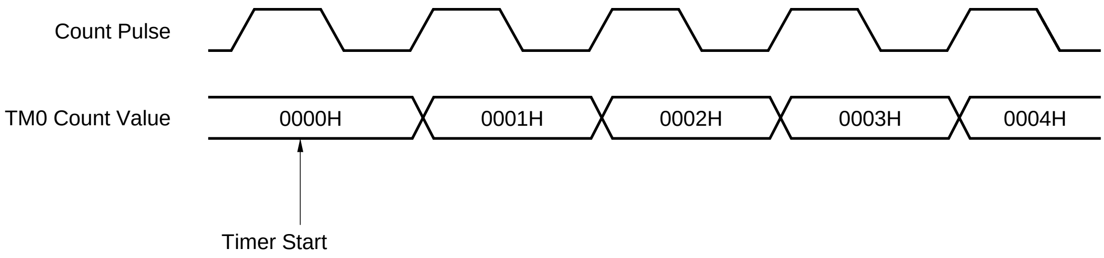
Note Initial value of TO0 output can be set at bits 2 and 3 (LVR0 and LVS0) of the 16-bit timer output control register (TOC0).

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 8-20. 16-Bit Timer Register Start Timings



(2) 16-bit compare register set

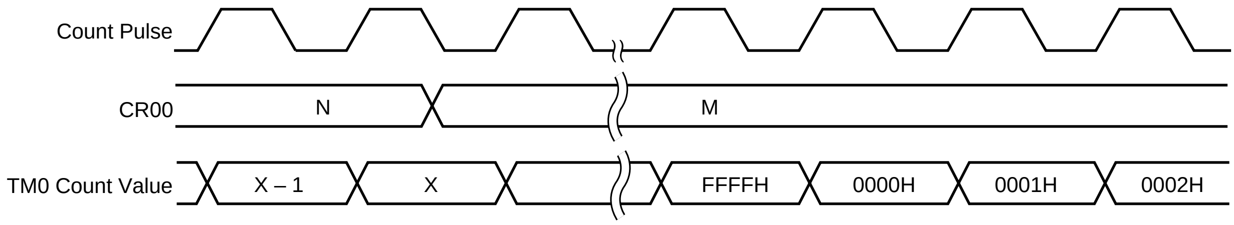
Set a value other than 0000H to the 16-bit compare register (CR00).

Thus, when using the 16-bit 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 the 16-bit compare register (CR00) is changed is smaller than that of the 16-bit timer register (TM0), TM0 continues counting and then restarts counting from 0. Therefore, if the value after CR00 changes (M) is smaller than the value before change (N), it is necessary to restart the timer after changing CR00.

Figure 8-21. Timings after Change of Compare Register during Timer Count Operation

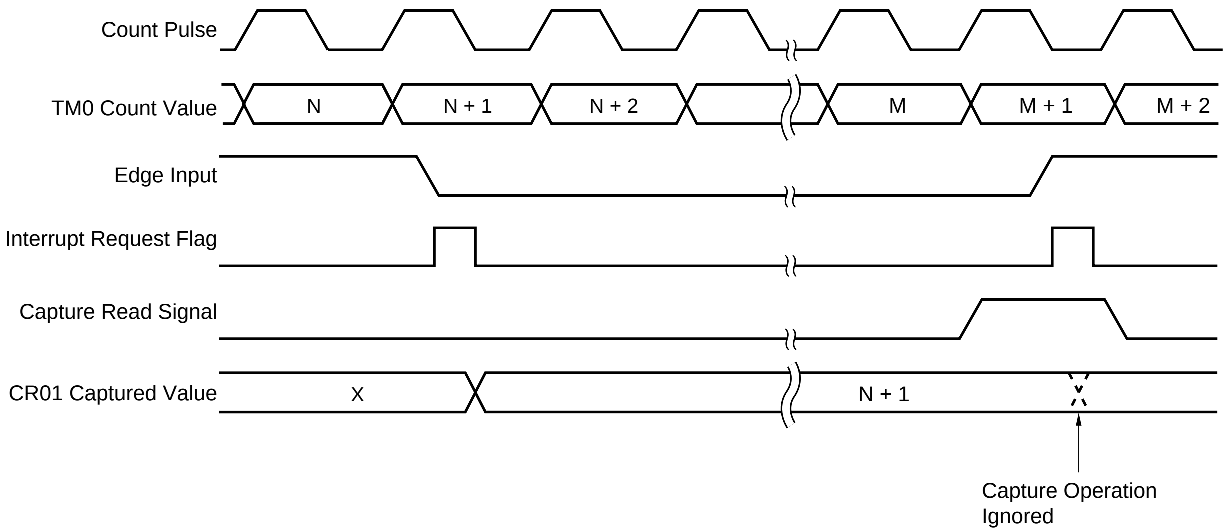


Remark N > X > M

(4) Capture register data retention timings

If the valid edge of the TI0/P00 pin is input during 16-bit capture register (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 8-22. Capture Register Data Retention Timings



(5) Valid edge set

Set the valid edge of the TI0/INTP0/P00 pin after setting bits 1 to 3 (TMC01 to TMC03) of the 16-bit timer mode control register (TMC0) 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 the external interrupt mode register (INTM0).

(6) OVF0 flag operation

OVF0 flag is set to 1:

When clear & start mode on match between TM0 and CR00 is selected

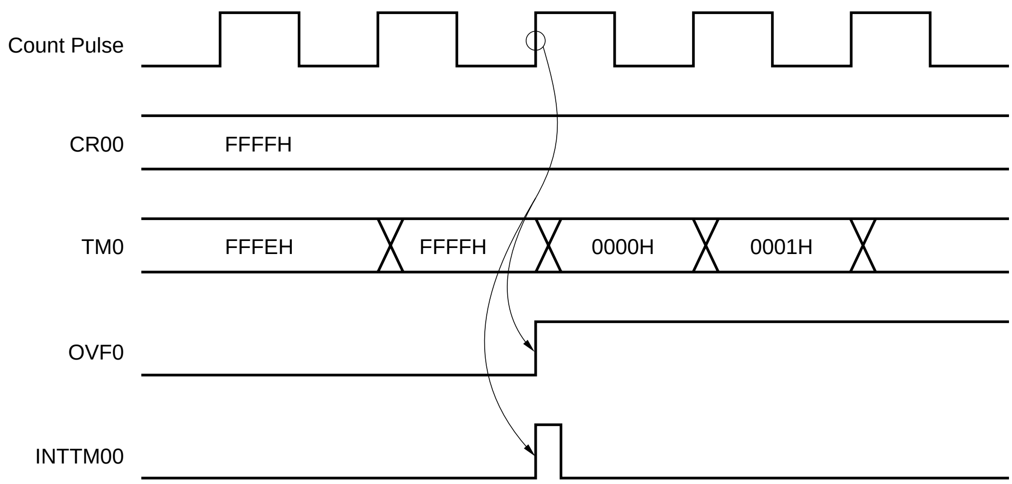
↓

CR00 is set to FFFFH

↓

TM0 is counted up from FFFFH to 0000H

Figure 8-23. OVF0 Flag Operation Timing



CHAPTER 9 8-BIT TIMER/EVENT COUNTER

9.1 8-Bit Timer/Event Counter Functions

For the 8-bit timer/event counter incorporated in the μPD78014 and 78014Y Subseries, the following two modes are available.

- 8-bit timer/event counter mode : two-channel 8-bit timer/event counters to be used separately
- 16-bit timer/event counter mode : two-channel 8-bit timer/event counters to be used together as 16-bit timer/ event counter

9.1.1 8-bit timer/event counter mode

The 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
$2^2 \times 1/f_X$ (400 ns)	$2^{10} \times 1/f_X$ (102.4 μs)	$2^2 \times 1/f_X$ (400 ns)
$2^3 \times 1/f_X$ (800 ns)	$2^{11} \times 1/f_X$ (204.8 μs)	$2^3 \times 1/f_X$ (800 ns)
$2^4 \times 1/f_X$ (1.6 μs)	$2^{12} \times 1/f_X$ (409.6 μs)	$2^4 \times 1/f_X$ (1.6 μs)
$2^5 \times 1/f_X$ (3.2 μs)	$2^{13} \times 1/f_X$ (819.2 μs)	$2^5 \times 1/f_X$ (3.2 μs)
$2^6 \times 1/f_X$ (6.4 μs)	$2^{14} \times 1/f_X$ (1.64 ms)	$2^6 \times 1/f_X$ (6.4 μs)
$2^7 \times 1/f_X$ (12.8 μs)	$2^{15} \times 1/f_X$ (3.28 ms)	$2^7 \times 1/f_X$ (12.8 μs)
$2^8 \times 1/f_X$ (25.6 μs)	$2^{16} \times 1/f_X$ (6.55 ms)	$2^8 \times 1/f_X$ (25.6 μs)
$2^9 \times 1/f_X$ (51.2 μs)	$2^{17} \times 1/f_X$ (13.1 ms)	$2^9 \times 1/f_X$ (51.2 μs)
$2^{10} \times 1/f_X$ (102.4 μs)	$2^{18} \times 1/f_X$ (26.2 ms)	$2^{10} \times 1/f_X$ (102.4 μs)
$2^{12} \times 1/f_X$ (409.6 μs)	$2^{20} \times 1/f_X$ (104.9 ms)	$2^{12} \times 1/f_X$ (409.6 μs)

Remarks 1. f_X: Main system clock oscillation frequency

2. Values in parentheses apply to operation with f_X = 10.0 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
$2^2 \times 1/f_X$ (400 ns)	$2^{10} \times 1/f_X$ (102.4 μs)	$2^2 \times 1/f_X$ (400 ns)
$2^3 \times 1/f_X$ (800 ns)	$2^{11} \times 1/f_X$ (204.8 μs)	$2^3 \times 1/f_X$ (800 ns)
$2^4 \times 1/f_X$ (1.6 μs)	$2^{12} \times 1/f_X$ (409.6 μs)	$2^4 \times 1/f_X$ (1.6 μs)
$2^5 \times 1/f_X$ (3.2 μs)	$2^{13} \times 1/f_X$ (819.2 μs)	$2^5 \times 1/f_X$ (3.2 μs)
$2^6 \times 1/f_X$ (6.4 μs)	$2^{14} \times 1/f_X$ (1.64 ms)	$2^6 \times 1/f_X$ (6.4 μs)
$2^7 \times 1/f_X$ (12.8 μs)	$2^{15} \times 1/f_X$ (3.28 ms)	$2^7 \times 1/f_X$ (12.8 μs)
$2^8 \times 1/f_X$ (25.6 μs)	$2^{16} \times 1/f_X$ (6.55 ms)	$2^8 \times 1/f_X$ (25.6 μs)
$2^9 \times 1/f_X$ (51.2 μs)	$2^{17} \times 1/f_X$ (13.1 ms)	$2^9 \times 1/f_X$ (51.2 μs)
$2^{10} \times 1/f_X$ (102.4 μs)	$2^{18} \times 1/f_X$ (26.2 ms)	$2^{10} \times 1/f_X$ (102.4 μs)
$2^{12} \times 1/f_X$ (409.6 μs)	$2^{20} \times 1/f_X$ (104.9 ms)	$2^{12} \times 1/f_X$ (409.6 μs)

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

2. Values in parentheses apply to operation with f_X = 10.0 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 Counters are Used as 16-Bit Timer/Event Counter

Minimum Interval Time	Maximum Interval Time	Resolution
$2^2 \times 1/f_X$ (400 ns)	$2^{18} \times 1/f_X$ (26.2 ms)	$2^2 \times 1/f_X$ (400 ns)
$2^3 \times 1/f_X$ (800 ns)	$2^{19} \times 1/f_X$ (52.4 ms)	$2^3 \times 1/f_X$ (800 ns)
$2^4 \times 1/f_X$ (1.6 μs)	$2^{20} \times 1/f_X$ (104.9 ms)	$2^4 \times 1/f_X$ (1.6 μs)
$2^5 \times 1/f_X$ (3.2 μs)	$2^{21} \times 1/f_X$ (209.7 ms)	$2^5 \times 1/f_X$ (3.2 μs)
$2^6 \times 1/f_X$ (6.4 μs)	$2^{22} \times 1/f_X$ (419.4 ms)	$2^6 \times 1/f_X$ (6.4 μs)
$2^7 \times 1/f_X$ (12.8 μs)	$2^{23} \times 1/f_X$ (838.9 ms)	$2^7 \times 1/f_X$ (12.8 μs)
$2^8 \times 1/f_X$ (25.6 μs)	$2^{24} \times 1/f_X$ (1.7 s)	$2^8 \times 1/f_X$ (25.6 μs)
$2^9 \times 1/f_X$ (51.2 μs)	$2^{25} \times 1/f_X$ (3.7 s)	$2^9 \times 1/f_X$ (51.2 μs)
$2^{10} \times 1/f_X$ (102.4 μs)	$2^{26} \times 1/f_X$ (6.7 s)	$2^{10} \times 1/f_X$ (102.4 μs)
$2^{12} \times 1/f_X$ (409.6 μs)	$2^{28} \times 1/f_X$ (26.8 s)	$2^{12} \times 1/f_X$ (409.6 μs)

Remarks 1. f_X: Main system clock oscillation frequency
2. Values in parentheses apply to operation with f_X = 10.0 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 Counters are Used as 16-Bit Timer/Event Counter

Minimum Pulse Width	Maximum Pulse Width	Resolution
$2^2 \times 1/f_X$ (400 ns)	$2^{18} \times 1/f_X$ (26.2 ms)	$2^2 \times 1/f_X$ (400 ns)
$2^3 \times 1/f_X$ (800 ns)	$2^{19} \times 1/f_X$ (52.4 ms)	$2^3 \times 1/f_X$ (800 ns)
$2^4 \times 1/f_X$ (1.6 μs)	$2^{20} \times 1/f_X$ (104.9 ms)	$2^4 \times 1/f_X$ (1.6 μs)
$2^5 \times 1/f_X$ (3.2 μs)	$2^{21} \times 1/f_X$ (209.7 ms)	$2^5 \times 1/f_X$ (3.2 μs)
$2^6 \times 1/f_X$ (6.4 μs)	$2^{22} \times 1/f_X$ (419.4 ms)	$2^6 \times 1/f_X$ (6.4 μs)
$2^7 \times 1/f_X$ (12.8 μs)	$2^{23} \times 1/f_X$ (838.9 ms)	$2^7 \times 1/f_X$ (12.8 μs)
$2^8 \times 1/f_X$ (25.6 μs)	$2^{24} \times 1/f_X$ (1.7 s)	$2^8 \times 1/f_X$ (25.6 μs)
$2^9 \times 1/f_X$ (51.2 μs)	$2^{25} \times 1/f_X$ (3.4 s)	$2^9 \times 1/f_X$ (51.2 μs)
$2^{10} \times 1/f_X$ (102.4 μs)	$2^{26} \times 1/f_X$ (6.7 s)	$2^{10} \times 1/f_X$ (102.4 μs)
$2^{12} \times 1/f_X$ (409.6 μs)	$2^{28} \times 1/f_X$ (26.8 s)	$2^{12} \times 1/f_X$ (409.6 μs)

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

2. Values in parentheses apply to operation with f_X = 10.0 MHz.

9.2 8-Bit Timer/Event Counter Configuration

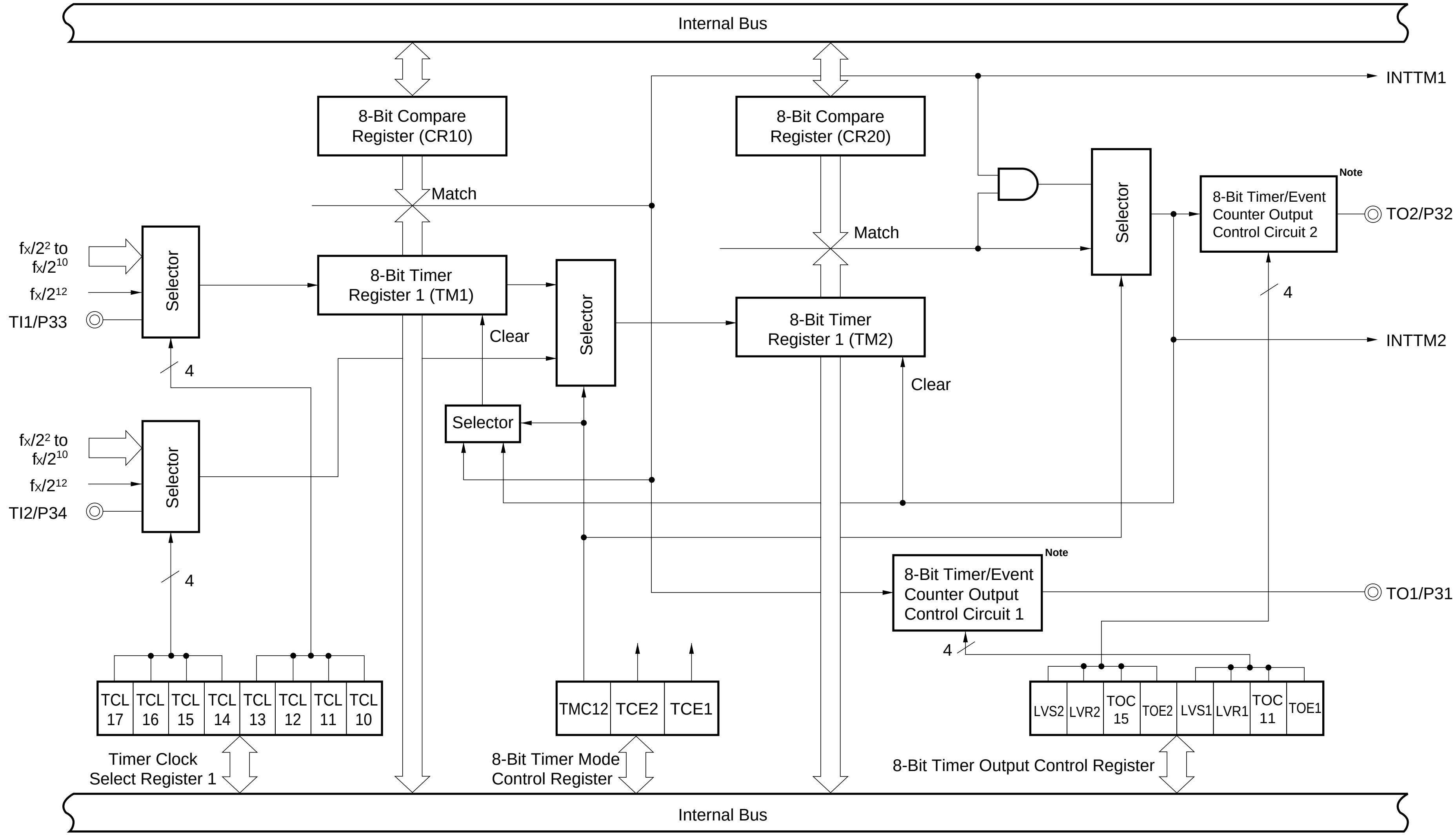
The 8-bit timer/event counter consists of the following hardware.

Table 9-5. 8-Bit Timer/Event Counter Configuration

Item	Configuration
Timer register	8-bits × 2 (TM1, TM2)
Register	8-bit compare register: 2 (CR10, CR20)
Timer output	2 (TO1, TO2)
Control registers	Timer clock select register 1 (TCL1) 8-bit timer mode control register (TMC1) 8-bit timer output control register (TOC1) Port mode register 3 (PM3)[Note]

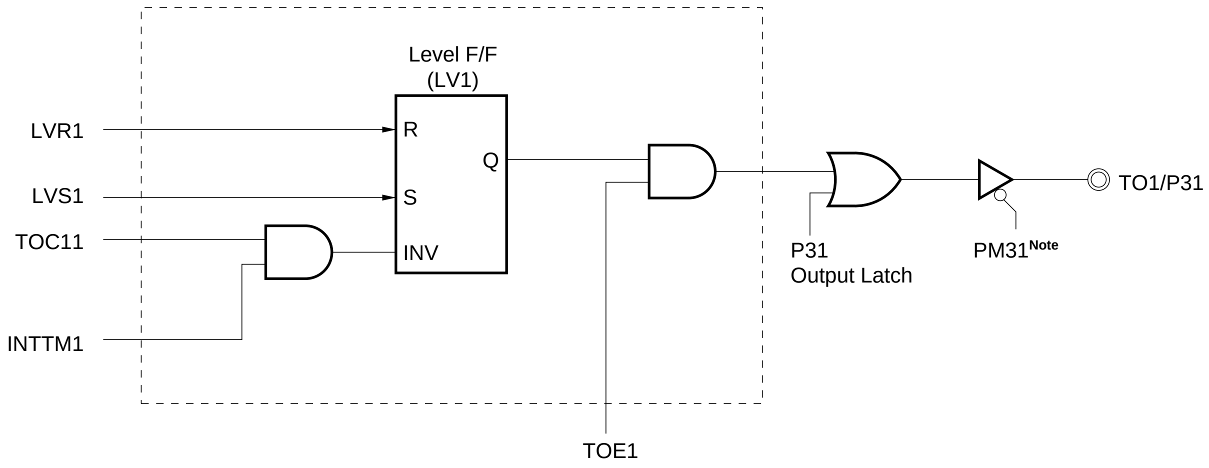
Note Refer to **Figure 6-10 P30 to P37 Block Diagrams**.

Figure 9-1. 8-Bit Timer/Event Counter Block Diagram



Note Refer to **Figures 9-2** and **9-3** for details of 8-bit timer/event counter output control circuits 1 and 2, respectively.

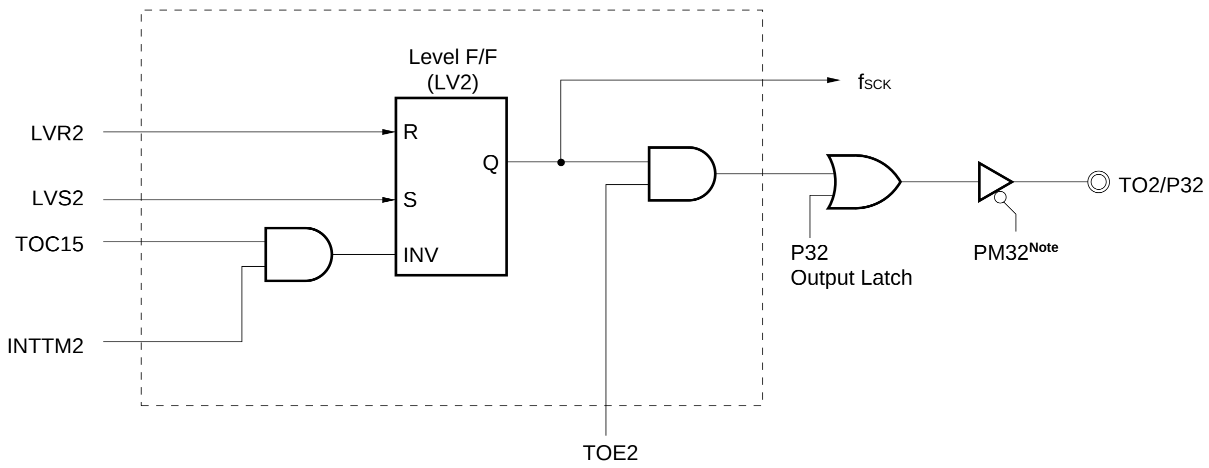
Figure 9-2. 8-Bit Timer/Event Counter Output Control Circuit 1 Block Diagram



Note Bit 1 of the port mode register 3 (PM3)

Remark The section in the broken line is an output control circuit.

Figure 9-3. 8-Bit Timer/Event Counter Output Control Circuit 2 Block Diagram



Note Bit 2 of the port mode register 3 (PM3)

Remarks 1. The section in the broken line is an output control circuit.

2. f_{SCK}: Serial clock frequency

(1) 8-bit compare registers (CR10, CR20)

These are 8-bit registers that compare the value set to CR10 with the 8-bit timer register 1 (TM1) count value, and the value set to CR20 with 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, they can be used as registers to hold interval time.

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 registers are used as 16-bit timer/event counter, the 0000H to FFFFH values can be set.

$\overline{\text{RESET}}$ input makes CR10 and CR20 undefined.

Cautions 1. When using the compare registers as 16-bit timer/event counter, be sure to set data after stopping timer operation.

2. When the values of CR10 and CR20 posterior to alteration are less than the values of the 8-bit timer registers (TM1 and TM2), TM1 and TM2 keep on counting and resume counting from 0 after an overflow. When the values of CR10 and CR20 posterior to alteration are less than the values prior to alteration, the timer must be restarted after CR10 and CR20 are altered.

(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 as 16-bit timer × 1-channel mode, 16-bit timer register (TMS) is read with a 16-bit memory manipulation instruction.

$\overline{\text{RESET}}$ input sets 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 (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 sets TCL1 to 00H.

Figure 9-4. Timer Clock Select Register 1 Format

Symbol	7	6	5	4	3	2	1	0	Address	When Reset	R/W
TCL1	TCL17	TCL16	TCL15	TCL14	TCL13	TCL12	TCL11	TCL10	FF41H	00H	R/W

TCL13	TCL12	TCL11	TCL10	8-bit Timer Register 1 Clock Selection
0	0	0	0	TI1 falling edge
0	0	0	1	TI1 rising edge
0	1	1	0	$f_X/2^2$ (2.5 MHz)
0	1	1	1	$f_X/2^3$ (1.25 MHz)
1	0	0	0	$f_X/2^4$ (625 kHz)
1	0	0	1	$f_X/2^5$ (313 kHz)
1	0	1	0	$f_X/2^6$ (156 kHz)
1	0	1	1	$f_X/2^7$ (78.1 kHz)
1	1	0	0	$f_X/2^8$ (39.1 kHz)
1	1	0	1	$f_X/2^9$ (19.5 kHz)
1	1	1	0	$f_X/2^{10}$ (9.8 kHz)
1	1	1	1	$f_X/2^{12}$ (2.4 kHz)
Other than above				Setting prohibited

TCL17	TCL16	TCL15	TCL14	8-bit Timer Register 2 Clock Selection
0	0	0	0	TI2 falling edge
0	0	0	1	TI2 rising edge
0	1	1	0	$f_X/2^2$ (2.5 MHz)
0	1	1	1	$f_X/2^3$ (1.25 MHz)
1	0	0	0	$f_X/2^4$ (625 kHz)
1	0	0	1	$f_X/2^5$ (313 kHz)
1	0	1	0	$f_X/2^6$ (156 kHz)
1	0	1	1	$f_X/2^7$ (78.1 kHz)
1	1	0	0	$f_X/2^8$ (39.1 kHz)
1	1	0	1	$f_X/2^9$ (19.5 kHz)
1	1	1	0	$f_X/2^{10}$ (9.8 kHz)
1	1	1	1	$f_X/2^{12}$ (2.4 kHz)
Other than above				Setting prohibited

Caution If data other than identical data is to be rewritten to TCL1, the timer operation must be stopped first.

Remarks
1. f_X : Main system clock oscillation frequency
2. TI1 : 8-bit timer register 1 input pin
3. TI2 : 8-bit timer register 2 input pin
4. Values in parentheses apply to operation with f_X = 10.0 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 sets TMC1 to 00H.

Figure 9-5. 8-Bit Timer Mode Control Register Format

Symbol	7	6	5	4	3	2	<1>	<0>	Address	When Reset	R/W
TMC1	0	0	0	0	0	TMC12	TCE2	TCE1	FF49H	00H	R/W

TCE1	8-Bit Timer Register 1 Operation Control
0	Operation stop (TM1 clear to 0)
1	Operation enable

TCE2	8-Bit Timer Register 2 Operation Control
0	Operation stop (TM2 clear to 0)
1	Operation enable

TMC12	Operating Mode Selection
0	8-Bit timer register × 2-channel mode (TM1, TM2)
1	16-Bit timer register × 1-channel mode (TMS)

Cautions 1. Switch the operating mode after stopping timer operation.

2. When used as 16-bit timer register (TMS), 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 control circuits 1 and 2.

It sets/resets the R-S flip-flops (LV1 and LV2) and enables/disables inversion and 8-bit 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 sets TOC1 to 00H.

Figure 9-6. 8-Bit Timer Output Control Register Format

Symbol	<7>	<6>	5	<4>	<3>	<2>	1	<0>	Address	When Reset	R/W
TOC1	LVS2	LVR2	TOC15	TOE2	LVS1	LVR1	TOC11	TOE1	FF4FH	00H	R/W

TOE1	8-Bit Timer/Event Counter 1 Output Control
0	Output disable (port mode)
1	Output enable

TOC11	8-Bit Timer/Event Counter 1 Timer Output F/F Control
0	Inverted operation disable
1	Inverted operation 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 (0)
1	0	Timer output F/F reset (1)
1	1	Setting prohibited

TOE2	8-Bit Timer/Event Counter 2 Output Control
0	Output disable (port mode)
1	Output enable

TOC15	8-Bit Timer/Event Counter 2 Timer Output F/F Status Control
0	Inverted operation disable
1	Inverted operation enable

LVS2	LVR2	8-Bit Timer/Event Counter 2 Timer Output F/F Status Set
0	0	Unchanged
0	1	Timer output F/F reset (0)
1	0	Timer output F/F reset (1)
1	1	Setting prohibited

Cautions 1. Be sure to set TOC1 after stopping timer operation.

2. After data setting, 0 can be read from LVS1, LVS2, LVR1, and LVR2.

(4) Port mode register 3 (PM3)

This register sets port 3 input/output bit-wise.

When using the P31/TO1 and P32/TO2 pins for timer output, set output latches PM31, PM32, and P31, 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	When Reset	R/W
PM3	PM37	PM36	PM35	PM34	PM33	PM32	PM31	PM30	FF23H	FFH	R/W

PM3n	P3n Pin Input/Output 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 (CR10 and CR20).

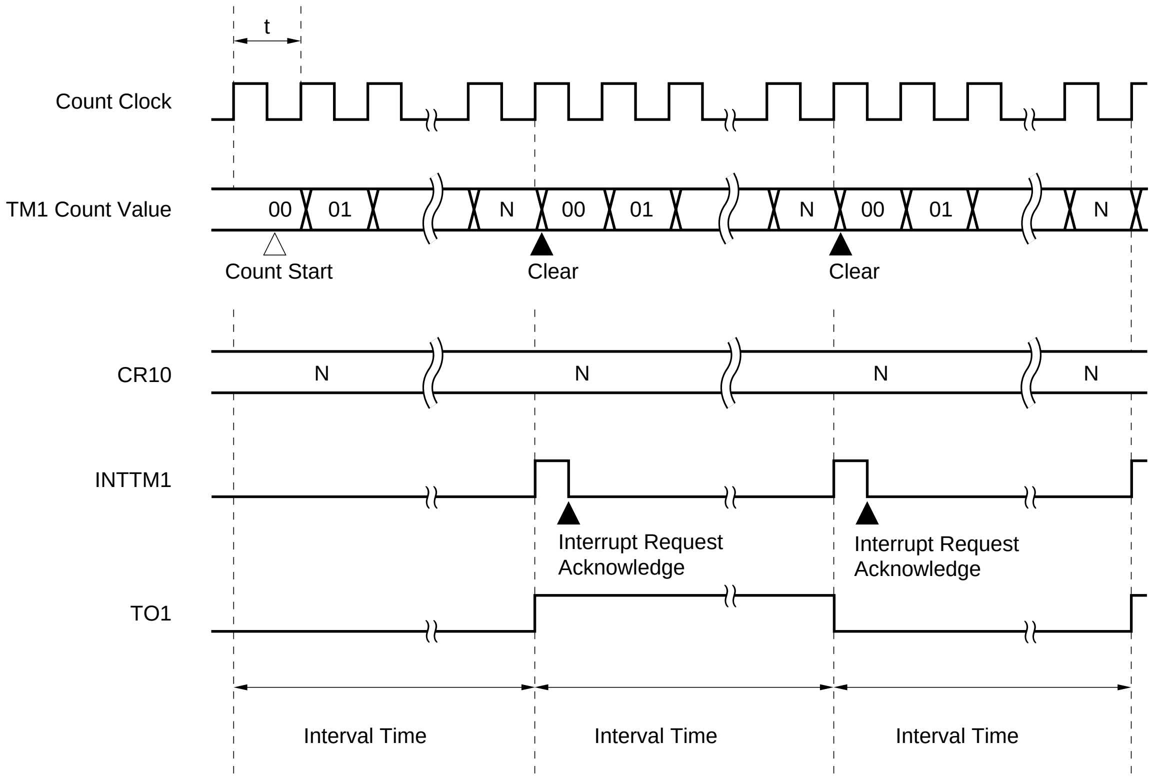
When the count values of the 8-bit timer registers 1 and 2 (TM1 and TM2) match the values set to CR10 and CR20, counting continues with the TM1 and TM2 values cleared to 0 and the interrupt request signals (INTTM1 and INTTM2) are generated.

Count clock of TM1 can be selected with bits 0 to 3 (TCL10 toTCL13) of the timer clock select register 1 (TCL1).

Count clock of TM2 can be selected with bits 4 to 7 (TCL14 toTCL17) of the timer clock select register 1 (TCL1).

For the operation after changing compare register value during timer count operation, refer to **9.5 Cautions on 8-Bit Timer/Event Counter (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 Times

TCL13	TCL12	TCL11	TCL10	Minimum Interval Time	Maximum Interval Time	Resolution
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	$2^2 \times 1/f_X$ (400 ns)	$2^{10} \times 1/f_X$ (102.4 μs)	$2^2 \times 1/f_X$ (400 ns)
0	1	1	1	$2^3 \times 1/f_X$ (800 ns)	$2^{11} \times 1/f_X$ (204.8 μs)	$2^3 \times 1/f_X$ (800 ns)
1	0	0	0	$2^4 \times 1/f_X$ (1.6 μs)	$2^{12} \times 1/f_X$ (409.6 μs)	$2^4 \times 1/f_X$ (1.6 μs)
1	0	0	1	$2^5 \times 1/f_X$ (3.2 μs)	$2^{13} \times 1/f_X$ (819.2 μs)	$2^5 \times 1/f_X$ (3.2 μs)
1	0	1	0	$2^6 \times 1/f_X$ (6.4 μs)	$2^{14} \times 1/f_X$ (1.64 ms)	$2^6 \times 1/f_X$ (6.4 μs)
1	0	1	1	$2^7 \times 1/f_X$ (12.8 μs)	$2^{15} \times 1/f_X$ (3.28 ms)	$2^7 \times 1/f_X$ (12.8 μs)
1	1	0	0	$2^8 \times 1/f_X$ (25.6 μs)	$2^{16} \times 1/f_X$ (6.55 ms)	$2^8 \times 1/f_X$ (25.6 μs)
1	1	0	1	$2^9 \times 1/f_X$ (51.2 μs)	$2^{17} \times 1/f_X$ (13.1 ms)	$2^9 \times 1/f_X$ (51.2 μs)
1	1	1	0	$2^{10} \times 1/f_X$ (102.4 μs)	$2^{18} \times 1/f_X$ (26.2 ms)	$2^{10} \times 1/f_X$ (102.4 μs)
1	1	1	1	$2^{12} \times 1/f_X$ (409.6 μs)	$2^{20} \times 1/f_X$ (104.9 ms)	$2^{12} \times 1/f_X$ (409.6 μs)
Other than above				Setting prohibited		

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

2. TCL10 to TCL13: Bits 0 to 3 of the timer clock select register 1 (TCL1)

3. Values in parentheses apply to operation with f_X = 10.0 MHz.

Table 9-7. 8-Bit Timer/Event Counter 2 Interval Times

TCL17	TCL16	TCL15	TCL14	Minimum Interval Time	Maximum Interval Time	Resolution
0	0	0	0	TI2 input cycle	$2^8 \times$ TI2 input cycle	TI2 input edge cycle
0	0	0	1	TI2 input cycle	$2^8 \times$ TI2 input cycle	TI2 input edge cycle
0	1	1	0	$2^2 \times 1/f_X$ (400 ns)	$2^{10} \times 1/f_X$ (102.4 μs)	$2^2 \times 1/f_X$ (400 ns)
0	1	1	1	$2^3 \times 1/f_X$ (800 ns)	$2^{11} \times 1/f_X$ (204.8 μs)	$2^3 \times 1/f_X$ (800 ns)
1	0	0	0	$2^4 \times 1/f_X$ (1.6 μs)	$2^{12} \times 1/f_X$ (409.6 μs)	$2^4 \times 1/f_X$ (1.6 μs)
1	0	0	1	$2^5 \times 1/f_X$ (3.2 μs)	$2^{13} \times 1/f_X$ (819.2 μs)	$2^5 \times 1/f_X$ (3.2 μs)
1	0	1	0	$2^6 \times 1/f_X$ (6.4 μs)	$2^{14} \times 1/f_X$ (1.64 ms)	$2^6 \times 1/f_X$ (6.4 μs)
1	0	1	1	$2^7 \times 1/f_X$ (12.8 μs)	$2^{15} \times 1/f_X$ (3.28 ms)	$2^7 \times 1/f_X$ (12.8 μs)
1	1	0	0	$2^8 \times 1/f_X$ (25.6 μs)	$2^{16} \times 1/f_X$ (6.55 ms)	$2^8 \times 1/f_X$ (25.6 μs)
1	1	0	1	$2^9 \times 1/f_X$ (51.2 μs)	$2^{17} \times 1/f_X$ (13.1 ms)	$2^9 \times 1/f_X$ (51.2 μs)
1	1	1	0	$2^{10} \times 1/f_X$ (102.4 μs)	$2^{18} \times 1/f_X$ (26.2 ms)	$2^{10} \times 1/f_X$ (102.4 μs)
1	1	1	1	$2^{12} \times 1/f_X$ (409.6 μs)	$2^{20} \times 1/f_X$ (104.9 ms)	$2^{12} \times 1/f_X$ (409.6 μs)
Other than above				Setting prohibited		

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

2. TCL14 to TCL17: Bits 4 to 7 of the timer clock select register 1 (TCL1)

3. Values in parentheses apply to operation with f_X = 10.0 MHz.

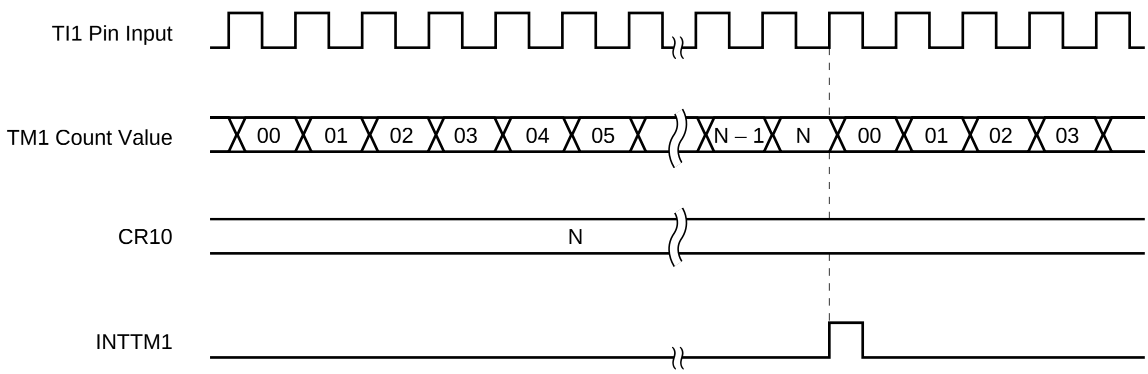
(2) External event counter operation

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 the 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 (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

The 8-bit timer/event counter operates as a square wave with any selected frequency which is output at intervals of the value preset to 8-bit compare registers (CR10 and CR20).

The TO1/P31 or TO2/P32 pin output status is inverted 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

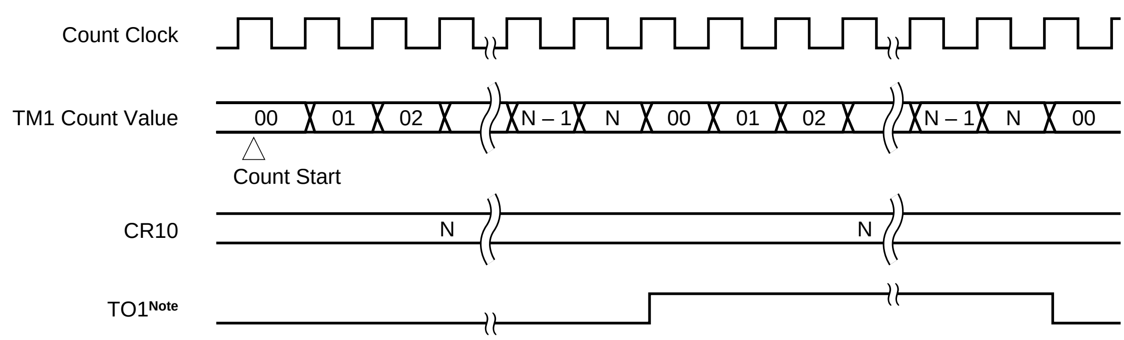
TCL13	TCL12	TCL11	TCL10	Minimum Pulse Width	Maximum Pulse Width	Resolution
0	1	1	0	$2^2 \times 1/f_X$ (400 ns)	$2^{10} \times 1/f_X$ (102.4 μs)	$2^2 \times 1/f_X$ (400 ns)
0	1	1	1	$2^3 \times 1/f_X$ (800 ns)	$2^{11} \times 1/f_X$ (204.8 μs)	$2^3 \times 1/f_X$ (800 ns)
1	0	0	0	$2^4 \times 1/f_X$ (1.6 μs)	$2^{12} \times 1/f_X$ (409.6 μs)	$2^4 \times 1/f_X$ (1.6 μs)
1	0	0	1	$2^5 \times 1/f_X$ (3.2 μs)	$2^{13} \times 1/f_X$ (819.2 μs)	$2^5 \times 1/f_X$ (3.2 μs)
1	0	1	0	$2^6 \times 1/f_X$ (6.4 μs)	$2^{14} \times 1/f_X$ (1.64 ms)	$2^6 \times 1/f_X$ (6.4 μs)
1	0	1	1	$2^7 \times 1/f_X$ (12.8 μs)	$2^{15} \times 1/f_X$ (3.28 ms)	$2^7 \times 1/f_X$ (12.8 μs)
1	1	0	0	$2^8 \times 1/f_X$ (25.6 μs)	$2^{16} \times 1/f_X$ (6.55 ms)	$2^8 \times 1/f_X$ (25.6 μs)
1	1	0	1	$2^9 \times 1/f_X$ (51.2 μs)	$2^{17} \times 1/f_X$ (13.1 ms)	$2^9 \times 1/f_X$ (51.2 μs)
1	1	1	0	$2^{10} \times 1/f_X$ (102.4 μs)	$2^{18} \times 1/f_X$ (26.2 ms)	$2^{10} \times 1/f_X$ (102.4 μs)
1	1	1	1	$2^{12} \times 1/f_X$ (409.6 μs)	$2^{20} \times 1/f_X$ (104.9 ms)	$2^{12} \times 1/f_X$ (409.6 μs)

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

2. TCL10 to TCL13: Bit 0 to bit 3 of timer clock select register 1 (TCL1)

3. Values in parentheses apply to operation with f_X = 10.0 MHz.

★ **Figure 9-10. Square-Wave Output Operation Timings**



Note Initial value of TO1 output can be set at bits 2 and 3 (LVS1 and LVR1) of the 8-bit timer output control register (TOC1).

9.4.2 16-bit timer/event counter mode

When bit 2 (TMC12) of the 8-bit timer mode control register (TMC1) is set to 1, the 16-bit timer/event counter mode is selected.

The count clocks are selected with bits 0 to 3 (TCL10 to TCL13) of timer clock select register (TCL1). The overflow signal of 8-bit timer/event counter 1 (TM1) becomes a count clock of 8-bit timer/event counter 2 (TM2). Count operation enable/disable is selected with bit 0 (TCE1) of TMC1.

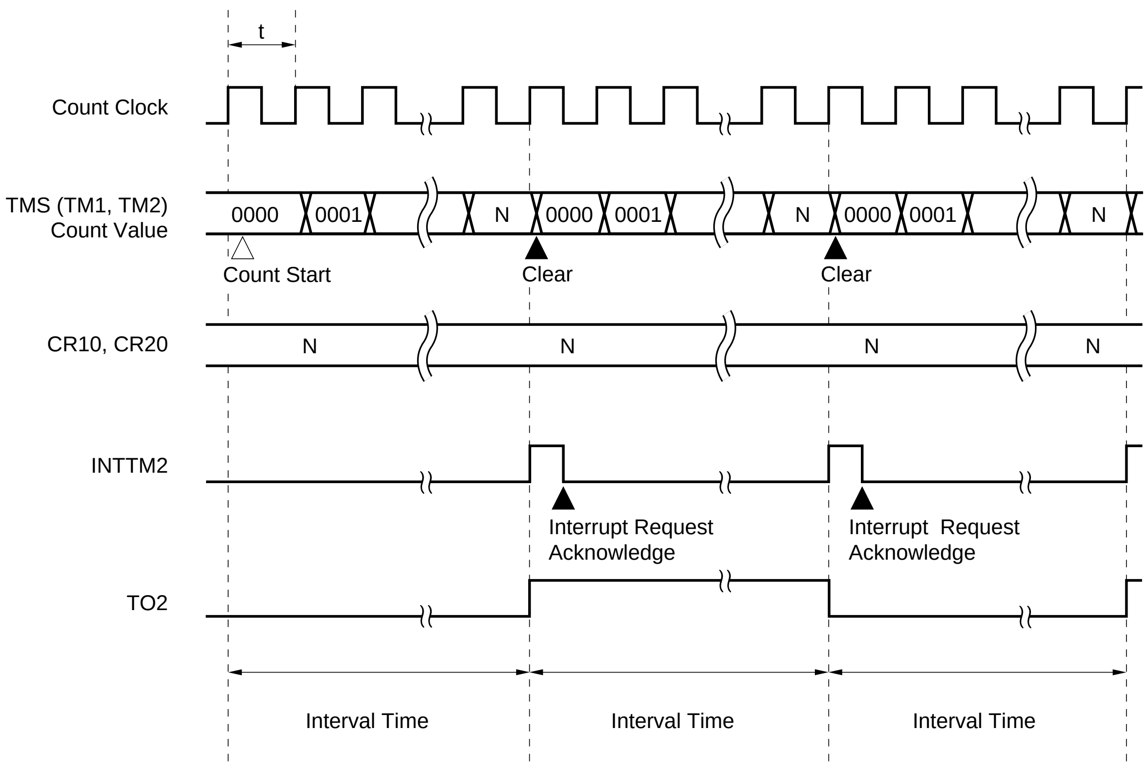
(1) Interval timer operation

The 8-bit timer/event counter operates as interval timer which generates interrupt requests repeatedly at intervals of the count value preset to 2-channel 8-bit compare registers (CR10 and CR20). When setting the count value, the upper 8-bit value is set as CR20 and the lower 8-bit value as CR10. For the count value (interval time) which can be set refer to **Table 9-9**.

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 timings of the interval timer, refer to **Figure 9-11**.

Count clock can be selected with bits 0 to 3 (TCL10 to TCL13) of the timer clock select register 1 (TCL1). The overflow signal of the TM1 becomes a count clock of the TM2.

Figure 9-11. Interval Timer Operation Timings



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 control circuit 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 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
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	$2^2 \times 1/f_X$ (400 ns)	$2^{18} \times 1/f_X$ (26.2 ms)	$2^2 \times 1/f_X$ (400 ns)
0	1	1	1	$2^3 \times 1/f_X$ (800 ns)	$2^{19} \times 1/f_X$ (52.4 ms)	$2^3 \times 1/f_X$ (800 ns)
1	0	0	0	$2^4 \times 1/f_X$ (1.6 μs)	$2^{20} \times 1/f_X$ (104.9 ms)	$2^4 \times 1/f_X$ (1.6 μs)
1	0	0	1	$2^5 \times 1/f_X$ (3.2 μs)	$2^{21} \times 1/f_X$ (209.7 ms)	$2^5 \times 1/f_X$ (3.2 μs)
1	0	1	0	$2^6 \times 1/f_X$ (6.4 μs)	$2^{22} \times 1/f_X$ (419.4 ms)	$2^6 \times 1/f_X$ (6.4 μs)
1	0	1	1	$2^7 \times 1/f_X$ (12.8 μs)	$2^{23} \times 1/f_X$ (838.9 ms)	$2^7 \times 1/f_X$ (12.8 μs)
1	1	0	0	$2^8 \times 1/f_X$ (25.6 μs)	$2^{24} \times 1/f_X$ (1.7 s)	$2^8 \times 1/f_X$ (25.6 μs)
1	1	0	1	$2^9 \times 1/f_X$ (51.2 μs)	$2^{25} \times 1/f_X$ (3.4 s)	$2^9 \times 1/f_X$ (51.2 μs)
1	1	1	0	$2^{10} \times 1/f_X$ (102.4 μs)	$2^{26} \times 1/f_X$ (6.7 s)	$2^{10} \times 1/f_X$ (102.4 μs)
1	1	1	1	$2^{12} \times 1/f_X$ (409.6 μs)	$2^{28} \times 1/f_X$ (26.8 s)	$2^{12} \times 1/f_X$ (409.6 μs)
Other than above				Setting prohibited		

Remarks **1.** f_X: Main system clock oscillation frequency
2. TCL10 to TCL13: Bits 0 to 3 of the timer clock select register 1 (TCL1)
3. Values in parentheses apply to operation with f_X = 10.0 MHz.

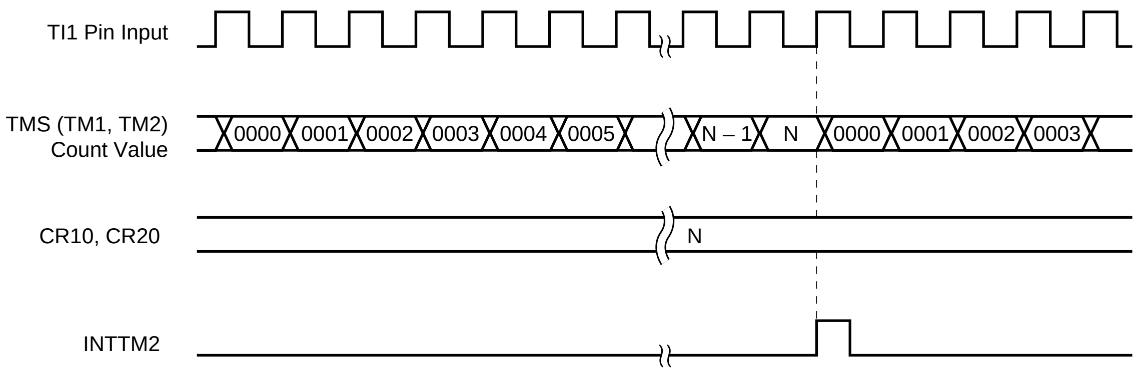
(2) External event counter operations

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 the timer clock select register 1 (TCL1) is input. When TM1 overflows, TM2 is incremented with the overflow signal as the 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 (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 control circuit 1 is inverted. Thus, when using 8-bit timer/event counter as 16-bit interval timer, set the mask flag TMMK1 to 1 to disable INTTM1 acknowledgment.

When reading 16-bit timer (TMS) count value, use the 16-bit memory manipulation instruction.

(3) Square-wave output operation

The 8-bit timer/event counter operates as a square wave with any selected frequency which is output at intervals of the value preset to 8-bit compare registers (CR10 and CR20). When setting the count value, the upper 8-bit value is set as CR20 and the lower 8-bit value as CR10.

The TO2/P32 pin output status is inverted 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 Tmer/Event Counter

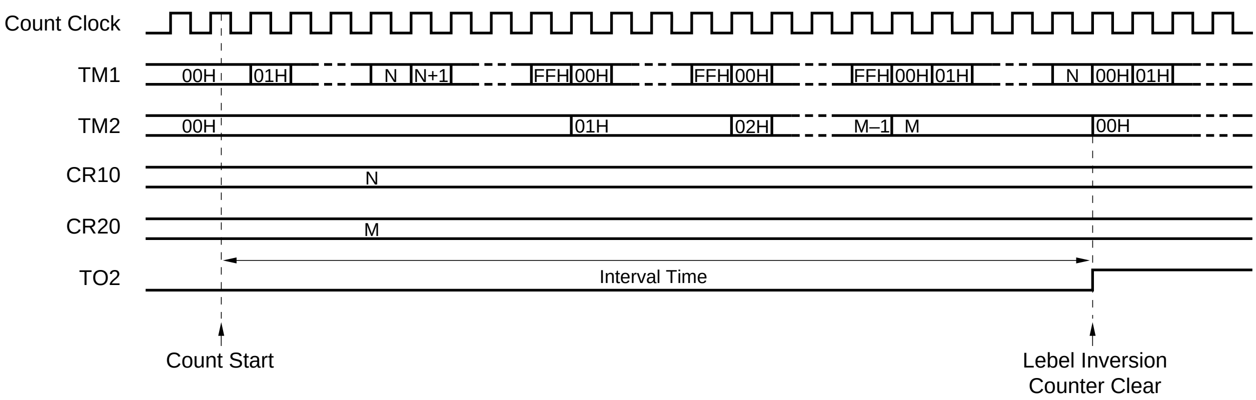
TCL13	TCL12	TCL11	TCL10	Minimum Pulse Width	Maximum Pulse Width	Resolution
0	1	1	0	$2^2 \times 1/f_X$ (400 ns)	$2^{18} \times 1/f_X$ (26.2 ms)	$2^2 \times 1/f_X$ (400 ns)
0	1	1	1	$2^3 \times 1/f_X$ (800 ns)	$2^{19} \times 1/f_X$ (52.4 ms)	$2^3 \times 1/f_X$ (800 ns)
1	0	0	0	$2^4 \times 1/f_X$ (1.6 μs)	$2^{20} \times 1/f_X$ (104.9 ms)	$2^4 \times 1/f_X$ (1.6 μs)
1	0	0	1	$2^5 \times 1/f_X$ (3.2 μs)	$2^{21} \times 1/f_X$ (209.7 ms)	$2^5 \times 1/f_X$ (3.2 μs)
1	0	1	0	$2^6 \times 1/f_X$ (6.4 μs)	$2^{22} \times 1/f_X$ (419.4 ms)	$2^6 \times 1/f_X$ (6.4 μs)
1	0	1	1	$2^7 \times 1/f_X$ (12.8 μs)	$2^{23} \times 1/f_X$ (838.9 ms)	$2^7 \times 1/f_X$ (12.8 μs)
1	1	0	0	$2^8 \times 1/f_X$ (25.6 μs)	$2^{24} \times 1/f_X$ (1.7 s)	$2^8 \times 1/f_X$ (25.6 μs)
1	1	0	1	$2^9 \times 1/f_X$ (51.2 μs)	$2^{25} \times 1/f_X$ (3.4 s)	$2^9 \times 1/f_X$ (51.2 μs)
1	1	1	0	$2^{10} \times 1/f_X$ (102.4 μs)	$2^{26} \times 1/f_X$ (6.7 s)	$2^{10} \times 1/f_X$ (102.4 μs)
1	1	1	1	$2^{12} \times 1/f_X$ (409.6 μs)	$2^{28} \times 1/f_X$ (26.8 s)	$2^{12} \times 1/f_X$ (409.6 μs)

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

2. TCL10 to TCL13: Bits 0 to 3 of the timer clock select register 1 (TCL1)

3. Values in parentheses apply to operation with f_X = 10.0 MHz.

Figure 9-13. Square-Wave Output Operation Timings

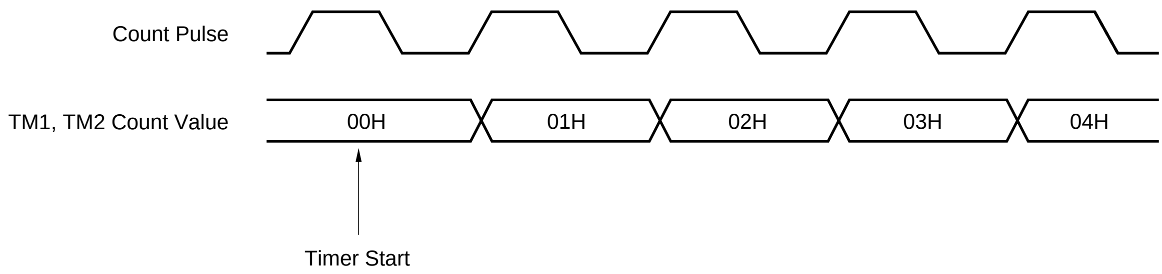


9.5 Cautions on 8-Bit Timer/Event Counter Operating

(1) Timer start errors

An error of one clock maximum 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 Timings



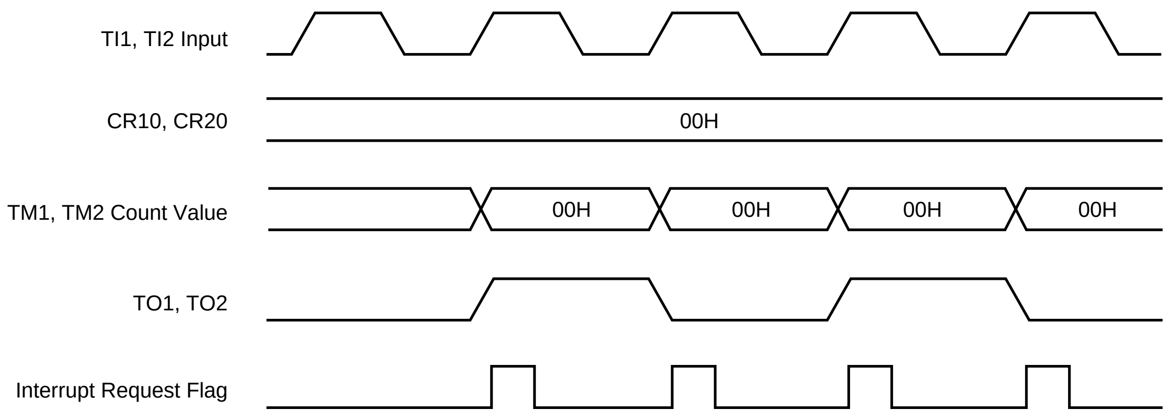
(2) 8-bit compare registers 1 and 2 sets

The 8-bit compare registers (CR10 and CR20) can be set to 00H.

Therefore, when the 8-bit compare register is used as event counter, one-pulse count operation can be carried out.

When the 8-bit compare registers are used as 16-bit timer/event counter, write data to CR10 and CR20 after setting bit 0 (TCE1) of the 8-bit timer mode control register (TMC1) to 0 and stopping timer operation.

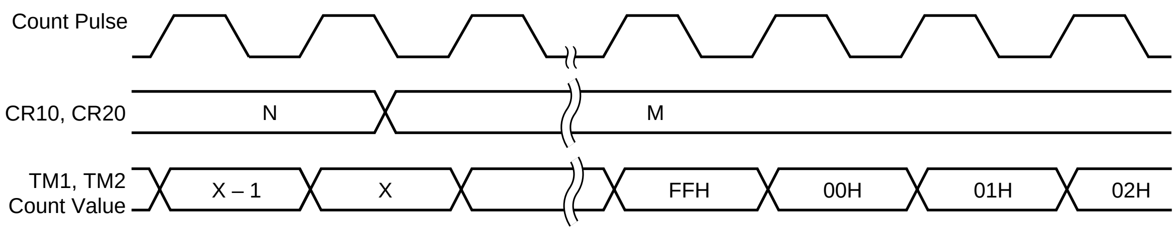
Figure 9-15. External Event Counter Operation Timings



(3) Operation after compare register change during timer count operation

If the values after the 8-bit compare registers (CR10 and CR20) are changed are smaller than those of 8-bit timer registers (TM1 and TM2), TM1 and TM2 continue counting, overflow and then restart counting from 0. Thus, if the value after CR10 and CR20 (M) change is smaller than that before change (N), it is necessary to restart the timer after changing CR10 and CR20.

Figure 9-16. Timings after Compare Register Change during Timer Count Operation



Remark N > X > M

[MEMO]

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 8.38 MHz main system clock is used, a flag (WTIF) is set at 0.5 second or 0.25 second intervals. In addition, when the 4.19 MHz (Standard: 4.194304 MHz) main system clock is used, a flag (WTIF) is set at 0.5 second or 1 second intervals.

When a frequency other than above is used, a flag (WTIF) is not set at 0.5/0.25 or 0.5/1.0 intervals.

Caution   When 8.38 MHz or 4.19 MHz frequency is used, a time interval has a little error.

(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_X = 10.0 MHz	When operated at f_X = 8.38 MHz	When operated at f_X = 4.19 MHz	When operated at f_{XT} = 32.768 kHz
$2^4 \times 1/f_W$	409.6 μs	489 μs	978 μs	488 μs
$2^5 \times 1/f_W$	819.2 μs	978 μs	1.96 ms	977 μs
$2^6 \times 1/f_W$	1.64 ms	1.96 ms	3.91 ms	1.95 ms
$2^7 \times 1/f_W$	3.28 ms	3.91 ms	7.82 ms	3.91 ms
$2^8 \times 1/f_W$	6.55 ms	7.82 ms	15.6 ms	7.81 ms
$2^9 \times 1/f_W$	13.1 ms	15.6 ms	31.3 ms	15.6 ms

Remarks f_X : Main system clock oscillation frequency

f_{XT} : Subsystem clock oscillation frequency

f_W : Watch timer clock frequency ($f_X/2^8$ 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)(Refer to **Figure 10-2.**)
This register sets the watch timer count clock.
TCL2 is set with an 8-bit memory manipulation instruction.
$\overline{\text{RESET}}$ input sets 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

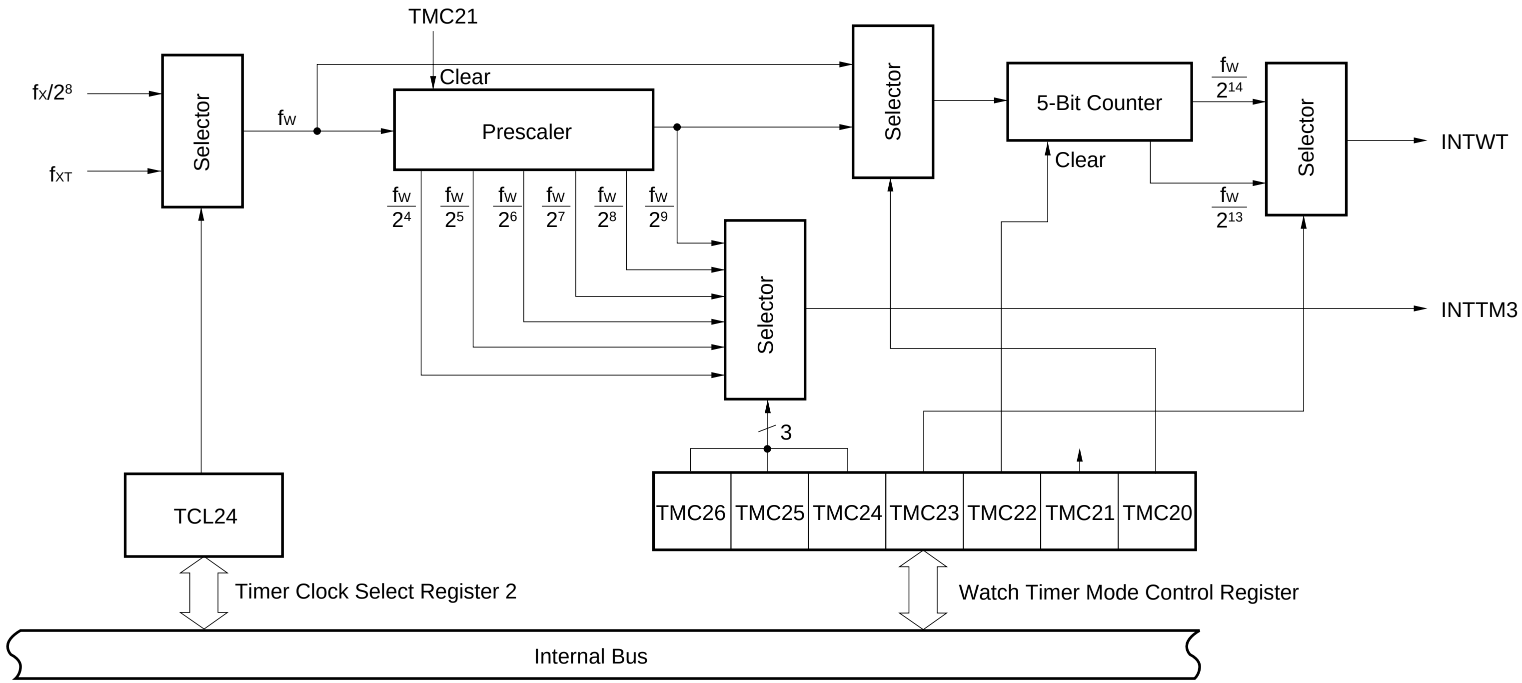


Figure 10-2. Timer Clock Select Register 2 Format

Symbol	7	6	5	4	3	2	1	0	Address	When Reset	R/W
TCL2	TCL27	TCL26	TCL25	TCL24	0	TCL22	TCL21	TCL20	FF42H	00H	R/W

TCL22	TCL21	TCL20	Watchdog Timer Count Clock Selection
0	0	0	$f_X/2^4$ (625 kHz)
0	0	1	$f_X/2^5$ (313 kHz)
0	1	0	$f_X/2^6$ (156 kHz)
0	1	1	$f_X/2^7$ (78.1 kHz)
1	0	0	$f_X/2^8$ (39.1 kHz)
1	0	1	$f_X/2^9$ (19.5 kHz)
1	1	0	$f_X/2^{10}$ (9.8 kHz)
1	1	1	$f_X/2^{12}$ (2.4 kHz)

TCL24	Watch Timer Count Clock Selection
0	$f_X/2^8$ (39.1 kHz)
1	f_{XT} (32.768 kHz)

TCL27	TCL26	TCL25	Buzzer Output Frequency Selection
0	×	×	Buzzer output disable
1	0	0	$f_X/2^{10}$ (9.8 kHz)
1	0	1	$f_X/2^{11}$ (4.9 kHz)
1	1	0	$f_X/2^{12}$ (2.4 kHz)
1	1	1	Setting prohibited

Caution **If data other than identical data is to be rewritten to TCL2, the timer operation must be stopped first.**

Remarks
1. f_X : Main system clock oscillation frequency
2. f_{XT} : Subsystem clock oscillation frequency
3. × : don't care
4. Values in parentheses apply to operation with f_X = 10.0 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 sets TMC2 to 00H.

Figure 10-3. Watch Timer Mode Control Register Format

Symbol	7	6	5	4	3	2	1	0	Address	When Reset	R/W
TMC2	0	TMC26	TMC25	TMC24	TMC23	TMC22	TMC21	TMC20	FF4AH	00H	R/W

TMC23	TMC20	Watch Flag Set Time Selection
0	0	$2^{14}/f_W$ (0.5 s)
1		$2^{13}/f_W$ (0.25 s)
0	1	$2^5/f_W$ (977 μs)
1		$2^4/f_W$ (488 μs)

TMC21	Prescaler Operation Control[Note]
0	Clear after operation stops
1	Operation enable

TMC22	5-Bit Counter Operation Control
0	Clear after operation stops
1	Operation enable

TMC26	TMC25	TMC24	Prescaler Interval Time Selection
0	0	0	$2^4/f_W$ (488 μs)
0	0	1	$2^5/f_W$ (977 μs)
0	1	0	$2^6/f_W$ (1.95 ms)
0	1	1	$2^7/f_W$ (3.91 ms)
1	0	0	$2^8/f_W$ (7.81 ms)
1	0	1	$2^9/f_W$ (15.6 ms)
Other than above			Setting prohibited

Note Do not frequently clear the prescaler when using the watch timer.

Remarks 1. f_W: Watch timer clock frequency ($f_X/2^8$ or f_{XT})

2. Values in parentheses apply to operation with f_W = 32.768 kHz.

10.4 Watch Timer Operations

10.4.1 Watch timer operation

When the 32.768 kHz subsystem clock or 8.38 kHz main system clock is used, the timer operates as a watch timer with a 0.5 second or 0.25 second interval. In addition, when the 4.19 MHz main system clock is used, the timer can operate as a watch timer with a 0.5 second or 1 second interval.

Caution When 8.38 MHz or 4.19 MHz frequency is used, the time interval is slightly off.
When f_X = 8.38 MHz frequency is used,

$$\frac{2^8}{f_X} \times 2^{14} = \frac{2^{22}}{8.38 \times 10^6} = 0.5005136\ldots \text{ (second)}$$

When f_X = 4.19 MHz frequency is used,

$$\frac{2^8}{f_X} \times 2^{13} = \frac{2^{21}}{4.19 \times 10^6} = 0.5005136\ldots \text{ (second)}$$

When f_{XT} = 32.768 MHz frequency is used,

$$\frac{1}{f_{XT}} \times 2^{14} = \frac{2^{14}}{32.768 \times 10^3} = 0.50000\ldots \text{ (second)}$$

When f_X = 10.0 MHz frequency is used (not intended),

$$\frac{2^8}{f_X} \times 2^{14} = \frac{2^{22}}{10.0 \times 10^6} = 0.4194304 \text{ (second)}$$

The watch timer sets the interrupt request 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 (TMC22) of the watch timer mode control register (TMC2) 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: 15.6 ms when operated at f_{XT} = 32.768 kHz).

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 (TMC2).

Table 10-3. Interval Timer Interval Time

TMC26	TMC25	TMC24	Interval Time	When operated at f_X = 10.0 MHz	When operated at f_X = 8.38 MHz	When operated at f_X = 4.19 MHz	When operated at f_{XT} = 32.768 kHz
0	0	0	$2^4 \times 1/f_W$	409.6 μs	489 μs	978 μs	488 μs
0	0	1	$2^5 \times 1/f_W$	819.2 μs	978 μs	1.96 ms	977 μs
0	1	0	$2^6 \times 1/f_W$	1.64 ms	1.96 ms	3.91 ms	1.95 ms
0	1	1	$2^7 \times 1/f_W$	3.28 ms	3.91 ms	7.82 ms	3.91 ms
1	0	0	$2^8 \times 1/f_W$	6.55 ms	7.82 ms	15.6 ms	7.81 ms
1	0	1	$2^9 \times 1/f_W$	13.1 ms	15.6 ms	31.3 ms	15.6 ms
Other than above			Setting prohibited				

Remarks f_X : Main system clock oscillation frequency

f_{XT} : Subsystem clock oscillation frequency

f_W : Watch timer clock frequency ($f_X/2^8$ or f_{XT})

TMC24 to TMC26: Bits 4 to 6 of the watch timer mode control register (TMC2)

[MEMO]

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) (the watchdog timer and the interval timer cannot be used simultaneously).

(1) Watchdog timer mode
An inadvertent program loop is detected. Upon detection of the inadvertent program loop, a non-maskable interrupt request or $\overline{RESET}$ can be generated.

Table 11-1. Watchdog Timer Inadvertent Program Loop Detection Time

Inadvertent Program Loop Detection Time	When operated at f_X = 10.0 MHz	Inadvertent Program Loop Detection Time	When operated at f_X = 10.0 MHz
$2^{12} \times 1/f_X$	409.6 μs	$2^{16} \times 1/f_X$	6.55 μs
$2^{13} \times 1/f_X$	819.2 μs	$2^{17} \times 1/f_X$	13.1 μs
$2^{14} \times 1/f_X$	1.64 ms	$2^{18} \times 1/f_X$	26.2 ms
$2^{15} \times 1/f_X$	3.28 ms	$2^{20} \times 1/f_X$	104.9 ms

Remark f_X: Main system clock oscillation frequency

(2) Interval timer mode
Interrupt requests are generated at the preset time intervals.

Table 11-2. Interval Time

Interval Time	When operated at f_X = 10.0 MHz	Interval Time	When operated at f_X = 10.0 MHz
$2^{12} \times 1/f_X$	409.6 μs	$2^{16} \times 1/f_X$	6.55 ms
$2^{13} \times 1/f_X$	819.2 μs	$2^{17} \times 1/f_X$	13.1 ms
$2^{14} \times 1/f_X$	1.64 ms	$2^{18} \times 1/f_X$	26.2 ms
$2^{15} \times 1/f_X$	3.28 ms	$2^{20} \times 1/f_X$	104.9 ms

Remark f_X: Main system clock oscillation frequency

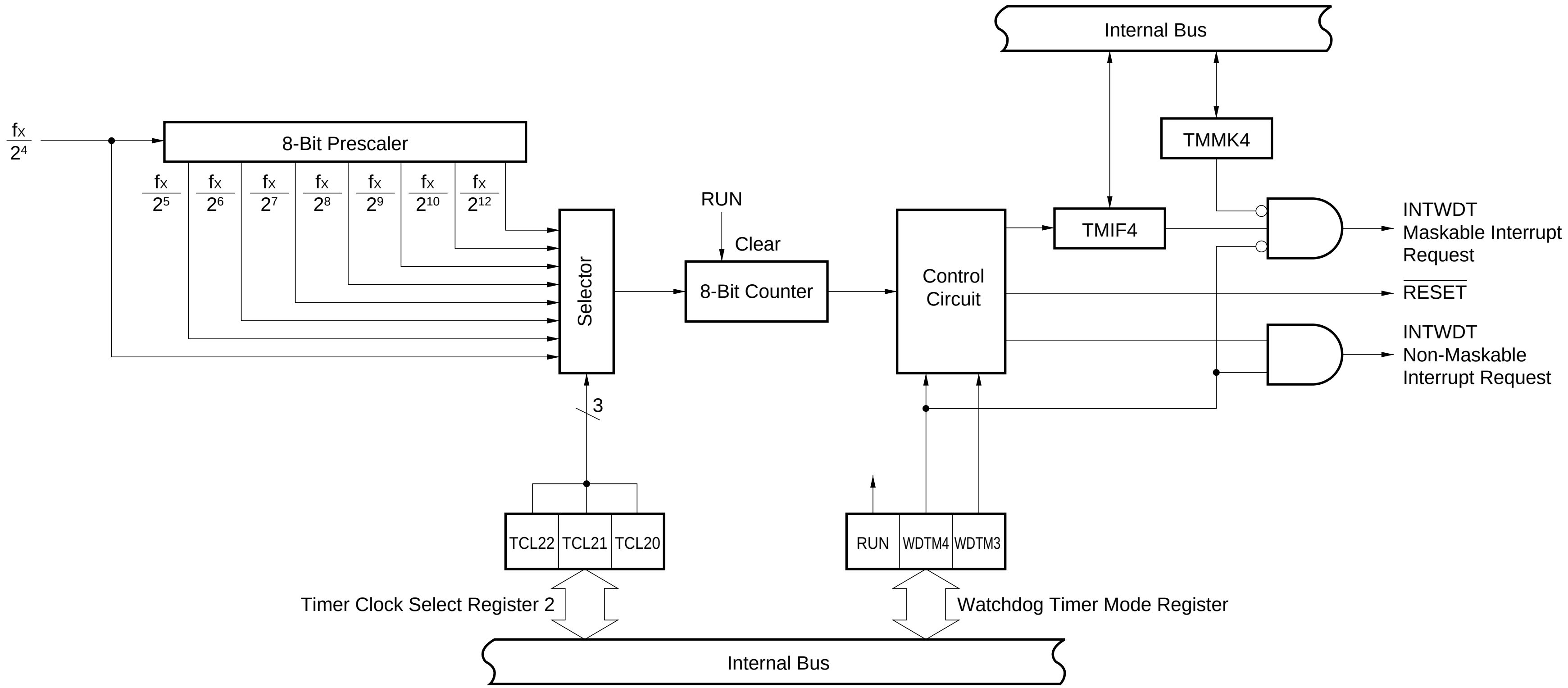
11.2 Watchdog Timer Configuration

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



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 sets 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	When Reset	R/W
TCL2	TCL27	TCL26	TCL25	TCL24	0	TCL22	TCL21	TCL20	FF42H	00H	R/W

TCL22	TCL21	TCL20	Watchdog Timer Count Clock Selection
0	0	0	$f_X/2^4$ (625 kHz)
0	0	1	$f_X/2^5$ (313 kHz)
0	1	0	$f_X/2^6$ (156 kHz)
0	1	1	$f_X/2^7$ (78.1 kHz)
1	0	0	$f_X/2^8$ (39.1 kHz)
1	0	1	$f_X/2^9$ (19.5 kHz)
1	1	0	$f_X/2^{10}$ (9.8 kHz)
1	1	1	$f_X/2^{12}$ (2.4 kHz)

TCL24	Watch Timer Count Clock Selection
0	$f_X/2^8$ (39.1 kHz)
1	f_{XT} (32.768 kHz)

TCL27	TCL26	TCL25	Buzzer Output Frequency Selection
0	×	×	Buzzer output disable
1	0	0	$f_X/2^{10}$ (9.8 kHz)
1	0	1	$f_X/2^{11}$ (4.9 kHz)
1	1	0	$f_X/2^{12}$ (2.4 kHz)
1	1	1	Setting prohibited

Caution **If data other than identical data is to be rewritten to TCL2, the timer operation must be stopped first.**

Remarks
1. f_X : Main system clock oscillation frequency
2. f_{XT} : Subsystem clock oscillation frequency
3. × : don't care
4. Values in parentheses apply to operation with f_X = 10.0 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 sets WDTM to 00H.

Figure 11-3. Watchdog Timer Mode Register Format

Symbol	<7>	6	5	4	3	2	1	0	Address	When Reset	R/W
WDTM	RUN	0	0	WDTM4	WDTM3	0	0	0	FFF9H	00H	R/W

WDTM4	WDTM3	Watchdog Timer Operating Mode Selection[Note 1]
0	x	Interval timer Mode[Note 2] (Maskable interrupt request occurs upon generation of an overflow.)
1	0	Watchdog timer mode 1 (Non-maskable interrupt request occurs upon generation of an overflow.)
1	1	Watchdog timer mode 2 (Reset operation is activated upon generation of an overflow.)

RUN	Watchdog Timer Operation Selection[Note 3]
0	Count stop
1	Counter is cleared and counting starts.

Notes 1. Once set to 1, WDTM3 and WDTM4 cannot be cleared to 0 by software.

2. Interval timer operation starts when the RUN bit is set to 1.

3. 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.

Cautions 1. When 1 is set in RUN so that the watchdog timer is cleared, the actual overflow time may be up to 0.5% shorter than the time set by timer clock select register 2 (TCL2).

2. In watchdog timer mode 1 or 2, make sure that the interrupt request flag (TMIF4) is set to 0 before setting WDTM4 to 1. If WDTM4 is set to 1 while TMIF4 is set to 1, a non-maskable interrupt request occurs regardless of the contents in WDTM3.

Remark x: 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 the 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 inadvertent program loop time interval to be set. 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.

Watchdog timer can be cleared by setting RUN to 1.

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 inadvertent program loop detection time may be shorter than the set time by a maximum of 0.5%.

2. When the subsystem clock is selected for CPU clock, watchdog timer count operation is stopped.

Table 11-4. Watchdog Timer Inadvertent Program Loop Detection Time

TCL22	TCL21	TCL20	Inadvertent Program Loop Detection Time	f_X = 10.0 MHz
0	0	0	$2^{12} \times 1/f_X$	409.6 μs
0	0	1	$2^{13} \times 1/f_X$	819.2 μs
0	1	0	$2^{14} \times 1/f_X$	1.64 ms
0	1	1	$2^{15} \times 1/f_X$	3.28 ms
1	0	0	$2^{16} \times 1/f_X$	6.55 ms
1	0	1	$2^{17} \times 1/f$	13.1 ms
1	1	0	$2^{18} \times 1/f_X$	26.2 ms
1	1	1	$2^{20} \times 1/f_X$	104.9 ms

Remark f_X: Main system clock oscillation frequency
TCL20 to TCL22: Bits 0 to 2 of the timer clock select register 2 (TCL2)

11.4.2 Interval timer operation

The watchdog timer operates as an interval timer which generates interrupt requests repeatedly at intervals of a preset count value when bit 4 (WDTM4) of the watchdog timer mode register (WDTM) is set to 0.

The count clock (or interval time) can be selected with bits 0 to 2 (TCL20 to TCL22) of the time clock select register (TCL2). When 1 is written into WDTM bit 7 (RUN), the interval timer operation starts.

When the watchdog timer 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 WDTM bit 7 (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 up to 0.5%.

3. When the subsystem clock is selected for CPU clock, watchdog timer count operation is stopped.

Table 11-5. Interval Timer Interval Time

TCL22	TCL21	TCL20	Interval Time	f_X = 10.0 MHz
0	0	0	$2^{12} \times 1/f_X$	409.6 μs
0	0	1	$2^{13} \times 1/f_X$	819.2 μs
0	1	0	$2^{14} \times 1/f_X$	1.64 ms
0	1	1	$2^{15} \times 1/f_X$	3.28 ms
1	0	0	$2^{16} \times 1/f_X$	6.55 ms
1	0	1	$2^{17} \times 1/f$	13.1 ms
1	1	0	$2^{18} \times 1/f_X$	26.2 ms
1	1	1	$2^{20} \times 1/f_X$	104.9 ms

Remark f_X: Main system clock oscillation frequency
TCL20 to TCL22: Bits 0 to 2 of the timer clock select register 2 (TCL2)

CHAPTER 12 CLOCK OUTPUT CONTROL CIRCUIT

12.1 Clock Output Control Circuit Functions

The clock output control circuit is intended for carrier output during remote controlled transmission and clock output for supply to peripheral LSI. Clocks selected with the timer clock select register 0 (TCL0) are output from the PCL/P35 pin.

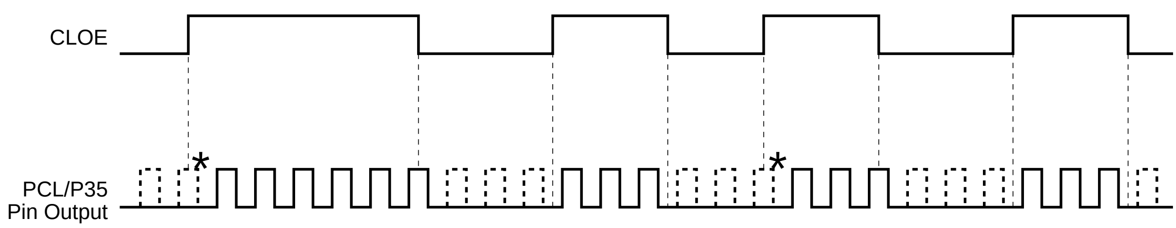
Follow the procedure below to output clock pulses.

(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 (PM3) to 0 (set to output mode).
(4) Set bit 7 (CLOE) of TCL0 to 1.

Caution Clock output cannot be used if P35 output latch is set to 1.

Remark When clock output enable/disable is switched, the clock output control circuit does not output pulses with small widths (See the mark * in Figure 12-1).

Figure 12-1. Remote Controlled Output Application Example



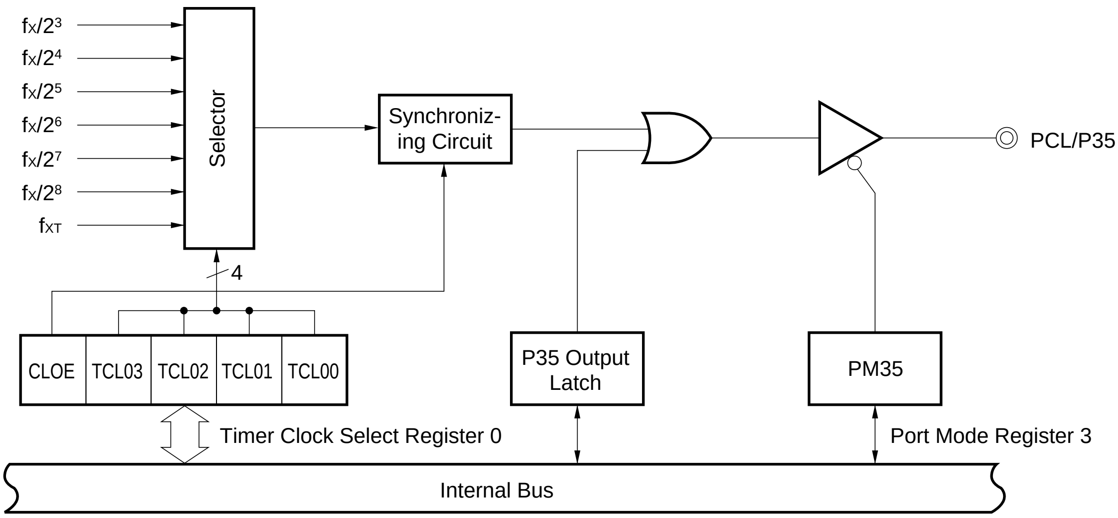
12.2 Clock Output Control Circuit Configuration

The clock output control circuit consists of the following hardware.

Table 12-1. Clock Output Control Circuit Configuration

Item	Configuration
Control register	Timer clock select register 0 (TCL0) Port mode register 3 (PM3)

Figure 12-2. Clock Output Control Circuit Block Diagram



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 sets 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	When Reset	R/W
TCL	CLOE	TCL06	TCL05	TCL04	TCL03	TCL02	TCL01	TCL00	FF40H	00H	R/W

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

TCL06	TCL05	TCL04	16-bit Timer Register Count Clock Selection
0	0	0	TI0 (Valid edge specifiable)
0	1	0	$f_X/2$ (5.0 MHz)
0	1	1	$f_X/2^2$ (2.5 MHz)
1	0	0	$f_X/2^3$ (1.25 MHz)
Other than above			Setting prohibited

CLOE	PCL Output Control
0	Output disable
1	Output enable

Cautions 1. Setting of the TI0/INTP0 pin valid edge is performed by external interrupt mode register (INTM0), and selection of the sampling clock frequency is performed by the sampling clock selection register (SCS).

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 TI0 has been specified as the TM0 count clock, the value should be read from TM0, not from the 16-bit capture register (CR01).

4. If data other than identical data is to be rewritten to TCL0, the timer operation must be stopped first.

Remarks 1. f_X : Main system clock oscillation frequency

2. f_{XT} : Subsystem clock oscillation frequency

3. TI0 : 16-bit timer/event counter input pin

4. TM0 : 16-bit timer register

5. Values in parentheses apply to operation with f_X = 10.0 MHz or f_{XT} = 32.768 kHz.

(2) Port mode register 3 (PM3)

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

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	When Reset	R/W
PM3	PM37	PM36	PM35	PM34	PM33	PM32	PM31	PM30	FF23H	00H	R/W

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

CHAPTER 13 BUZZER OUTPUT CONTROL CIRCUIT

13.1 Buzzer Output Control Circuit Functions

The buzzer output control circuit outputs a square wave with a frequency of either 2.4 kHz, 4.9 kHz, or 9.8 kHz. 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 (PM3) to 0 (Set to output mode).

Caution Buzzer output cannot be used if P36 output latch is set to 1.

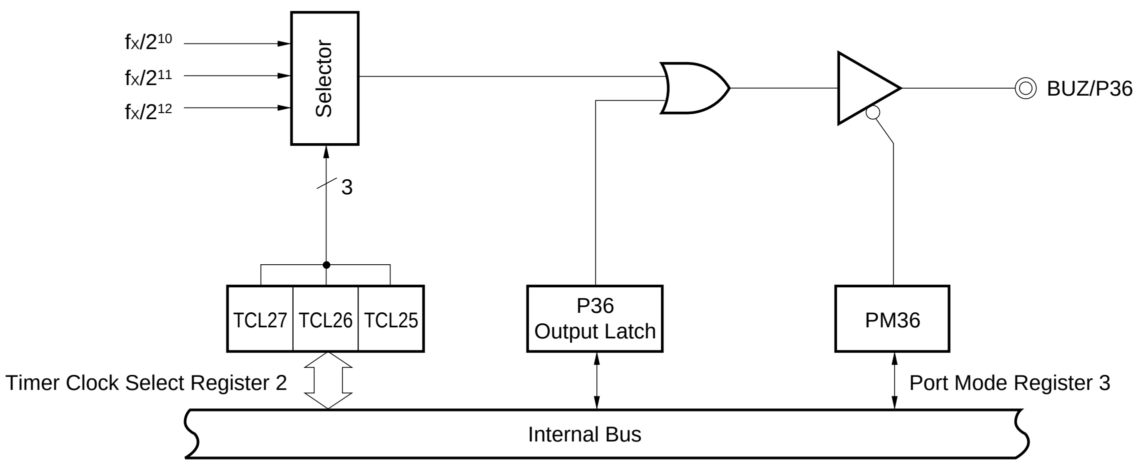
13.2 Buzzer Output Control Circuit Configuration

The buzzer output control circuit consists of the following hardware.

Table 13-1. Buzzer Output Control Circuit Configuration

Item	Configuration
Control register	Timer clock select register 2 (TCL2) Port mode register 3 (PM3)

Figure 13-1. Buzzer Output Control Circuit Block Diagram



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 sets 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	When Reset	R/W
TCL2	TCL27	TCL26	TCL25	TCL24	0	TCL22	TCL21	TCL20	FF42H	00H	R/W

TCL22	TCL21	TCL20	Watchdog Timer Count Clock Selection
0	0	0	$f_X/2^4$ (625 kHz)
0	0	1	$f_X/2^5$ (313 kHz)
0	1	0	$f_X/2^6$ (156 kHz)
0	1	1	$f_X/2^7$ (78.1 kHz)
1	0	0	$f_X/2^8$ (39.1 kHz)
1	0	1	$f_X/2^9$ (19.5 kHz)
1	1	0	$f_X/2^{10}$ (9.8 kHz)
1	1	1	$f_X/2^{12}$ (2.4 kHz)

TCL24	Watch Timer Count Clock Selection
0	$f_X/2^8$ (39.1 kHz)
1	f_{XT} (32.768 kHz)

TCL27	TCL26	TCL25	Buzzer Output Frequency Selection
0	×	×	Buzzer output disable
1	0	0	$f_X/2^{10}$ (9.8 kHz)
1	0	1	$f_X/2^{11}$ (4.9 kHz)
1	1	0	$f_X/2^{12}$ (2.4 kHz)
1	1	1	Setting prohibited

Caution **If data other than identical data is to be rewritten to TCL2, the timer operation must be stopped first.**

Remarks
1. f_X : Main system clock oscillation frequency
2. f_{XT} : Subsystem clock oscillation frequency
3. × : don't care
4. Values in parentheses apply to operation with f_X = 10.0 MHz or f_{XT} = 32.768 kHz.

(2) Port mode register 3 (PM3)

This register sets port 3 input/output bit-wise.

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	When Reset	R/W
PM3	PM37	PM36	PM35	PM34	PM33	PM32	PM31	PM30	FF23H	00H	R/W

PM3n	P3n Pin Input/Output 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 (ADM).

One channel of analog input is selected from ANI0 to ANI7 and A/D conversion is carried out. In the case of hardware start, A/D conversion operation stops when it terminates and an interrupt request (INTAD) is generated. In the case of software start, the A/D conversion operation is repeated. Each time an A/D conversion operation ends, an interrupt request (INTAD) is generated.

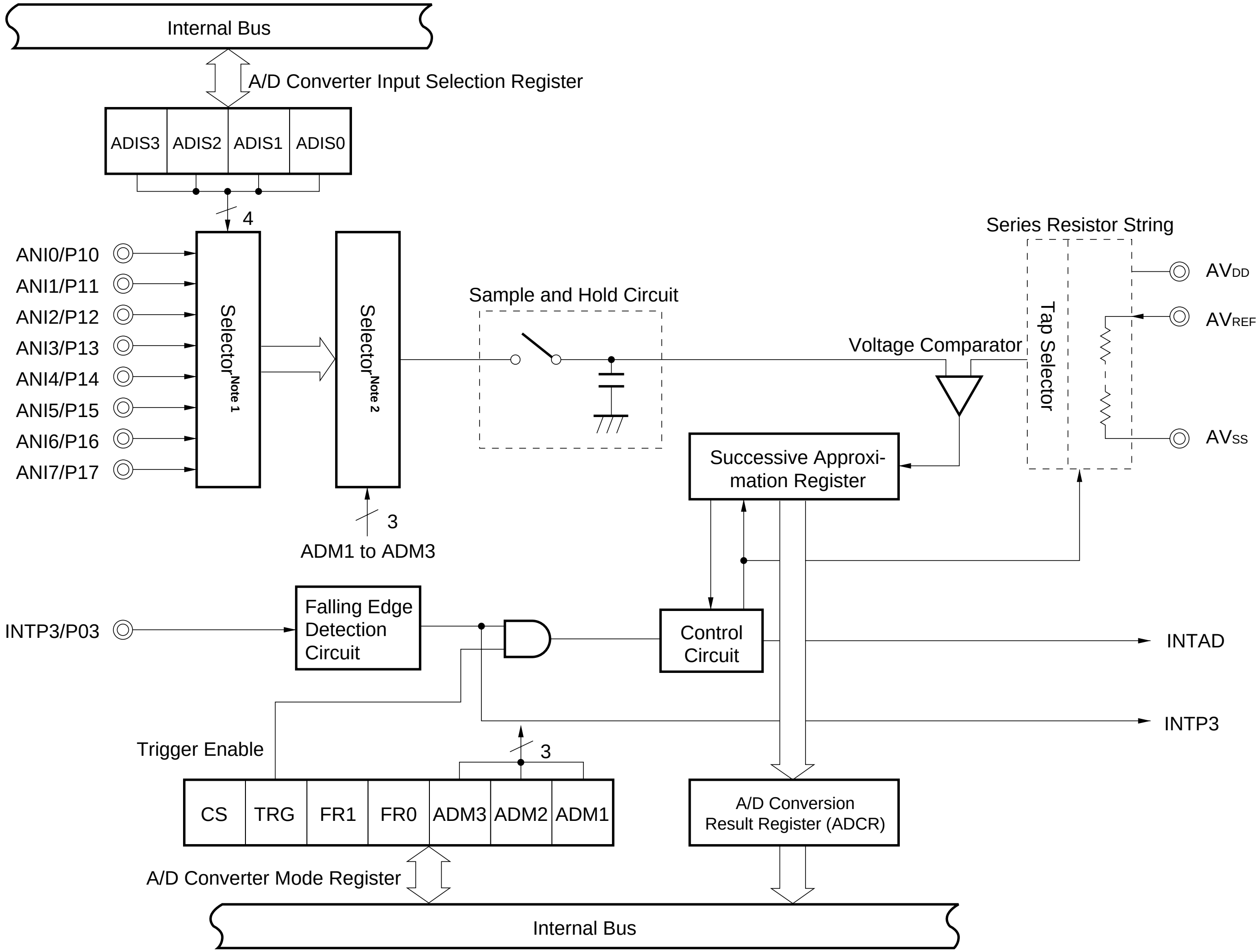
14.2 A/D Converter Configuration

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)
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

(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 held (termination of A/D conversion), the SAR contents are transferred to the A/D conversion result register (ADCR).

(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 (SAR).
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 to among AV_{REF} to AV_{SS} and generates a voltage to be compared to 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.
These pins except analog input pins selected with the A/D converter input select register (ADIS) can be used as the input/output port.

Cautions 1. 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 will be undefined and may adversely affect the converted values of other channels.

2. Pins ANI0/P10 to ANI7/P17
The analog input pins ANI0 to ANI7 also function as input/output port (PORT1) pins. Pins used as the analog input should be specified to the input mode.
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.

(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}. If the voltage to be input to the AV_{REF} pin is adjusted to the AV_{SS} level in the standby mode, the current in the series resistor string will be decreased.

Caution 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.

(8) AV_{SS} pin

Ground potential pin of the A/D converter. It must be at the same level as the V_{SS} pin even if the A/D converter is not used.

(9) AV_{DD} pin

Analog power supply pin of the A/D converter. It must be at the same level as the V_{DD} pin even if the A/D converter is not used.

14.3 A/D Converter Control Registers

The following two types of registers are used to control the A/D converter.

- A/D converter mode register (ADM)
- A/D converter input select register (ADIS)

(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	When Reset	R/W
ADM	CS	TRG	FR1	FR0	ADM3	ADM2	ADM1	1	FF80H	01H	R/W

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

FR1	FR0	A/D Conversion Time Selection[Note 1]			
			When operated at f_X = 10.0 MHz	When operated at f_X = 8.38 MHz	When operated at f_X = 4.19 MHz
0	0	160/f_X	Setting prohibited[Note 2]	19.1 μs	38.1 μs
0	1	80/f_X	Setting prohibited[Note 2]	Setting prohibited[Note 2]	19.1 μs
1	0	200/f_X	20.0 μs	23.9 μs	47.7 μs
1	1	Setting prohibited			

TRG	External Trigger Selection
0	No external trigger (software starts mode)
1	Conversion started by external trigger (hardware starts mode)

CS	A/D Conversion Operation Control
0	Operation stop
1	Operation start

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. Set bit 0 to 1.

2. To reduce power dissipation in the A/D converter when standby functions used, the bit 7 (CS) should be cleared to 0 to stop the A/D conversion operation before executing a HALT or STOP instruction.

3. To restart the stopped A/D conversion operation, the interrupt request flag (ADIF) should be cleared to 0 before starting the A/D conversion operation.

Remark f_X: Main system clock oscillation frequency

(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. The pins which are not selected for analog input pins can be used as the input/output port.

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

$\overline{\text{RESET}}$ input sets 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 the A/D converter mode register (ADM), select one channel to undergo A/D conversion among the channels which is set for analog input with ADIS.

2. On-chip pull-up resistor is not used for the channels set for analog input with ADIS, irrespective of the value of bit 1 (PUO1) of the pull-up resistor option register.

Figure 14-3. A/D Converter Input Select Register Format

Symbol	7	6	5	4	3	2	1	0	Address	When 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 channels (ANI0, ANI1, P12 to P17)
0	0	1	1	3 channels (ANI0 to ANI2, P13 to P17)
0	1	0	0	4 channels (ANI0 to ANI3, P14 to P17)
0	1	0	1	5 channels (ANI0 to ANI4, P15 to P17)
0	1	1	0	6 channels (ANI0 to ANI5, P16, P17)
0	1	1	1	7 channels (ANI0 to ANI6, P17)
1	0	0	0	8 channels (ANI0 to ANI7)
Other than above				Setting prohibited

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) Sample the voltage input to the selected analog input channel with the sample & hold circuit.

(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.

(5) Bit 7 of successive approximation register (SAR) is set and the tap selector sets the series resistor string voltage tap to (1/2) AV_{REF}.

(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 larger 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.

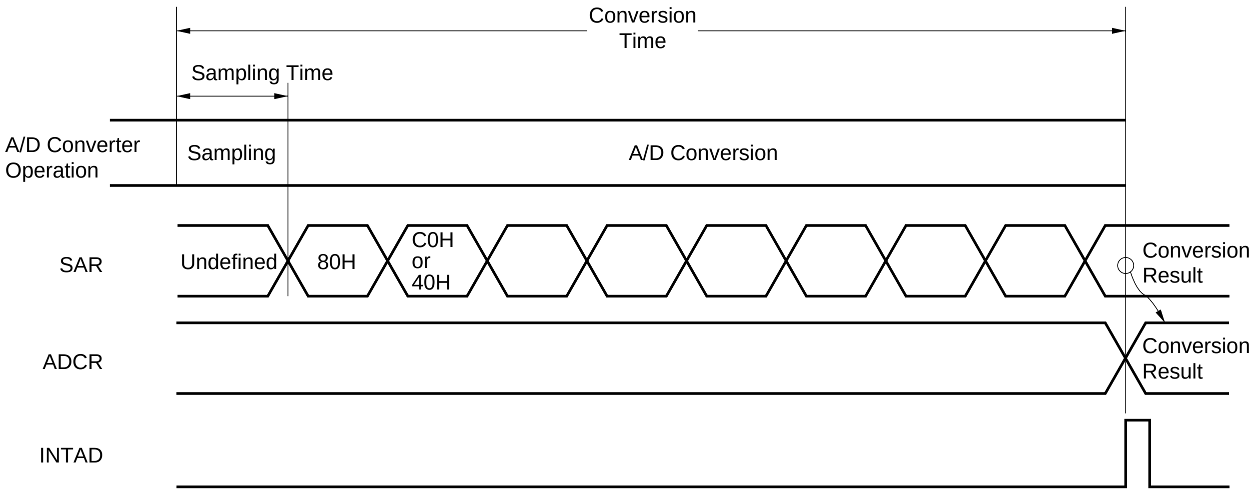
(7) Next, bit 6 of SAR is automatically set and the operation proceeds to the next comparison. In this case, the series resistor string voltage tap is selected according to the preset value of bit 7 as described below.
   - Bit 7 = 1: (3/4) AV_{REF}
   - Bit 7 = 0: (1/4) AV_{REF}

   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 resulting effective digital value remains in SAR and the resulting 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 14-4. A/D Converter Basic Operation



A/D conversion operations are performed continuously until bit 7 (CS) of the A/D converter mode register (ADM) is reset (0) by software.

If a write to the ADM is performed during an A/D conversion operation, the conversion operation is initialized, and if the CS bit is set (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 relationship between the analog input voltage input to the analog input pins (ANI0 to ANI7) and the A/D conversion result (the value stored in the A/D conversion result register (ADCR)) is expressed by the following expression.

$$ADCR = INT\left(\frac{V_{IN}}{AV_{REF}} \times 256 + 0.5\right)$$

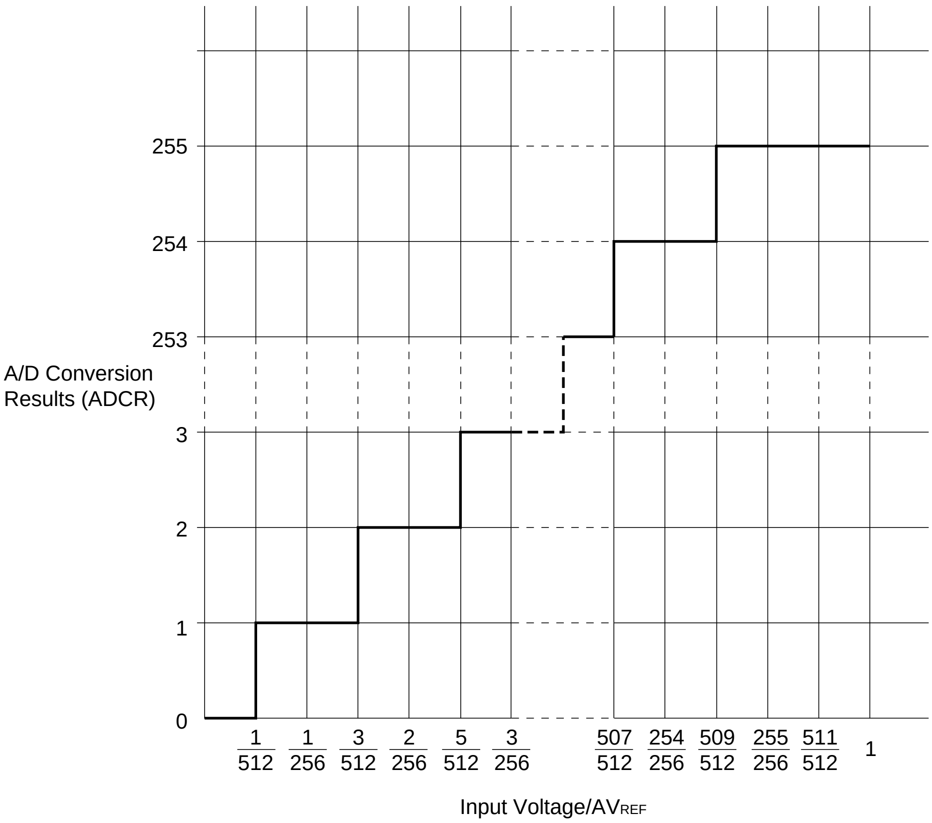
or

$$(ADCR - 0.5) \times \frac{AV_{REF}}{256} \leq V_{IN} < (ADCR + 0.5) \times \frac{AV_{REF}}{256}$$

where 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 14-5 shows the relationship between the analog input voltage and the A/D conversion result.

Figure 14-5. Relationship between Analog Input Voltage and A/D Conversion Result



14.4.3 A/D converter operating mode

The operating mode is a select 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 start the 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

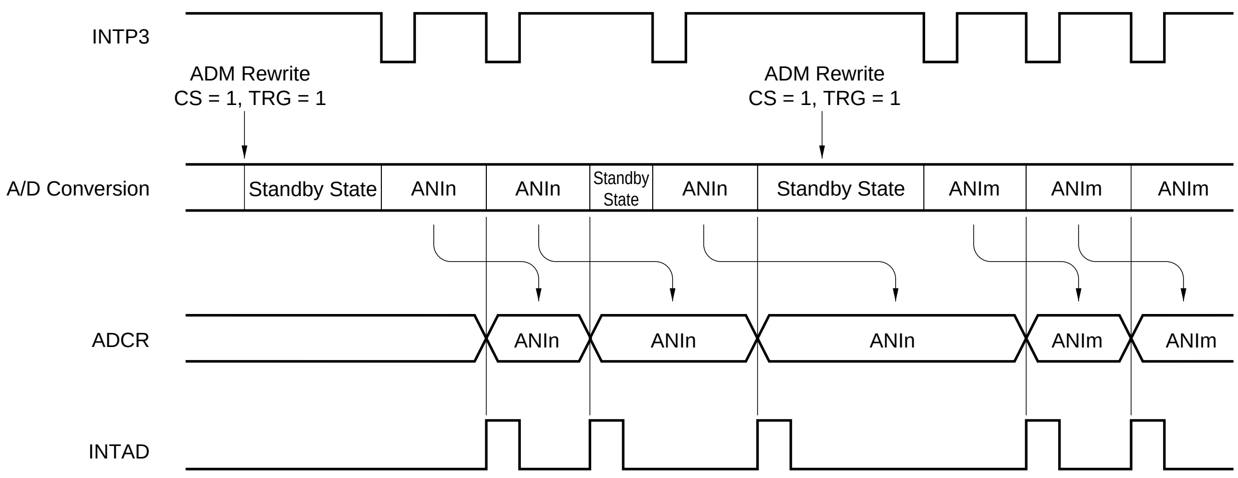
When bit 6 (TRG) and bit 7 (CS) of A/D converter mode register (ADM) are set to 1, the A/D conversion standby state is set. When the external trigger signal (INTP3) is input, the A/D conversion starts on the voltage applied to the analog input pins specified with bits 1 to 3 (ADM1 to ADM3) of ADM.

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 14-6. A/D Conversion by Hardware Start



Remark n = 0, 1, ..., 7
m = 0, 1, ..., 7

(2) A/D conversion by software start

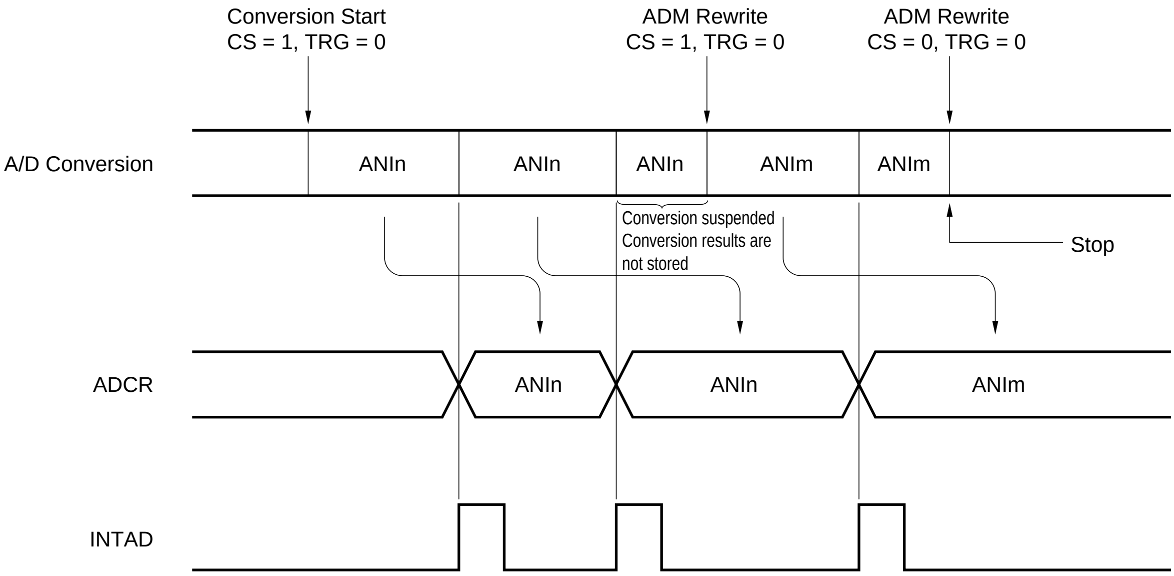
When bit 6 (TRG) and bit 7 (CS) of the A/D converter mode register (ADM) are set to 0 and 1, respectively, A/D conversion starts on the voltage applied to the analog input pins specified with bits 1 to 3 (ADM1 to ADM3) of ADM.

Upon termination of 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 14-7. A/D Conversion by Software Start



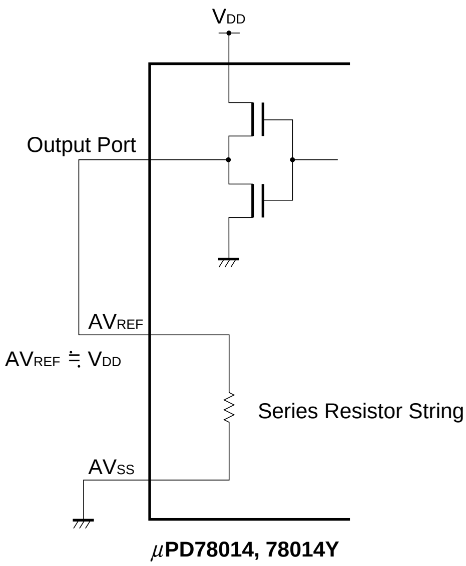
Remark n = 0, 1, ..., 7
m = 0, 1, ..., 7

14.5 Cautions on A/D Converter

(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 this example, the power dissipation can be reduced if a low level is output to the output port in the standby mode. However, the actual AV_{REF} voltage is not so accurate and, accordingly, the converted value is not accurate and should be used for relative comparison only.

Figure 14-8. Example of Method of Reducing Power Dissipation in Standby Mode



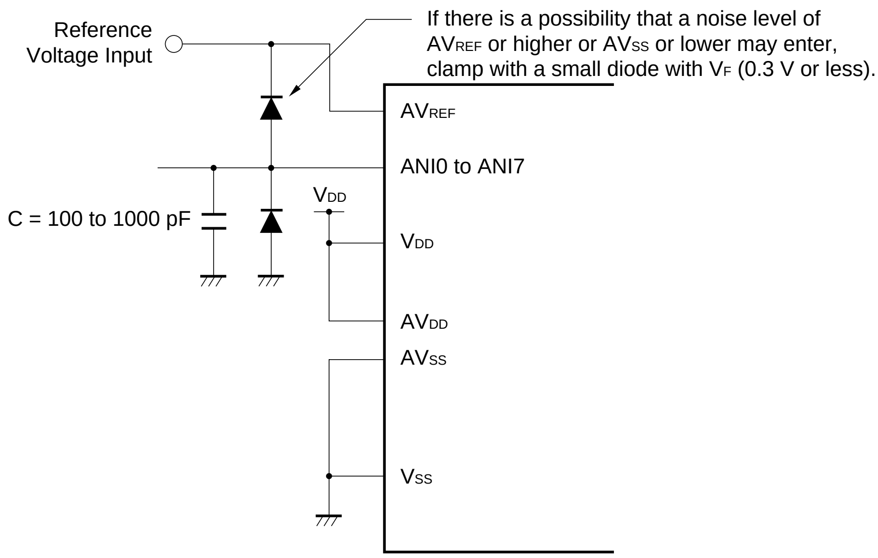
(2) Input range of ANI0 to ANI7

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 is connected externally as shown in Figure 14-9 in order to reduce noise.

Figure 14-9. Analog Input Pin Disposition



(4) Pins ANI0/P10 to ANI7/P17

The analog input pins ANI0 to ANI7 also function as input/output port (PORT1) pins. Pins used as the analog input should be specified to the input mode.

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.

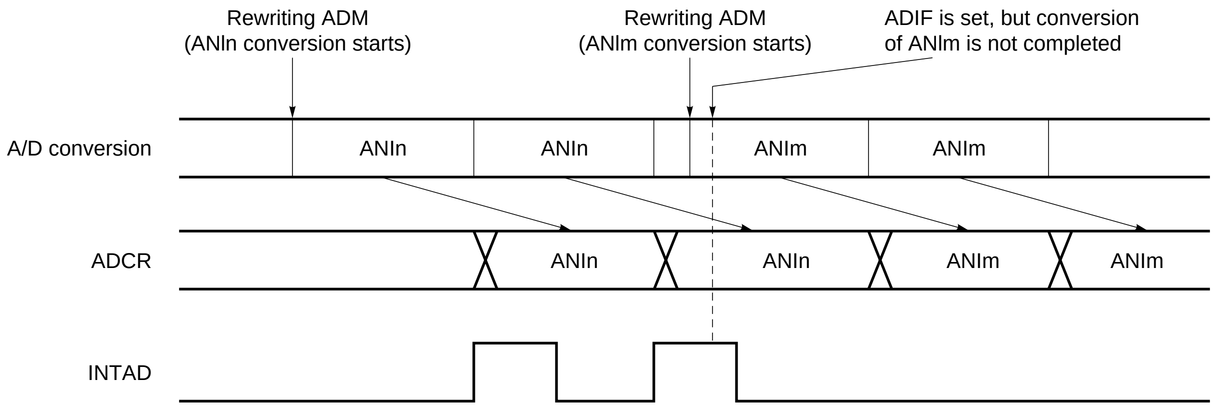
Therefore, if the output impedance of the reference voltage source is high, this will result in parallel connection to the series resistor string between the AV_{REF} pin and the AV_{SS} pin, and there will be a large reference voltage error.

(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 ADIF 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. (Refer to **Figure 14-10.**)

When the A/D conversion is stopped, the ADIF must be cleared before restarting.

Figure 14-10. A/D Conversion End Interrupt Request Generation Timing

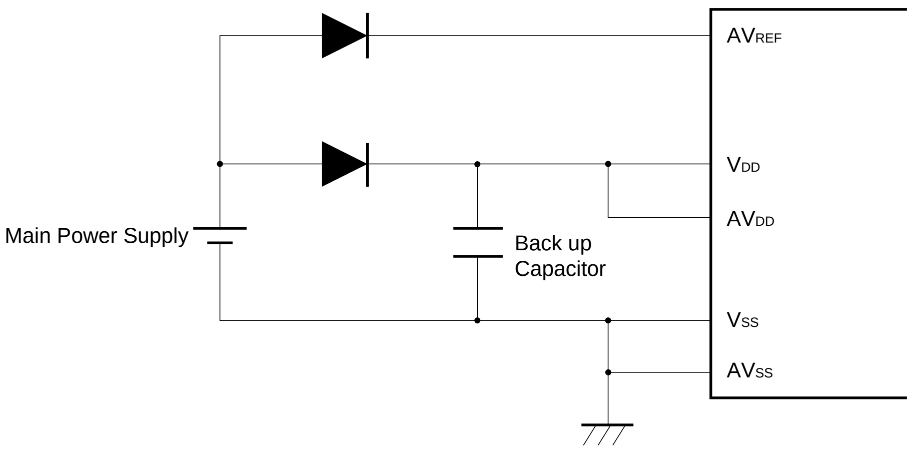


(7) AV_{DD} pin

The AV_{DD} pin is the analog circuit power supply pin, and supplies power to the input circuits of ANI0/P10 to ANI7/P17.

Therefore, be sure to apply the voltage at the same level as V_{DD} as shown in Figure 14-11, even in an application where the power supply is switched to the back-up power supply.

Figure 14-11. AV_{DD} Pin Connection



[MEMO]

CHAPTER 15 SERIAL INTERFACE CHANNEL 0 (μPD78014 Subseries)

The μPD78014 Subseries incorporates two channels of clock synchronous serial interfaces.

Differences between channels 0 and 1 are as follows (Refer to **CHAPTER 17 SERIAL INTERFACE CHANNEL 1** for details of the serial interface channel 1).

Table 15-1. Differences between Channels 0 and 1

Serial Transfer Mode		Channel 0	Channel 1
3-wire serial I/O	Clock selection	$f_X/2^2$ **Note**, $f_X/2^3$, $f_X/2^4$, $f_X/2^5$, $f_X/2^6$, $f_X/2^7$, $f_X/2^8$, $f_X/2^9$, external clock, TO2 output	
	Transfer method	MSB/LSB switchable as the start bit	MSB/LSB switchable as the start bit Automatic transmit/receive function
	Transfer end flag	Serial interface channel 0 transfer end interrupt request flag (CSIIF0)	Serial interface channel 1 transfer end interrupt request flag (CSIIF1 and TRF)
SBI (serial bus interface)		Use possible	None
2-wire serial I/O			

Note Can be set only when the main system clock oscillates at 4.19 MHz or less.

Differences of Serial interface channel 0 modes are shown in Table 15-2.

Table 15-2. Difference of Serial Interface Channel 0 Modes

Operation mode	Used pin	Features	Usage
3-wire serial I/O	$\overline{SCK0}$, SO0, SI0	• Input and output lines are independent and they can transfer/receive at the same time, so the data transfer processing time is fast. • NEC single-chip microcontrollers provide as before.	Serial interface as is the case with the 75X/XL, 78K and 17K Series.
SBI mode	$\overline{SCK0}$, SB0 or SB1	• Enables to configure serial bus with two signal lines, thus, even when connect to some microcontrollers, the number of ports can be cut and reduced the wiring and drawing around on a board. • High-speed serial interface to be complianced with the NEC standard bus format. • Address and command information onto the serial bus	
2-wire serial I/O	$\overline{SCK0}$, SB0 or SB1	• Enables to configure serial bus with two signal lines, thus, even when connect to some microcontrollers, the number of ports can be cut and reduced the wiring and drawing around on a board. • Enables to cope with any data transfer format by program.	

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 switch the operation mode (3-wire serial I/O/2-wire serial I/O/SBI) during the serial interface channel 0 operation enable. The operation mode should be switched after stopping the serial operation.**

(1) Operation stop mode
This mode is used when serial transfer is not carried out. Power dissipation can be reduced.

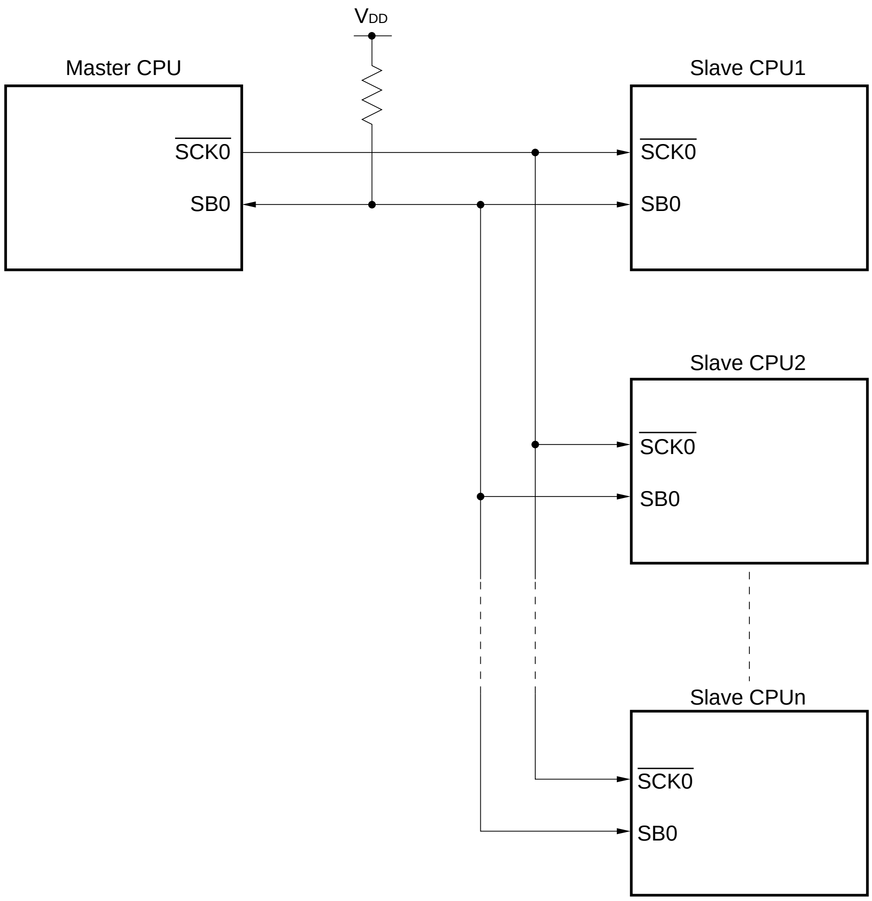
(2) 3-wire serial I/O mode (MSB-/LSB-first selectable)
This mode is used to 8-bit data transfer using three lines, one each for serial clock ($\overline{SCK0}$), serial output (SO0) and serial input (SI0).
This mode enables simultaneous transmission/reception and therefore reduces the data transfer processing 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 ($\overline{SCK0}$) and serial data bus (SB0 or SB1) (See **Figure 15-1**).
The SBI mode complies with the NEC serial bus format and distinguishes the transfer data into “address”, “command”, and “data” to transmit or receive the data.

- Address : Data to select objective devices in serial communication
- Command : Data to instruct objective devices
- Data : Data to be actually transferred

In the actual transfer, the master device first outputs the “address” to the serial bus, and selects the slave device as communication target from among two or more devices. The serial transfer is then performed by transmitting and receiving “command” and “data” between the master and slave devices. The receiver automatically distinguishes the received data into “address”, “command”, or “data”, by hardware.
This function enable to use input/output ports effectively and to simplify a serial interface controller of application programs.
In addition, wake-up function for handshake, acknowledge signal, and busy signal output function can be used.

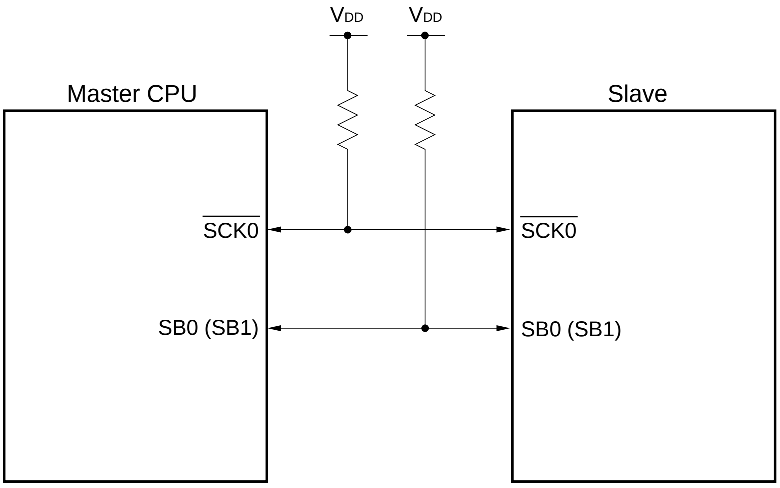
Figure 15-1. Serial Bus Interface (SBI) System Configuration Example



(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 an increased number of available input/output ports.

Figure 15-2. Serial Bus Configuration Example with 2-Wire Serial I/O



15.2 Serial Interface Channel 0 Configuration

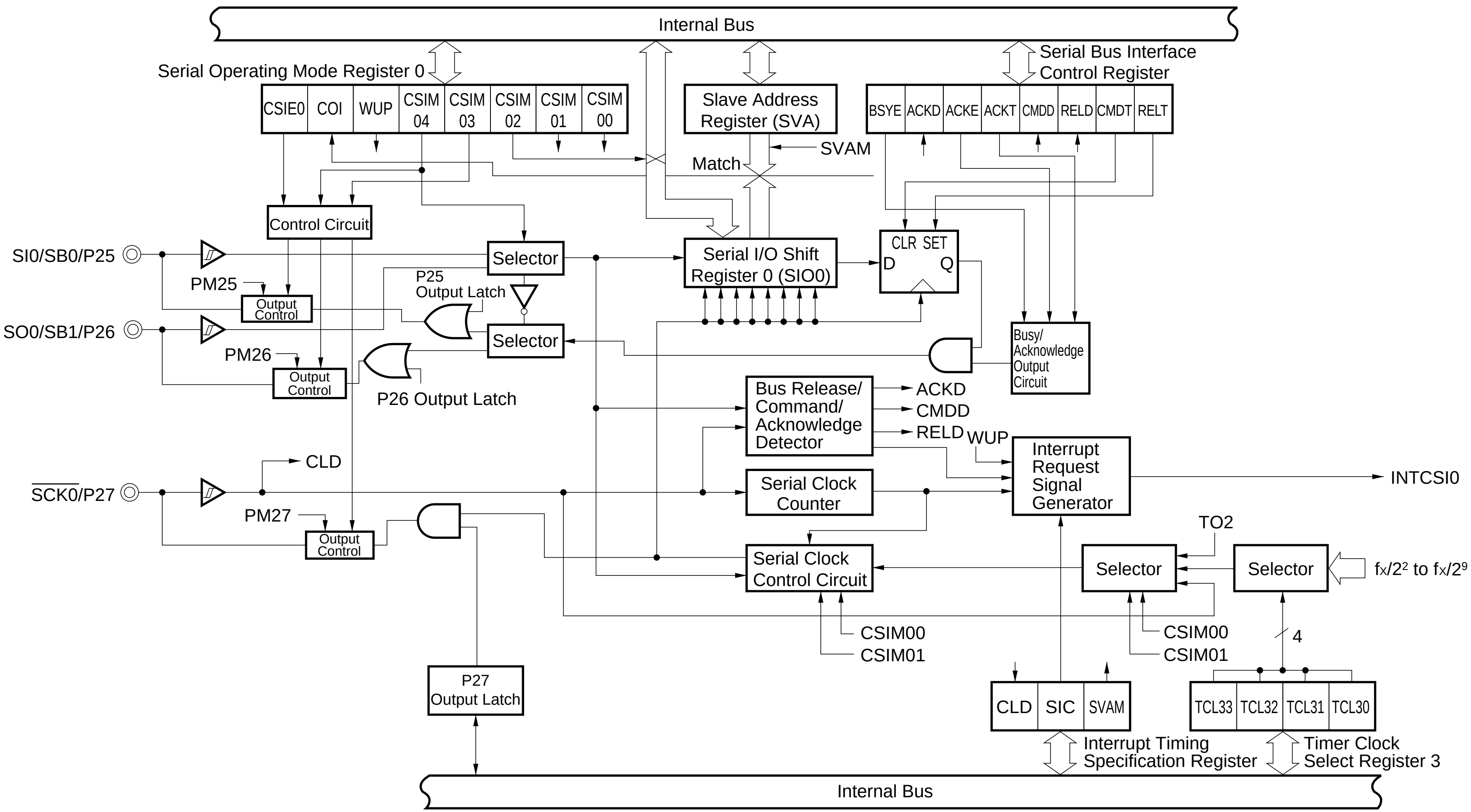
Serial interface channel 0 consists of the following hardware.

Table 15-3. 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 6-6 P20, P21, P23 to P26 Block Diagrams (μPD78014 Subseries)** and **Figure 6-7 P22 and P27 Block Diagrams (μPD78014 Subseries)**.

Figure 15-3. Serial Interface Channel 0 Block Diagram



Remark Output control performs selection between CMOS output and N-ch open-drain output.

(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.

The SBI mode and 2-wire serial I/O mode bus configurations enables the pin to 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 does not be 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.

Address comparison can also be executed on the data of LSB-masked high-order 7 bits when bit 4 (SVAM) of the interrupt timing specify register (SINT) is 1.

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, when bit 5 (WUP) of CSIM0 is 1, the wake-up function is available. In this case, the interrupt request signal (INTCSI0) is generated only if the the slave address output from the master device matches the value of SVA. With this interrupt request, the slave device acknowledges that a communication request is sent from the master device. When bit 5 (SIC) of the interrupt timing specification register has been set to 1, the wake-up function is not available even if WUP is 1. (The interrupt request signal is generated at the bus release in the SBI mode) The SIC must be cleared to 0 while in use of the wake-up function.

Further, when SVA transmits data as the master or slave device in the SBI mode or 2-wire serial I/O mode, SVA can be used to detect errors.

$\overline{\text{RESET}}$ 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 control circuit

This circuit controls serial clock supply to the 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 a wake-up function specification bit. It is bit 5 of the serial operating mode register 0 (CSIM0). Bit 5 (SIC) of the interrupt timing select register (SINT) must be 0 when the wake-up function (WUP = 1) is selected.

(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) (See **Figure 15-4**.)
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.

Remark TCL3 has functions to set the serial clock of serial interface channel 1 besides setting the serial clock of serial interface channel 0.

(2) Serial operating mode register 0 (CSIM0) (See **Figure 15-5**.)
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 sets CSIM0 to 00H.

★ **Caution Do not switch the operation mode (3-wire serial I/O/2-wire serial I/O/SBI) during the serial interface channel 0 operation enable. The operation mode should be switched after stopping the serial operation.**

Figure 15-4. Timer Clock Select Register 3 Format

Symbol	7	6	5	4	3	2	1	0	Address	When Reset	R/W
TCL3	TCL37	TCL36	TCL35	TCL34	TCL33	TCL32	TCL31	TCL30	FF43H	88H	R/W

TCL33	TCL32	TCL31	TCL30	Serial Interface Channel 0 Serial Clock Selection
0	1	1	0	$f_X/2^2$ Note
0	1	1	1	$f_X/2^3$ (1.25 MHz)
1	0	0	0	$f_X/2^4$ (625 kHz)
1	0	0	1	$f_X/2^5$ (313 kHz)
1	0	1	0	$f_X/2^6$ (156 kHz)
1	0	1	1	$f_X/2^7$ (78.1 kHz)
1	1	0	0	$f_X/2^8$ (39.1 kHz)
1	1	0	1	$f_X/2^9$ (19.5 kHz)
Other than above				Setting prohibited

TCL37	TCL36	TCL35	TCL34	Serial Interface Channel 1 Serial Clock Selection
0	1	1	0	$f_X/2^2$ Note
0	1	1	1	$f_X/2^3$ (1.25 MHz)
1	0	0	0	$f_X/2^4$ (625 kHz)
1	0	0	1	$f_X/2^5$ (313 kHz)
1	0	1	0	$f_X/2^6$ (156 kHz)
1	0	1	1	$f_X/2^7$ (78.1 kHz)
1	1	0	0	$f_X/2^8$ (39.1 kHz)
1	1	0	1	$f_X/2^9$ (19.5 kHz)
Other than above				Setting prohibited

Note Can be set only when the main system clock oscillate at 4.19 MHz or less.

Caution If TCL3 is to be rewritten in data other than identical data, the timer operation must be stopped first.

Remarks 1. f_X: Main system clock oscillation frequency

2. Values in parentheses apply to operation with f_X = 10.0 MHz.

Figure 15-5. Serial Operating Mode Register 0 Format (1/2)

Symbol	<7>	<6>	<5>	4	3	2	1	0	Address	When Reset	R/W
CSIM0	CSIE0	COI	WUP	CSIM04	CSIM03	CSIM02	CSIM01	CSIM00	FF60H	00H	R/W[Note 1]

R/W

CSIM01	CSIM00	Serial Interface Channel 0 Clock Selection
0	×	Input clock to $\overline{SCK0}$ 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)

R/W

CSIM 04	CSIM 03	CSIM 02	PM25	P25	PM26	P26	PM27	P27	Operating Mode	Start Bit	SI0/SB0/P25 Pin Function	SO0/SBI/P26 Pin Function	$\overline{SCK0}$/P27 Pin Function
0	×	0	1	×	0	0	0	1	3-wire serial I/O mode	MSB	SI0[Note 2] (input)	SO0 (CMOS output)	$\overline{SCK0}$ (CMOS input/output)
		1								LSB			
1	0	0	× Note 3	× Note 3	0	0	0	1	SBI mode	MSB	P25 (CMOS input/output)	SB1 (N-ch open-drain input/output)	$\overline{SCK0}$ (CMOS input/output)
		1	0	0	× Note 3	× Note 3	0	1			SB0 (N-ch open-drain input/output)	P26 (CMOS input/ output)	
1	1	0	× Note 3	× Note 3	0	0	0	1	2-wire serial I/O mode	MSB	P25 (CMOS input/output)	SB1 (N-ch open-drain input/output)	$\overline{SCK0}$ (N-ch open-drain input/output)
		1	0	0	× Note 3	× Note 3	0	1			SB0 (N-ch open-drain input/output)	P26 (CMOS input/output)	

R/W

WUP	Wake-up Function Control[Note 4]
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 (SVA) data in SBI mode

Notes
1. Bit 6 (COI) is a Read-Only bit.
2. Can be used as P25 (CMOS input) when used only for transmission.
3. Can be used freely as port function.
4. When the wake-up function is used (WUP = 1), bit 5 (SIC) of the interrupt timing select register (SINT) must be set to 0.

Remark × : don't care
PMxx: Port mode register
Pxx : Output latch of port

Figure 15-5. Serial Operating Mode Register 0 Format (2/2)

R

COI	Slave Address Comparison Result Flag[Note]
0	Slave address register (SVA) not equal to serial I/O shift register 0 (SIO0) data
1	Slave address register (SVA) equal to serial I/O shift register 0 (SIO0) data

R/W

CSIE0	Serial Interface Channel 0 Operation Control
0	Operation stopped
1	Operation enabled

Note When CSIE0 = 0, COI becomes 0.

(3) Serial bus interface control register (SBIC)
This register sets serial bus interface operation and displays status.
SBIC is set with a 1-bit or 8-bit memory manipulation instruction.
$\overline{\text{RESET}}$ input sets SBIC to 00H.

Figure 15-6. Serial Bus Interface Control Register Format (1/2)

Symbol	<7>	<6>	<5>	<4>	<3>	<2>	<1>	<0>	Address	When Reset	R/W
SBIC	BSYE	ACKD	ACKE	ACKT	CMDD	RELD	CMDT	RELT	FF61H	00H	R/W[Note]

R/W

RELT	Use 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.

R/W

CMDT	Use 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

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

Note Bits 2, 3 and 6 (RELD, CMDD and ACKD) are Read-Only bits.

Remark CSIE0: Bit 7 of the serial operating mode register 0 (CSIM0)

Figure 15-6. Serial Bus Interface Control Register Format (2/2)

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/W

ACKT	Acknowledge signal is output in synchronization with the falling edge of $\overline{\text{SCK0}}$ clock immediately after execution of the instruction to be set to 1, and after acknowledge signal output, automatically cleared to 0. Also cleared to 0 upon start of serial interface transfer or when CSIE0 = 0

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 $\overline{\text{SCK0}}$ clock immediately 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.

R

ACKD	Acknowledge Detection
Clear Conditions (ACKD = 0)	Set Conditions (ACKD = 1)
• At the falling edge of $\overline{\text{SCK0}}$ immediately after the busy mode has been released when a transfer start instruction is executed • When CSIE0 = 0 • When $\overline{\text{RESET}}$ input is applied	• When acknowledge signal (ACK) is detected at the rising edge of $\overline{\text{SCK0}}$ clock after completion of transfer

R/W

BSYE[Note]	Synchronizing Busy Signal Output Control
0	Disables busy signal which is output in synchronization with the falling edge of $\overline{\text{SCK0}}$ clock immediately 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.

★ **Note** Busy mode can be cleared by start of serial interface transfer. However, BSYE flag is not cleared to 0.

Remarks **1.** Zeros will be returned from bits 0, 1, and 4 (or RELT, CMDT, ACKT, respectively) if users read these bits after data setting is completed.

2. CSIE0: Bit 7 of the serial operating mode register 0 (CSIM0)

(4) Interrupt timing specification register (SINT)

This register sets the bus release interrupt and address mask functions and displays the $\overline{\text{SCK0}}$/P27 pin level status.

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

$\overline{\text{RESET}}$ input sets SINT to 00H.

Figure 15-7. Interrupt Timing Specification Register Format

Symbol	7	<6>	<5>	<4>	3	2	1	0	Address	When Reset	R/W
SINT	0	CLD	SIC	SVAM	0	0	0	0	FF63H	00H	R/W[Note 1]

R/W

SVAM	SVA Bit to be Used as Slave Address
0	Bit 0 to Bit 7
1	Bit 1 to Bit 7

R/W

SIC	INTCSI0 Interrupt Factor Selection[Note 2]
0	CSIIF0 is set (1) upon termination of serial channel 0 transfer
1	CSIIF0 is set (1) upon bus release detection termination of serial interface channel

R

CLD	$\overline{\text{SCK0}}$/P27 Pin Level[Note 3]
0	Low Level
1	High Level

Notes **1.** Bit 6 (CLD) is a Read-Only bit.

2. When using wake-up function, set SIC to 0.

3. When CSIE0 = 0, CLD becomes 0.

Caution **Be sure to set bit 0 to bit 3 to 0.**

Remark SVA : Slave address register

CSIIF0: Interrupt request flag supports the INTCSI0

CSIE0 : Bit 7 of the serial operating mode register 0 (CSIM0)

15.4 Serial Interface Channel 0 Operations

The following four operating modes are available to the 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 dissipation can be reduced. The serial I/O shift register 0 (SIO0) does not carry out shift operation either and thus it can be used as normal 8-bit register.

In the operation stop mode, the P25/SI0/SB0, P26/SO0/SB1 and P27/$\overline{SCK0}$ pins can be used as normal input/output ports.

(1) Register setting

The operation stop mode is set with the serial operating mode register 0 (CSIM0).

CSIM0 is set with a 1-bit or 8-bit memory manipulation instruction.

$\overline{RESET}$ input sets CSIM0 to 00H.

Symbol	<7>	<6>	<5>	4	3	2	1	0	Address	When 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 enable

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 the serial operating mode register 0 (CSIM0) and 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 sets CSIM0 to 00H.

Symbol	<7>	<6>	<5>	4	3	2	1	0	Address	When Reset	R/W
CSIM0	CSIE0	COI	WUP	CSIM04	CSIM03	CSIM02	CSIM01	CSIM00	FF60H	00H	R/W[Note 1]

R/W

CSIM01	CSIM00	Serial Interface Channel 0 Clock Selection
0	×	Input clock to $\overline{SCK0}$ 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)

R/W

CSIM 04	CSIM 03	CSIM 02	PM25	P25	PM26	P26	PM27	P27	Operating Mode	Start Bit	SI0/SB0/P25 Pin Function	SO0/SBI/P26 Pin Function	$\overline{SCK0}$/P27 Pin Function
0	×	0	1	×	0	0	0	1	3-wire serial I/O mode	MSB	SI0[Note 2] (Input)	SO0 (CMOS output)	$\overline{SCK0}$ (CMOS input/output)
		1								LSB			
1	0	SBI mode (Refer to **15.4.3 SBI mode operation**)											
1	0	2-wire serial I/O mode (Refer to **15.4.4 2-wire serial I/O mode operation**)											

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 (SVA) in SBI mode

R/W

CSIE0	Serial Interface Channel 0 Operation Control
0	Operation stopped
1	Operation enable

Notes
1. Bit 6 (COI) is a Read-Only bit.
2. Can be used as P25 (CMOS input) when used only for transmission.
3. Be sure to set WUP to 0 when the 3-wire serial I/O mode is selected.

Remark × : don't care
PMxx: Port mode register
Pxx : Output latch of port

(b) Serial bus interface control register (SBIC)
SBIC is set with a 1-bit or 8-bit memory manipulation instruction.
$\overline{\text{RESET}}$ input sets SBIC to 00H.

Symbol	<7>	<6>	<5>	<4>	<3>	<2>	<1>	<0>	Address	When Reset	R/W
SBIC	BSYE	ACKD	ACKE	ACKT	CMDD	RELD	CMDT	RELT	FF61H	00H	R/W

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.

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.

CSIE0: Bit 7 of the serial operating mode register 0 (CSIM0)

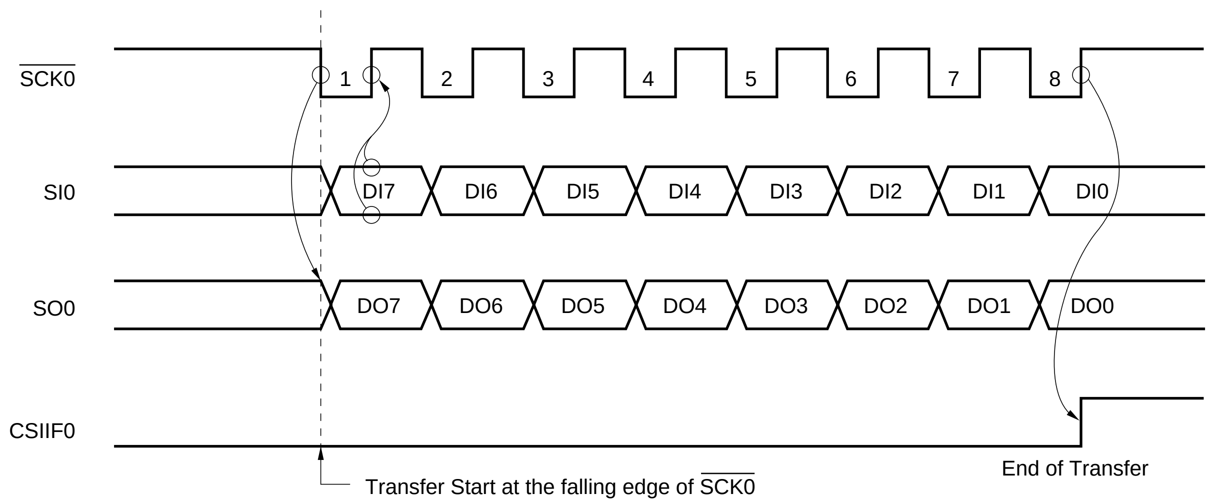
(2) Communication operation

The 3-wire serial I/O mode is used for data transmission/reception in 8-bit units. Data transmission/reception is carried out bit-wise in synchronization of the serial clock.

Shift operation of the 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 15-8. 3-Wire Serial I/O Mode Timings



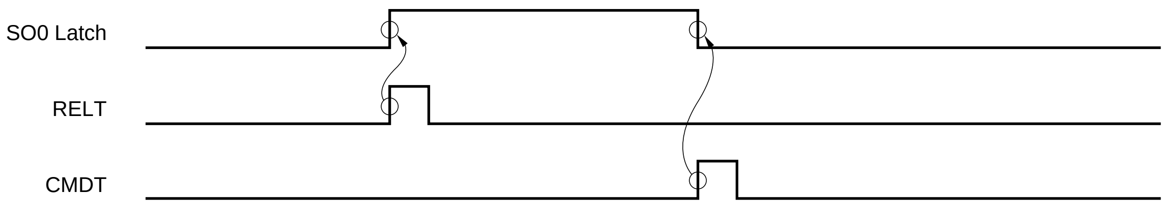
The SO0 pin serves for CMOS output and generates the SO0 latch status. Thus, the SO0 pin output status can be manipulated by setting bit 0 (RELT) and bit 1 (CMDT) of the serial bus interface control register (SBIC). However, do not carry out this manipulation during serial transfer.

The $\overline{\text{SCK0}}$ pin output level is controlled by manipulating the P27 output latch in the output mode (internal system clock mode) (refer to **15.4.5 $\overline{\text{SCK0}}$/P27 pin output manipulation**).

(3) Various signals

Figure 15-9 shows RELT and CMDT operations.

Figure 15-9. RELT and CMDT Operations



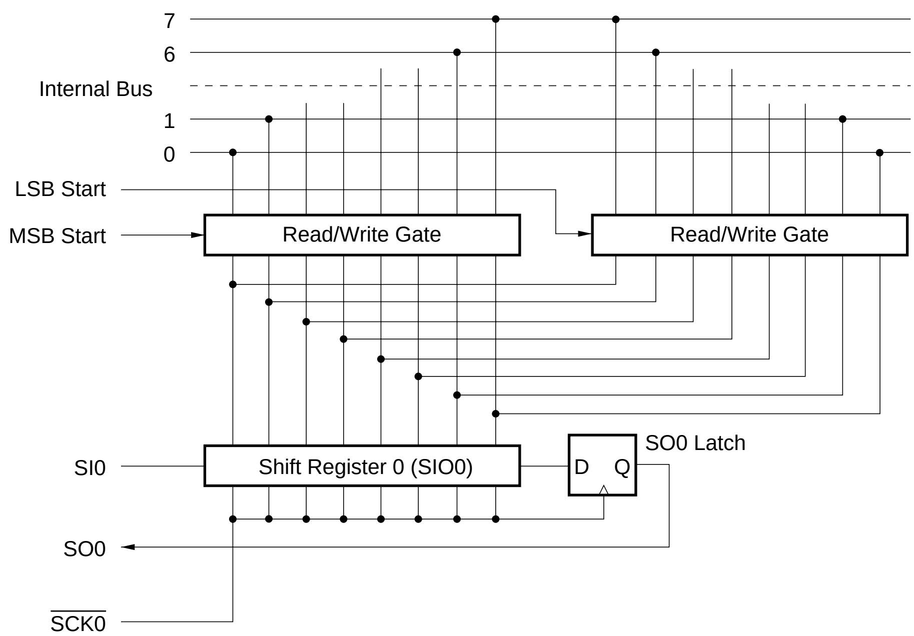
(4) MSB/LSB switching as the start bit

The 3-wire serial I/O mode enables to select transfer to start at MSB or LSB.

Figure 15-10 shows the configuration of the serial I/O shift register 0 (SIO0) and internal bus. As shown in the figure, MSB/LSB can be read/written in inverted form.

MSB/LSB switching as the start bit can be specified with bit 2 (CSIM02) of the serial operating mode register 0 (CSIM0).

Figure 15-10. 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, switch the MSB/LSB start bit before writing data to the shift register.

(5) Transfer start

Serial transfer is started by setting transfer data to the 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 the 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 has a format with the bus configuration function added to the clocked serial I/O method so that it can carry out communication with two or more devices with two signal conductors on the single-master high-speed serial bus. 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 can output to the serial data bus of the slave device "addresses" for selection of the serial communication target device, "commands" to instruct the target device and actual "data". The slave device can identify the received data into "address", "command" or "data", by hardware. This function enables the application program to control serial interface channel 0 to be simplified.

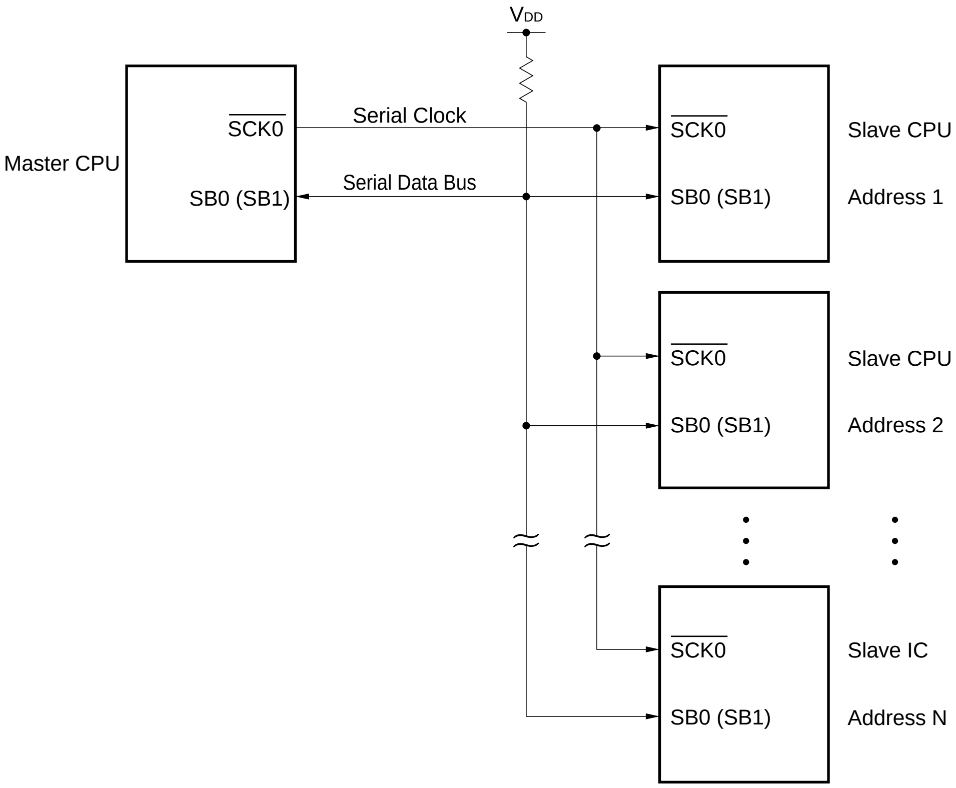
The SBI function is incorporated into various devices including 75X/XL Series devices and 78K Series.

Figure 15-11 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 (or SB1) serial data bus pin serves for open-drain output and so the serial data bus line is in wired-OR state. A pull-up resistor is necessary for the serial data bus line.

Refer to **(11) Cautions on SBI mode (d)** described later when the SBI mode is used.

Figure 15-11. Example of Serial Bus Configuration with SBI



Caution When replacing 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 method, when a serial bus is constructed by connecting two or more devices, many ports and wiring are necessary to distinguish chip select signals and command/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 constructed with two signal conductors of serial clock $\overline{\text{SCK0}}$ and serial data bus SB0 (SB1). Thus, SBI is effective to decrease 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, a CPU other than those of the selected slave devices can operate regardless of serial communication.

(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.

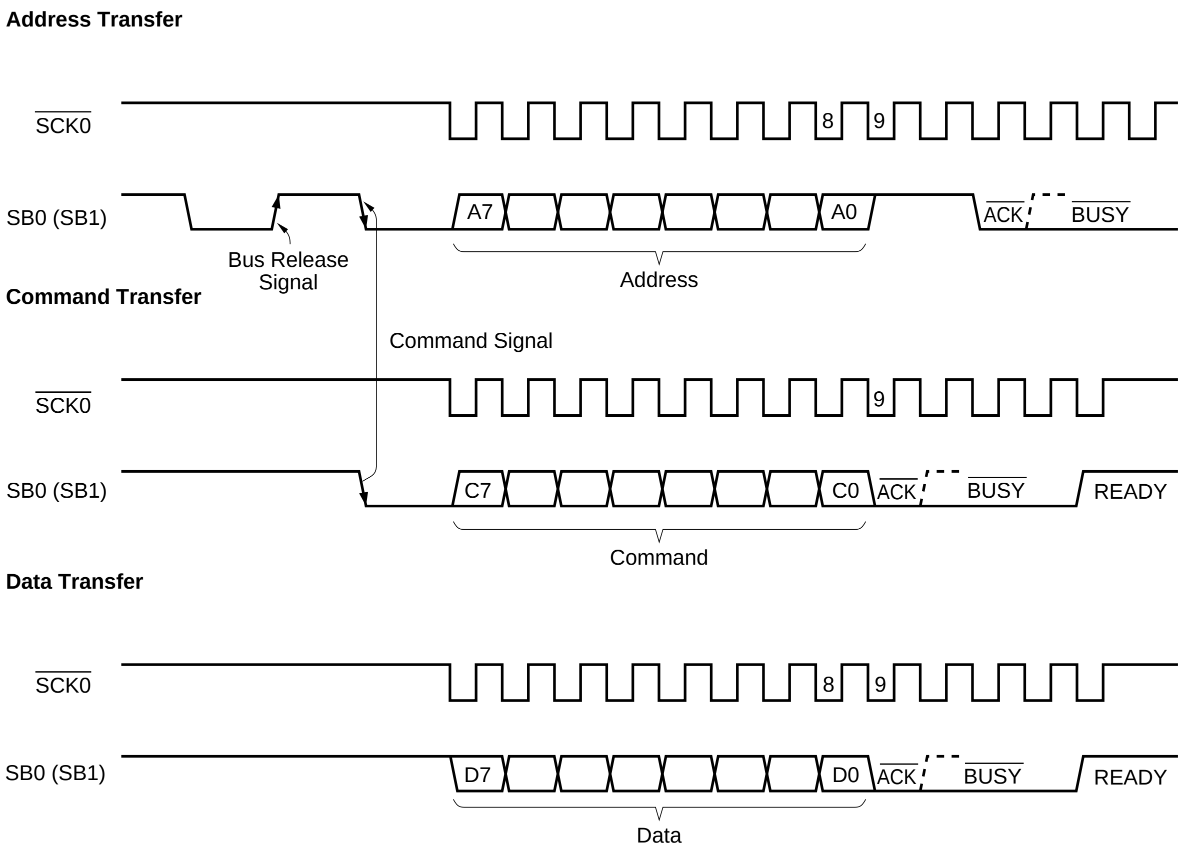
(2) SBI definition

The SBI serial data format and implication of signals to be used are defined as follows.

Serial data to be transferred with SBI is distinguished into three types, “address”, “command” and “data”.

Figure 15-12 shows the address, command and data transfer timings.

Figure 15-12. SBI Transfer Timings



Remark The broken line indicates the READY state.

The bus release signal and the command signal are output by the master device. BUSY is output by the slave signal. ACK can be output by either the master or slave device (normally, the 8-bit data receiver outputs).

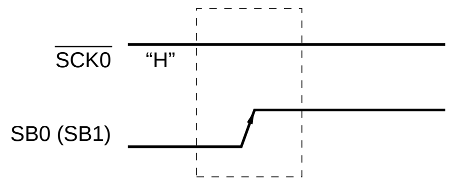
Serial clocks continue to be output by the master device from 8-bit data transfer start to BUSY reset.

(a) Bus release signal (REL)

The bus release signal is generated when the $\overline{SCK0}$ line is in high level (a serial clock is not output) and the SB0 (SB1) line changes from low level to high level.

The bus release signal is output by the master.

Figure 15-13. Bus Release Signal



The bus release signal indicates that the master will send the address to the slave. The slave contains hardware to detect the bus release signal.

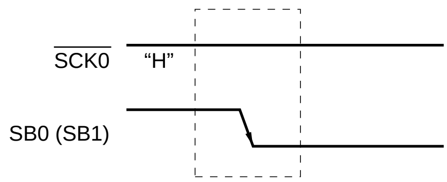
★ **Caution The bus release signal is acknowledged when the $\overline{SCK0}$ line is in high level, and the SB0 (SB1) line changes from low level to high level. Thus, if the timing at which bus changes deviates due to effects such as board capacity, it may be determined as the bus release signal even if data is sent. Therefore perform wiring carefully.**

(b) Command signal (CMD)

The command signal is generated when the $\overline{SCK0}$ line is in high level (a serial clock is not output) and the SB0 (SB1) line changes from high level to low level.

The command signal is output by the master.

Figure 15-14. Command Signal



The command signal indicates that the master will send the command to the slave (However, the command signal following the bus release signal indicates that address will be sent).

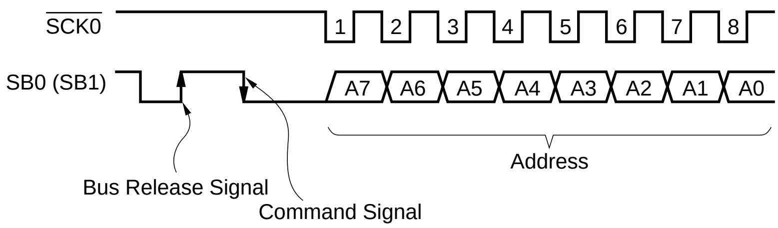
The slave contains hardware to detect the command signal.

★ **Caution The command signal is acknowledged when the $\overline{SCK0}$ line is in high level, and the SB0 (SB1) line changes from high level to low level. Thus, if the timing at which bus changes deviates due to effects such as board capacity, it may be determined as the command signal even if data is sent. Therefore perform wiring carefully.**

(c) Address

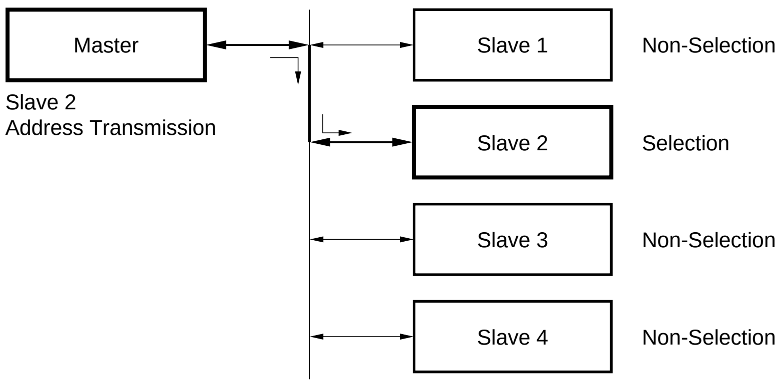
The address is 8-bit data that the master outputs to the slave connected to the bus line to select a specific slave.

Figure 15-15. Address



8-bit data following the bus release signal and the command signal is defined as the address. The slave detects the condition and checks by hardware if 8-bit data matches its specified number (the slave address). When 8-bit data matches the slave address, which means the slave is selected, the slave communicates with the master until the master instructs disconnection.

Figure 15-16. Slave Selection with Address



(d) Command and Data

The master sends commands and sends/receives data to the slave selected by sending the address.

Figure 15-17. Command

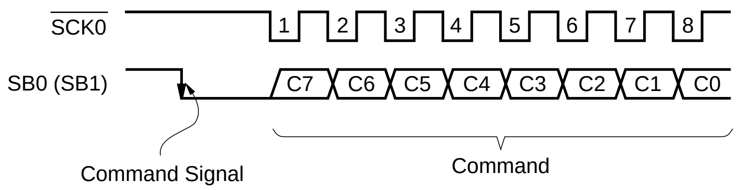
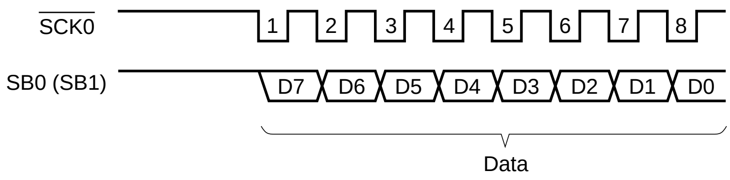


Figure 15-18. Data



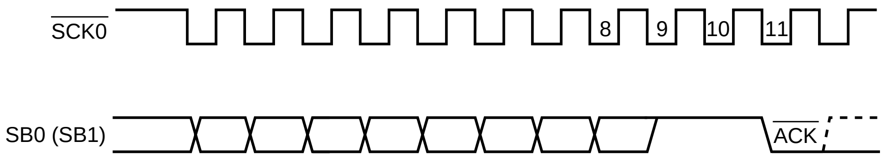
8-bit data following the command signal is defined as a command. 8-bit data wihtout the command signal is defined as data. How to use the command and data can be determined based on communication specifications.

(e) Acknowledge signal ($\overline{\text{ACK}}$)

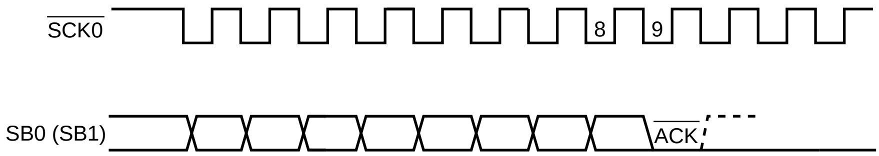
This signal is used between the sending side and receiving side devices for confirmation of correct serial data sending.

Figure 15-19. Acknowledge Signal

(When output synchronously with $\overline{\text{SCK0}}$ in 11th clock.)



(When output synchronously with $\overline{\text{SCK0}}$ in 9th clock.)



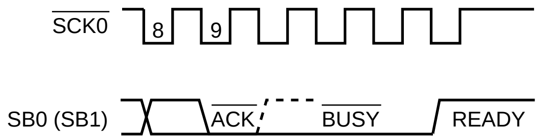
Remark The broken line indicates the READY state.

The acknowledge signal is a one-shot pulse synchronous with $\overline{\text{SCK0}}$ falling, whose position can be synchronized with $\overline{\text{SCK0}}$ in any clock.

The sending side that has transferred 8-bit data checks if the acknowledge signal has been sent back by the receiving side. If this signal is not sent back by the slave device for a period after data sending, this means that the data sent has not been received correctly by the slave device.

(f) Busy signal ($\overline{\text{BUSY}}$), Ready signal (READY)
The busy signal informs the master that the slave is busy transmitting/receiving data.
The ready signal informs the master that the slave is ready to transmit/receive data.

Figure 15-20. Busy Signal and Ready Signal



Remark The broken line indicates the READY state.

In the SBI mode, the slave informs the master of the busy state by setting the SB0 (SB1) line to low level. The busy signal is output following the acknowledge signal output by the slave. The busy signal is set/cleared synchronously with the falling edge of $\overline{\text{SCK0}}$. The master terminates automatically to output the serial clock $\overline{\text{SCK0}}$ when the busy signal is cleared.
The master can start subsequent transmissions when the busy signal is cleared and changes to the ready state.

Caution In the SBI mode, the $\overline{\text{BUSY}}$ signal is output until the falling of the next serial clock after the $\overline{\text{BUSY}}$ release indication. If WUP = 1 is set by mistake during this period, $\overline{\text{BUSY}}$ will not be released. Thus, after releasing $\overline{\text{BUSY}}$, be sure to check that the SB0 (SB1) has become high level before setting WUP = 1.

(3) Register setting
The SBI mode is set with the serial operating mode register 0 (CSIM0), the serial bus interface control register (SBIC) and the interrupt timing specification 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 sets CSIM0 to 00H.

Symbol	<7>	<6>	<5>	4	3	2	1	0	Address	When Reset	R/W
CSIM0	CSIE0	COI	WUP	CSIM04	CSIM03	CSIM02	CSIM01	CSIM00	FF60H	00H	R/W[Note 1]

R/W

CSIM01	CSIM00	Serial Interface Channel 0 Clock Selection
0	×	Input clock to $\overline{SCK0}$ 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)

R/W

<table>
<tr><th>CSIM 04</th><th>CSIM 03</th><th>CSIM 02</th><th>PM25</th><th>P25</th><th>PM26</th><th>P26</th><th>PM27</th><th>P27</th><th>Operating Mode</th><th>Start Bit</th><th>SI0/SB0/P25 Pin Function</th><th>SO0/SB1/P26 Pin Function</th><th>$\overline{SCK0}$/P27 Pin Function</th></tr>
<tr><td>0</td><td>×</td><td colspan="12">3-wire serial I/O mode (Refer to 15.4.2 3-wire serial I/O mode operation)</td></tr>
<tr><td rowspan="2">1</td><td rowspan="2">0</td><td>0</td><td>Note 2 ×</td><td>Note 2 ×</td><td>0</td><td>0</td><td>0</td><td>1</td><td rowspan="2">SBI mode</td><td rowspan="2">MSB</td><td>P25 (CMOS input/output)</td><td>SB1 (N-ch open-drain input/output)</td><td rowspan="2">$\overline{SCK0}$ (CMOS input/output)</td></tr>
<tr><td>1</td><td>0</td><td>0</td><td>Note 2 ×</td><td>Note 2 ×</td><td>0</td><td>1</td><td>SB0 (N-ch open-drain input/output)</td><td>P26 (CMOS input/ output)</td></tr>
<tr><td>1</td><td>1</td><td colspan="12">2-wire serial I/O mode (Refer to 15.4.4 2-wire serial I/O mode operation)</td></tr>
</table>

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 (SVA) data when SBI mode is used

R

COI	Slave Address Comparison Result Flag[Note 4]
0	Slave address register (SVA) not equal to serial I/O shift register 0 (SIO0) data
1	Slave address register (SVA) equal to serial I/O shift register 0 (SIO0) data

R/W

CSIE0	Serial Interface Channel 0 Operation Control
0	Operation stopped
1	Operation enable

Notes
1. Bit 6 (COI) is a Read-Only bit.
2. Can be used freely as port function.
3. When the wake-up function is used (WUP = 1), set bit 5 (SIC) of the interrupt timing specification register (SINT) to 0.
4. When CSIE0 = 0, COI becomes 0.

Remark × : don't care
PMxx: Port mode register
Pxx : Output latch of port

(b) Serial bus interface control register (SBIC)

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

RESET input sets SBIC to 00H.

Symbol	<7>	<6>	<5>	<4>	<3>	<2>	<1>	<0>	Address	When Reset	R/W
SBIC	BSYE	ACKD	ACKE	ACKT	CMDD	RELD	CMDT	RELT	FF61H	00H	R/W[Note]

R/W	RELT	Use 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.

R/W	CMDT	Use 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

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 (only if WUP = 1) • When CSIE0 = 0 • When RESET input is applied	• When bus release signal (REL) is detected

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 RESET input is applied	• When command signal (CMD) is detected

R/W	ACKT	Acknowledge signal is output in synchronization with the falling edge clock of SCK0 just after execution of the instruction to be set to 1, and after acknowledge signal output, automatically cleared to 0. Used as ACKE = 0 Also cleared to 0 upon start of serial interface transfer or when CSIE0 = 0.

(continued)

Note Bits 2, 3, and 6 (RELD, CMDD and ACKD) are Read-Only bits.

Remarks **1.** Zeros will be returned from bits 0, 1, and 4 (or RELT, CMDT, ACKT, respectively) if users read these bits after data setting is completed.

2. CSIE0: Bit 7 of the serial operating mode register 0 (CSIM0)

(continued)

R/W

ACKE	Acknowledge Signal Automatic 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 $\overline{SCK0}$ (automatically output when ACKE = 1).
	After completion of transfer	Acknowledge signal is output in synchronization with the falling edge of $\overline{SCK0}$ clock immediately 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.

R

ACKD	Acknowledge Detection
Clear Conditions (ACKD = 0)	Set Conditions (ACKD = 1)
• At the falling edge of $\overline{SCK0}$ clock immediately after the busy mode has been released when a transfer start instruction is executed • When CSIE0 = 0 • When $\overline{RESET}$ input is applied	• When acknowledge signal ($\overline{ACK}$) is detected at the rising edge of $\overline{SCK0}$ clock after completion of transfer

R/W

BSYE[Note]	Synchronizing Busy Signal Output Control
0	Disables busy signal which is output in synchronization with the falling edge of $\overline{SCK0}$ clock immediately after execution of the instruction to be cleared to (0) (with READY state).
1	Outputs busy signal at the falling edge of $\overline{SCK0}$ clock following the acknowledge signal.

★ **Note** Busy mode can be cleared by start of serial interface transfer. However, the BSYE flag is not cleared to 0.

Remark CSIE0: Bit 7 of the serial operating mode register 0 (CSIM0)

(c) Interrupt timing specification register (SINT)

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

$\overline{\text{RESET}}$ input sets SINT to 00H.

Symbol	7	<6>	<5>	<4>	3	2	1	0	Address	When Reset	R/W
SINT	0	CLD	SIC	SVAM	0	0	0	0	FF63H	00H	R/W[Note 1]

R/W

SVAM	SVA Bit to be Used as Slave Address
0	Bits 0 to 7
1	Bits 1 to 7

R/W

SIC	INTCSI0 Interrupt Factor Selection[Note 2]
0	CSIIF0 is set (1) upon termination of serial channel 0 transfer
1	CSIIF0 is set (1) upon bus release detection

R

CLD	$\overline{\text{SCK0}}$/P27 Pin Level[Note 3]
0	Low Level
1	High Level

Notes **1.** Bit 6 (CLD) is a Read-Only bit.

2. When using wake-up function, set SIC to 0.

3. When CSIE0 = 0, CLD becomes 0.

Caution **Be sure to set bits 0 to 3 to 0.**

Remark SVA : Slave address register

CSIIF0: Interrupt request flag supports the INTCSI0

CSIE0 : Bit 7 of the serial operating mode register 0 (CSIM0)

(4) Various signals

Figures 15-21 to 15-26 show various signals in SBI and flag operations of the serial bus interface control register (SBIC). Table 15-4 lists various signals in SBI.

Figure 15-21. RELT, CMDT, RELD and CMDD Operations (Master)

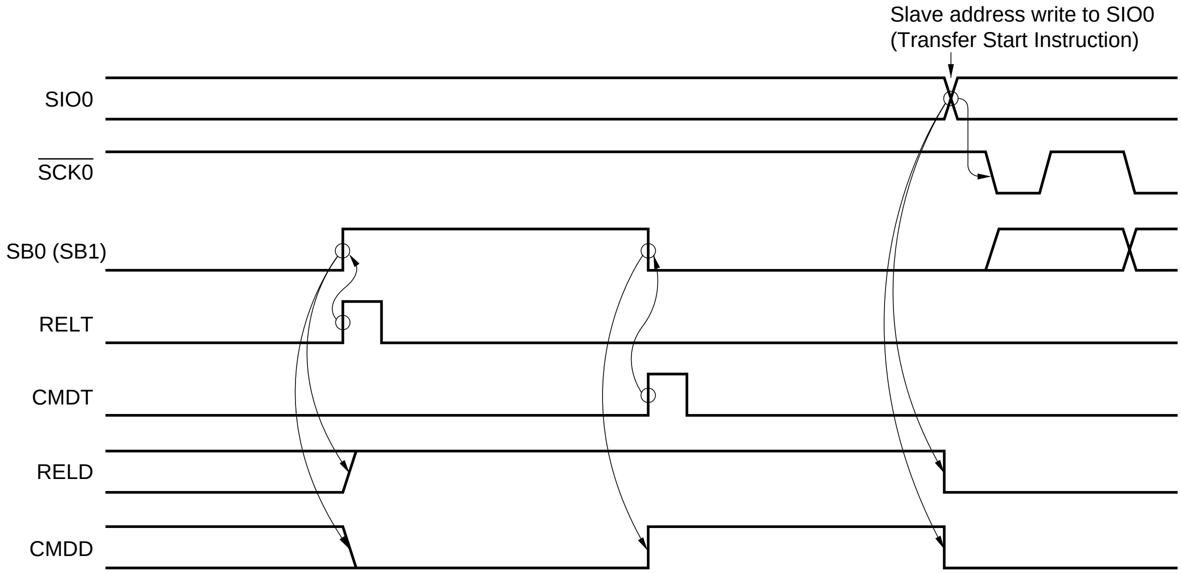


Figure 15-22. RELD and CMDD Operations (Slave)

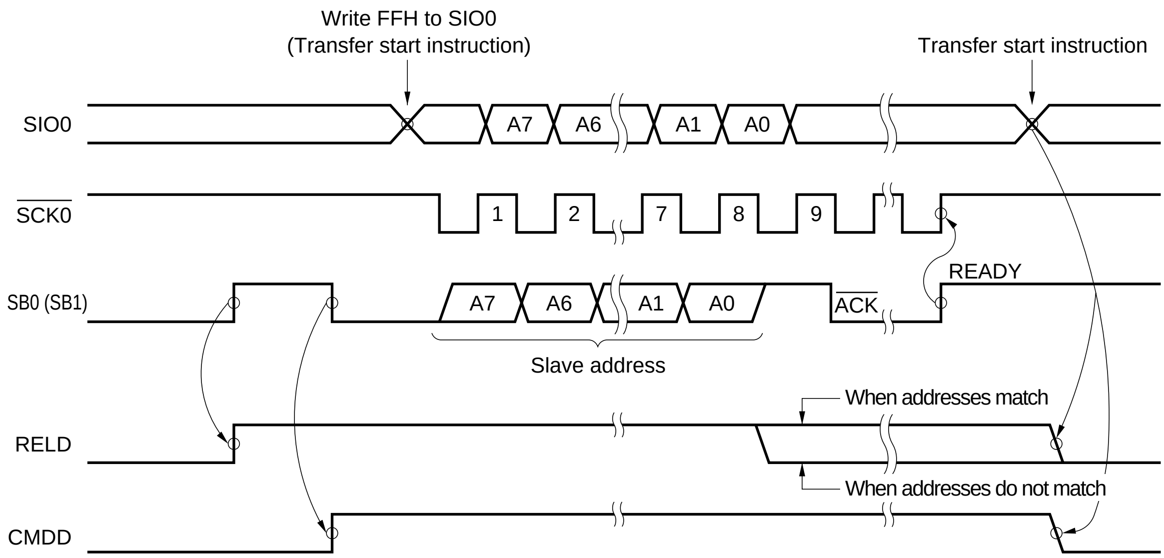
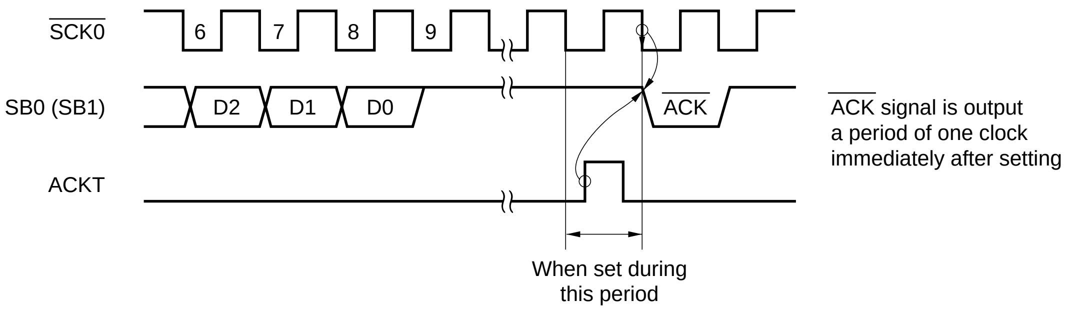


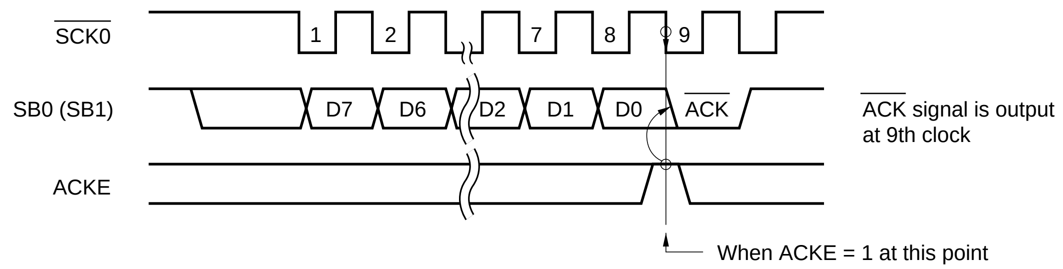
Figure 15-23. ACKT Operation



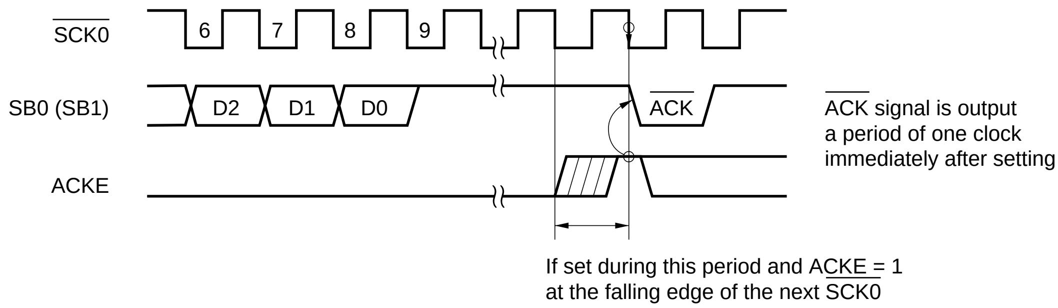
Caution Do not set ACKT before termination of transfer.

Figure 15-24. ACKE Operations

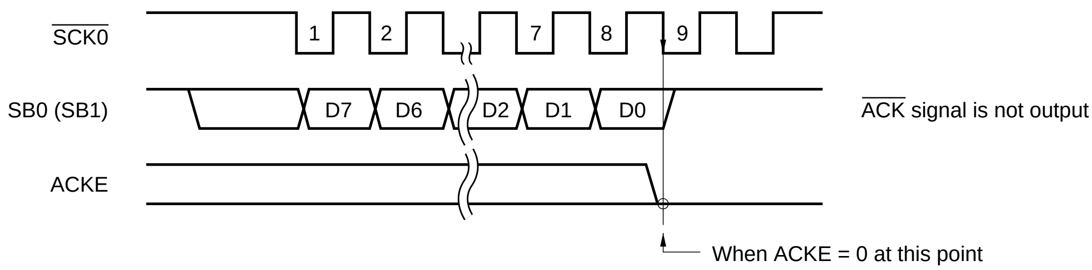
(a) When ACKE = 1 upon completion of transfer



(b) When set after completion of transfer



(c) When ACKE = 0 upon completion of transfer



(d) When ACKE = 1 period is short

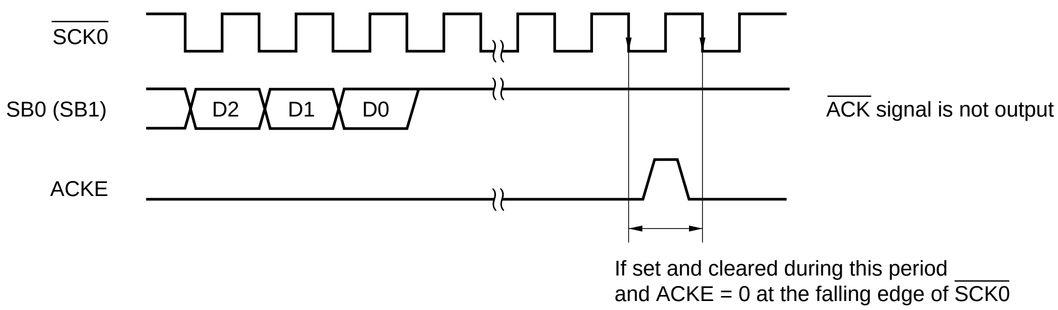
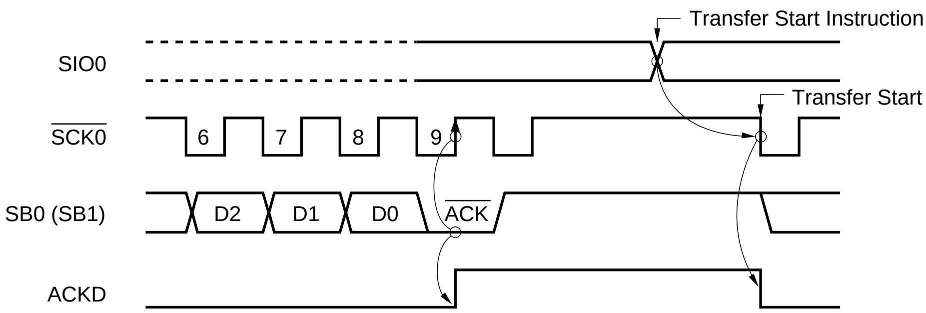
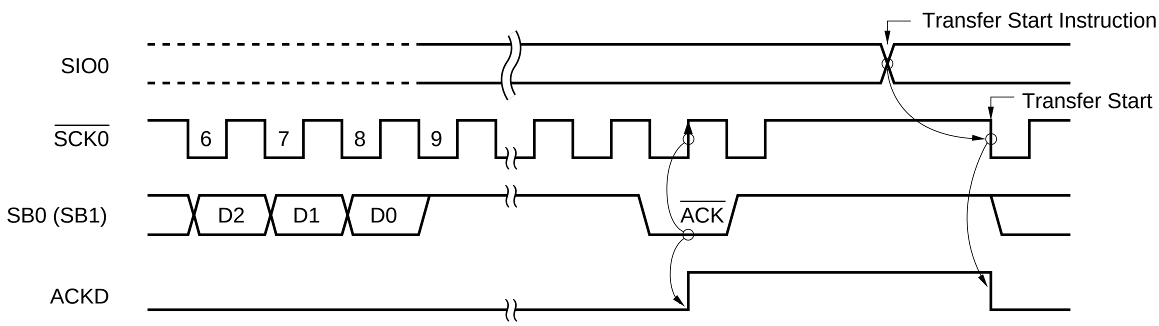


Figure 15-25. ACKD Operations

(a) When $\overline{ACK}$ signal is output at 9th clock of $\overline{SCK0}$



(b) When $\overline{ACK}$ signal is output after 9th clock of $\overline{SCK0}$



(c) Clear timing when transfer start is instructed in BUSY

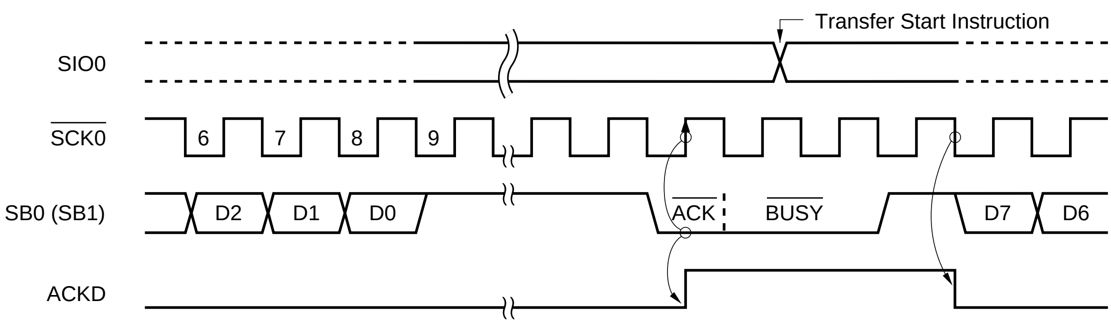


Figure 15-26. BSYE Operation

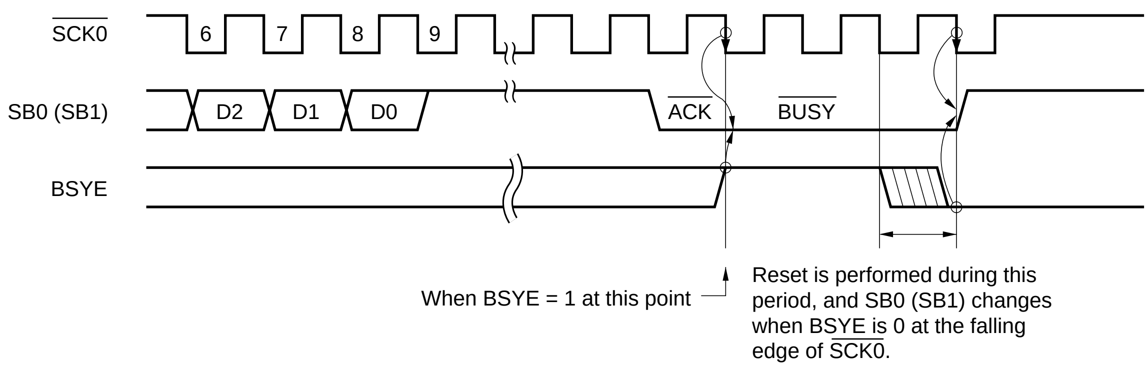


Table 15-4. Various Signals in SBI Mode (1/2)

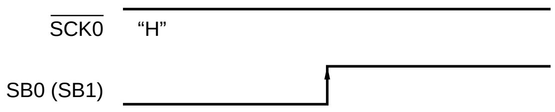
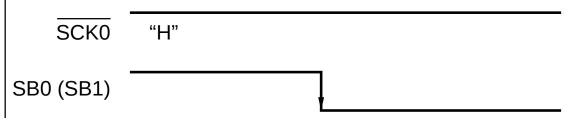
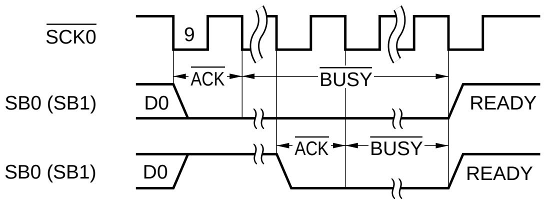
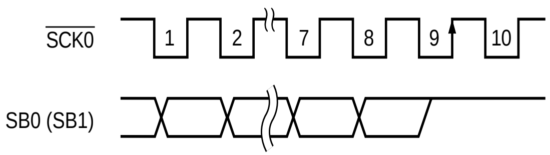
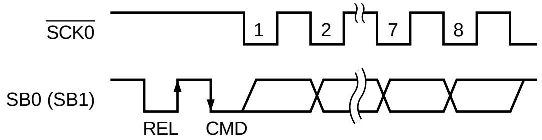
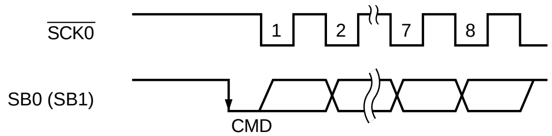
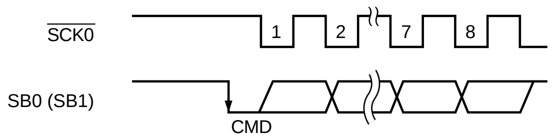
Signal Name	Output Device	Definition	Timing Chart	Output Condition	Effects on Flag	Meaning of Signal
Bus release signal (REL)	Master	SB0 (SB1) rising edge when $\overline{SCK0}$ = 1		• RELT set	• RELD set • CMDD clear	CMD signal is output to indicate that transmit data is an address.
Command signal (CMD)	Master	SB0 (SB1) falling edge when $\overline{SCK0}$ = 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 ($\overline{ACK}$)	Master/slave	Low-level signal to be output to SB0 (SB1) during one-clock period of $\overline{SCK0}$ after completion of serial reception		<1> ACKE = 1 <2> ACKT set	• ACKD set	Completion of reception
Busy signal ($\overline{BUSY}$)	Slave	[Synchronous BUSY signal] Low-level signal to be output to SB0 (SB1) following Acknowledge signal	[Synchronous BUSY output] 	• 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		<1> BSYE = 0 <2> Execution of instruction data write SIO0 (trans start instruction)	—	Serial receive enable

Table 15-4. 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, synchronization $\overline{\text{BUSY}}$ 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
Address (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				Instruction messages to the slave device
Address (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 always 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), CSIIF0 is set (when the address does not match, RELD is cleared).

2. In $\overline{\text{BUSY}}$ state, transfer starts after the READY state is set.

(5) Pin configuration

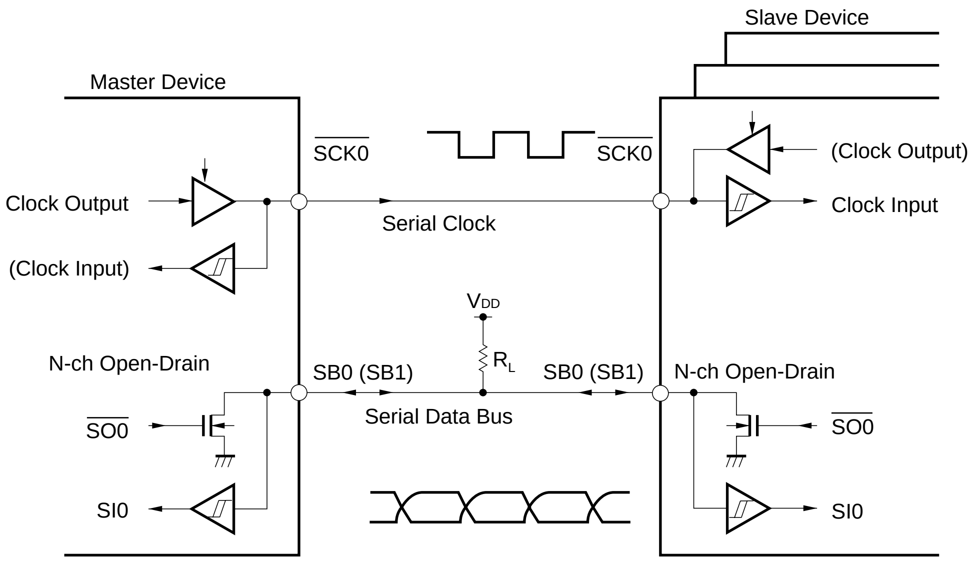
The serial clock pin SCK0 and serial data bus pin SB0 (SB1) have the following configurations.

(a) SCK0 .... Serial clock input/output pin
  <1> Master .. CMOS and push-pull output
  <2> Slave .... Schmitt input

(b) SB0 (SB1) .... Serial data input/output 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 15-27. Pin Configuration



Caution Because the N-ch open-drain output must be set to high-impedance at the time of data reception, write FFH to the serial I/O shift register 0 (SIO0) in advance. However, when the wake-up function specification bit (WUP) = 1, the N-ch open-drain output will always be set to high-impedance. Thus, it is not necessary to write FFH to SIO0 before reception.

(6) Address match detection method

In the SBI mode, a particular slave device can be selected by sending a slave address from the master device. Address match detection is automatically executed by hardware. With the slave address register (SVA), and if the wake-up function specification bit (WUP) = 1, CSIIF0 is set only when the selve address transmitted from the master device matches the value set in SVA.

If bit 5 (SIC) of the interrupt timing specification register (SINT) is set to 1, the wake-up function dose not operate even with WUP = 1 (an interrupt request signal is generated in detecting a bus release). When the wake-up function is used, 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 request (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 request 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, the serial I/O shift register 0 (SIO0). Thus, transmit errors can be detected in the following ways:

(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, the COI bit (match signal coming from the address comparator) of the serial operating mode register 0 (CSIM0) is tested. If it is “1”, normal transmission is judged to have been carried out. If it is “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 15-28 to 15-31 show data communication timing charts.

Shift operation of the serial I/O shift register 0 (SIO0) 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 SIO0.

Figure 15-28. Address Transmission from Master Device to Slave Device (WUP = 1)

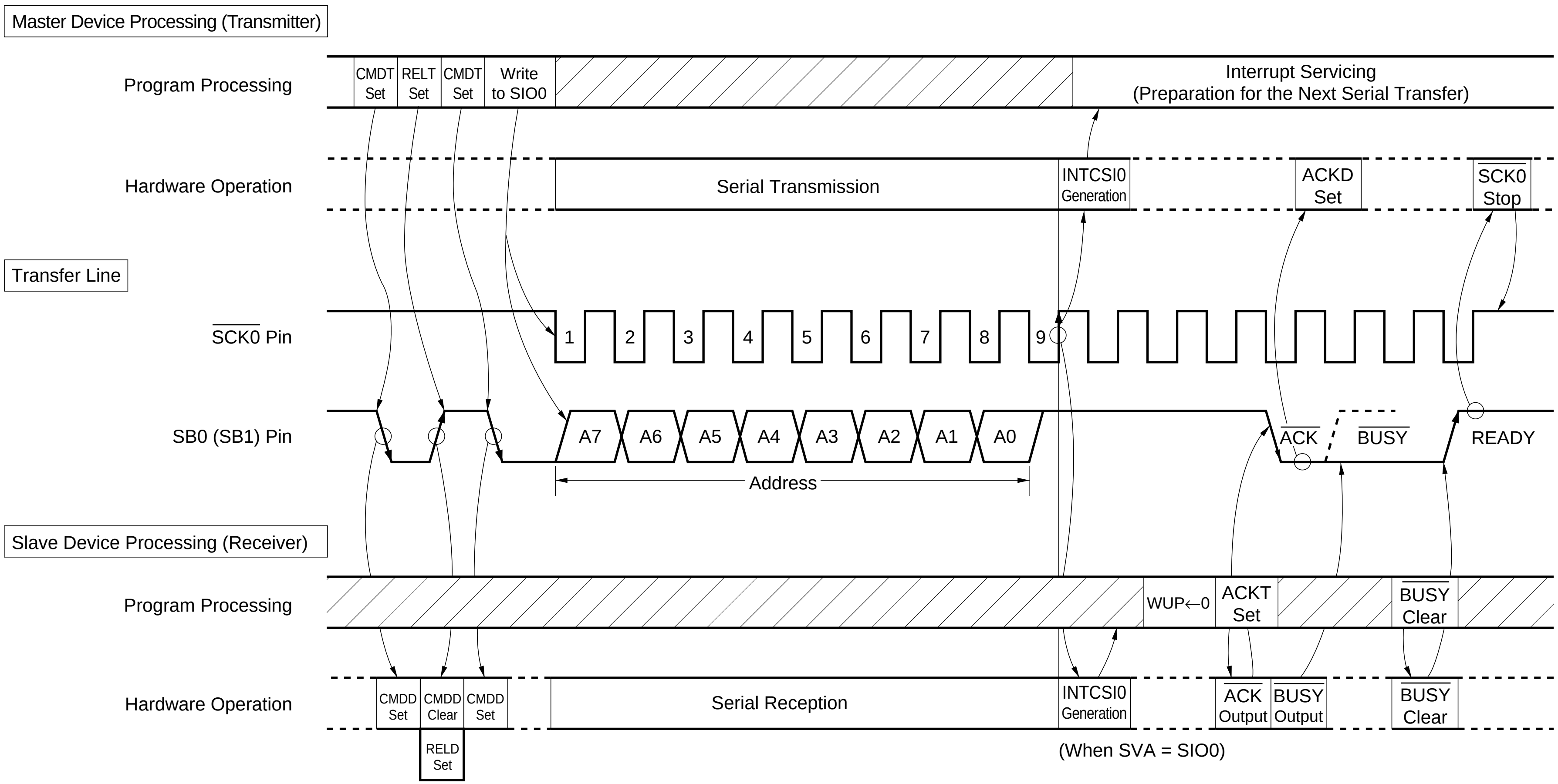


Figure 15-29. Command Transmission from Master Device to Slave Device

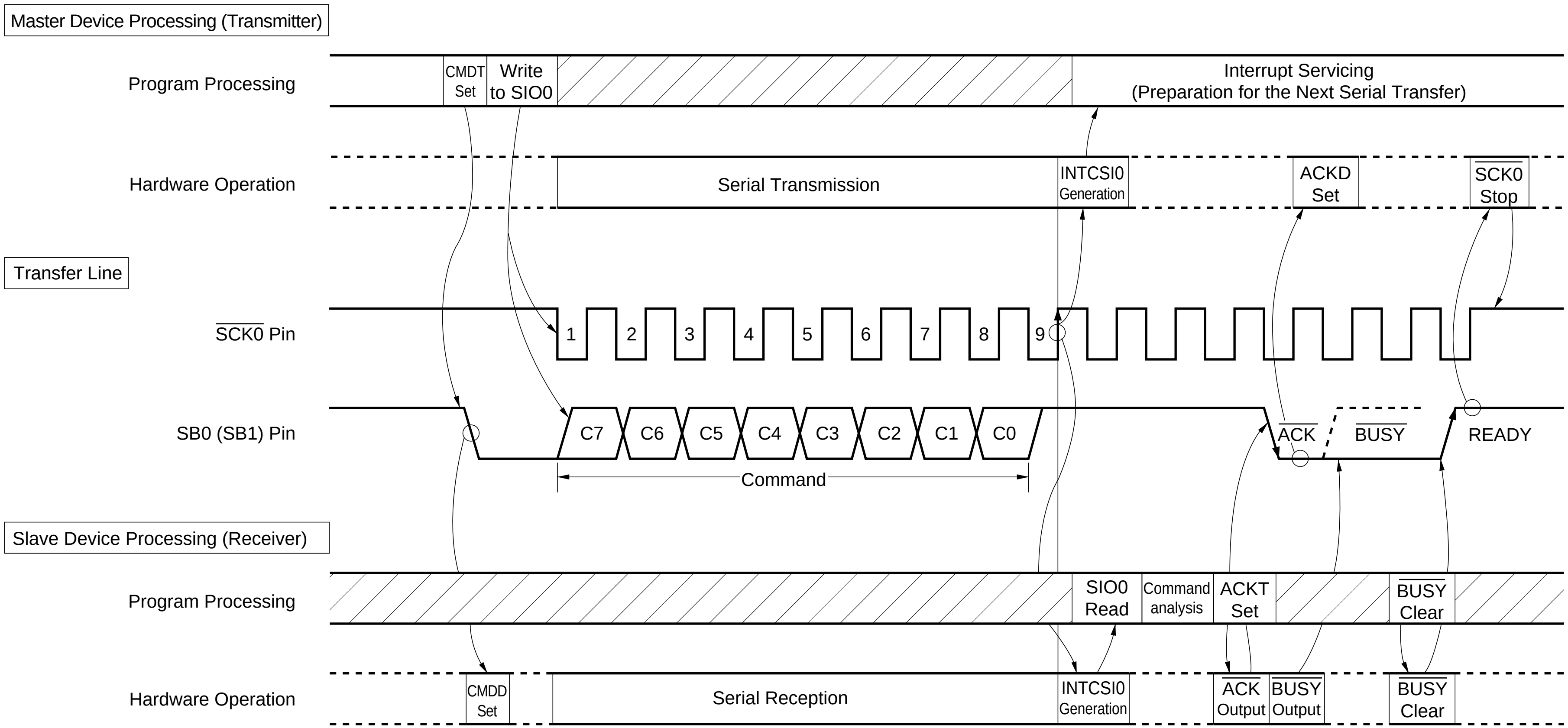


Figure 15-30. Data Transmission from Master Device to Slave Device

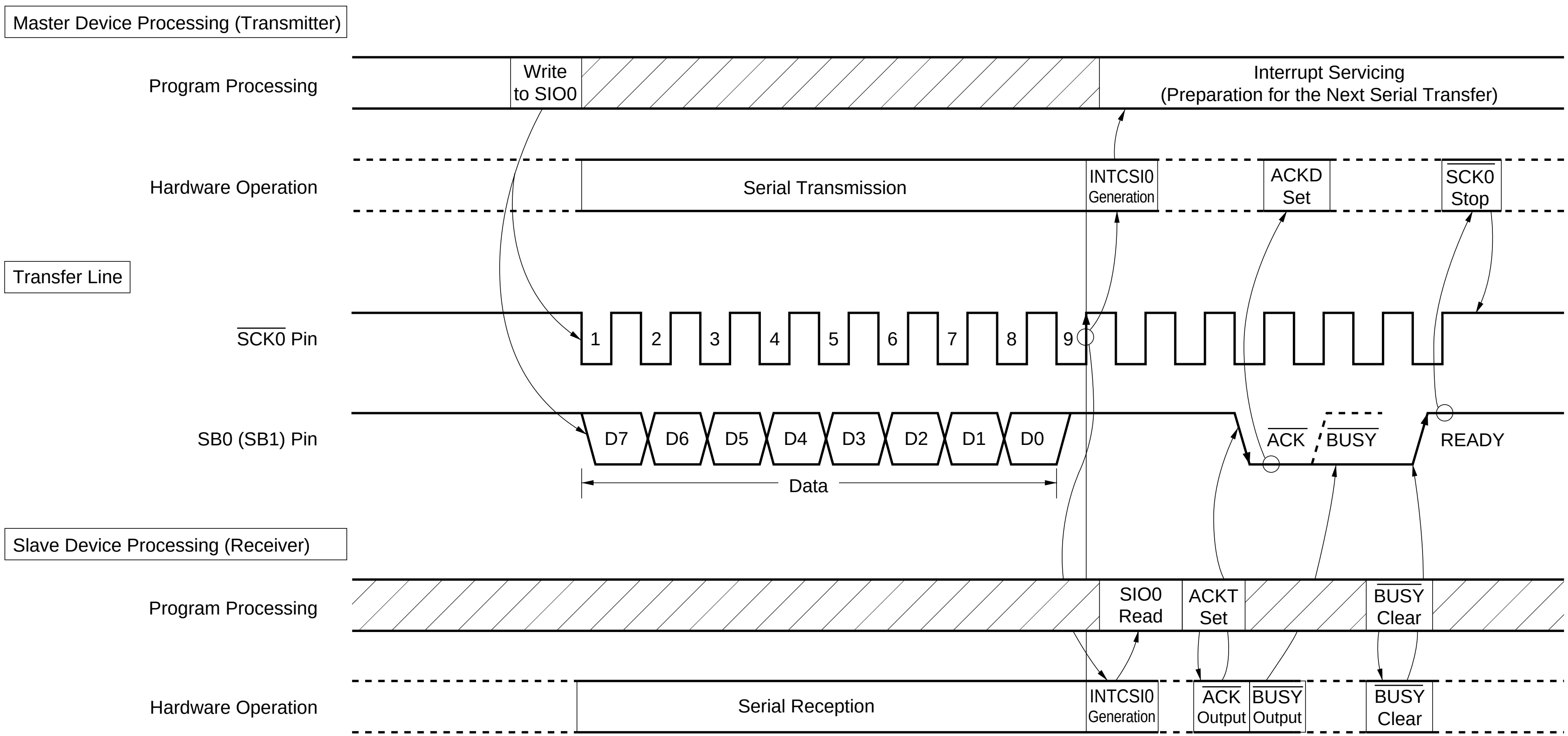
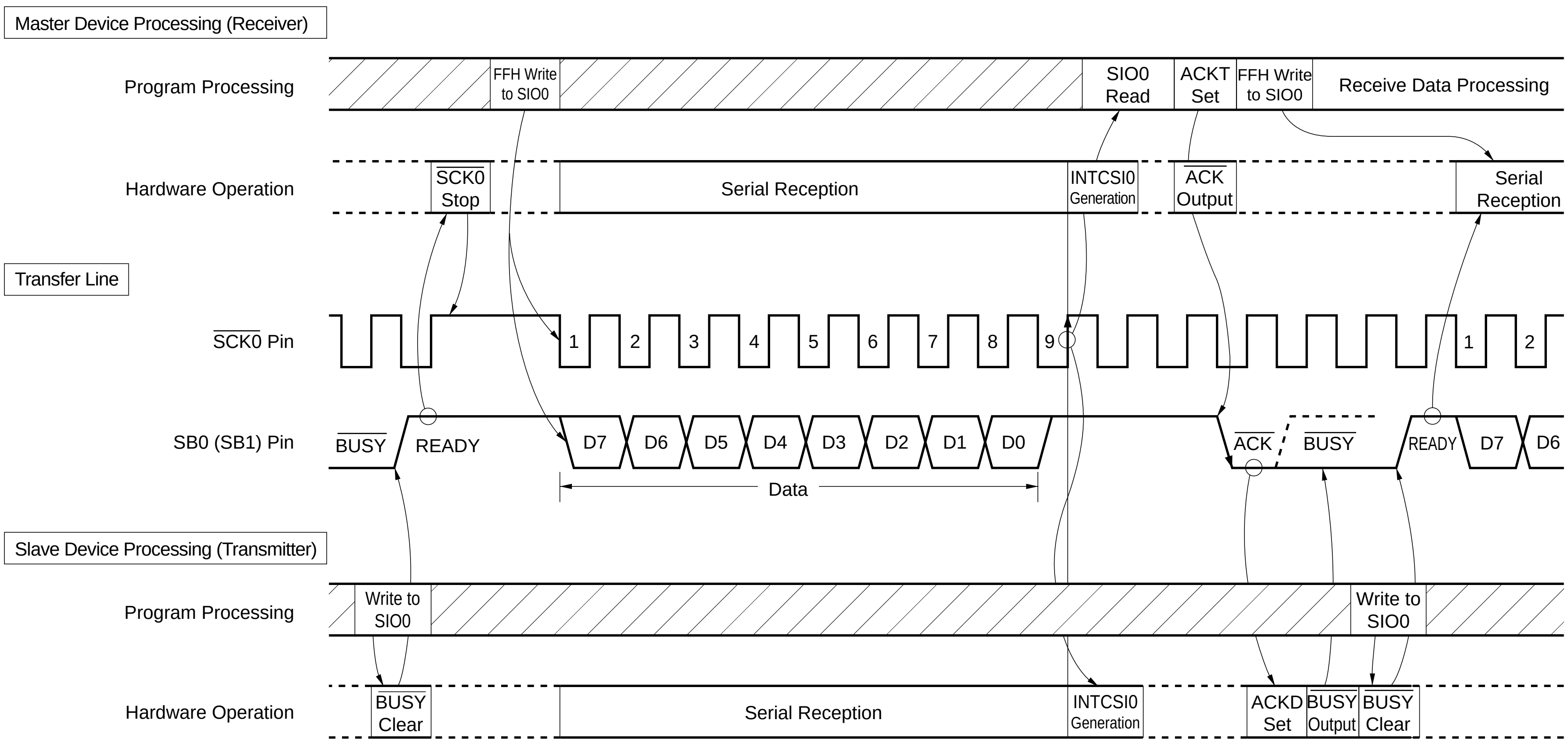


Figure 15-31. Data Transmission from Slave Device to Master Device



(9) Transfer start

Serial transfer is started by setting transfer data to the 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 $\overline{SCK0}$ 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 open-drain output must be set to high-impedance at the time of data reception, write FFH to SIO0 in advance. However, when the wake-up function specification bit (WUP) = 1, the N-ch open-drain output will always be set to high-impedance. 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 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 $\overline{RESET}$ 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) Distinction method of slave busy state

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 ($\overline{ACK}$) 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) Cautions on SBI mode

(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 request (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) In the SBI mode, the $\overline{\text{BUSY}}$ signal is output until the falling of the next serial clock after the $\overline{\text{BUSY}}$ release indication. If WUP = 1 is set by mistake during this period, $\overline{\text{BUSY}}$ will not be released. Thus, after releasing $\overline{\text{BUSY}}$, be sure to check that the SB0 (SB1) has become high level before setting WUP = 1.

(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 $\overline{\text{RESET}}$ 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.

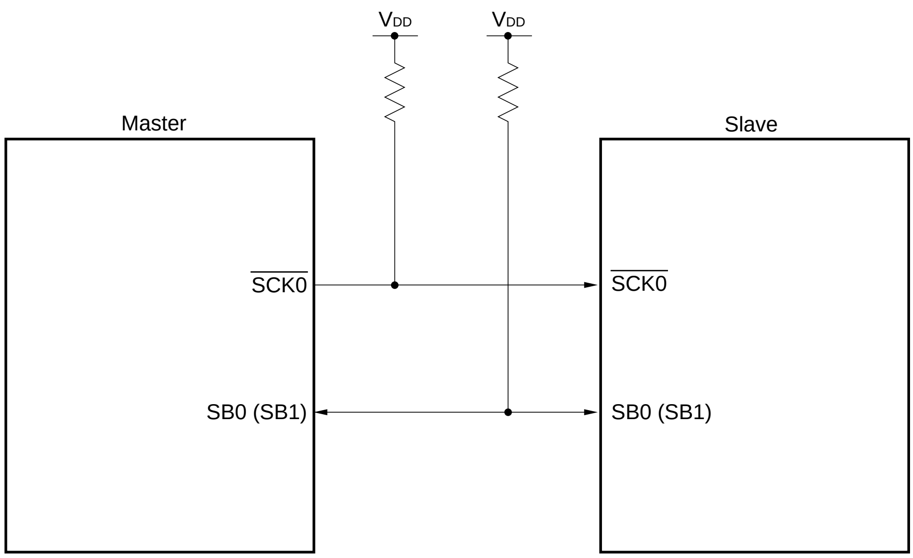
★ (e) The bus release signal or the command signal is acknowledged when the $\overline{\text{SCK0}}$ line is in high level, and the SB0 (SB1) line changes from low level to high level or from high level to low level. Thus, if the timing at which bus changes deviates due to effects such as board capacity, it may be determined as the bus release signal (or the command signal) even if data is sent. Therefore perform wiring carefully.

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 15-32. Example of Serial Bus Configuration with 2-Wire Serial I/O



(1) Register setting

The 2-wire serial I/O mode is set with the serial operating mode register 0 (CSIM0), the serial bus interface control register (SBIC) and the interrupt timing specification 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 sets CSIM0 to 00H.

Symbol	<7>	<6>	<5>	4	3	2	1	0	Address	When Reset	R/W
CSIM0	CSIE0	COI	WUP	CSIM04	CSIM03	CSIM02	CSIM01	CSIM00	FF60H	00H	R/W Note 1

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)

R/W

<table>
<tr><th>CSIM 04</th><th>CSIM 03</th><th>CSIM 02</th><th>PM25</th><th>P25</th><th>PM26</th><th>P26</th><th>PM27</th><th>P27</th><th>Operating Mode</th><th>Start Bit</th><th>SI0/SB0/P25 Pin Function</th><th>SO0/SB1/P26 Pin Function</th><th>$\overline{\text{SCK0}}$/P27 Pin Function</th></tr>
<tr><td>0</td><td>×</td><td colspan="12">3-wire serial I/O mode (Refer to 15.4.2 3-wire serial I/O mode operation)</td></tr>
<tr><td>1</td><td>0</td><td colspan="12">SBI mode (Refer to 15.4.3 SBI mode operation)</td></tr>
<tr><td rowspan="2">1</td><td rowspan="2">1</td><td>0</td><td>× Note 2</td><td>× Note 2</td><td>0</td><td>0</td><td>0</td><td>1</td><td rowspan="2">2-wire serial I/O mode</td><td rowspan="2">MSB</td><td>P25 (CMOS input/output)</td><td>SB1 (N-ch open-drain input/output)</td><td rowspan="2">$\overline{\text{SCK0}}$ (N-ch open-drain input/output)</td></tr>
<tr><td>1</td><td>0</td><td>0</td><td>× Note 2</td><td>× Note 2</td><td>0</td><td>1</td><td>SB0 (N-ch open-drain input/output)</td><td>P26 (CMOS input/output)</td></tr>
</table>

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 (SVA) data whenthe SBI mode is used

R

COI	Slave Address Comparison Result Flag Note 4
0	Slave address register (SVA) not equal to serial I/O shift register 0 (SIO0) data
1	Slave address register (SVA) equal to serial I/O shift register 0 (SIO0) data

R/W

CSIE0	Serial Interface Channel 0 Operation Control
0	Operation stopped
1	Operation enable

Notes
1. Bit 6 (COI) is a Read-Only bit.
2. Can be used freely as port function.
3. When 2-wire serial I/O mode is used, be sure to set WUP to 0.
4. When CSIE0 = 0, COI becomes 0.

Remark × : don't care
PMxx: Port mode register
Pxx : Output latch of port

(b) Serial bus interface control register (SBIC)

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

$\overline{\text{RESET}}$ input sets SBIC to 00H.

Symbol	<7>	<6>	<5>	<4>	<3>	<2>	<1>	<0>	Address	When Reset	R/W
SBIC	BSYE	ACKD	ACKE	ACKT	CMDD	RELD	CMDT	RELT	FF61H	00H	R/W

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.

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.

CSIE0: Bit 7 of the serial operating mode register 0 (CSIM0)

(c) Interrupt timing specification register (SINT)

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

$\overline{\text{RESET}}$ input sets SINT to 00H.

Symbol	7	<6>	<5>	<4>	3	2	1	0	Address	When Reset	R/W
SINT	0	CLD	SIC	SVAM	0	0	0	0	FF63H	00H	R/W[Note 1]

R/W

SIC	INTCSI0 Interrupt Factor Selection
0	CSIIF0 is set (1) upon termination of serial channel 0 transfer
1	CSIIF0 is set (1) upon bus release detection

R

CLD	$\overline{\text{SCK0}}$/P27 Pin Level[Note 2]
0	Low Level
1	High Level

Notes **1.** Bit 6 (CLD) is a Read-Only bit.

2. When CSIE0 = 0, CLD becomes 0.

Caution Be sure to set bits 0 to 3 to 0.

Remark CSIIF0: Interrupt request flag supports the INTCSI0

CSIE0 : Bit 7 of the 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. Data transmission/reception is carried out bit-wise in synchronization with the serial clock.

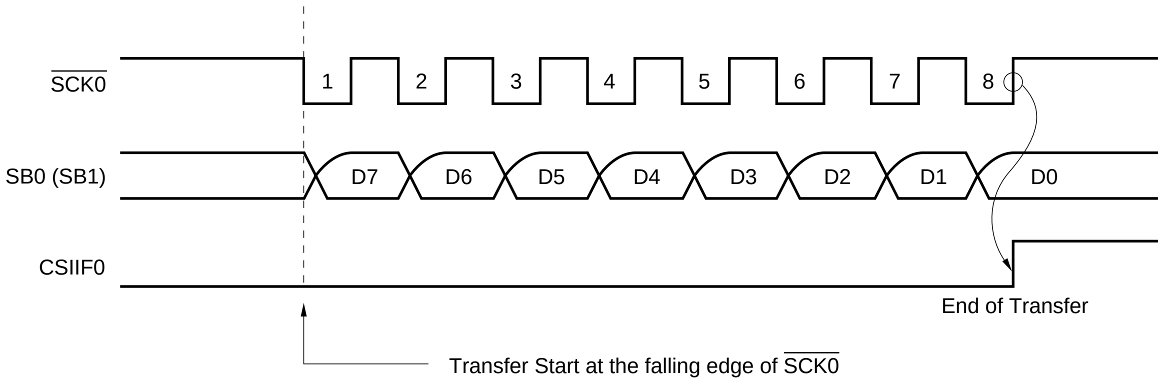
Shift operation of the serial I/O shift register 0 (SIO0) is carried out in synchronization with the falling edge of the serial clock ($\overline{SCK0}$).

The transmit data is held in the SO0 latch and is output from the SB0/P25 (or SB1/P26) pin with MSB set at start.

The receive data input from the SB0 (or SB1) pin is latched into the SIO0 at the rising edge of $\overline{SCK0}$.

Upon termination of 8-bit transfer, the SIO0 operation stops automatically and the interrupt request flag (CSIIF0) is set.

Figure 15-33. 2-Wire Serial I/O Mode Timings



The SB0 (or SB1) pin specified for the serial data bus serves for N-ch open-drain input/output and thus it must be externally pulled up. Because it is necessary to be set to high-impedance the N-ch open-drain output for data reception, write FFH to SIO0 in advance.

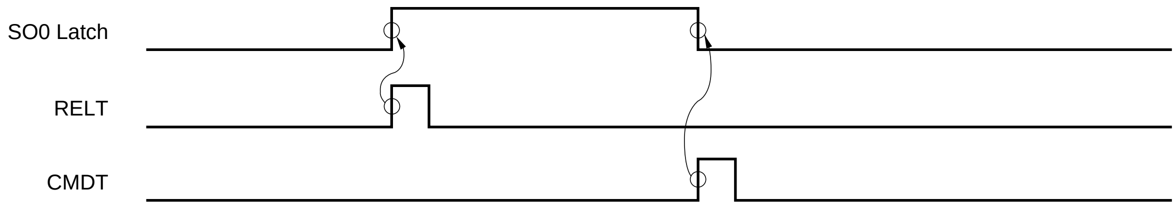
The SB0 (or SB1) pin generates the SO0 latch status and thus the SB0 (or SB1) pin output status can be manipulated by setting bit 0 (RELT) or bit 1 (CMDT) of the serial bus interface control register (SBIC). However, do not carry out this manipulation during serial transfer.

Control the $\overline{SCK0}$ pin output level in the output mode (internal system clock mode) by manipulating the P27 output latch (refer to **15.4.5 $\overline{SCK0}$/P27 pin output manipulation**).

(3) Various signals

Figure 15-34 shows RELT and CMDT operations.

Figure 15-34. RELT and CMDT Operations



(4) Transfer start

Serial transfer is started by setting transfer data to the 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 open-drain output must be set to high-impedance 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 (SBI) 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, the COI bit (match signal coming from the address comparator) of the serial operating mode register 0 (CSIM0) is tested. If it is “1”, normal transmission is judged to have been carried out. If it is “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.

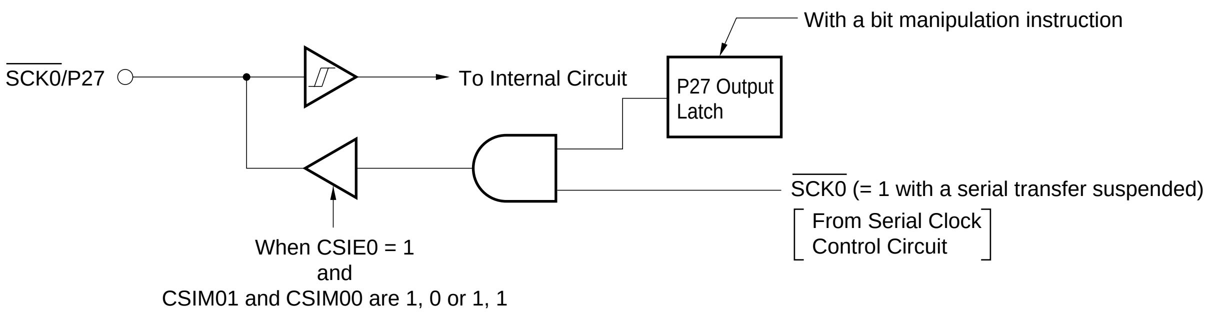
P27 output latch manipulation enables any value of $\overline{\text{SCK0}}$ to be set by software (SI0/SB0 and SO0/SB1 pin to be controlled with bit 0 (RELD) or bit 1 (CMDT) of the serial bus interface control register (SBIC)).

The $\overline{\text{SCK0}}$/P27 pin output manipulating procedure is described below.

<1> Set the serial operating mode register 0 (CSIM0) ($\overline{\text{SCK0}}$ pin enabled for serial operation in the output mode). $\overline{\text{SCK0}}$ = 1 with serial transfer suspended.

<2> Manipulate the P27 output latch with a bit manipulation instruction.

Figure 15-35. $\overline{\text{SCK0}}$/P27 Pin Configuration



[MEMO]

CHAPTER 16 SERIAL INTERFACE CHANNEL 0 (μPD78014Y Subseries)

The μPD78014Y Subseries incorporates two channels of clock synchronous serial interfaces.

Differences between channels 0 and 1 are as follows (Refer to **CHAPTER 17 SERIAL INTERFACE CHANNEL 1** for details of the serial interface channel 1) .

Table 16-1. Differences between Channels 0 and 1

Serial Transfer Mode		Channel 0	Channel 1
3-wire serial I/O	Clock selection	$f_X/2^2$**Note**, $f_X/2^3$, $f_X/2^4$, $f_X/2^5$, $f_X/2^6$, $f_X/2^7$, $f_X/2^8$, $f_X/2^9$ external clock, TO2 output clock	
	Transfer method	MSB/LSB switchable as the start bit	MSB/LSB switchable as the start bit Automatic transmit/receive function
	Transfer end flag	Serial interface channel transfer end interrupt request flag (CSIIF0)	Serial interface channel transfer end interrupt request flag (CSIIF1 and TRF)
SBI (serial bus interface)		Use possible	None
2-wire serial I/O			
I^2C bus (Inter IC Bus)			

Note Can be set only when the main system clock oscillates at 4.19 MHz or less.

Differences of serial interface channel 0 are shown in Table 16-2.

Table 16-2. Difference of Serial Interface Channel 0 Mode

Operation mode	Used pin	Features	Usage
3-wire serial I/O mode	$\overline{SCK0}$, SO0 or SI0	• Input and output lines are independent and they can transfer/receive at the same time, so the data transfer processing time is fast. • NEC single-chip microcontrollers provided as before.	Serial interface as is the case with the 75X/XL, 78K and 17K series.
SBI mode	$\overline{SCK0}$, SB0 or SB1	• Enables configuration of serial bus with two signal lines, thus, even when connected to some microcontrollers, the number of ports can be cut and wiring and routing on a board can be reduced. • High-speed serial interface compliant with the NEC standard bus format. • Address and command information onto the serial bus	
2-wire serial mode	$\overline{SCK0}$, SB0 or SB1	• Enables configuration of serial bus with two signal lines, thus, even when connected to some microcontrollers, the number of ports can be cut and wiring and routing on a board can be reduced. • Supports any data transfer format by program.	
I^2C	SCL, SDA0 or SDA1	• Supports I^2C bus format	Application sets using I^2C bus (such as TV, VCR, and audio products)

16.1 Serial Interface Channel 0 Functions

Serial interface channel 0 employs the following five modes.

- Operation stop mode
- 3-wire serial I/O mode
- SBI (Serial bus interface) mode
- 2-wire serial I/O mode
- I^2C (Inter IC) bus mode

★ **Caution Do not switch the operation mode (3-wire serial I/O/2-wire serial I/O/SBI/I^2C bus) during the serial interface channel 0 operation enable. The operation mode should be switched after stopping the serial operation.**

(1) Operation stop mode
Operation stop mode is used when serial transfer is not performed, thus reducing power dissipation.

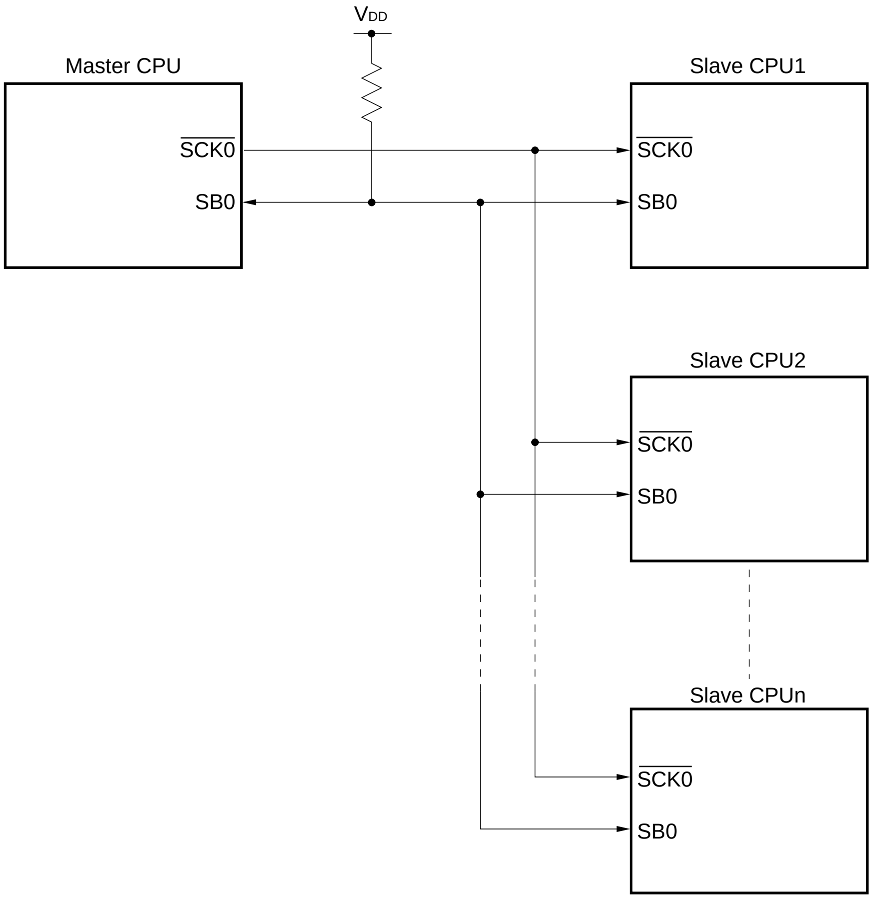
(2) 3-wire serial I/O mode (MSB/LSB-first switchable)
3-wire serial I/O mode transfers 8-bit data with three lines; serial clock ($\overline{SCK0}$), serial output (SO0), and serial input (SI0).
3-wire serial I/O mode can transfer/receive at the same time, so the data transfer processing time is fast.
The start bit of 8-bit data to undergo serial transfer is switchable between MSB and LSB, so it is possible to connect devices of any start bit.
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.

(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 ($\overline{SCK0}$) and serial data bus (SB0 or SB1) (see **Figure 16-1**).
The SBI mode complies with the NEC serial bus format and distinguishes the transfer data into "address", "command", and "data" to transmit or receive the data.

- Address : Data to select objective devices in serial communication
- Command : Data to instruct objective devices
- Data : Data to be actually transferred

In the actual transfer, the master device first outputs the "address" to the serial bus, and selects the slave device as communication target from among two or more devices. The serial transfer is then performed by transmitting and receiving "command" and "data" between the master and slave devices. The receiver automatically distinguishes the received data as "address", "command", or "data", by hardware.
This function enables to use input/output ports effectively and to simplify a serial interface controller of application programs.
In addition, the wake-up function for handshake, and the acknowledge signal and busy signal output function can be used.

Figure 16-1. Serial Bus Interface (SBI) System Configuration Example

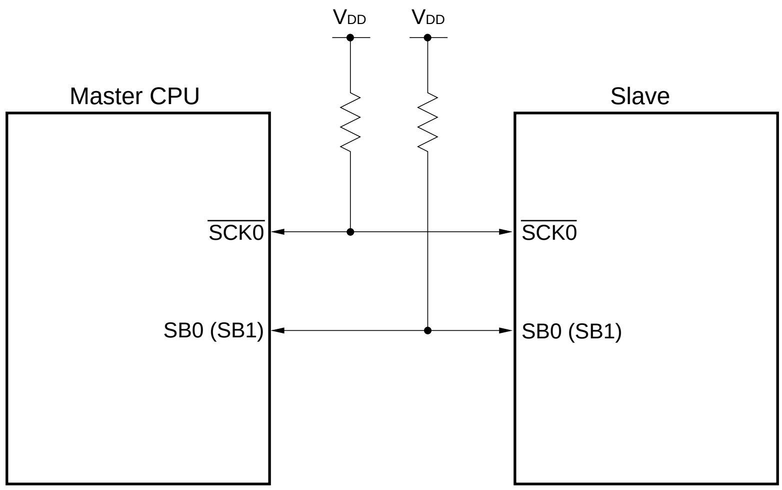


(4) 2-wire serial I/O mode (MSB-first)

This mode is used for 8-bit data transfer using two lines of the serial clock ($\overline{SCK0}$) and serial data bus (SB0 or SB1).

This mode supports any one of the possible data transfer formats by controlling the $\overline{SCK0}$ level and the SB0 or SB1 output level. Thus, the handshake line previously necessary for connecting two or more devices can be removed, resulting in an increased number of available input/output ports.

Figure 16-2. Serial Bus Configuration Example with 2-Wire Serial I/O

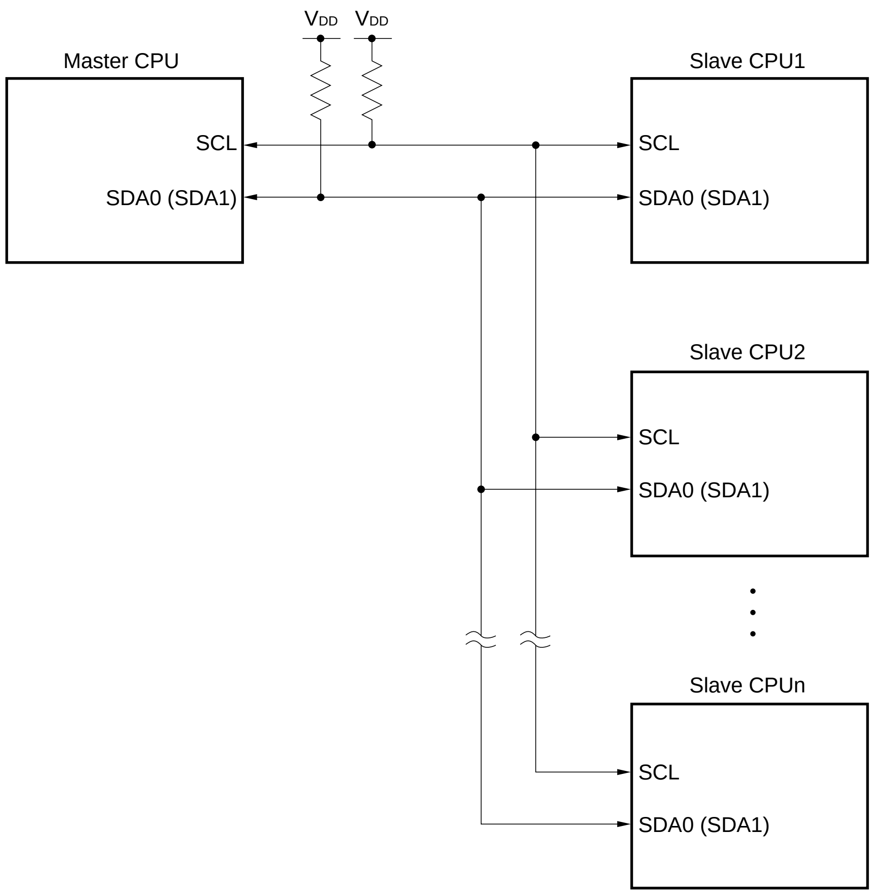


(5) I^2C bus mode (MSB first)

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 complies with the NEC I^2C bus format. In this mode, the transmitter outputs three kinds of data onto the serial data bus "start condition", "data", and "stop condition". The receiver automatically detects the received data by hardware.

Figure 16-3. Serial Bus Configuration Example Using I^2C Bus



16.2 Serial Interface Channel 0 Configuration

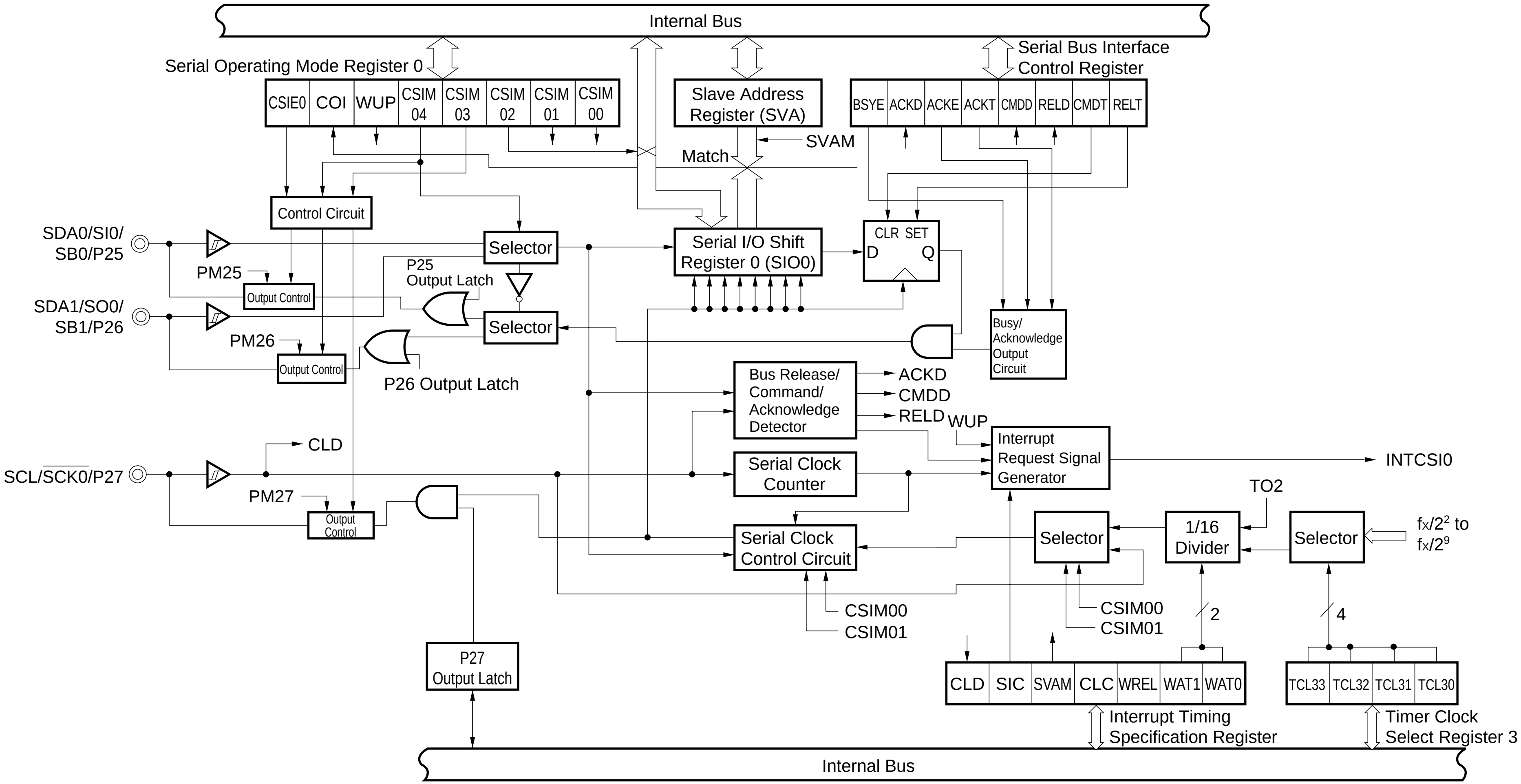
Serial interface channel 0 consists of the following hardware.

Table 16-3. 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 6-8 P20, P21, P23 to P26 Block Diagrams (μPD78074Y Subseries)** and **Figure 6-9 P22 and P27 Block diagrams (μPD78074Y Subseries)**.

Figure 16-4. Serial Interface Channel 0 Block Diagram



Remark Output Control performs selection between CMOS output and N-ch open-drain output.

(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.

The SBI mode, 2-wire serial I/O mode, and I^2C bus mode bus configurations enable the pin to 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.

Caution In the I^2C bus mode, do not write data to SIO0 during WUP (bit 5 of serial operation mode register 0 (CSIM0)) = 1. When wake-up function is used, data reception is available without writing data to SIO0. For details about wake-up function, refer to 16.4.5 (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 does not be 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.

Address comparison can also be executed on the data of LSB-masked high-order 7 bits when bit 4 (SVAM) of the interrupt timing specification register (SINT) is 1.

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 or the I^2C bus mode, when bit 5 (WUP) of CSIM0 is 1, the wake-up function can be used. In this case, the interrupt request signal (INTCSI0) is generated only if the slave address output from the master device matches the SVA value. With this interrupt request, the slave device acknowledges that a communication request is sent from the master device. When bit 5 (SIC) of the interrupt timing specification register (SINT) has been set to 1, the wake-up function is not available even if WUP IS 1.

(The interrupt request signal is generated at the bus release in the SBI mode, and at the stop condition in the I^2C mode. The SIC must be cleared to 0 while in use of the wake-up function.

Further, when SVA transmits data as the master or slave device in the SBI mode, 2-wire serial I/O mode, or I^2C bus mode, SVA can be used to detect errors.

$\overline{\text{RESET}}$ 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. 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 control circuit

This circuit controls serial clock supply to the 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 the interrupt request signal by setting bits 0, 1 (WAT0, WAT1) of the interrupt timing specify register (SINT) and bit 5 (WUP) of the serial operation mode register 0 (CSIM0) as shown in Table 16-4.

(7) Busy/acknowledge output circuit and bus release/command/acknowledge detector

These two circuits output and detect various control signals when the SBI mode or I^2C bus mode is used. These do not operate in the 3-wire serial I/O mode and 2-wire serial I/O mode.

Table 16-4. Serial Interface Channel 0 Interrupt Request Signal Generation

Serial Transfer Mode	WUP	WAT1	WAT0	ACKE	Description
3-wire or 2-wire serial I/O mode	0	0	0	0	An interrupt request signal is generated each time 8 serial clocks are counted.
	Other than above				Setting prohibited
SBI mode	0	0	0	0/1	An interrupt request signal is generated each time 8 serial clocks are counted (8-clock wait).
	1				After address is received, if the values of the serial I/O shift register 0 (SIO0) and the slave address register (SVA) match, an interrupt request signal is generated.
	Other than above				Setting prohibited
I^{2}C bus mode (transmit)	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 WAT1, 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^{2}C bus mode (receive)	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 is generated.
		1	1	0/1	An interrupt request signal is generated each time 9 serial clocks are counted (9-clock wait). To automati cally generate ACK information, preset ACKE to 1 (enable) before transfer start. However, in the case of the master, set ACKE to 0 (disable) before receiving the last data.
	1	1	1	1	After address is received, if the values of the serial I/O shift register 0 (SIO0) and the slave address register (SVA) match, an interrupt request signal is generated. To automatically generate ACK information, preset ACKE to 1 (enable) before transfer start.
	Other than above				Setting prohibited

Remark 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 specification register (SINT)

(1) Timer clock select register 3 (TCL3) (See **Figure 16-5**)
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.

Remark TCL3 has functions to set the serial clock of serial interface channel 1 except to set the serial clock of serial interface channel 0.

(2) Serial operating mode register 0 (CSIM0) (See **Figure 16-6**)
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 sets CSIM0 to 00H.

★ **Caution Do not switch the operation mode (3-wire serial I/O/2-wire serial I/O/SBI I^{2}C bus) during the serial interface channel 0 operation enable. The operation mode should be switched after stopping the serial operation.**

(3) Serial bus interface control register (SBIC) (See **Figure 16-7**)
This register sets the serial bus interface operation and displays the status.
SBIC is set with a 1-bit or 8-bit memory manipulation instruction.
$\overline{\text{RESET}}$ input sets SBIC to 00H.

(4) Interrupt timing specify register (SINT) (See **Figure 16-8**)
This register sets interrupt, wait, clock level control, address mask function and displays the level status of $\overline{\text{SCK0}}$/SCL/P27 pin.
SINT is set with 1-bit or 8-bit memory manipulation instruction.
$\overline{\text{RESET}}$ input sets SINT to 00H.

Figure 16-5. Timer Clock Select Register 3 Format

Symbol	7	6	5	4	3	2	1	0	Address	When Reset	R/W
TCL3	TCL37	TCL36	TCL35	TCL34	TCL33	TCL32	TCL31	TCL30	FF43H	88H	R/W

TCL33	TCL32	TCL31	TCL30	Serial Interface Channel 0 Serial Clock Seletion	
				Serial Clock in I²C bus mode	Serial Clock in 3-wire/SBI/2-wire mode
0	1	1	0	$f_X/2^6$ (156 kHz)	$f_X/2^2$ Note
0	1	1	1	$f_X/2^7$ (78.1 kHz)	$f_X/2^3$ (1.25 MHz)
1	0	0	0	$f_X/2^8$ (39.1 kHz)	$f_X/2^4$ (625 kHz)
1	0	0	1	$f_X/2^9$ (19.5 kHz)	$f_X/2^5$ (313 kHz)
1	0	1	0	$f_X/2^{10}$ (9.8 kHz)	$f_X/2^6$ (156 kHz)
1	0	1	1	$f_X/2^{11}$ (4.9 kHz)	$f_X/2^7$ (78.1 kHz)
1	1	0	0	$f_X/2^{12}$ (2.4 kHz)	$f_X/2^8$ (39.1 kHz)
1	1	0	1	$f_X/2^{13}$ (1.2 kHz)	$f_X/2^9$ (19.5 kHz)
Other than above				Setting prohibited	

TCL37	TCL36	TCL35	TCL34	Serial Interface Channel 1 Serial Clock Seletion
0	1	1	0	$f_X/2^2$ Note
0	1	1	1	$f_X/2^3$ (1.25 MHz)
1	0	0	0	$f_X/2^4$ (625 kHz)
1	0	0	1	$f_X/2^5$ (313 kHz)
1	0	1	0	$f_X/2^6$ (156 kHz)
1	0	1	1	$f_X/2^7$ (78.1 kHz)
1	1	0	0	$f_X/2^8$ (39.1 kHz)
1	1	0	1	$f_X/2^9$ (19.5 kHz)
Other than above				Setting prohibited

Note Can be set only when the main system clock oscillate at 4.19 MHz or less.

Caution If TCL3 is to be rewritten in data other than identical data, the serial transfer must be stopped first.

Remarks 1. f_X: Main system clock oscillation frequency

2. Value in parentheses apply to operation with f_X = 10.0 MHz.

Figure 16-6. Serial Operating Mode Register 0 Format (1/2)

Symbol	<7>	<6>	<5>	4	3	2	1	0	Address	When Reset	R/W
CSIM0	CSIE0	COI	WUP	CSIM04	CSIM03	CSIM02	CSIM01	CSIM00	FF60H	00H	R/W Note 1

R/W

CSIM01	CSIM00	Serial Interface Channel 0 Clock Selection
0	×	Input clock to $\overline{SCK0}$ pin from off-chip
1	0	8-bit timer register 2 (TM2) output Note 2
1	1	Clock specified with bits 0 to 3 of timer clock select register 3 (TCL3)

R/W

CSIM 04	CSIM 03	CSIM 02	PM25	P25	PM26	P26	PM27	P27	Operating Mode	Start Bit	SI0/SB0/SDA0/ P25 Pin Function	SO0/SBI/SDA1/ P26 Pin Function	$\overline{SCK0}$/SCL/P27 Pin Function
0	×	0	1	×	0	0	0	1	3-wire serial I/O mode	MSB	SI0 Note 3 (Input)	SO0 (CMOS output)	$\overline{SCK0}$ (CMOS input/output)
		1								LSB			
1	0	0	Note 4 ×	Note 4 ×	0	0	0	1	SBI mode	MSB	P25 (CMOS input/output)	SB1 (N-ch open-drain input/output)	$\overline{SCK0}$ (CMOS input/output)
		1	0	0	Note 4 ×	Note 4 ×	0	1			SB0 (N-ch open-drain input/output)	P26 (CMOS input/ output)	
1	1	0	Note 4 ×	Note 4 ×	0	0	0	1	2-wire serial I/O mode or I^2C Bus Mode	MSB	P25 (CMOS input/output)	SB1/SDA1 (N-ch open-drain input/output)	$\overline{SCK0}$/SCL (N-ch open-drain input/ output)
		1	0	0	Note 4 ×	Note 4 ×	0	1			SB0/SDA0 (N-ch open-drain input/output)	P26 (CMOS input/output)	

R/W

WUP	Wake-up Function Control Note 5
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 in the SBI mode, CMDD = 1 in the I^2C bus mode) matches the slave address register (SVA) data when SBI mode or I^2C bus mode is used.

Notes
1. Bit 6 (COI) is a Read-Only bit.
2. When the I^2C mode is used, the clock gets to 1/16 clock frequency which TO2 outputs.
3. Can be used as P25 (CMOS input) when used only for transmission.
4. Can be used freely as port function.
5. When the wake-up function is used (WUP = 1), set bit 5 of the interrupt timing specification register (SINT) to 0.
   Do not write data to serial I/O shift register 0 (SIO0) during WUP = 1.

Remark × : Don't care
PM××: Port mode register
P×× : Output latch of port

Figure 16-6. Serial Operating Mode Register 0 Format (2/2)

R

COI	Slave Address Comparison Result Flag[Note]
0	Slave address register (SVA) not equal to serial I/O shift register 0 (SIO0) data
1	Slave address register (SVA) equal to serial I/O shift register 0 (SIO0) data

R/W

CSIE0	Serial Interface Channel 0 Operation Control
0	Operation stopped
1	Operation enabled

Note When CSIE0 = 0, COI becomes 0.

Figure 16-7. Serial Bus Interface Control Register Format (1/2)

Symbol	<7>	<6>	<5>	<4>	<3>	<2>	<1>	<0>	Address	When Reset	R/W
SBIC	BSYE	ACKD	ACKE	ACKT	CMDD	RELD	CMDT	RELT	FF61H	00H	R/W[Note]

R/W	RELT	
		Use for bus release signal output when the SBI mode is used. Use for stop condition output when the I^2C bus mode is used. When RELT = 1, SO latch is set to 1. After SO latch setting, automatically cleared to 0. Also cleared to 0 when CSIE0 = 0.

R/W	CMDT	
		Use for command signal output when the SBI mode is used. Use for start condition output in the I^2C bus mode. 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

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 in the SBI mode • When stop condition is detected in the I^2C bus mode

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 stop condition is detected in the I^2C bus mode • When CSIE0 = 0 • When $\overline{\text{RESET}}$ input is applied	• When command signal (CMD) is detected in the SBI mode • When stop condition is detected in the I^2C mode

R/W	ACKT	
		When the SBI mode is used, acknowledge signal is output in synchronization with the falling edge of $\overline{\text{SCK0}}$ clock immediately after execution of the instruction to be set to 1, and after acknowledge signal output, automatically cleared to 0. Used as ACKE = 0. Also cleared to 0 upon start of serial interface transfer or when CSIE0 = 0. When the I^2C bus mode is used, SDA0 (SDA1) is made low-level until the next SCL falling edge immediately after executionof the set instruction (ACKT = 1). Used to generate ACK signal by software when 8-clock wait is selected. Cleared to (0) upon start of serial interface transfer or when CSIE = 0.

Note Bits 2, 3, and 6 (RELD, CMDD and ACKD) are read-only bits.

Remarks 1. Zeros will be returned form bits 0, 1, and 4 (or RELT, CMDT, ACKT, respectively) if users read these bits after data setting is completed.

2. CSIE0: Bit 7 of the serial operating mode register 0 (CSIM0)

Figure 16-7. Serial Bus Interface Control Register Format (2/2)

R/W

<table>
<tr><td>ACKE</td><td colspan="2">Acknowledge Signal Automatic Output Control (in SBI mode)</td></tr>
<tr><td>0</td><td colspan="2">Acknowledge signal automatic output disable (output with ACKT enable)</td></tr>
<tr><td rowspan="2">1</td><td>Before completion of transfer</td><td>Acknowledge signal is output in synchronization with the 9th clock falling edge of $\overline{SCK0}$ (automatically output when ACKE = 1).</td></tr>
<tr><td>After completion of transfer</td><td>Acknowledge signal is output in synchronization with the falling edge of $\overline{SCK0}$ clock immediately 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.</td></tr>
</table>

R/W

ACKE	Acknowledge Signal Automatic Output Control[Note 1] (in the I^2C bus mode)
0	Disables acknowledge signal automatic output. (However, output with ACKT possible) Use for reception when 8-clock wait mode is selected or for transmission[Note 2].
1	Enables acknowledge signal automatic output. Outputs acknowledge signal in synchronization with the 9th clock falling edge of SCL (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

ACKD	Acknowledge Detection
Clear Conditions (ACKD = 0)	Set Conditions (ACKD = 1)
• At the falling edge of $\overline{SCK0}$ clock immediately after the busy mode has been released when a transfer start instruction is executed • Upon execution of transfer start instruction in the I^2C bus mode • When CSIE0 = 0 • When $\overline{RESET}$ input is applied	• When acknowledge signal is detected at the rising edge of $\overline{SCK0}$/SCL clock after completion of transfer

R/W

BSYE[Note 3]	Synchronizing Busy Signal Output Control
0	When the SBI mode is used, disables busy signal which is output in synchronization with the falling edge of SCK0 clock immediately after execution of the instruction to be cleared to 0. Be sure to set BSYE to 0 in the I^2C bus mode.
1	Outputs busy signal at the falling edge of $\overline{SCK0}$ clock following the acknowledge signal when the SBI mode is used.

Notes **1.** Setting should be performed before transfer.

2. If 8-clock wait mode is selected, the acknowledge signal at reception time must be output using ACKT.

★ **3.** The busy mode can be cancelled by start of serial interface.
However, the BSYE flag is not cleared to 0.

Remark CSIE0: Bit 7 of the serial operating mode register 0 (CSIM0)

Figure 16-8. Interrupt Timing Specification Register Format (1/2)

Symbol	7	<6>	<5>	<4>	<3>	<2>	1	0	Address	When Reset	R/W
SINT	0	CLD	SIC	SVAM	CLC	WREL	WAT1	WAT0	FF63H	00H	R/W[Note 1]

R/W

WAT1	WAT0	Wait and Interrupt Control
0	0	Generates interrupt service request at rising edge of 8th $\overline{SCK0}$ 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 output. In the case of slave device, makes SCL output low to request wait 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 output. In the case of slave device, makes SCL output low to request waits pulses are input.)

R/W

WREL	Wait State 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

CLC	Clock Level Control[Note 2]
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) Use to enable master device to generate start condition and stop condition signal.

Notes **1.** Bit 6 (CLD) is a Read-Only bit.

2. When not using the I^{2}C bus mode, set CLC to 0.

Figure 16-8. Interrupt Timing Specification Register Format (2/2)

R/W

SVAM	SVA Bit to be Used as Slave Address
0	Bits 0 to 7
1	Bits 1 to 7

R/W

SIC	INTCSI0 Interrupt Cause Selection[Note 1]
0	CSIIF0 is set to 1 upon termination of serial interface channel 0 transfer
1	CSIIF0 is set to 1 upon stop condition detection in the I²C bus mode or termination of serial interface in the SBI mode

R

CLD	$\overline{\text{SCK0}}$/SCL/P27 Pin Level[Note 2]
0	Low level
1	High level

Notes 1. When using wake-up function, set SIC to 0.

2. When CSIE0 = 0, CLD becomes 0.

Remark SVA : Slave address register

CSIIF0 : Interrupt request flag supports the INTCSI0

CSIE0 : Bit 7 of the serial operating mode register 0 (CSIM0)

16.4 Serial Interface Channel 0 Operations

The following four operating modes are available to the serial interface channel 0.

- Operation stop mode
- 3-wire serial I/O mode
- SBI mode
- 2-wire serial I/O mode
- I^2C (Inter IC) bus mode

16.4.1 Operation stop mode

Serial transfer is not carried out in the operation stop mode. Thus, power dissipation can be reduced. The serial I/O shift register 0 (SIO0) does not carry out shift operation either and thus it can be used as normal 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 normal input/output ports.

(1) Register setting

The operation stop mode is set with the serial operating mode register 0 (CSIM0).

CSIM0 is set with a 1-bit or 8-bit memory manipulation instruction.

$\overline{\text{RESET}}$ input sets CSIM0 to 00H.

Symbol	<7>	<6>	<5>	4	3	2	1	0	Address	When 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 enable

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 ($\overline{SCK0}$), serial output (SO0), and serial input (SI0).

(1) Register setting

The 3-wire serial I/O mode is set with the serial operating mode register 0 (CSIM0) and 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.

$\overline{RESET}$ input sets CSIM0 to 00H.

Symbol	<7>	<6>	<5>	4	3	2	1	0	Address	When Reset	R/W
CSIM0	CSIE0	COI	WUP	CSIM04	CSIM03	CSIM02	CSIM01	CSIM00	FF60H	00H	R/W Note 1

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)

R/W

CSIM04	CSIM03	CSIM02	PM25	P25	PM26	P26	PM27	P27	Operating 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	×	0	0	0	1	3-wire serial I/O mode	MSB	SI0 Note 2 (Input)	SO0 (CMOS output)	$\overline{\text{SCK0}}$ (CMOS input/output)
		1								LSB			
1	0	SBI mode (Refer to **16.4.3 SBI mode operation**)											
1	1	2-wire serial I/O mode (Refer to **16.4.4 2-wire serial I/O mode operation**) or I^{2}C bus mode (Refer to **16.4.5 I^{2}C bus mode operation**)											

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 in the SBI mode or CMDD = 1 in the I^{2}C bus mode) matches the slave address register data (SVA) when the SBI or I^{2}C bus mode is used.

R/W

CSIE0	Serial Interface Channel 0 Operation Control
0	Operation stopped
1	Operation enable

Notes **1.** Bit 6 (COI) is a Read-Only bit.
2. Can be used as P25 (CMOS input) when used only for transmission.
3. Be sure to set WUP to 0 when the 3-wire serial I/O mode is selected.

Remark × : don't care
PM×× : Port mode register
P×× : Output latch of port

(b) Serial bus interface control register (SBIC)
SBIC is set with a 1-bit or 8-bit memory manipulation instruction.
$\overline{\text{RESET}}$ input sets SBIC to 00H.

Symbol	<7>	<6>	<5>	<4>	<3>	<2>	<1>	<0>	Address	When Reset	R/W
SBIC	BSYE	ACKD	ACKE	ACKT	CMDD	RELD	CMDT	RELT	FF61H	00H	R/W

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.

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.

CSIE0: Bit 7 of the serial operating mode register 0 (CSIM0)

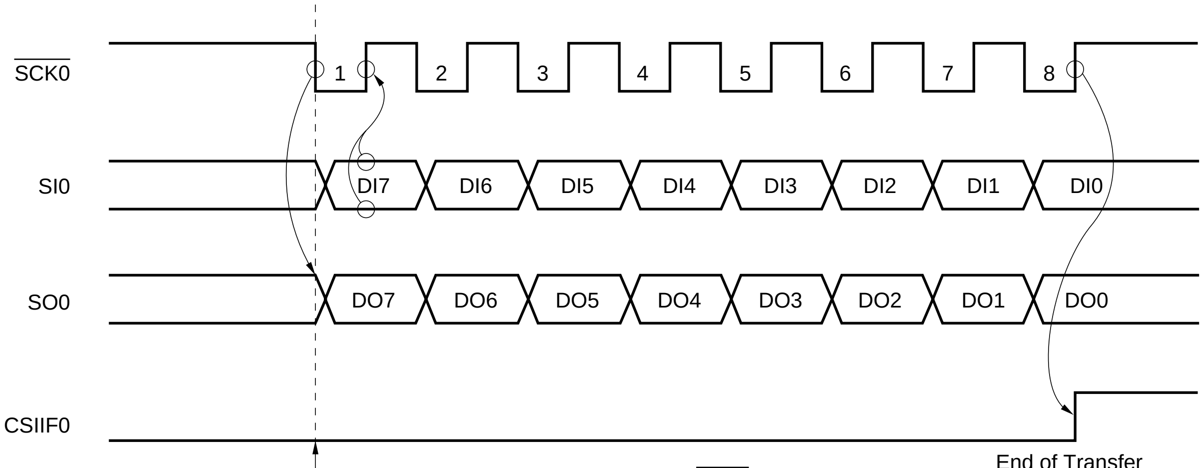
(2) Communication operation

The 3-wire serial I/O mode is used for data transmission/reception in 8-bit units. Data transmission/reception is carried out bit-wise in synchronization of the serial clock.

Shift operation of the serial I/O shift register 0 (SIO0) is carried out at the falling edge of the serial clock ($\overline{SCK0}$). 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 $\overline{SCK0}$.

Upon termination of 8-bit transfer, SIO0 operation stops automatically and the interrupt request flag (CSIIF0) is set.

Figure 16-9. 3-Wire Serial I/O Mode Timings



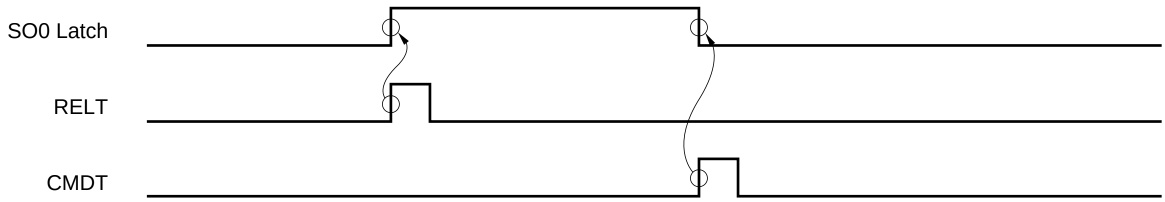
The SO0 pin serves for CMOS output and generates the SO0 latch status. Thus, the SO0 pin output status can be manipulated by setting bit 0 (RELT) and bit 1 (CMDT) of the serial bus interface control register (SBIC). However, do not carry out this manipulation during serial transfer.

Control the $\overline{SCK0}$ pin output level in the output mode (internal system clock mode) by manipulating the P27 output latch (refer to **16.4.8 $\overline{SCK0}$/SCL/P27 pin output manipulation**).

(3) Various signals

Figure 16-10 shows RELT and CMDT operations.

Figure 16-10. RELT and CMDT Operations



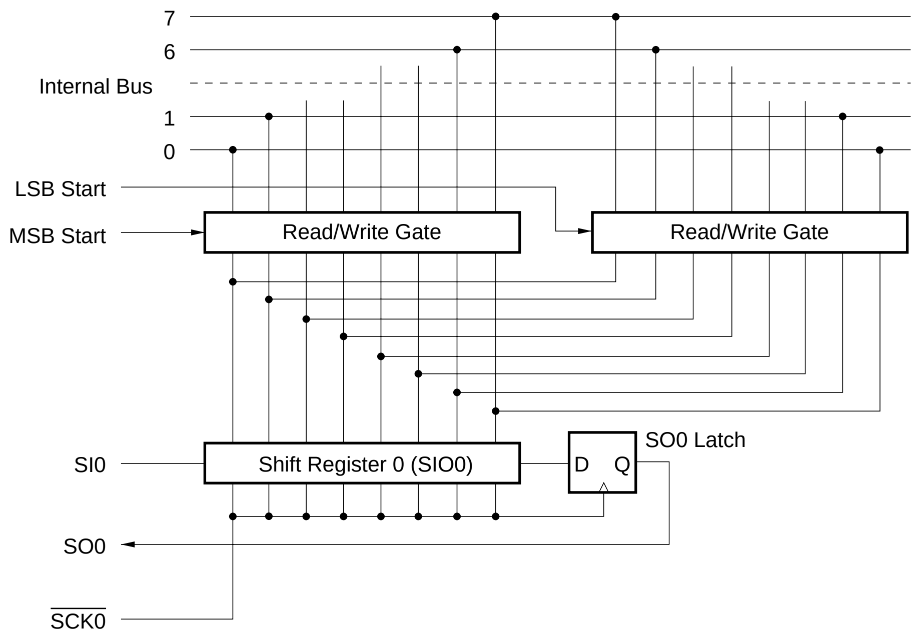
(4) MSB/LSB switching as the start bit

The 3-wire serial I/O mode enables to select transfer to start at MSB or LSB.

Figure 16-11 shows the configuration of the serial I/O shift register 0 (SIO0) and internal bus. As shown in the figure, MSB/LSB can be read/written in inverted form.

MSB/LSB switching as the start bit can be specified with bit 2 (CSIM02) of the serial operating mode register 0 (CSIM0).

Figure 16-11. 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, switch the MSB/LSB start bit before writing data to the shift register.

(5) Transfer start

Serial transfer is started by setting transfer data to the serial I/O shift register 0 (SIO0) when the following two conditions are satisfied.

- Serial interface 0 operation control bit (CSIE0) = 1
- Internal serial clock is stopped or $\overline{SCK0}$ is the 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 SBI mode operation

SBI (Serial Bus Interface) is a high-speed serial interface in compliance with the NEC serial bus format.

SBI has a format with the bus configuration function added to the clocked serial I/O method so that it can carry out communication with two or more devices with two signal conductors on the single-master high-speed serial bus. Thus, when making up a serial bus with two or more microcomputers and peripheral ICs, the number of ports to be used and the number of wires on the board can be decreased.

The master device can output to the serial data bus of the slave device “addresses” for selection of the serial communication target device, “commands” to instruct the target device and actual “data”. The slave device can identify the received data into “address”, “command” or “data”, by hardware. This function enables the application program to control serial interface channel 0 to be simplified.

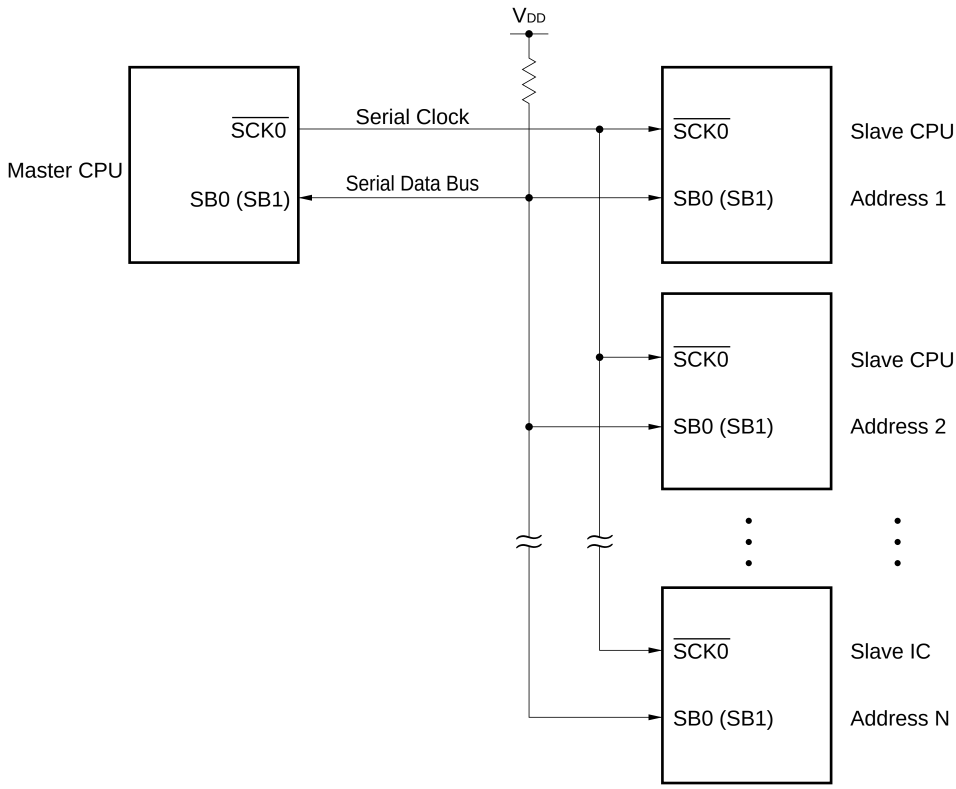
The SBI function is incorporated into various devices including 75X/XL Series devices and 78K Series.

Figure 16-12 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 pin serves for open-drain output and so the serial data bus line is in wired-OR state. A pull-up resistor is necessary for the serial data bus line.

Refer to **(11) Cautions on SBI mode (d)** described later when the SBI mode is used.

Figure 16-12. Example of Serial Bus Configuration with SBI



Caution **When replacing 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 method, when a serial bus is constructed by connecting two or more devices, many ports and wiring are necessary to distinguish chip select signals and command/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 constructed with two signal conductors of serial clock $\overline{\text{SCK0}}$ and serial data bus SB0 (SB1). Thus, SBI is effective to decrease the number of microcontroller ports and that of wiring and routing 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 serial communication.

(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.

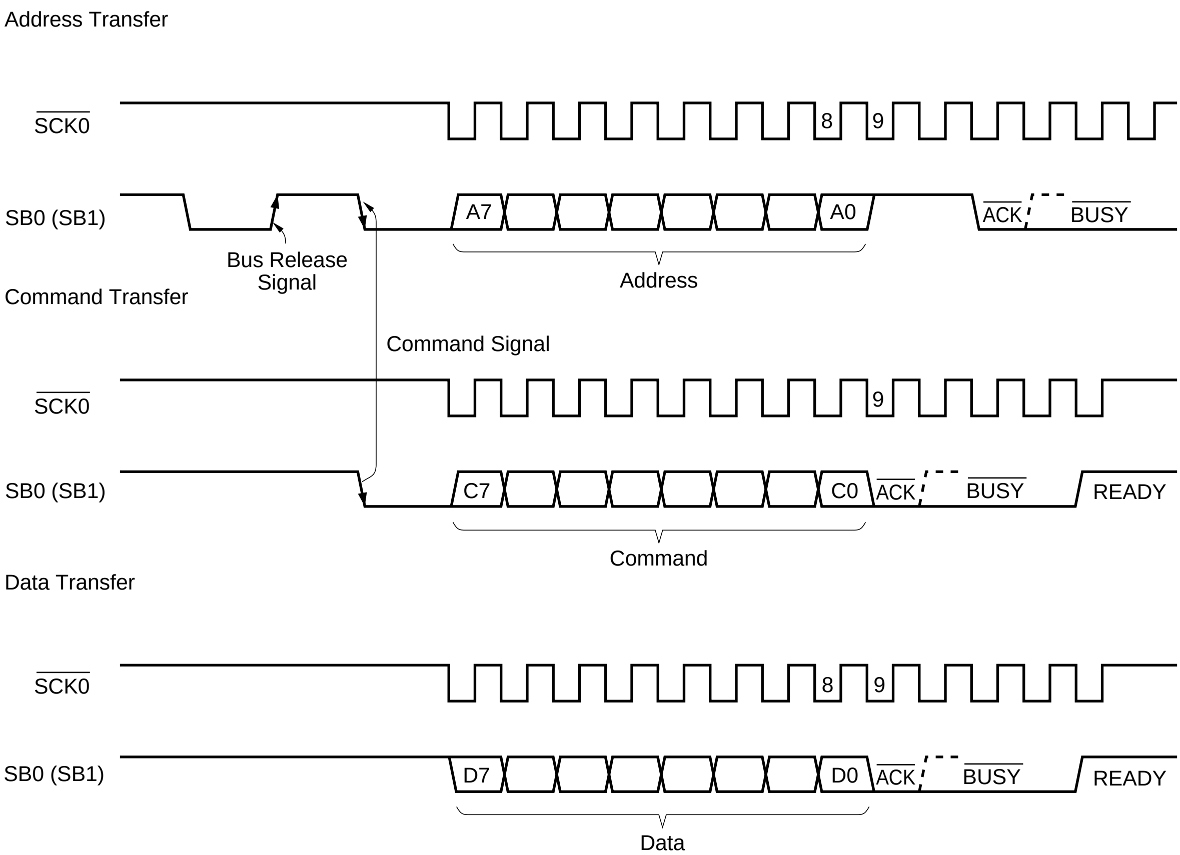
(2) SBI definition

The SBI serial data format is defined as follows.

Serial data to be transferred with SBI is distinguished into three types, “address”, “command” and “data”.

Figure 16-13 shows the address, command and data transfer timings.

Figure 16-13. SBI Transfer Timings



Remark The broken line indicates the READY state.

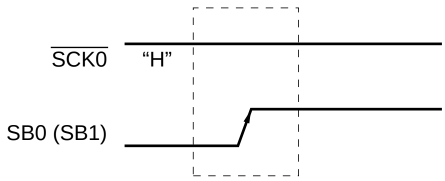
The bus release signal and the command signal are output by the master device. $\overline{\text{BUSY}}$ is output by the slave signal. $\overline{\text{ACK}}$ can be output by either the master or slave device (normally, the 8-bit data receiver outputs). Serial clocks continue to be output by the master device from 8-bit data transfer start to $\overline{\text{BUSY}}$ reset.

(a) Bus release signal (REL)

The bus release signal is generated when the $\overline{SCK0}$ line is in high level (a serial clock is not output) and the SB0 (SB1) line changes from low level to high level.

The bus release signal is output by the master.

Figure 16-14. Bus Release Signal



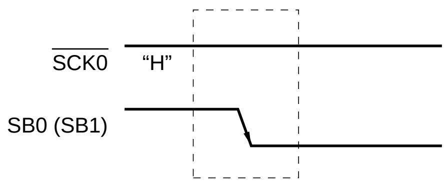
The bus release signal indicates the master will send the address to the slave. The slave contains hardware to detect the bus release signal.

★ **Caution The bus release signal is acknowledged when the $\overline{SCK0}$ line is in high level, and the SB0 (SB1) line changes from low level to high level. Thus, if the timing at which bus changes deviates due to effects such as board capacity, it may be determined as the bus release signal even if data is sent. Therefore perform wiring carefully.**

(b) Command Signal (CMD)

The command signal is generated when the $\overline{SCK0}$ line is in high level (a serial clock is not output) and the SB0 (SB1) line changes from high level to low level. The command signal is output by the master.

Figure 16-15. Command Signal



The command signal indicates that master will send the command to the slave (However, the command signal following the bus release signal indicates that address will be sent).

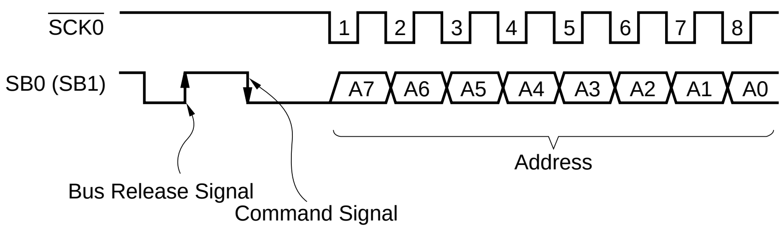
The slave contains hardware to detect the command signal.

★ **Caution The command signal is acknowledged when the $\overline{SCK0}$ line is in high level, and the SB0 (SB1) line changes from high level to low level. Thus, if the timing at which bus changes deviates due to effects such as board capacity, it may be determined as the command signal even if data is sent. Therefore perform wiring carefully.**

(c) Address

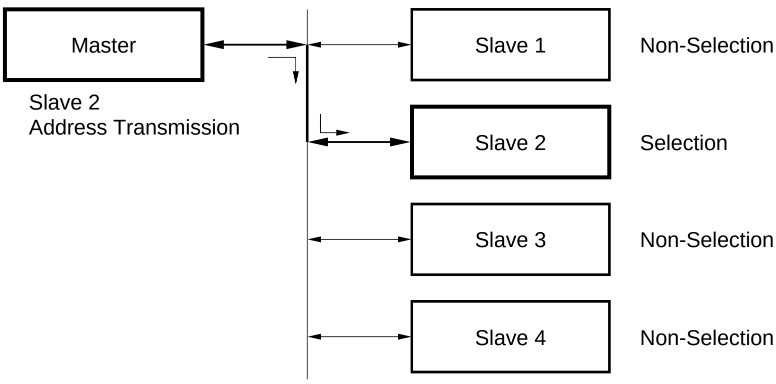
The address is 8-bit data that the master outputs to the slave connected to the bus line to select a specific slave.

Figure 16-16. Address



8-bit data following the bus release signal and the command signal is defined as the address. The slave detects the condition and checks by hardware if 8-bit data matches its specified number (the slave address). When 8-bit data matches the slave address, which means the slave is selected, the slave communicates with the master until the master instructs disconnection.

Figure 16-17. Slave Selection with Address



(d) Command and Data

The master sends commands and sends/receives data to the slave selected by sending the address.

Figure 16-18. Command

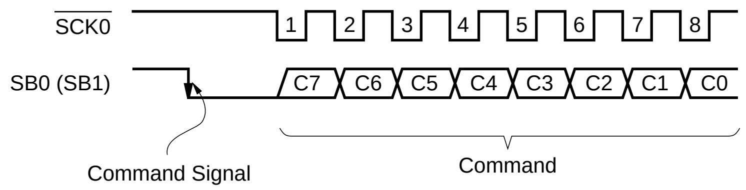
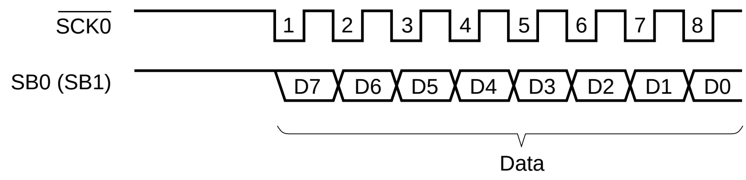


Figure 16-19. Data



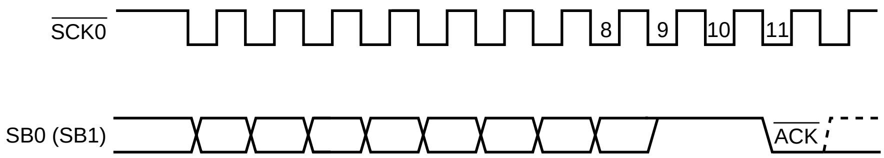
8-bit data following the command signal is defined as a command. 8-bit data without the command signal is defined as data. How to use the command and data can be determined depending on the communication specifications.

(e) Acknowledge signal ($\overline{\text{ACK}}$)

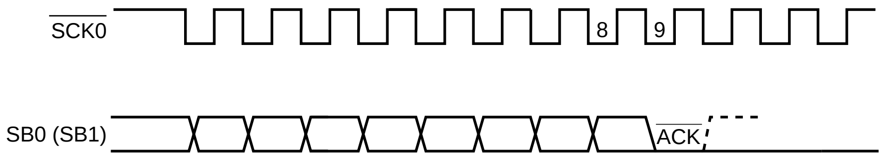
This signal is used between the sending side and receiving side devices for confirmation of correct serial data sending.

Figure 16-20. Acknowledge Signal

[When output synchronously with $\overline{\text{SCK0}}$ in 11th clock]



[When output synchronously with $\overline{\text{SCK0}}$ in 9th clock]



Remark The broken line indicates the READY state.

The acknowledge signal is a one-shot pulse synchronous with $\overline{\text{SCK0}}$ falling, whose position can be synchronized with $\overline{\text{SCK0}}$ in any clock.

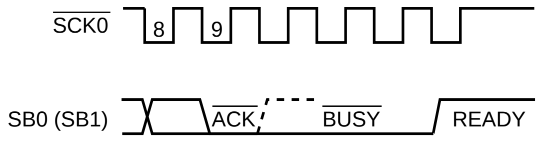
The sending side that has transferred 8-bit data checks if the acknowledge signal has been sent from the receiving side. If this signal is not sent back from the slave device for a given period after data sending, this means that the data sent has not been received correctly by the slave device.

(f) Busy signal (BUSY), Ready signal (READY)

The busy signal informs the master that the slave is busy transmitting/receiving data.

The ready signal informs the master that the slave is ready to transmit/receive data.

Figure 16-21. Busy Signal, Ready Signal



Remark The broken line indicates the ready state.

In the SBI mode, the slave informs the master of the busy state by setting SB0 (SB1) line to low level.
The busy signal is output following the acknowledge signal the slave outputs. The busy signal is set/cleared synchronously with the falling edge of SCK0. The master terminates automatically to output the serial clock SCK0 when the busy signal is cleared.
The master can start subsequent transmissions when the busy signal is cleared and changes to the ready state.

Caution In the SBI mode, the BUSY signal is output until the falling of the next serial clock after the BUSY release indication. If WUP = 1 is set by mistake during this period, BUSY will not be released. Thus, after releasing BUSY, be sure to check that the SB0 (SB1) has become high level before setting WUP = 1.

(3) Register setting

The SBI mode is set with the serial operating mode register 0 (CSIM0), the serial bus interface control register (SBIC), and the interrupt timing specification register (SINT).

(a) Serial operating mode register 0 (CSIM0)

CSIM0 is set with a 1-bit or 8-bit memory manipulation instruction.

RESET input sets CSIM0 to 00H.

Symbol	<7>	<6>	<5>	4	3	2	1	0	Address	When Reset	R/W
CSIM0	CSIE0	COI	WUP	CSIM04	CSIM03	CSIM02	CSIM01	CSIM00	FF60H	00H	R/W Note 1

R/W

CSIM01	CSIM00	Serial Interface Channel 0 Clock Selection
0	×	Input clock to $\overline{SCK0}$ 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 selection register 3 (TCL3)

R/W

<table>
<tr><th>CSIM 04</th><th>CSIM 03</th><th>CSIM 02</th><th>PM25</th><th>P25</th><th>PM26</th><th>P26</th><th>PM27</th><th>P27</th><th>Operating Mode</th><th>Start Bit</th><th>SI0/SB0/SDA0/ P25 Pin Function</th><th>SO0/SB0/SDA1/ P26 Pin Function</th><th>$\overline{SCK0}$/SCL/P27 Pin Function</th></tr>
<tr><td>0</td><td>×</td><td colspan="12">3-wire serial I/O mode (Refer to 16.4.2 3-wire serial I/O mode operation)</td></tr>
<tr><td rowspan="2">1</td><td rowspan="2">0</td><td>0</td><td>× Note 2</td><td>× Note 2</td><td>0</td><td>0</td><td>0</td><td>1</td><td rowspan="2">SBI mode</td><td rowspan="2">MSB</td><td>P25 (CMOS input/output)</td><td>SB1 (N-ch open-drain input/output)</td><td rowspan="2">$\overline{SCK0}$ (CMOS input/output)</td></tr>
<tr><td>1</td><td>0</td><td>0</td><td>× Note 2</td><td>× Note 2</td><td>0</td><td>1</td><td>SB0 (N-ch open-drain input/output)</td><td>P26 (CMOS input/ output)</td></tr>
<tr><td>1</td><td>1</td><td colspan="12">2-wire serial I/O mode (Refer to 16.4.4 2-wire serial I/O mode operation)
or I²C bus mode (Refer to 16.4.5 I²C bus mode operation)</td></tr>
</table>

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 (SVA) data when SBI mode is used

R

COI	Slave Address Comparison Result Flag Note 4
0	Slave address register (SVA) not equal to serial I/O shift register 0 (SIO0) data
1	Slave address register (SVA) equal to serial I/O shift register 0 (SIO0) data

R/W

CSIE0	Serial Interface Channel 0 Operation Control
0	Operation stopped
1	Operation enable

Notes
1. Bit 6 (COI) is a Read-Only bit.
2. Can be used freely as port function.
3. When the wake-up function is used, set bit 5 of the interrupt timing specification register (SINT) to 0. Do not write data to serial I/O shift register 0 (SIO0) during WUP = 1.
4. When CSIE0 = 0, COI becomes 0.

Remark × : don't care
PM××: Port mode register
P×× : Output latch of port

(b) Serial bus interface control register (SBIC)

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

$\overline{\text{RESET}}$ input sets SBIC to 00H.

Symbol	<7>	<6>	<5>	<4>	<3>	<2>	<1>	<0>	Address	When Reset	R/W
SBIC	BSYE	ACKD	ACKE	ACKT	CMDD	RELD	CMDT	RELT	FF61H	00H	R/W[Note]

R/W

RELT	Use for bus release signal output when the SBI mode is used. When RELT = 1, SO latch is set to 1. After SO latch setting, automatically cleared to 0. Also cleared to 0 when CSIE0 = 0.

R/W

CMDT	Use for command signal output when the SBI mode is used. 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

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 (only if WUP = 1) • When CSIE0 = 0 • When $\overline{\text{RESET}}$ input is applied	• When bus release signal (REL) is detected in the SBI mode

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 in the SBI mode • When CSIE0 = 0 • When $\overline{\text{RESET}}$ input is applied	• When command signal (CMD) is detected in the SBI mode

R/W

ACKT	When the SBI mode is used, acknowledge signal is output in synchronization with the falling edge of $\overline{\text{SCK0}}$ clock immediately after execution of the instruction to be set to 1, and after acknowledge signal output, automatically cleared to 0. Used as ACKE = 0. Also cleared to 0 upon start of serial interface transfer or when CSIE0 = 0.

(continued)

Note Bits 2, 3 and 6 (RELD, CMDD, and ACKD) are Read-Only bits.

Remarks 1. Zeros will be returned form bits 0, 1, and 4 (or RELT, CMDT, ACKT, respectively) if users read these bits after data setting is completed.

2. CSIE0: Bit 7 of the serial operating mode register 0 (CSIM0)

(continued)

R/W

ACKE	Acknowledge Signal Automatic Output Control (in the SBI mode)	
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 $\overline{SCK0}$ (automatically output when ACKE = 1).
	After completion of transfer	Acknowledge signal is output in synchronization with the falling edge of $\overline{SCK0}$ clock immediately after execution of the instruction to be set to 1 (automatically output when ACKE = 1). However, not automatically cleared to 0 after acknowl edge signal output.

R

ACKD	Acknowledge Detection
Clear Conditions (ACKD = 0)	Set Conditions (ACKD = 1)
• In the SBI mode, at the falling edge of $\overline{SCK0}$ clock immediately after the busy mode has been released when a transfer start instruction is executed • When CSIE0 = 0 • When $\overline{RESET}$ input is applied	• When acknowledge signal ($\overline{ACK}$) is detected at the rising edge of $\overline{SCK0}$ clock after completion of transfer

R/W

BSYE[Note]	Synchronizing Busy Signal Output Control
0	When the SBI mode is used, disables busy signal which is output in synchronization with the falling edge of $\overline{SCK0}$ clock immediately after execution of the instruction to be cleared to 0 (with READY state).
1	When the SBI mode is used, outputs busy signal at the falling edge of $\overline{SCK0}$ clock following the acknowledge signal.

★ **Note** The busy mode can be cleared by start of serial interface transfer. However, the BSYE flag is not cleared to 0.

Remark CSIE0: Bit 7 of the serial operating mode register 0 (CSIM0)

(c) Interrupt timing specification register (SINT)
SINT is set with a 1-bit or 8-bit memory manipulation instruction.
$\overline{\text{RESET}}$ input sets SINT to 00H.

Symbol	7	<6>	<5>	<4>	<3>	<2>	1	0	Address	When Reset	R/W
SINT	0	CLD	SIC	SVAM	CLC	WREL	WAT1	WAT0	FF63H	00H	R/W[Note 1]

R/W

SVAM	SVA Bit to be Used as Slave Address
0	Bits 0 to 7
1	Bits 1 to 7

R/W

SIC	INTCSI0 Interrupt Factor Selection[Note 2]
0	CSIIF0 is set (1) upon termination of serial channel 0 transfer
1	CSIIF0 is set (1) upon bus release detection in SBI mode.

R

CLD	$\overline{\text{SCK0}}$/SCL/P27 Pin Level[Note 3]
0	Low Level
1	High Level

Notes **1.** Bit 6 (CLD) is a Read-Only bit.
2. When using wake-up function, set SIC to 0.
3. When CSIE0 = 0, CLD becomes 0.

Caution Be sure to set bits 0 to 3 to 0 when the SBI mode is used.

Remark SVA : Slave address register
CSIIF0 : Interrupt request flag supports the INTCSI0
CSIE0 : Bit 7 of the serial operating mode register 0 (CSIM0)

(4) Various signals

Figures 16-22 to 16-27 show various signals in the SBI and flag operations of the serial bus interface control register (SBIC). Table 16-5 lists various signals in SBI.

Figure 16-22. RELT, CMDT, RELD and CMDD Operations (Master)

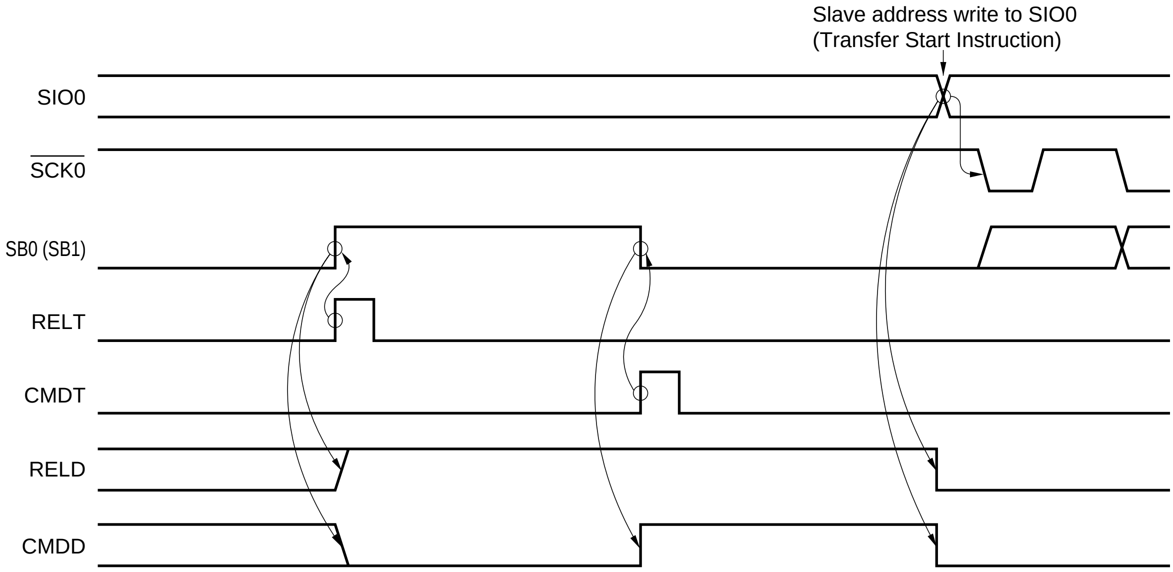


Figure 16-23. RELD and CMDD Operations (Slave)

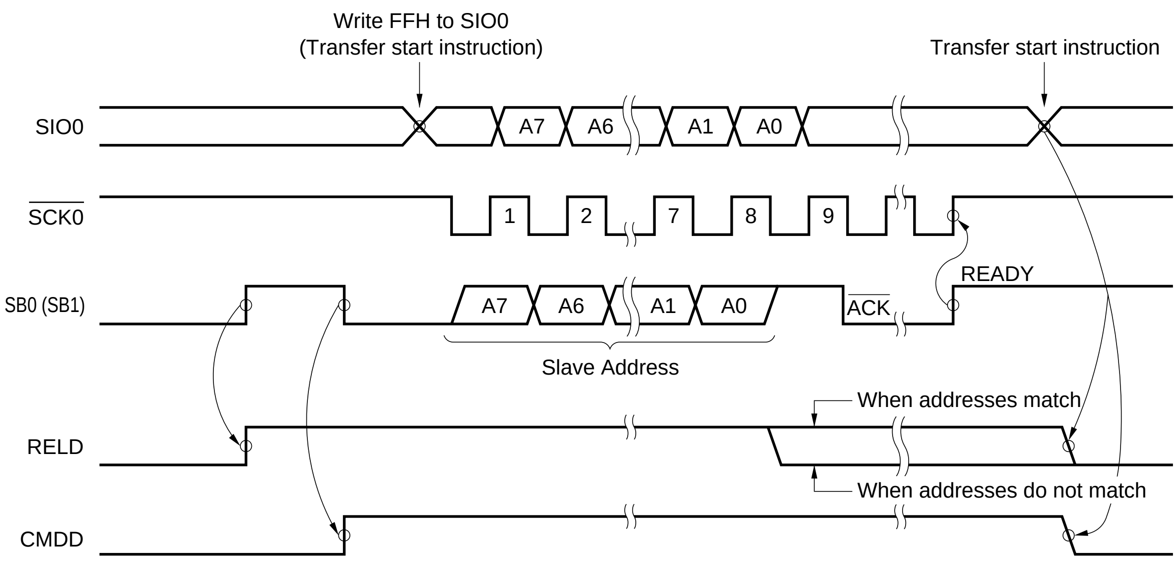
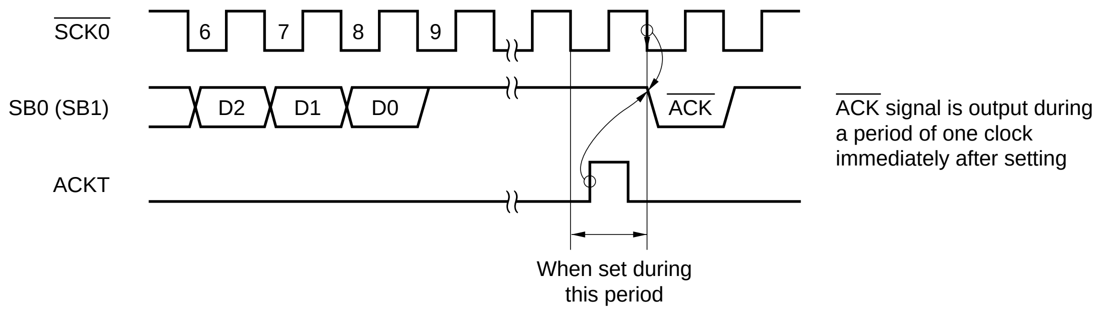


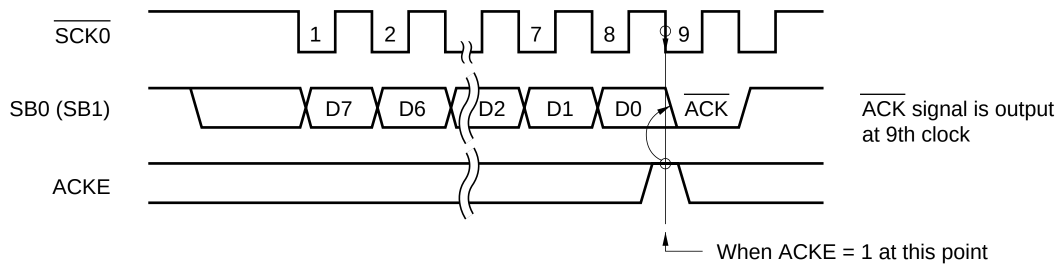
Figure 16-24. ACKT Operation



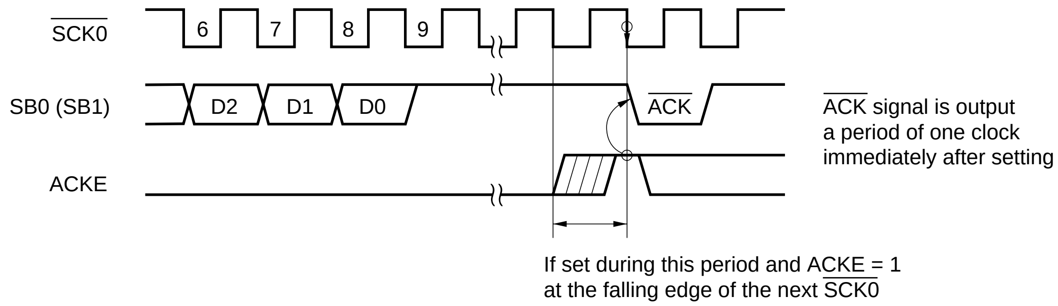
Caution Do not set ACKT before termination of transfer.

Figure 16-25. ACKE Operations

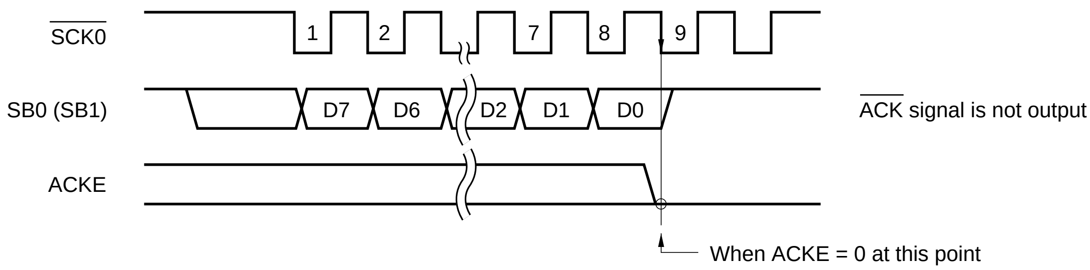
(a) When ACKE = 1 upon completion of transfer



(b) When set after completion of transfer



(c) When ACKE = 0 upon completion of transfer



(d) When ACKE = 1 period is short

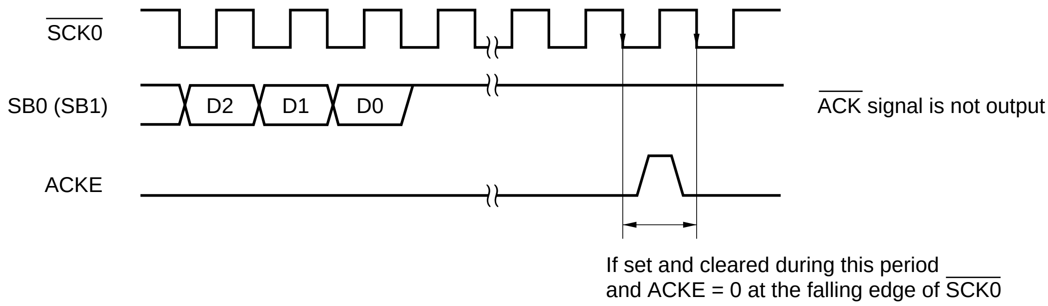
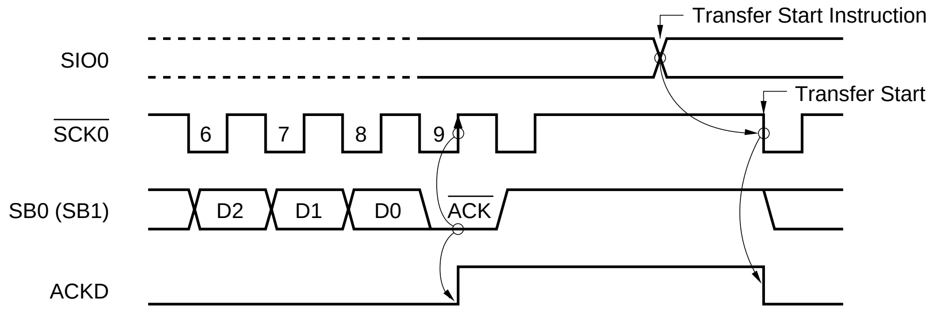
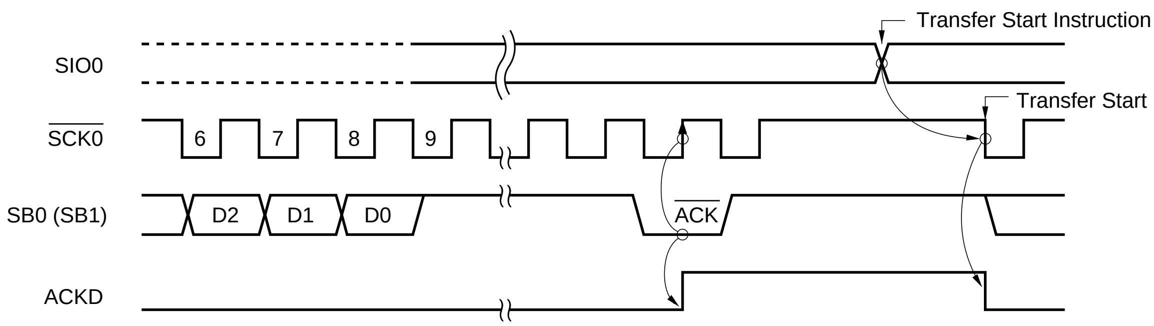


Figure 16-26. ACKD Operations

(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

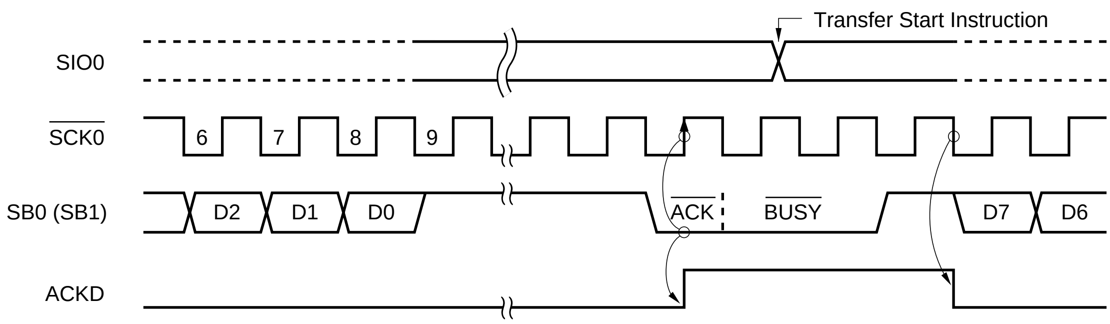


Figure 16-27. BSYE Operation

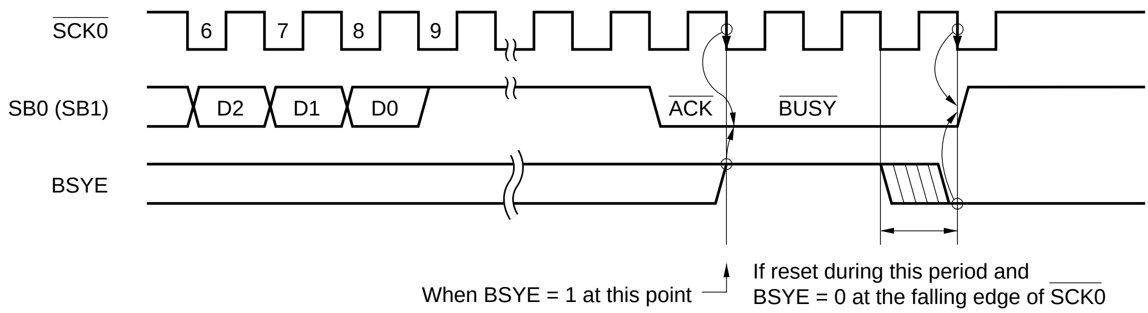


Table 16-5. Various Signals in SBI Mode (1/2)

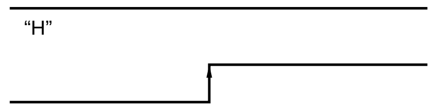
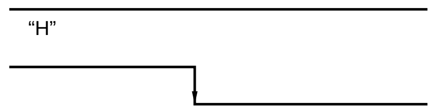
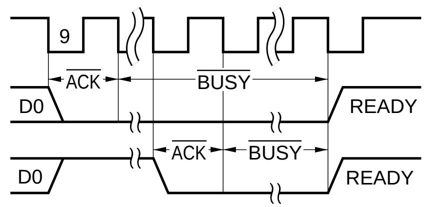
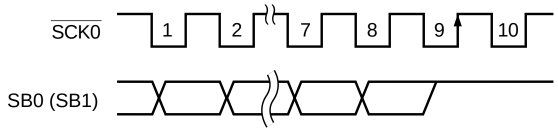
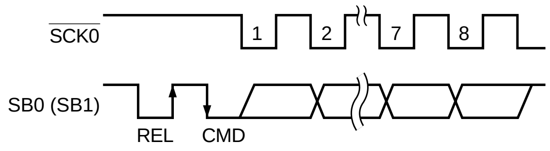
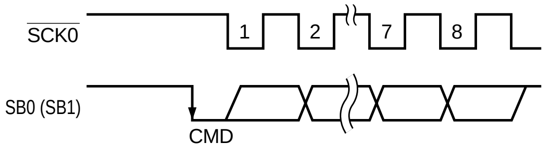
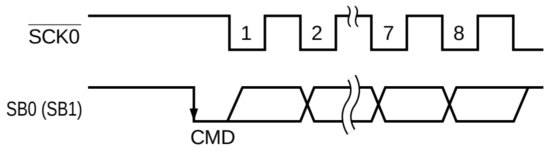
Signal Name	Output Device	Definition	Timing Chart	Output Condition	Effects on Flag	Meaning of Signal
Bus release signal (REL)	Master	SB0 (SB1) rising edge when $\overline{SCK0}$ = 1		• RELT set	• RELD set • CMDD clear	CMD signal is output to indicate that transmit data is an address.
Command signal (CMD)	Master	SB0 (SB1) falling edge when $\overline{SCK0}$ = 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 ($\overline{ACK}$)	Master/ slave	Low-level signal to be output to SB0 (SB1) during one-clock period of $\overline{SCK0}$ after completion of serial reception		(1) ACKE = 1 (2) ACKT set	• ACKD set	Completion of reception
Busy signal ($\overline{BUSY}$)	Slave	[Synchronous BUSY signal] Low-level signal to be output to SB0 (SB1) following acknowledge signal	[Synchronous BUSY output] 	• 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		(1) BSYE = 0 (2) Execution of instruction data write SIO0 (transfer start instruction)	—	Serial receive enable

Table 16-5. Various Signals in SBI Mode (2/2)

Signal Name	Output Device	Definition	Timing Chart	Output Condition	Effects on Flag	Meaning of Signal
Serial clock ($\overline{SCK0}$)	Master	Synchronous clock to output address/command/data, $\overline{ACK}$ signal, synchronization $\overline{BUSY}$ 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 $\overline{SCK0}$) Note 1	Timing of signal output to serial data bus
Address (A7 to A0)	Master	8-bit data to be transferred in synchronization with $\overline{SCK0}$ after output of REL and CMD signals				Address value of slave device on the serial bus
Address (C7 to C0)	Master	8-bit data to be transferred in synchronization with $\overline{SCK0}$ after output of only CMD signal without REL signal output				Instruction messages to the slave device
Address (D7 to D0)	Master/ slave	8-bit data to be transferred in synchronization with $\overline{SCK0}$ 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 $\overline{SCK0}$.
When WUP = 1, an address is received. Only when the address matches the slave address register (SVA), CSIIF0 is set (when the address does not match, RELD is cleared).

2. In $\overline{BUSY}$ state, transfer starts after the READY state is set.

(5) Pin configuration

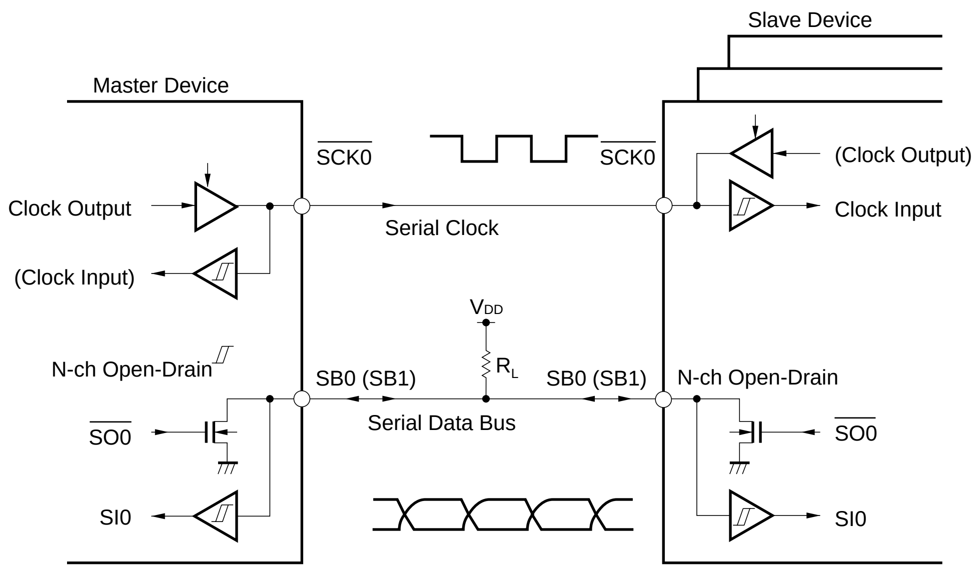
The serial clock pin $\overline{SCK0}$ and serial data bus pin SB0 (SB1) have the following configurations.

(a) $\overline{SCK0}$: Serial clock input/output pin
<1> Master : CMOS and push-pull output
<2> Slave : Schmitt input

(b) SB0 (SB1) : Serial data input/output 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 16-28. Pin Configuration



Caution Because the N-ch open-drain output must be set to high-impedance at the time of data reception, write FFH to the serial I/O shift register 0 (SIO0) in advance. However, when the wake-up function specification bit (WUP) = 1, the N-ch open-drain output will always be set to high-impedance. Thus, it is not necessary to write FFH to SIO0 before reception.

(6) Address match detection method

In the SBI mode, a particular slave device can be selected by sending a slave address from the master device. Address match detection is automatically executed by hardware.

With the slave address register (SVA), and if the wake-up function specification bit (WUP) = 1, CSIIF0 is set only when the slave address transmitted from the master device matches the value set in SVA.

If bit 5 (SIC) of the interrupt timing specification register (SINT) is set to 1, the wake-up function does not operate even with WUP = 1 (an interrupt request signal is generated in detecting a bus release). When the wake-up function is used, 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 request (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 request 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, the 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 the serial operating mode register 0 (CSIM0) is tested. If it is “1”, normal transmission is judged to have been carried out. If it is “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 16-29 to 16-32 show data communication timing charts.

Shift operation of the serial I/O shift register 0 (SIO0) 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 SIO0.

Figure 16-29. Address Transmission from Master Device to Slave Device (WUP = 1)

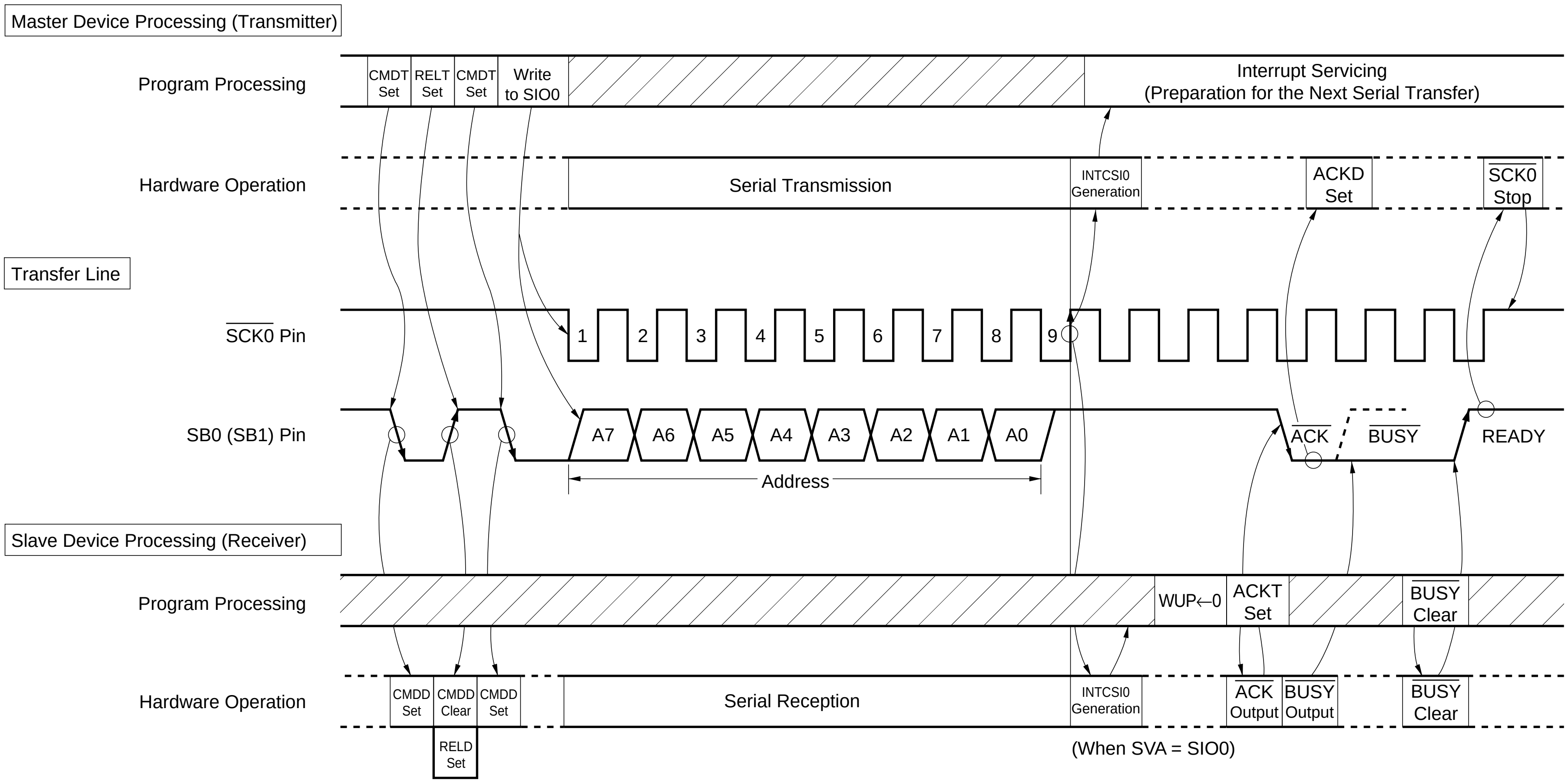


Figure 16-30. Command Transmission from Master Device to Slave Device

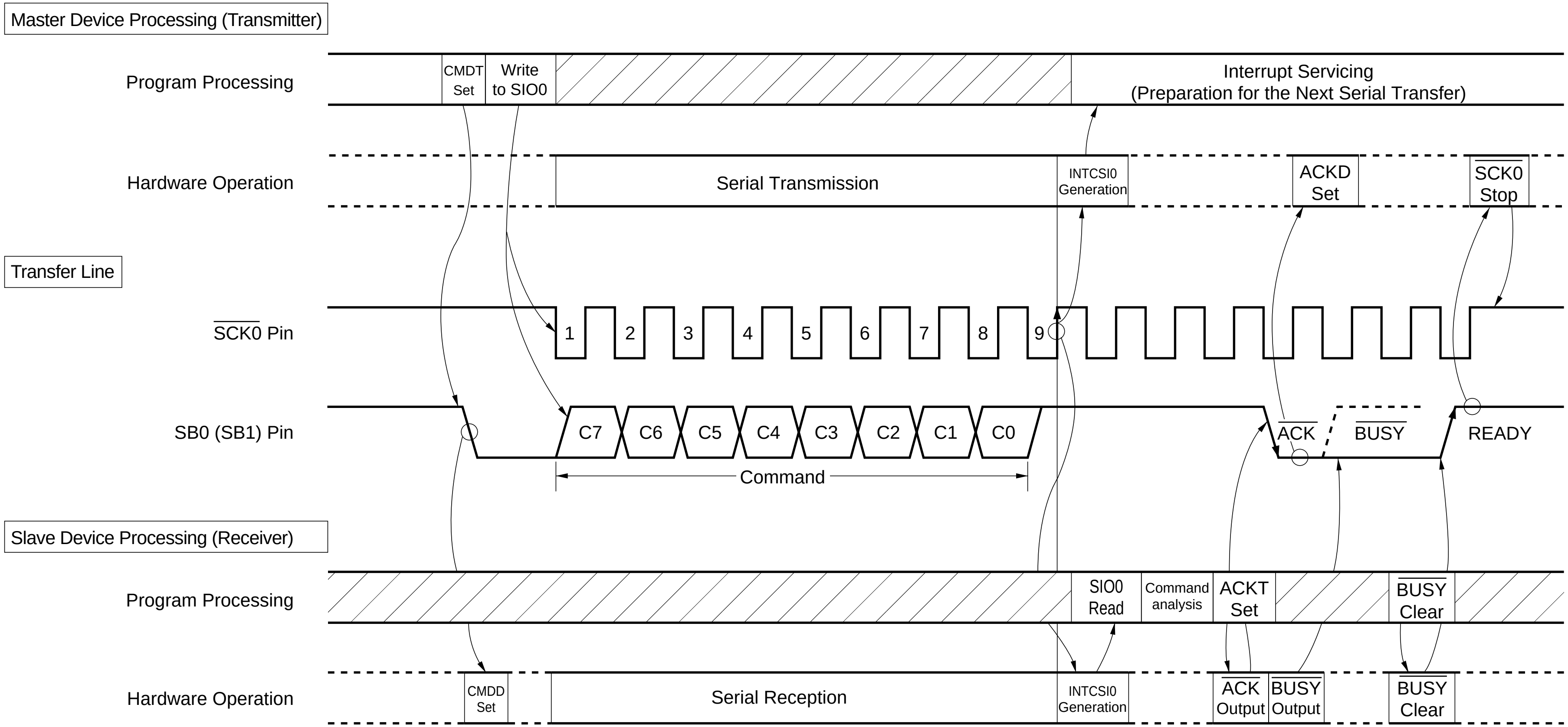


Figure 16-31. Data Transmission from Master Device to Slave Device

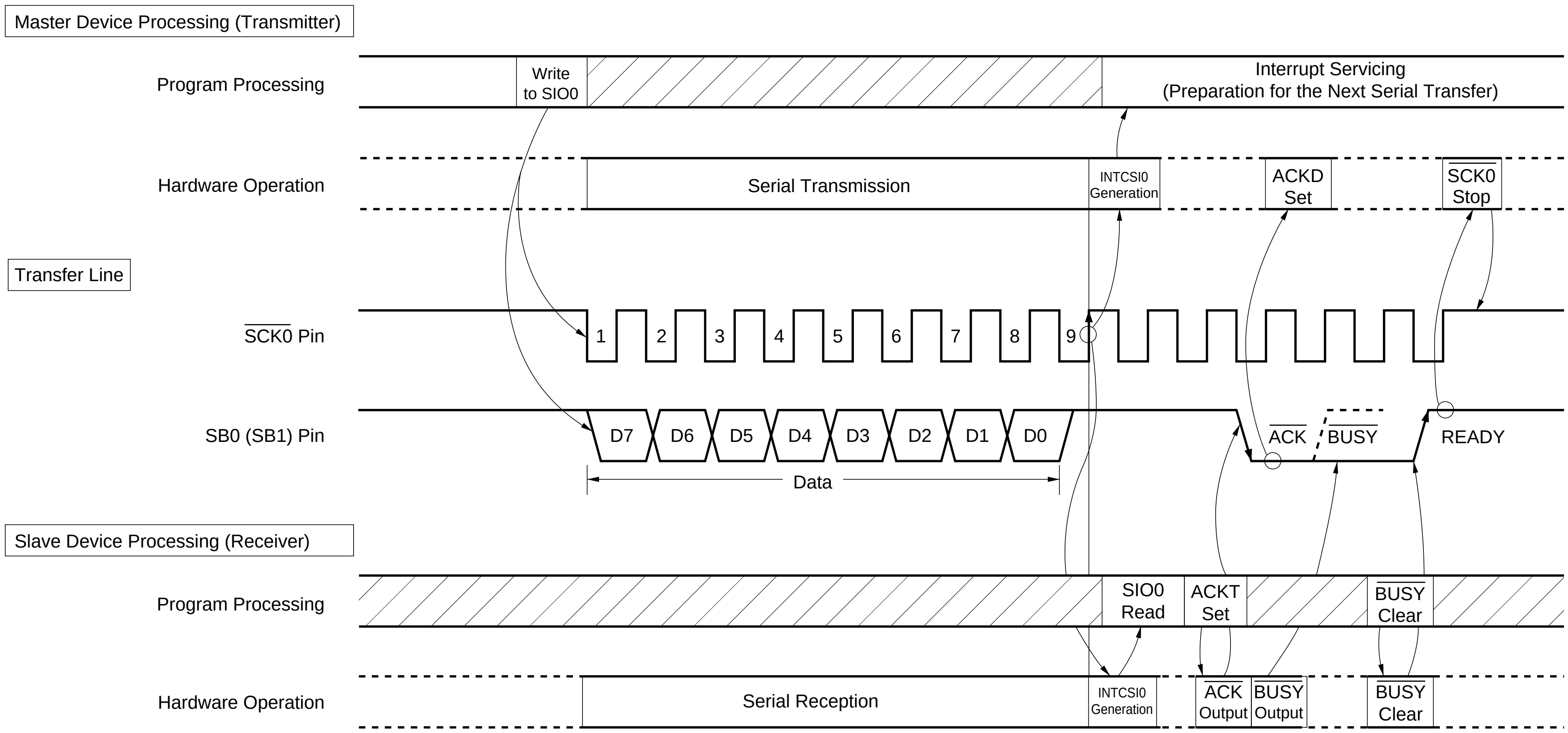
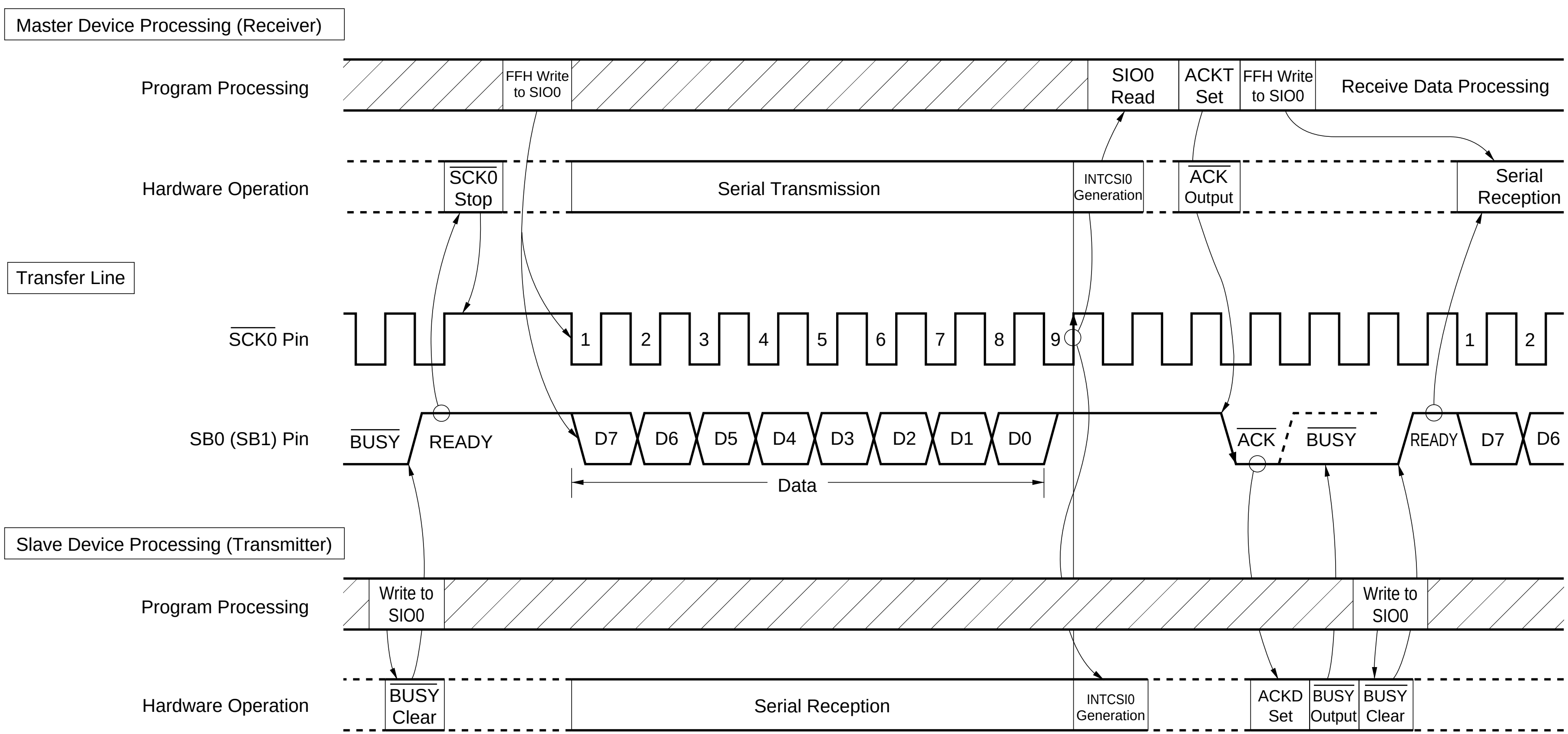


Figure 16-32. Data Transmission from Slave Device to Master Device



(9) Transfer start

Serial transfer is started by setting transfer data to the 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 open-drain output must be set to high-impedance at the time of data reception, write FFH to SIO0 in advance. However, when the wake-up function specification bit (WUP) = 1, the N-ch open-drain output will always be set to high-impedance. 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 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 control register (SBIC) to 1.
<3> Reset the P25 and P26 output latches from 1 to 0.

(10) Distinction method of slave busy state

When device is in the master mode, follow the method below to judge whether the slave device is in the busy state or not.

<1> Detect acknowledge signal ($\overline{\text{ACK}}$) 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) Cautions on SBI mode

(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 request (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) In the SBI mode, the $\overline{\text{BUSY}}$ signal is output until the falling of the next serial clock after the $\overline{\text{BUSY}}$ release indication. If WUP = 1 is set by mistake during this period, $\overline{\text{BUSY}}$ will not be released. Thus, after releasing $\overline{\text{BUSY}}$, be sure to check that the SB0 (SB1) has become high level before setting WUP = 1.

(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 $\overline{\text{RESET}}$ 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.

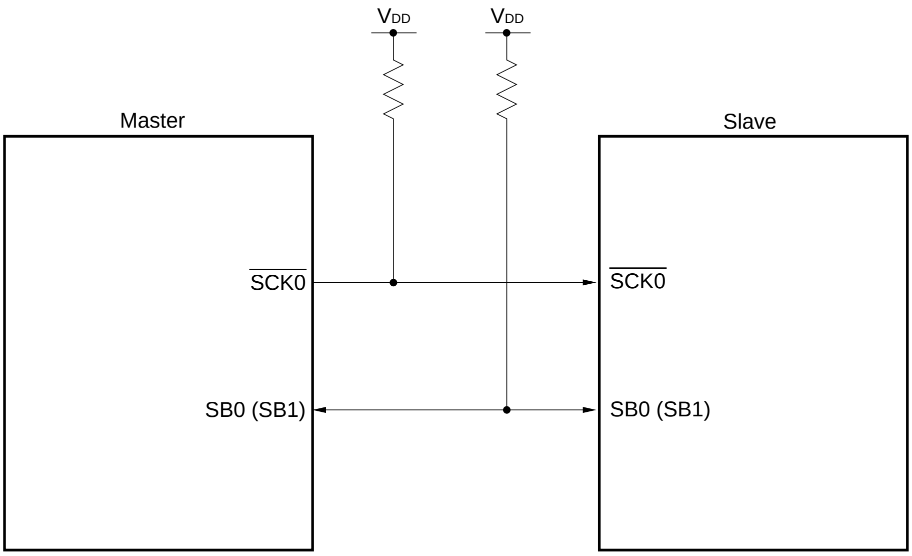
★ (e) The bus release signal or the command signal is acknowledged when the $\overline{\text{SCK0}}$ line is in high level, and the SB0 (SB1) line changes from low level to high level or from high level to low level. Thus, if the timing at which bus changes deviates due to effects such as board capacity, it may be determined as the bus release signal (or the command signal) even if data is sent. Therefore perform wiring carefully.

16.4.4 2-wire serial I/O mode operation

The 2-wire serial I/O mode supports 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 16-33. Example of Serial Bus Configuration with 2-Wire Serial I/O



(1) Register setting

The 2-wire serial I/O mode is set with the serial operating mode register 0 (CSIM0), the serial bus interface control register (SBIC) and the interrupt timing specification 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 sets CSIM0 to 00H.

Symbol	<7>	<6>	<5>	4	3	2	1	0	Address	When Reset	R/W
CSIM0	CSIE0	COI	WUP	CSIM04	CSIM03	CSIM02	CSIM01	CSIM00	FF60H	00H	R/W^Note 1

R/W

CSIM01	CSIM02	Serial Interface Channel 0 Clock Selection
0	×	Input clock to $\overline{SCK0}$ 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)

R/W

CSIM 04	CSIM 03	CSIM 02	PM25	P25	PM26	P26	PM27	P27	Operating Mode	Start Bit	SI0/SB0/SDA0/ P25 Pin Function	SO0/SB1/SDA1/ P26 Pin Function	$\overline{SCK0}$/SCL/ P27 Pin Function
0	×	3-wire serial I/O mode (Refer to **16.4.2 3-wire serial I/O mode operation**)											
1	0	SBI mode (Refer to **16.4.3 SBI mode operation**)											
1	1	0	× Note 2	× Note 2	0	0	0	1	2-wire serial I/O mode or I²C bus mode	MSB	P25 (CMOS input/output)	SB1/SDA1 (N-ch open-drain input/output)	$\overline{SCK0}$/SCL (N-ch open-drain input/output)
		1	0	0	× Note 2	× Note 2	0	1			SB0/SDA0 (N-ch open-drain input/output)	P26 (CMOS input/output)	

R/W

WUP	Wake-up Function Control^Note 3
0	Interrupt request signal generation with each serial transfer in all modes
1	Interrupt request signal generation when the address received after bus release (when CMDD = RELD = 1 when the SBI mode is used or when CMDD = 1 when the I²C bus mode is used) matches the slave address register (SVA) data in the SBI mode and I²C bus mode

R

COI	Slave Address Comparison Result Flag^Note 4
0	Slave address register (SVA) not equal to serial I/O shift register 0 (SIC0) data
1	Slave address register (SVA) equal to serial I/O shift register 0 (SIC0) data

R/W

CSIE0	Serial Interface Channel 0 Operation Control
0	Operation stopped
1	Operation enable

Notes **1.** Bit 6 (COI) is a Read-Only bit.
2. Can be used freely as port function.
3. When 2-wire serial I/O mode is used, be sure to set WUP to 0.
4. When CSIE0 = 0, COI becomes 0.

Remark × : don't care
PM××: Port mode register
P×× : Output latch of port

(b) Serial bus interface control register (SBIC)

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

$\overline{\text{RESET}}$ input sets SBIC to 00H.

Symbol	<7>	<6>	<5>	<4>	<3>	<2>	<1>	<0>	Address	When Reset	R/W
SBIC	BSYE	ACKD	ACKE	ACKT	CMDD	RELD	CMDT	RELT	FF61H	00H	R/W

R/W		
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.
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.

CSIE0: Bit 7 of the serial operating mode register 0 (CSIM0)

(c) Interrupt timing specification register (SINT)

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

$\overline{\text{RESET}}$ input sets SINT to 00H.

Symbol	7	<6>	<5>	<4>	<3>	<2>	1	0	Address	When Reset	R/W
SINT	0	CLD	SIC	SVAM	CLC	WREL	WAT1	WAT0	FF63H	00H	R/W[Note 1]

R/W

SIC	INTCSI0 Interrupt Factor Selection
0	CSIIF0 is set (1) upon termination of serial channel 0 transfer
1	CSIIF0 is set (1) upon bus release detection

R

CLD	$\overline{\text{SCK0}}$/SCL/P27 Pin Level[Note 2]
0	Low Level
1	High Level

Notes 1. Bit 6 (CLD) is a Read-Only bit.
2. When CSIE0 = 0, CLD becomes 0.

Caution Be sure to set bits 0 to 3 to 0 when 2-wire serial I/O mode is used.

Remark CSIIF0 : Interrupt request flag supports the INTCSI0
CSIE0 : Bit 7 of the serial operating mode register 0 (CSIM0)

(2) Communication operation

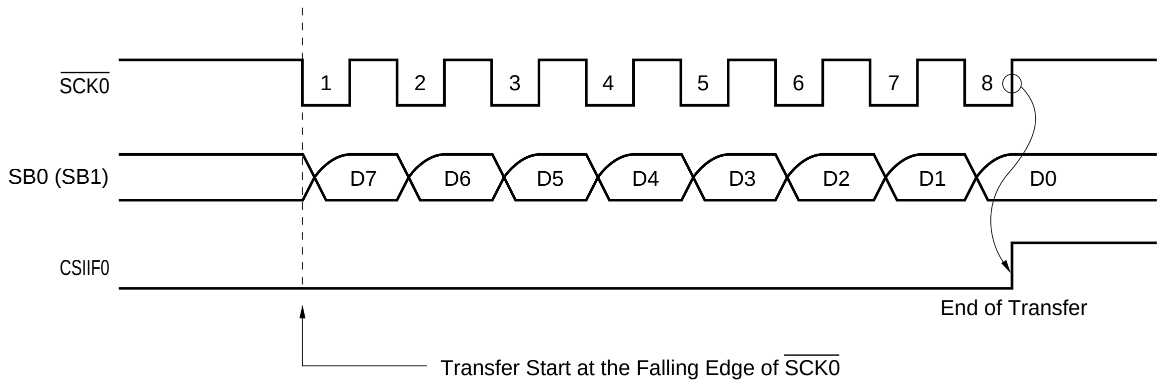
The 2-wire serial I/O mode is used for data transmission/reception in 8-bit units. Data transmission/reception is carried out bit-wise in synchronization with the serial clock.

Shift operation of the serial I/O shift register 0 (SIO0) is carried out in synchronization with the falling edge of the serial clock ($\overline{SCK0}$).

The transmit data is held in the SO0 latch and is output from the SB0/SDA0/P25 (or SB1/SDA1/P26) pin with MSB set at start. The receive data input from the SB0 (or SB1) pin is latched into the SIO0 at the rising edge of $\overline{SCK0}$.

Upon termination of 8-bit transfer, the SIO0 operation stops automatically and the interrupt request flag (CSIIF0) is set.

Figure 16-34. 2-Wire Serial I/O Mode Timings



The SB0 (or SB1) pin specified for the serial data bus serves for N-ch open-drain input/output and thus it must be externally pulled up. Because it is necessary to be set to high-impedance the N-ch open-drain output for data reception, write FFH to SIO0 in advance.

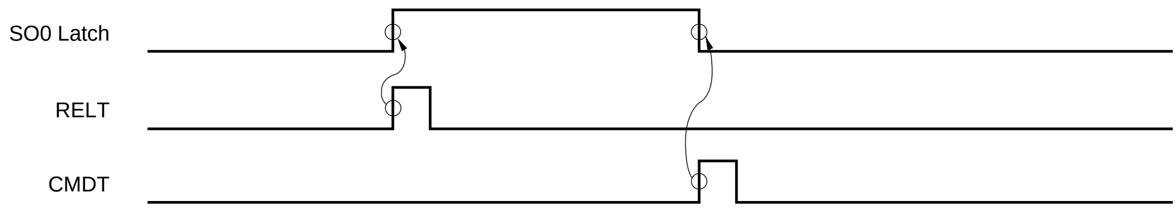
The SB0 (or SB1) pin generates the SO0 latch status and thus the SB0 (or SB1) pin output status can be manipulated by setting bit 0 (RELT) and bit 1 (CMDT) of the serial bus interface control register (SBIC). However, do not carry out this manipulation during serial transfer.

Control the $\overline{SCK0}$ pin output level in the output mode (internal system clock mode) by manipulating the P27 output latch (refer to **16.4.8 $\overline{SCK0}$/SCL/P27 pin output manipulation**).

(3) Various signals

Figure 16-35 shows RELT and CMDT operations.

Figure 16-35. RELT and CMDT Operations



(4) Transfer start

Serial transfer is started by setting transfer data to the 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 $\overline{SCK0}$ 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 open-drain output must be set to high-impedance 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, the COI bit (match signal coming from the address comparator) of the serial operating mode register 0 (CSIM0) is tested. If it is "1", normal transmission is judged to have been carried out. If it is "0", a transmit error is judged to have occurred.

16.4.5 I²C bus mode operation

The I²C bus mode is used 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 I²C bus specification, the master sends start condition, data, and stop condition signals to slave devices through the serial data bus.

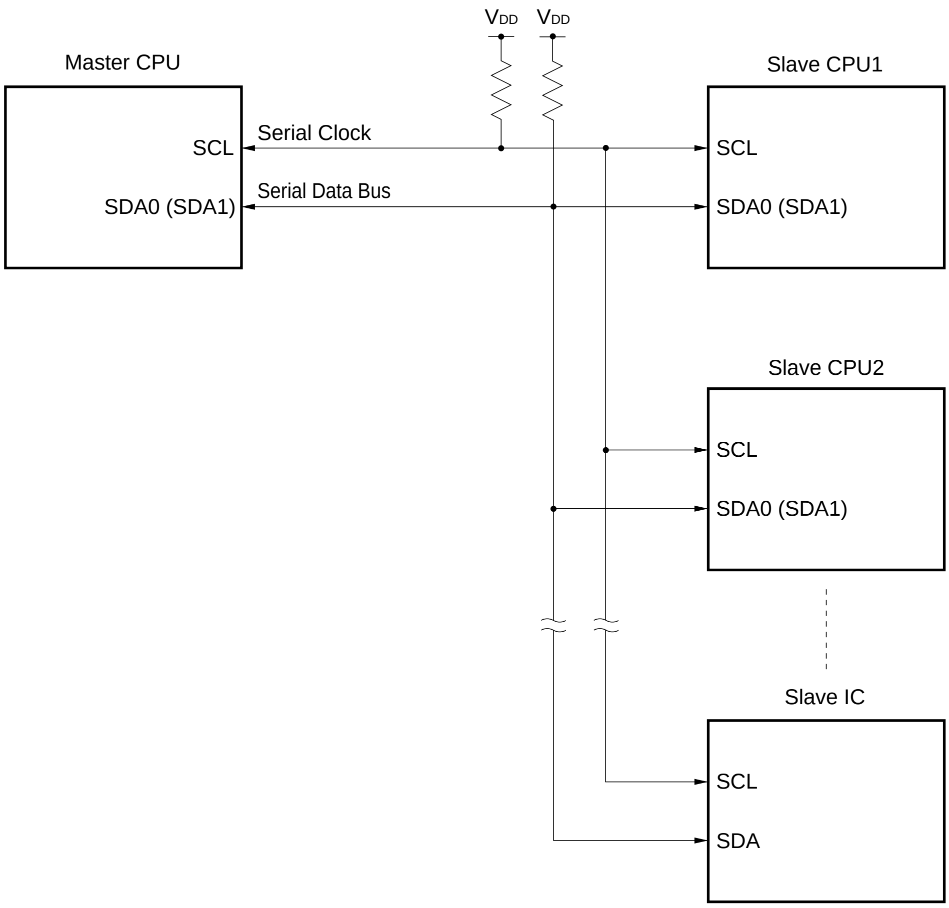
Slave devices automatically detect and distinguish the type of signals due to the signal detection function incorporated as hardware. This simplifies the application program to control I²C bus.

An example of a serial bus configuration is shown in Figure 16-36. This system below is composed of CPUs and peripheral ICs having serial interface hardware that complies with the I²C 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 (SDA1) on the I²C bus.

The signals used in the I²C bus mode are described in Table 16-6.

Figure 16-36. Serial Bus Configuration Example Using I²C Bus



(1) I²C bus mode functions

In the I²C 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 I²C bus and communicate with it by sending in advance the address data corresponding to the destination device.

(c) Wake-up function

Interrupt request occurs only if the received address equal to the value of the slave address register (SVA) during slave operation. Therefore, CPUs other than the selected slave device on the I²C 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

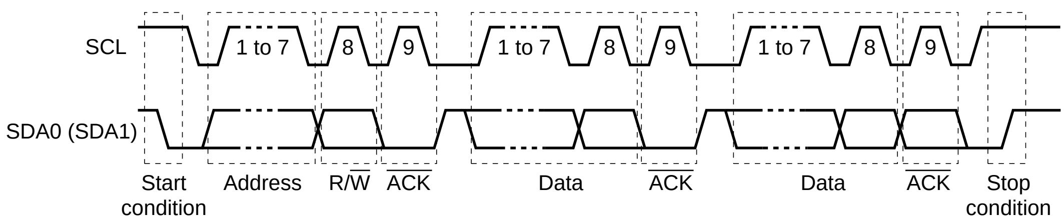
The slave device controls a wait signal on the bus to inform the master device of the wait status.

(2) I²C bus definition

This section describes the format of serial data communications and functions of the signals used in the I²C bus mode.

The transfer timings of the start condition, data, and stop condition signals, which are output onto the signal data bus of the I²C bus, are shown in Figure 16-37.

Figure 16-37. I²C Bus Serial Data Transfer Timing



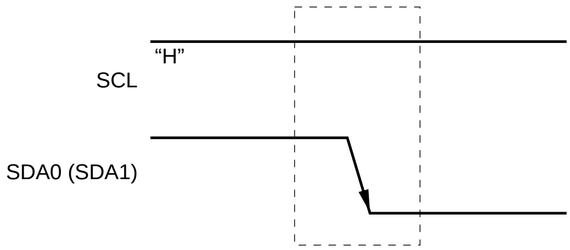
The start condition, slave address, and stop condition signals are output by the master.
The acknowledge signal ($\overline{ACK}$) is output by either the master or the slave device (normally by the device which has received the 8-bit data that was sent).
A serial clock (SCL) is continuously supplied 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 section **16.4.6 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 16-38. Start Condition



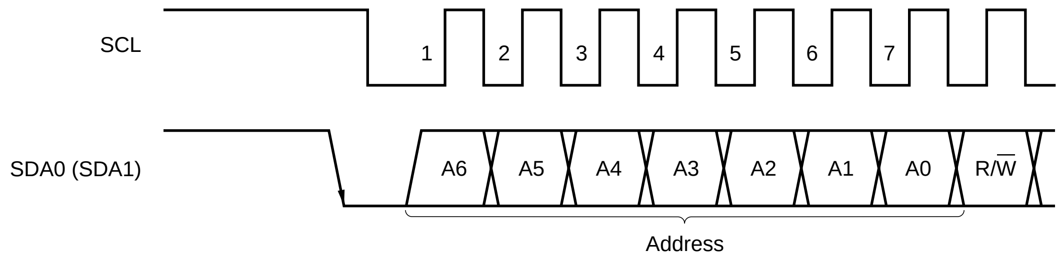
(b) Address

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 16-39. Address

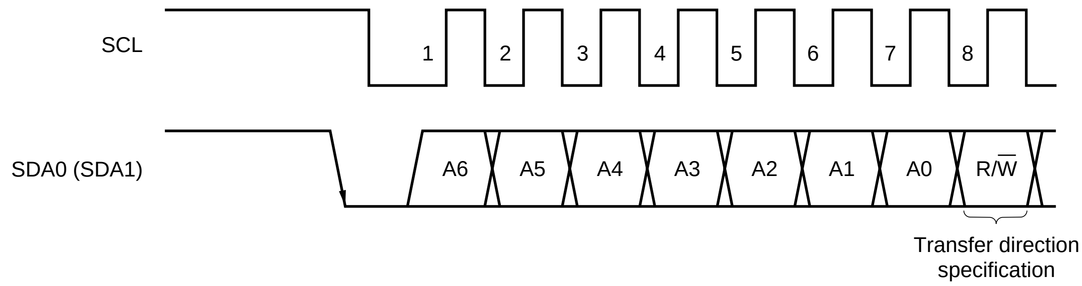


(c) Transfer direction specification

The 1-bit data 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 16-40. Transfer Direction Specification



(d) Acknowledge signal ($\overline{ACK}$)

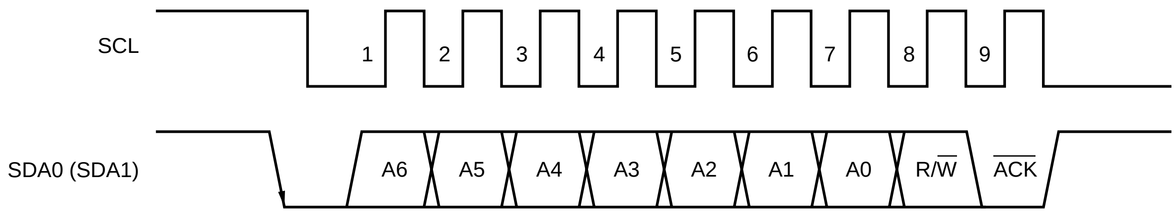
The acknowledge signal indicates that the transferred serial data has definitely been received. The receiving side returns an acknowledge signal each time it receives 8-bit data.

The receiving side usually outputs after 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 checks if the acknowledge signal has been 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 correctly by the slave device and therefore the master device outputs a stop condition signal to terminate subsequent transmissions.

Figure 16-41. Acknowledge Signal



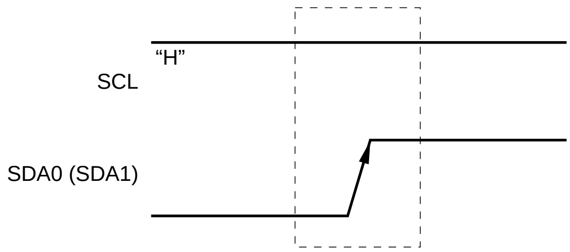
(e) Stop condition

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 16-42. Stop Condition



(f) Wait signal ($\overline{\text{WAIT}}$)

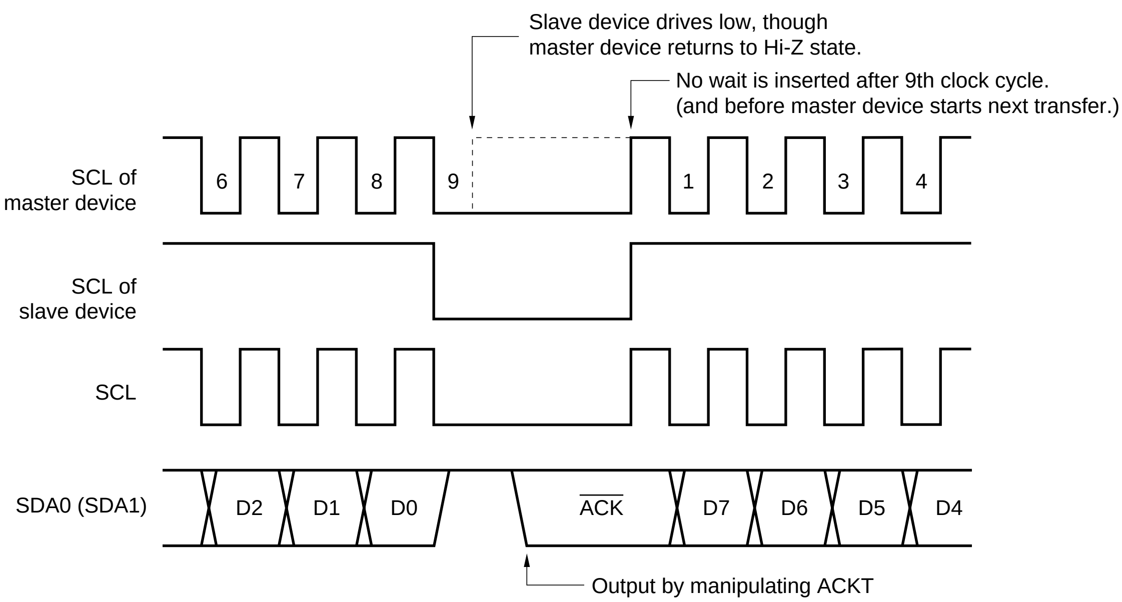
The wait signal is output by a slave device to inform the master device that the slave device is in wait state due to preparing for transmitting or receiving data.

The slave device notifies the master device about the wait state by keeping the SCL pin low.

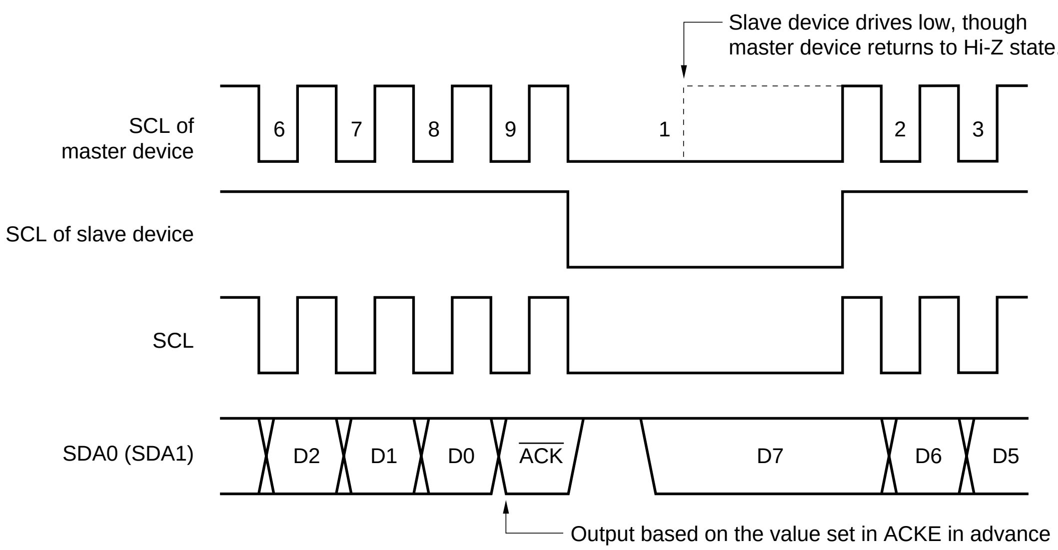
When the wait state is released, the master device can start the next transfer. For the releasing operation of slave devices, see section **16.4.6 Cautions on use of I^{2}C bus mode.**

Figure 16-43. Wait Signal

(a) Wait of 8 Clock Cycles



(b) Wait of 9 Clock Cycles



(3) Register setting

The I²C bus mode setting is performed by the 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 by a 1-bit or 8-bit memory manipulation instruction.

$\overline{\text{RESET}}$ input sets 00H.

Symbol	<7>	<6>	<5>	4	3	2	1	0	Address	When Reset	R/W
CSIM0	CSIE0	COI	WUP	CSIM04	CSIM03	CSIM02	CSIM01	CSIM00	FF60H	00H	R/W Note 1

R/W

CSIM01	CSIM02	Serial Interface Channel 0 Clock Selection
0	×	Input clock to SCK0 pin from off-chip
1	0	8-bit timer register 2 (TM2) output Note 2
1	1	Clock specified with bits 0 to 3 of timer clock select register 3 (TCL3)

R/W

<table>
<tr><th>CSIM 04</th><th>CSIM 03</th><th>CSIM 02</th><th>PM25</th><th>P25</th><th>PM26</th><th>P26</th><th>PM27</th><th>P27</th><th>Operating Mode</th><th>Start Bit</th><th>SI0/SB0/SDA0/ P25 Pin Function</th><th>SO0/SB1/SDA1/ P26 Pin Function</th><th>$\overline{SCK0}$/SCL/ P27 Pin Function</th></tr>
<tr><td>0</td><td>×</td><td colspan="12">3-wire serial I/O mode (Refer to 16.4.2 3-wire serial I/O mode operation)</td></tr>
<tr><td>1</td><td>0</td><td colspan="12">SBI mode (Refer to 16.4.3 SBI mode operation)</td></tr>
<tr><td rowspan="2">1</td><td rowspan="2">1</td><td>0</td><td>× Note 3</td><td>× Note 3</td><td>0</td><td>0</td><td>0</td><td>1</td><td rowspan="2">2-wire serial I/O mode or I²C bus mode</td><td rowspan="2">MSB</td><td>P25 (CMOS input/output)</td><td>SB1/SDA1 (N-ch open-drain input/output)</td><td rowspan="2">$\overline{SCK0}$/SCL (N-ch open-drain input/output)</td></tr>
<tr><td>1</td><td>0</td><td>0</td><td>× Note 3</td><td>× Note 3</td><td>0</td><td>1</td><td>SB0/SDA0 (N-ch open-drain input/output)</td><td>P26 (CMOS input/output)</td></tr>
</table>

Notes **1.** Bit 6 (COI) is a Read-Only bit.

2. When the I²C bus mode is used, the clock frequency is 1/16 of the clock frequency output by TO2.

3. Can be used freely as a port.

Remark × : don't care

PM×× : Port mode register

P×× : Output latch of port

R/W

WUP	Wake-up Function Control[Note 1]
0	Interrupt request signal generation with each serial transfer in all modes
1	Interrupt request signal generation when the address received after bus release (when CMDD = RELD = 1 when the SBI mode is used or when CMDD = 1 when the I^2C bus mode is used) matches the slave address register (SVA) data in the SBI mode and the I^2C bus mode

R

COI	Slave Address Comparison Result Flag[Note 2]
0	Slave address register (SVA) not equal to serial I/O shift register 0 (SIC0) data
1	Slave address register (SVA) equal to serial I/O shift register 0 (SIC0) data

R/W

CSIE0	Serial Interface Channel 0 Operation Control
0	Operation stopped
1	Operation enable

Notes
1. When the wake-up function is used, set bit 5 of the interrupt timing specification register (SINT) to 0. Do not write data to serial I/O shift register 0 (SIO0) during WUP = 1.
2. When CSIE0 = 0, COI is 0.

(b) Serial bus interface control register (SBIC)

SBIC is set by a 1-bit or 8-bit memory manipulation instruction.

RESET input sets SBIC to 00H.

Symbol	<7>	<6>	<5>	<4>	<3>	<2>	<1>	<0>	Address	When Reset	R/W
SBIC	BSYE	ACKD	ACKE	ACKT	CMDD	RELD	CMDT	RELT	FF61H	00H	R/W[Note]

R/W

RELT	
	Use for stop condition output when the I²C mode is used. When RELT = 1, SO latch is set to 1. After SO latch setting, automatically cleared to 0. Also cleared to 0 when CSIE0 = 0.

R/W

CMDT	
	Use for start condition output when the I²C mode is used. 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

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 RESET input is applied	• When stop condition is detected in the I²C bus mode

R

CMDD	Start Condition Detection
Clear Conditions (CMDD = 0)	Set Conditions (CMDD = 1)
• When transfer start instruction is executed • When stop condition is detected in the I²C bus mode • When CSIE0 = 0 • When RESET input is applied	• When start condition is detected in the I²C bus mode

R/W

ACKT	
	When the I²C bus mode is used, SDA0 (SDA1) is made low-level until the next SCL falling edge immediately after execution of the set instruction (ACKT = 1). Used to generate ACK signal by software when 8-clock wait is selected. Also cleared to 0 upon start of serial interface transfer or when CSIE0 = 0.

(continued)

Note Bits 2, 3, and 6 (RELD, CMDD, ACKD) are Read-Only bits.

Caution Be sure to set bit 7 to 0 when the I²C bus is used.

Remark CSIE0: Bit 7 of the serial operating mode register 0 (CSIM0)

R/W

ACKE	Acknowledge Signal Automatic Output Control[Note 1] (in the I^2C bus mode)
0	Disables acknowledge signal automatic output. (However, output with ACKT enable) Use for reception when 8-clock wait mode is selected or for transmission[Note 2].
1	Enables acknowledge signal automatic output. Outputs acknowledge signal in synchronization with the 9th clock falling edge of SCL (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

ACKD Acknowledge Detection	
Clear Conditions (ACKD = 0)	Set Conditions (ACKD = 1)
• Upon execution of a transfer start instruction in the I^2C mode • 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

Notes **1.** Should be set before starting transfer.

2. Output acknowledge signal in reception with ACKT when 8-clock wait is selected.

Remark CSIE0: Bit 7 of the serial operating mode register 0 (CSIM0)

(c) Interrupt timing specification register (SINT)

SINT is set by the 1-bit or 8-bit memory manipulation instruction.

RESET input sets SINT to 00H.

Symbol	7	<6>	<5>	<4>	<3>	<2>	1	0	Address	When Reset	R/W
SINT	0	CLD	SIC	SVAM	CLC	WREL	WAT1	WAT0	FF63H	00H	R/W[Note 1]

R/W

WAT1	WAT0	Wait and Interrupt Control[Note 2]
0	0	Generates interrupt service request at rising edge of 8th SCK0 clock cycle. (Keeping clock output in high impedance)
0	1	Setting prohibited
1	0	Used in the I²C 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 output. In the case of slave device, makes SCL output low to request wait pulses are input.)
1	1	Used in the I²C 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 output. In the case of slave device, makes SCL output low to request waits pulses are input.)

R/W

WREL	Wait State 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

CLC	Clock Level Control
0	Used in the I²C bus mode. Make output level of SCL pin low unless serial transfer is being performed.
1	Used in I²C bus mode. Make SCL pin enter high-impedance state unless serial transfer is being performed (except for clock line which is kept high) Use to enable master device to generate start condition and stop condition signal.

(continued)

Notes **1.** Bit 6 (CLD) is Read-Only bit.

2. When the I²C bus mode is used, be sure to set 1 and 0, or 1 and 1 in WAT0 and WAT1, respectively.

R/W

SVAM	SVA Bit to be Used as Slave Address
0	Bits 0 to 7
1	Bits 1 to 7

R/W

SIC	INTCSI0 Interrupt Cause Selection[Note 1]
0	CSIIF0 is set to 1 upon termination of serial interface channel 0 transfer
1	CSIIF0 is set to 1 upon stop condition detection in I^2C bus mode

R

CLD	$\overline{SCK0}$/SCL/P27 Pin Level[Note 2]
0	Low level
1	High level

Notes **1.** When using the wake-up function in the I^2C mode, be sure to set SIC to 1.

2. When CSIE0 = 0, CLD is 0.

Remark SVA : Slave address register

CSIIF0 : Interrupt request flag supports the INTCSI0

CSIE0 : Bit 7 of the serial operating mode register 0 (CSIM0)

(4) Various signals

A list of signals in the I²C bus mode is given in Table 16-6.

Table 16-6. Signals in the I²C Bus Mode

Signal name	Signaled by	Definition	Signaled when	Affected flag(s)	Function
Start condition	Master	SDA0 (SDA1) falling edge when SCL is high[Note 1]	CMDT is set.	CMDD is set.	Indicates that serial communication starts and subsequent data are address data.
Stop condition	Master	SDA0 (SDA1) rising edge when SCL is high[Note 1]	RELT is set.	RELD is set and CMDD is cleared	Indicates end of serial transmission.
Acknowledge signal ($\overline{ACK}$)	Master or slave	Low level of SDA0 (SDA1) pin during one SCL clock cycle after serial reception	• ACKE = 1. • ACKT is set.	ACKD is set.	Indicates completion of reception of 1 byte.
Wait ($\overline{WAIT}$)	Slave	Low-level signal output to SCL	WAT1, WAT0 = 1×.	—	Indicates state in which serial reception is not possible.
Serial Clock (SCL)	Master	Synchronization clock for output of various signals	Execution of data write instruction to SIO0 when CSIE0 = 1 (instruction of serial transfer start)[Note 2]	CSIIF0 is set.[Note 3]	Serial communication synchronization signal.
Address (A6 to A0)	Master	7-bit data synchronized with SCL immediately after start condition signal			Indicates address value for specification of slave on serial bus.
Transfer direction (R/$\overline{W}$)	Master	1-bit data output in synchronization with SCL after address output			Indicates whether data transmission or reception is to be performed.
Data (D7 to D0)	Master or slave	8-bit data synchronized with SCL, not immediately after start condition			Contains data actually to be sent.

Notes **1.** The level of the serial clock can be controlled by bit 3 (CLC) of the interrupt timing specification register (SINT).

2. 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 only when an address is received and the address matches the slave address register (SVA) value.

(5) Pin configurations

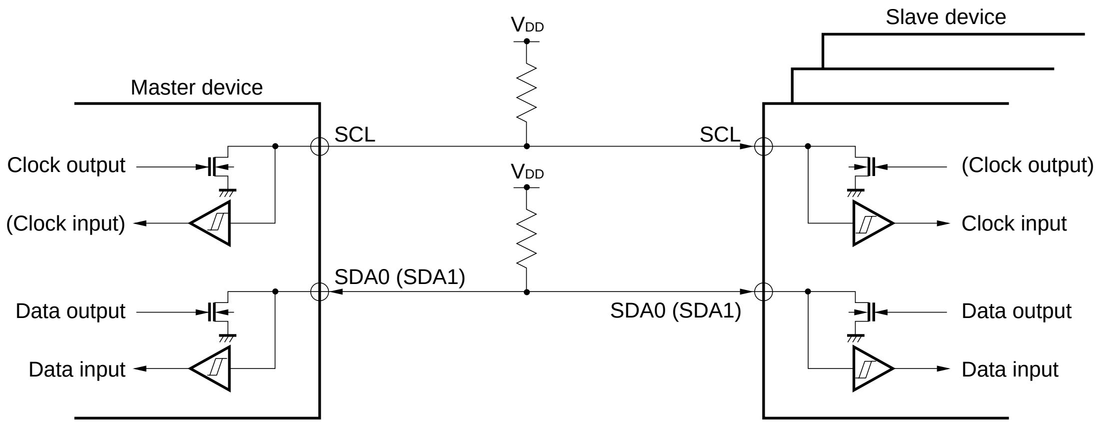
The configurations of the serial clock pin SCL and the serial data bus pins SDA0 (SDA1) are shown below.

(a) SCL .... Pin for serial clock input/output.
<1> Master .... N-ch open-drain output
<2> Slave .... Schmitt input

(b) SDA0 (SDA1) .... Serial data input/output dual-function pin.
Uses N-ch open-drain output and Schmitt-input buffers for both master and slave devices.

Both serial clock and serial data bus require the external pull-up resistors to be output by N-ch open drain.

Figure 16-44. Pin Configuration



Caution Because the N-ch open-drain output must be set to high-impedance at the time of data reception, write FFH to the serial I/O shift register 0 (SIO0) in advance. However, when wake-up function is used (that is, bit 5 (WUP) of serial operating mode register 0 (CSIM0) is set), do not write FFH to SIO0 before data reception. Without writing FFH to SIO0, the N-ch open-drain output is always high-impedance state.

(6) Address match detection method

In the I^2C 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. CSIIF0 is set only when a slave device address has a slave register (SVA), the wake-up function specification bit (WUP) is 1, and the slave address sent from the master device matches with the address set in SVA.

When bit 5 (SIC) of the interrupt timing specification register (SINT) is set to 1, the wake-up function does not operate if WUP is set to 1. (In the detection of stop condition, an interrupt request signal is generated.) When wake-up function is used, clear SIC to 0.

Caution Slave selection/non-selection is detected by matching of the slave address received after bus release.

For this match detection, match interrupt request (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.

(7) Error detection

In the I^2C 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 the 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^2C 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 $R/\overline{W}$ bit which indicates the data transfer direction, and starts serial communication with the selected slave device.

Data communication timing charts are shown in Figures 16-45 and 16-46.

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 SIO0 in synchronization with the rising edge of SCL.

Figure 16-45. Data Transmission from Master to Slave (Both Master and Slave Selected 9-Clock Wait) (1/3)

(a) Start Condition to Address

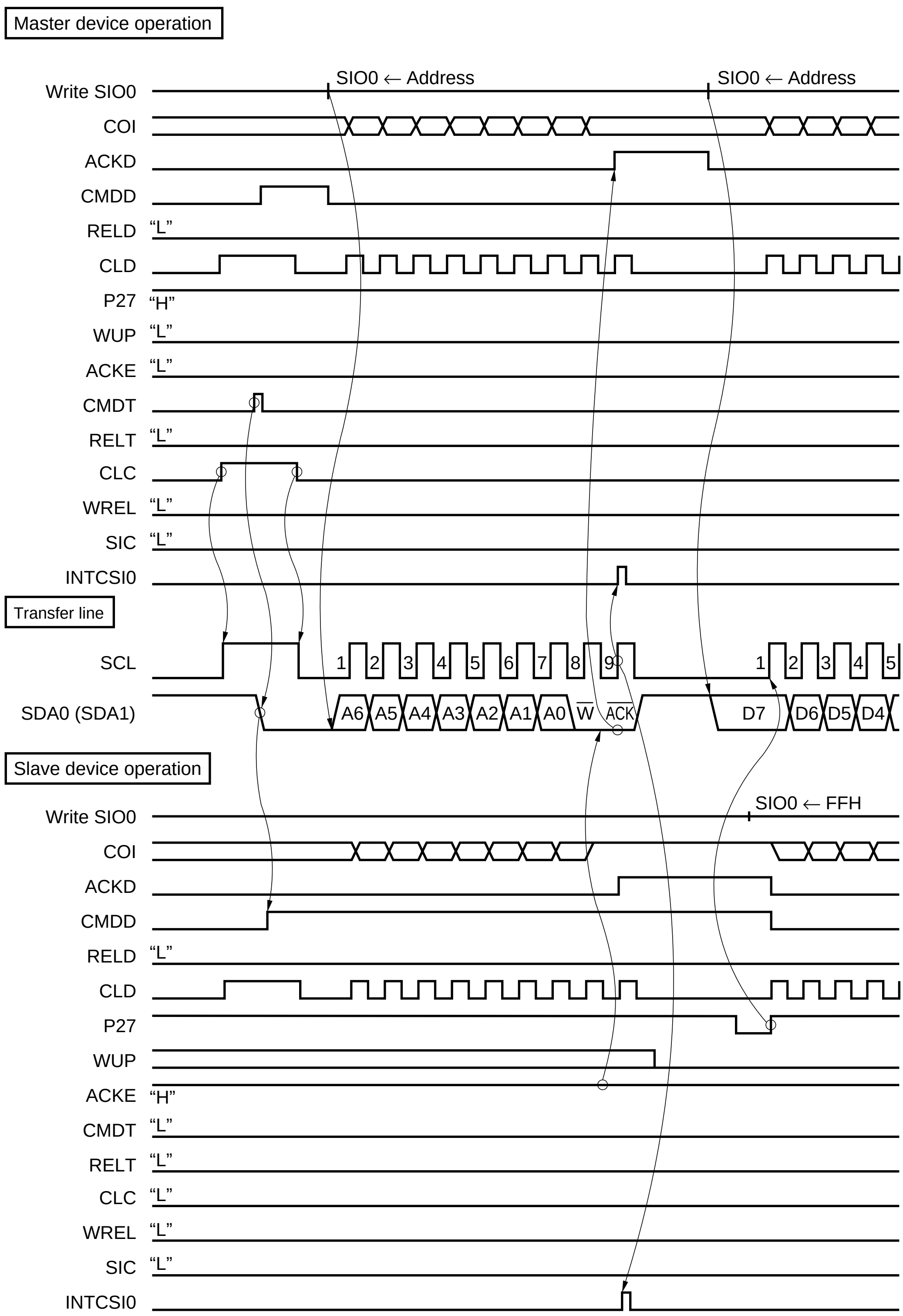


Figure 16-45. Data Transmission from Master to Slave (Both Master and Slave Selected 9-Clock Wait) (2/3)

(b) Data

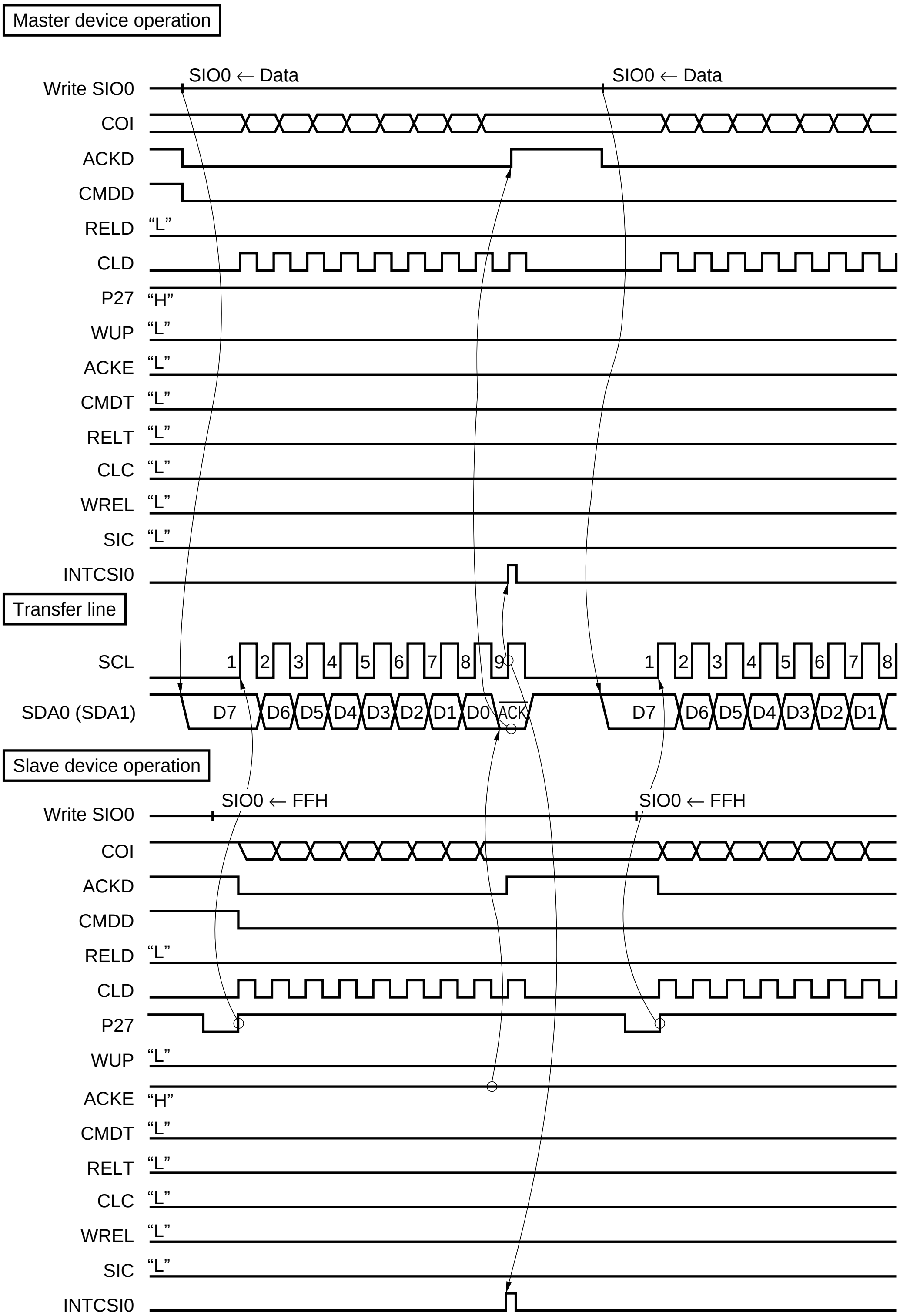
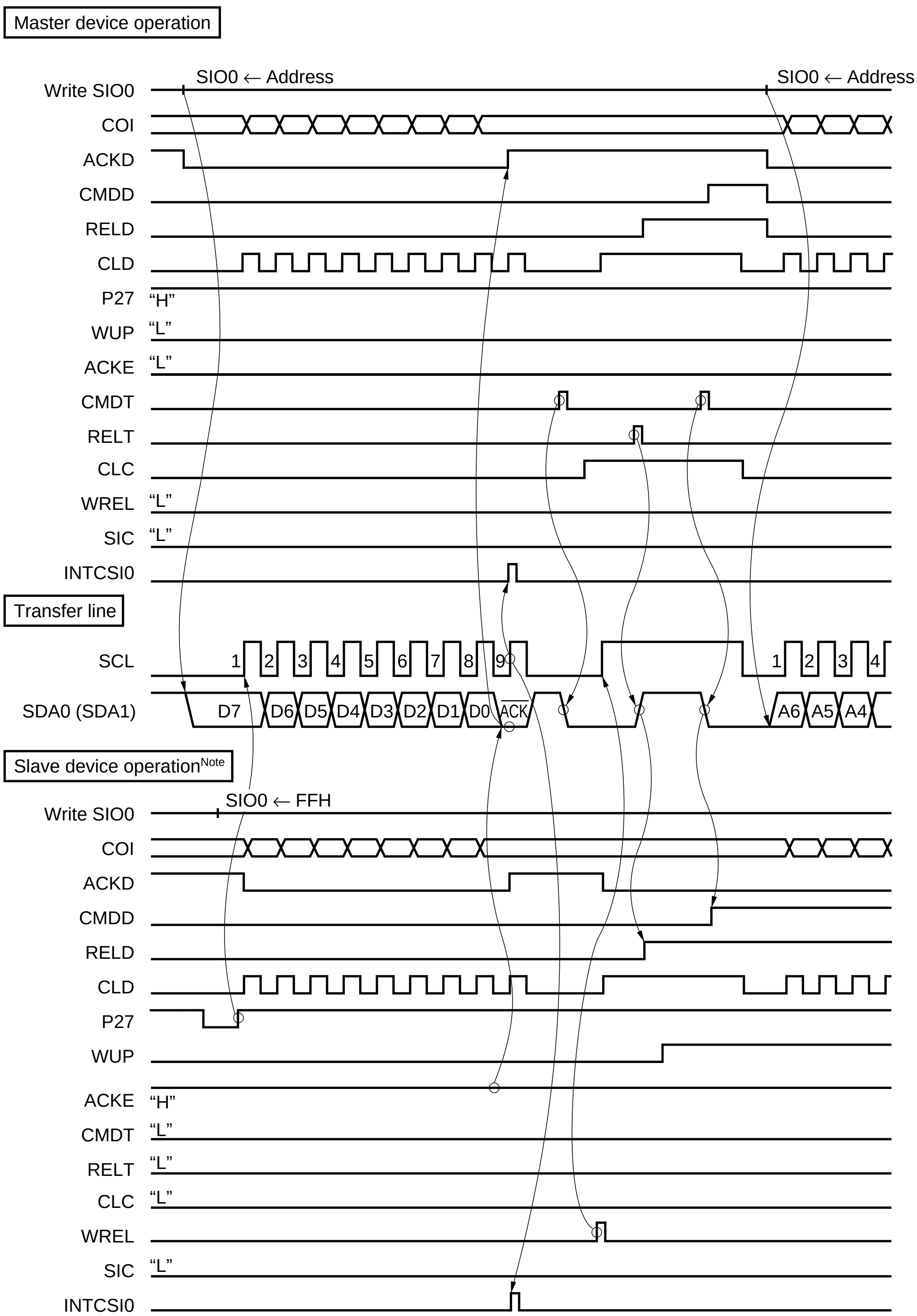


Figure 16-45. Data Transmission from Master to Slave (Both Master and Slave Selected 9-Clock Wait) (3/3)

(c) Stop Condition

★



Note This operation corresponds to the timing chart if it meets the specifications described in **16.4.7 (2) Avoidance.** Refer to **16.4.7 (2) Limitation when used as the slave device in the I²C bus mode,** for details.

Figure 16-46. Data Transmission from Slave to Master (Both Master and Slave Selected 9-Clock Wait) (1/3)

(a) Start Condition to Address

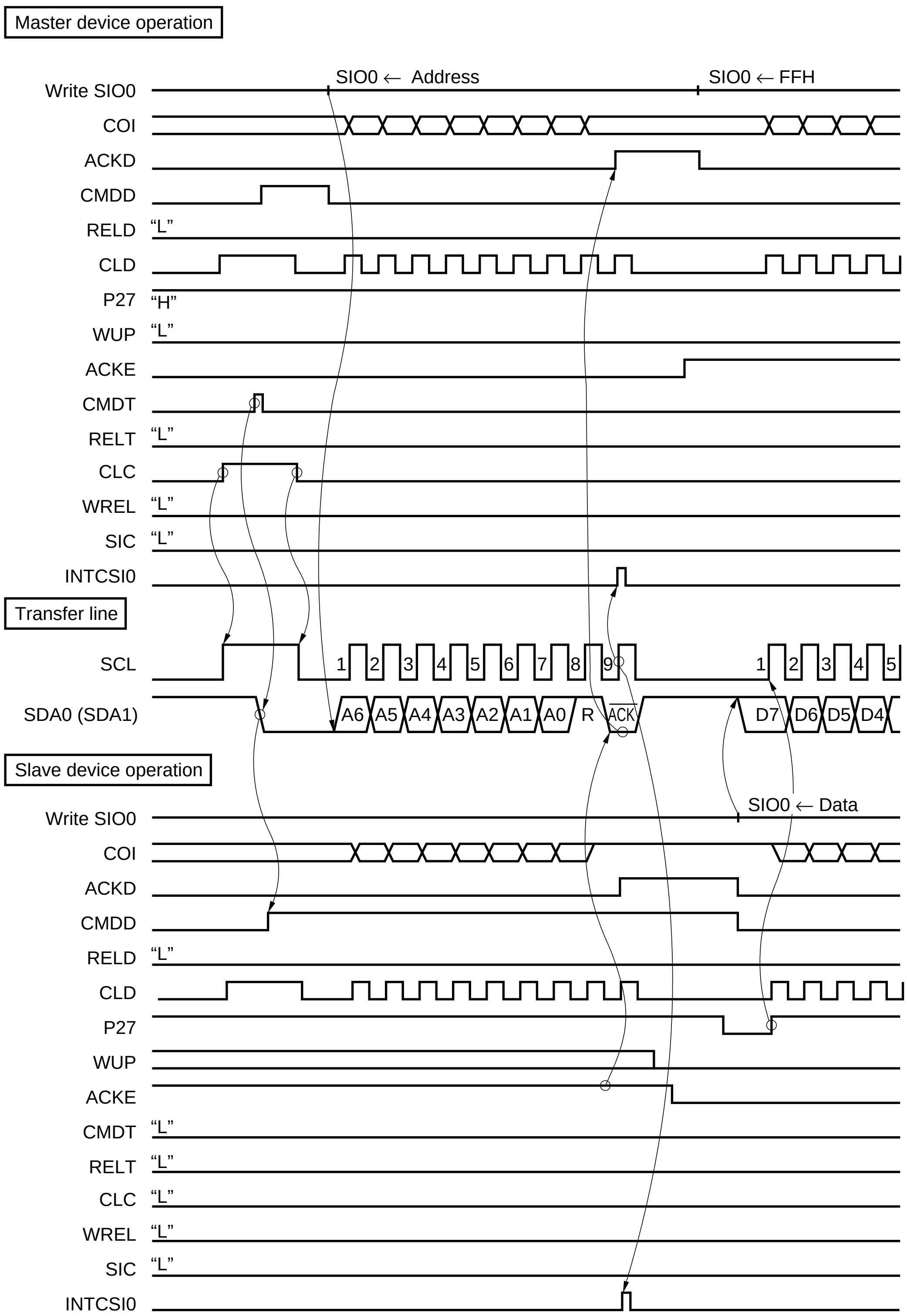


Figure 16-46. Data Transmission from Slave to Master (Both Master and Slave Selected 9-Clock Wait) (2/3)

(b) Data

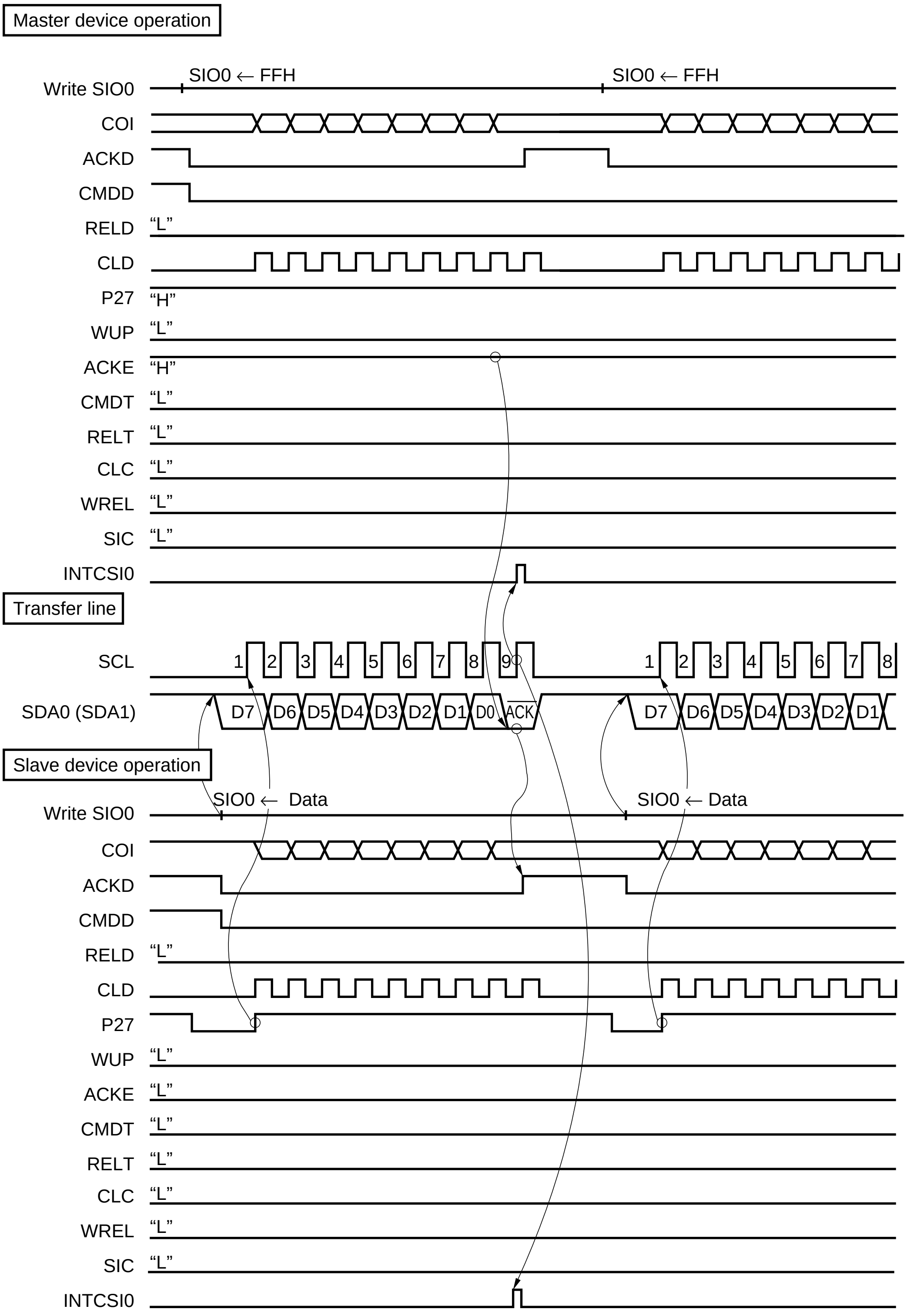
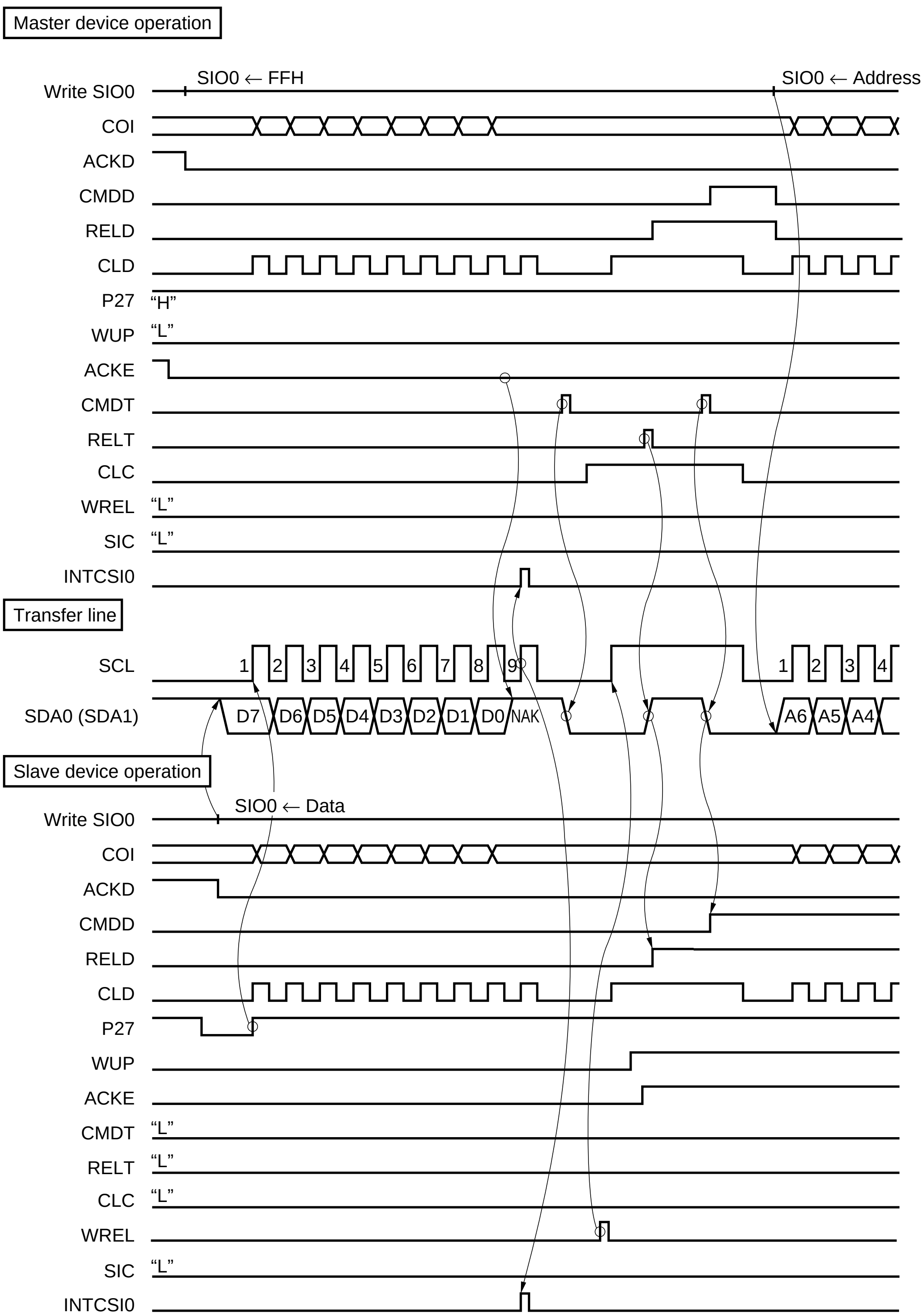


Figure 16-46. Data Transmission from Slave to Master (Both Master and Slave Selected 9-Clock Wait) (3/3)

(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:

- The 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. Setting CSIE0 to 1 after writing data in SIO0 does not initiate transfer operation.

2. Because the N-ch open-drain output must be set to high-impedance at the time of data reception, write FFH to the serial I/O shift register 0 (SIO0) in advance. However, when wake-up function is used (that is, bit 5 (WUP) of serial operating mode register 0 (CSIM0) is set), do not write FFH to SIO0 before data reception. Without writing FFH to SIO0, the N-ch open-drain output is always high-impedance state.

3. If data is written to SIO0 while the slave is in the wait state, that data is held. The 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.

16.4.6 Cautions on use of I²C bus mode

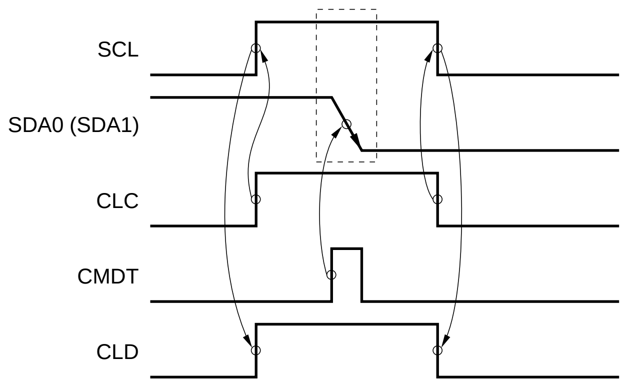
(1) Start condition output (master)

The SCL pin normally outputs the 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 bit 3 (CLC) of the interrupt timing specification 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. This is because a slave device may have set SCL to low (wait state).

Figure 16-47. Start Condition Output



(2) Slave wait release (slave transmission)

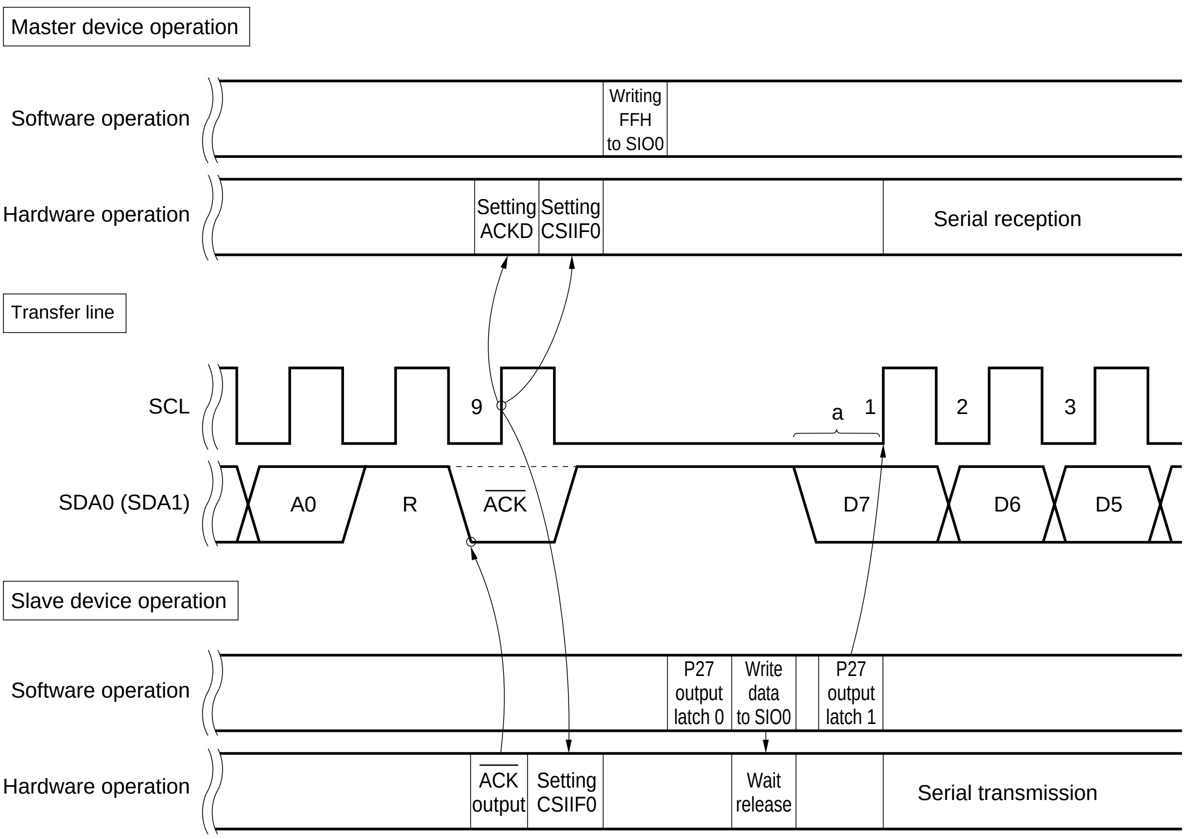
Slave wait release operation is performed by WREL flag (bit 2 of interrupt timing specification register (SINT)) setting or execution of a serial I/O shift register 0 (SIO0) write instruction.

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, as shown in Figure 16-48, data should be transmitted by manipulating the P27 output latch through the program. At this time, control the low-level width ("a" in Figure 16-48) 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 16-46**.

Figure 16-48. Slave Wait Release (Transmission)



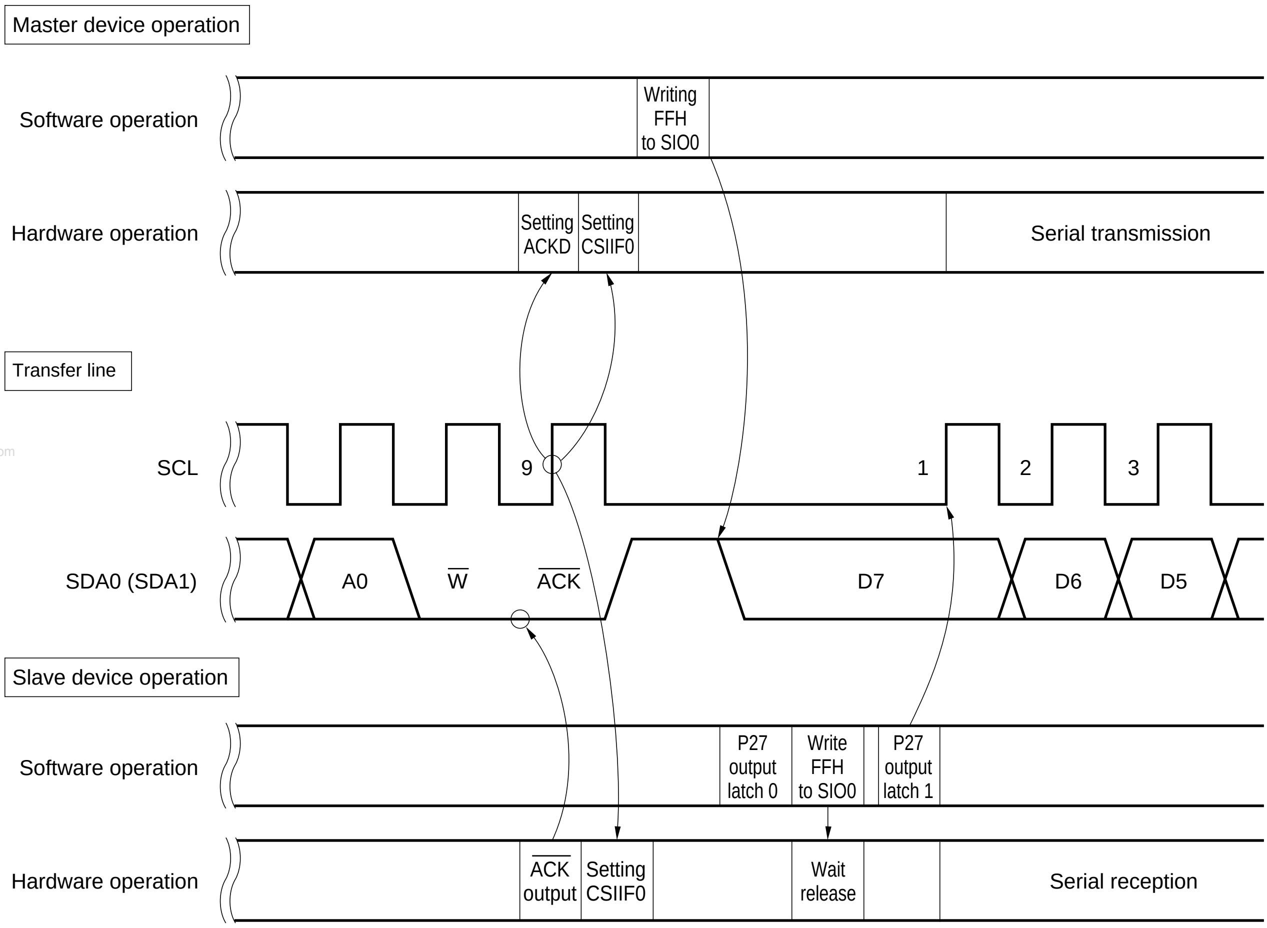
(3) Slave wait release (slave reception)

Slave wait release operation is performed by WREL flag (bit 2 of interrupt timing specification register (SINT)) setting or execution of a serial I/O shift register 0 (SIO0) write instruction.

When the slave receives a data, if the SCL line will immediately become high-impedance state by executing of write instruction to the SIO0, 1st bit data from the master may not be received. This is because if SCL line is being high-impedance state during execution of write instruction to the SIO0 (until next instruction execution), SIO0 does not start the operation. Therefore receive the data by manipulating the P27 output latch using program as shown in the Figure 16-49.

For these timings, see **Figure 16-45**.

Figure 16-49. Slave Wait Release (Reception)



(4) Reception completion processing by a slave

During processing of reception completion by a slave device (interrupt servicing etc.), confirm the status of bit 3 (CMDD) of the serial bus interface control register (SBIC) and bit 6 (COI) of serial operating mode register 0 (CSIM0) (when CMDD = 1). This procedure is necessary to use the wake-up function normally, because if an uncertain amount of data is sent from the master device, the slave device cannot determine whether the start condition signal or data will be sent from the master. This may disable use the wake-up function.

16.4.7 Restrictions on use of I^{2}C bus mode

The μPD78014Y subseries devices have the following restrictions.

(1) Restriction on master device operation in the I^{2}C bus mode

Applied device: μPD78P014Y
IE-78014-R-EM

Description: When the master device outputs the serial clock via the SCL pin, if the SCL rise time takes more than 1/32 of serial clock period, then the master device sometimes suspends serial clock output or outputs impulse signal via the SCL pin.
"Rise time" is the period of time that elapses between the moment which the master device starts communication and the moment which the potential of SCL rises to 0.8V_{DD}. Therefore a period during which the slave device outputs the wait signal by keeping the SCL pin at low level although the master device is ready for communication is included in the "rise time".

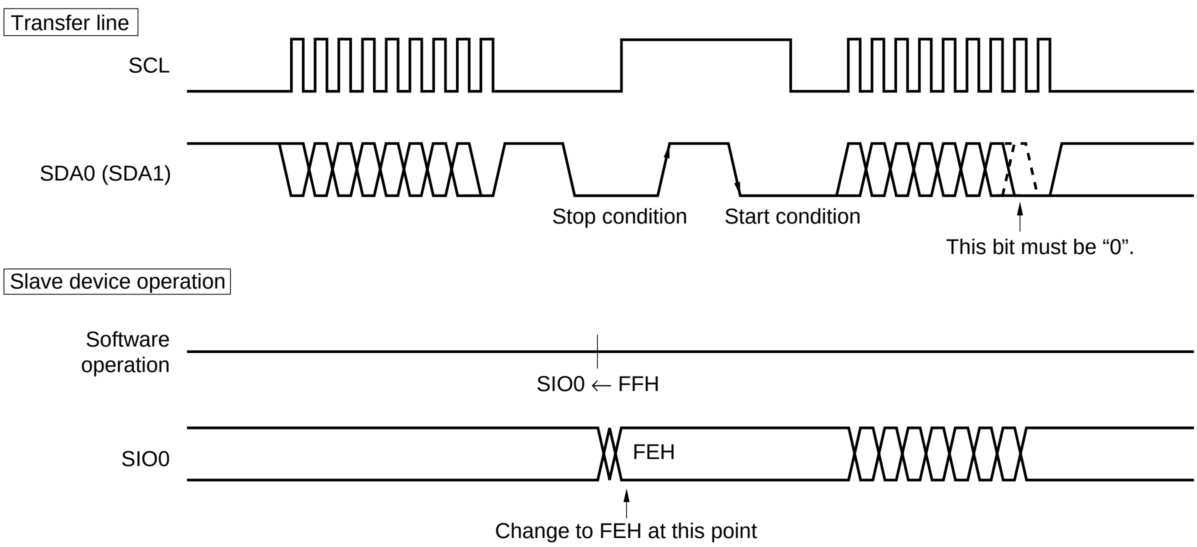
(2) Restriction on slave device operation in the I^2C bus mode

Applied devices: μPD78011BY, 78012BY, 78013Y, 78014Y, 78P014Y
IE-78014-R-EM

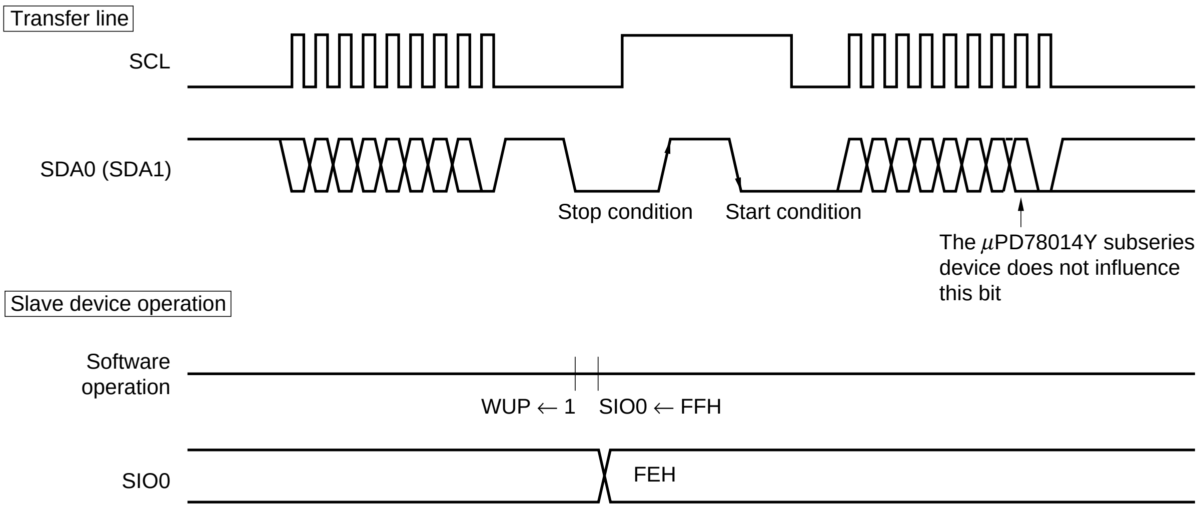
Description: If all of the following conditions are satisfied, all slave devices on the transfer line cannot transmit data.

- The μPD78014Y subseries device is used as one of the slave devices in the I^2C bus mode.
- The master device outputs the stop condition signal when it terminates transmission to the μPD78014Y Subseries device (i.e. slave reception).
- Following the master transmission operation to the μPD78014Y Subseries device (i.e. slave reception), the master reception (i.e. slave transmission) request is sent to any unit.

In the μPD78014Y Subseries, communication is started by writing data in serial I/O shift register 0 (SIO0). In data reception operation, write FFH in SIO0 to be high-impedance state the N-ch open-drain output.
After writing FFH into SIO0 of the μPD78014Y Subseries device, if the master device drives the SCL line to high level to output the start condition or stop condition signals, then SIO0 shift operation is carried out in the μPD78014Y Subseries device (slave device). As a result, written FFH is shifted and LSB of SIO0 becomes equal to the level of SDA0 (SDA1).
If the master device drives SCL to high level after driving SDA0 (SDA1) to low level to output a stop condition signal as shown in the following figure, then the contents of SIO0 change to FEH (LSB = 0) according to the above-mentioned operation. Therefore the LSB of next reception data must be "0".
The reception data which follows the start condition signal is defined as the slave address field, and the LSB of the slave address field is defined as the transfer direction specification bit. The LSB of the slave address field must be "0", so that it indicates the slave reception operation regardless of the data output from the master device.



Avoidance: If the stop condition output timing for the μPD78014Y Subseries device has been determined previously (i.e. amount of communication data between the μPD78014Y Subseries device and the master device is fixed), then it is possible to avoid this restriction by software.
Set bit 5 (WUP) of serial operating mode register 0 (CSIM0) and serial I/O shift register 0 (SIO0) of the slave device to 1 and FFH respectively before a stop condition signal is output. Then the wake-up function is enabled for the next slave address field which is sent from the master device, and the N-ch open-drain output is high-impedance state automatically. As a result, there is no influence on slave reception data.



16.4.8 $\overline{SCK0}$/SCL/P27 pin output manipulation

The $\overline{SCK0}$/SCL/P27 pin incorporates an output latch. Therefore, in addition to normal serial clock output, static output from this pin is also possible by controlling the output latch with an instruction.

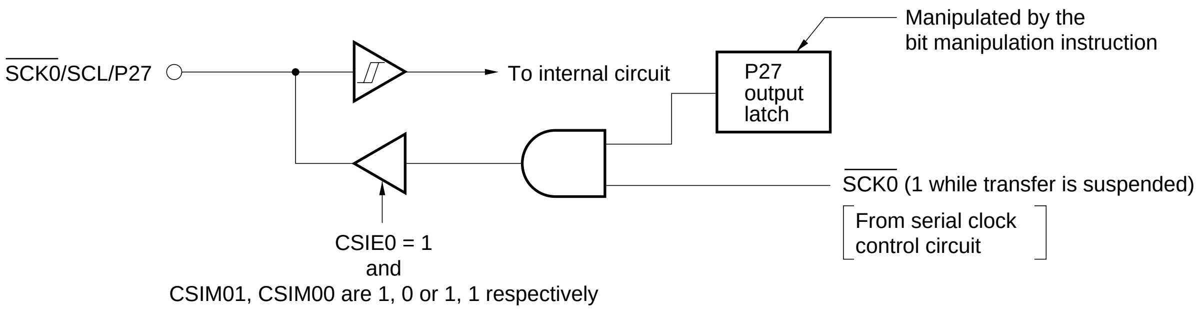
Manipulating the output latch through the software for the P27 pin, the value of serial clock can be selected by software. (SI0/SB0/SDA0 and SO0/SB1/SDA1 pins are controlled with bit 0 (RELT) or bit 1 (CMDT) of SBIC.)

The $\overline{SCK0}$/SCL/P27 pin output should be manipulated as described below.

(1) In the 3-wire serial I/O mode and the 2-wire serial I/O mode
Output level of $\overline{SCK0}$/SCL/P27 pin is manipulated by the P27 output latch.

<1> Set serial operating mode register 0 (CSIM0) ($\overline{SCK0}$ pin is set in the output mode and serial operation is enabled). While serial transfer is suspended, $\overline{SCK0}$ is set to 1.
<2> Manipulate the content of the P27 output latch by executing the bit manipulation instruction.

Figure 16-50. $\overline{SCK0}$/SCL/P27 Pin Configuration



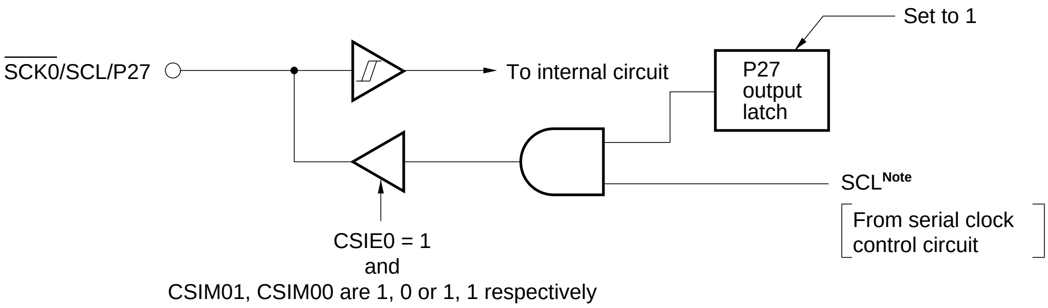
(2) In the I²C bus mode

The output level of the $\overline{\text{SCK0}}$/SCL/P27 pin is manipulated by the CLC bit of the interrupt timing specification register (SINT).

<1> Set the serial operating mode register 0 (CSIM0) (SCL pin is set in the output mode and serial operation is enabled). Set the P27 output latch to 1. While serial transfer is suspended, SCL is set to 0.

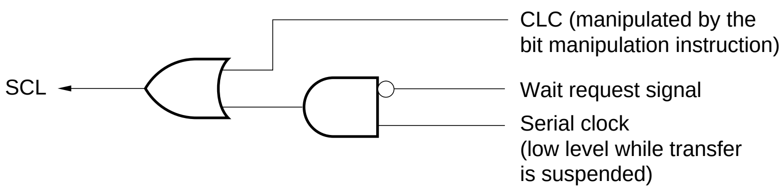
<2> Manipulate the CLC bit of SINT by executing the bit manipulation instruction.

Figure 16-51. $\overline{\text{SCK0}}$/SCL/P27 Pin Configuration



Note Level of SCL signal is determined by the following logic in the Figure 16-52.

Figure 16-52. SCL Signal Logic



Remarks 1. This figure shows the relationship between each signal and does not show the internal circuit.

2. CLC: Bit 3 of the interrupt timing specification register (SINT)

[MEMO]

CHAPTER 17 SERIAL INTERFACE CHANNEL 1

17.1 Serial Interface Channel 1 Functions

Serial interface channel 1 employs the following three modes.

- Operation stop mode
- 3-wire serial I/O mode
- 3-wire serial I/O mode with automatic transmit/receive function

(1) Operation stop mode
Operation stop 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)
3-wire serial I/O mode transfer 8-bit data with 3-wires; serial clock ($\overline{SCK1}$), serial output (SO1), and serial input (SI1).
3-wire serial I/O mode can transfer/receive at the same time, so the data transfer processing time is fast.
The start bit of 8-bit data to undergo serial transfer is switchable between MSB and LSB, so it is possible to connect to devices of any start bit.
3-wire serial I/O mode is valid for connection of peripheral I/O units and display controllers which incorporate a conventional synchronous clock serial interface as is the case with the 75X/XL, 78K and 17K Series.

(3) 3-wire serial I/O mode with automatic transmit/receive function

This mode with the automatic transmit/receive function added to (2) 3-wire serial I/O mode functions.
The automatic transmit/receive function transfers/receives up to 32-byte data. This function enables the hardware to transmit/receive data to/from the OSD (On Screen Display) device and device with on-chip display controller/driver independently of the CPU, thus the software load can be reduced.

17.2 Serial Interface Channel 1 Configuration

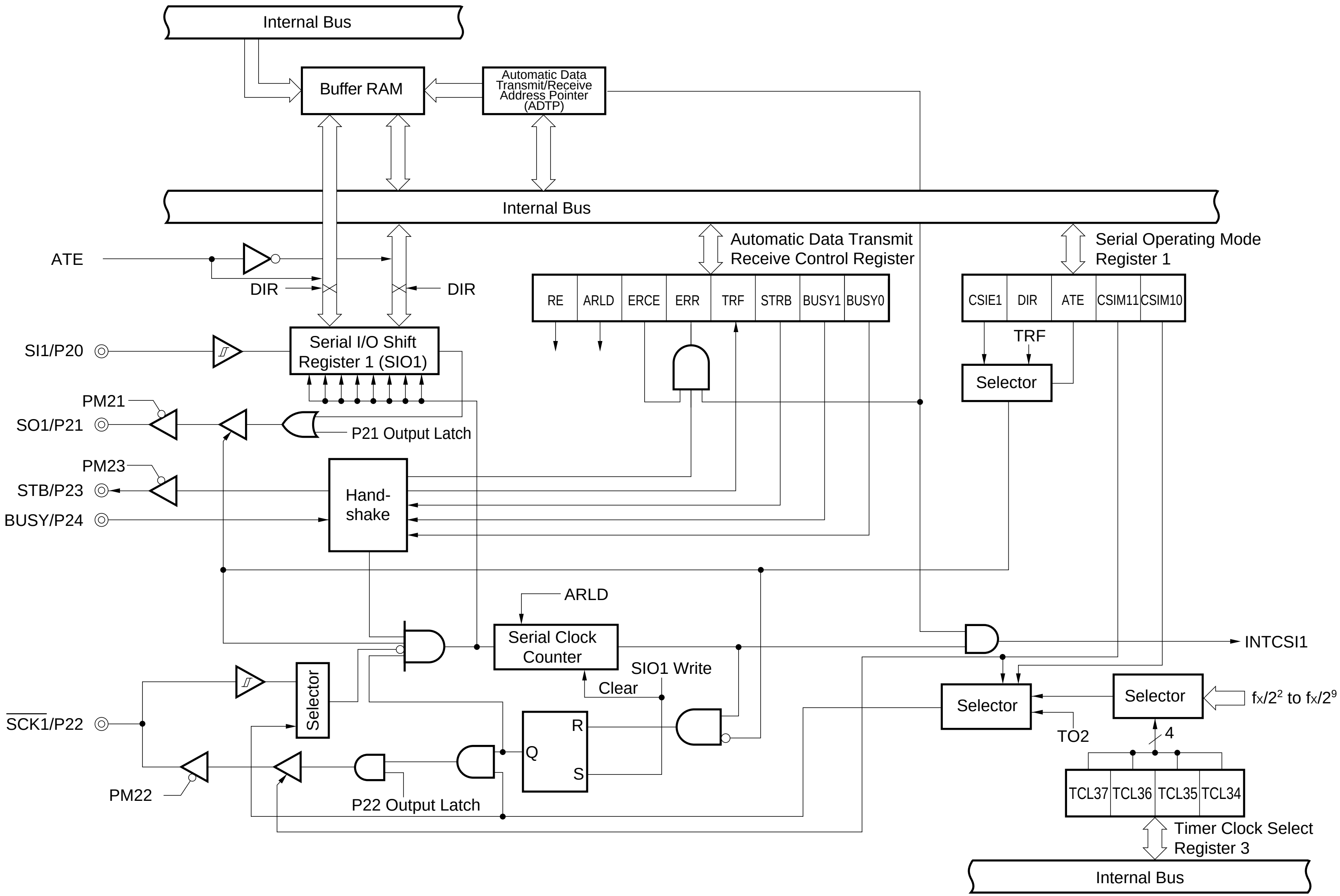
Serial interface channel 1 consists of the following hardware.

Table 17-1. Serial Interface Channel 1 Configuration

Item	Configuration
Register	Serial I/O shift register 1 (SIO1) Automatic data transmit/receive address pointer (ADTP)
Control register	Timer clock select register 3 (TCL3) Serial operating mode register 1 (CSIM1) Automatic data transmit/receive control register (ADTC) Port mode register 2 (PM2)[Note]

Note Refer to **Figures 6-6** and **6-8 P20, P21, P23 to P26 Block Diagrams** and **Figures 6-7** and **6-9 P22 and P27 Block Diagrams**.

Figure 17-1. Serial Interface Channel 1 Block Diagram



(1) Serial I/O shift register 1 (SIO1)

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.

SIO1 is set with an 8-bit memory manipulation instruction.

When value in bit 7 (CSIE1) of serial operating mode register 1 (CSIM1) is 1, writing data to SIO1 starts serial operation.

In transmission, data written to SIO1 is output to the serial output (SO1). In reception, data is read from the serial input (SI1) to SIO1.

$\overline{\text{RESET}}$ input makes SIO1 undefined.

Caution Do not write data to SIO1 while the automatic transmit/receive function is activated.

(2) Automatic data transmit/receive address pointer (ADTP)

This register stores the value of (the number of transmit data bytes – 1) while the automatic transmit/receive function is activated. It is decremented automatically with data transmission/reception.

ADTP is set with an 8-bit memory manipulation instruction. The high-order 3 bits must be set to 0.

$\overline{\text{RESET}}$ input sets ADTP to 00H.

Caution Do not write data to ADTP while the automatic transmit/receive function is activated.

(3) 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.

17.3 Serial Interface Channel 1 Control Registers

The following three types of registers are used to control serial interface channel 1.

- Timer clock select register 3 (TCL3)
- Serial operating mode register 1 (CSIM1)
- Automatic data transmit/receive control register (ADTC)

(1) Timer clock select register 3 (TCL3)
This register sets the serial clock of serial interface channel 1.
TCL3 is set with an 8-bit memory manipulation instruction.
$\overline{\text{RESET}}$ input sets TCL3 to 88H.

Remark Besides setting the serial clock of serial interface channel 1, TCL3 sets the serial clock of serial interface channel 0.

Figure 17-2. Timer Clock Select Register 3 Format

Symbol	7	6	5	4	3	2	1	0	Address	When Reset	R/W
TCL3	TCL37	TCL36	TCL35	TCL34	TCL33	TCL32	TCL31	TCL30	FF43H	88H	R/W

TCL33	TCL32	TCL31	TCL30	Serial Interface Channel 0 Serial Clock Selection
0	1	1	0	$f_X/2^2$ **Note**
0	1	1	1	$f_X/2^3$ (1.25 MHz)
1	0	0	0	$f_X/2^4$ (625 kHz)
1	0	0	1	$f_X/2^5$ (313 kHz)
1	0	1	0	$f_X/2^6$ (156 kHz)
1	0	1	1	$f_X/2^7$ (78.1 kHz)
1	1	0	0	$f_X/2^8$ (39.1 kHz)
1	1	0	1	$f_X/2^9$ (19.5 kHz)
Other than above				Setting prohibited

TCL37	TCL36	TCL35	TCL34	Serial Interface Channel 1 Serial Clock Selection
0	1	1	0	$f_X/2^2$ **Note**
0	1	1	1	$f_X/2^3$ (1.25 MHz)
1	0	0	0	$f_X/2^4$ (625 kHz)
1	0	0	1	$f_X/2^5$ (313 kHz)
1	0	1	0	$f_X/2^6$ (156 kHz)
1	0	1	1	$f_X/2^7$ (78.1 kHz)
1	1	0	0	$f_X/2^8$ (39.1 kHz)
1	1	0	1	$f_X/2^9$ (19.5 kHz)
Other than above				Setting prohibited

Note Can be set only when the main system clock oscillate at 4.19 MHz or less.

Caution If TCL3 is to be rewritten in data other than identical data, the serial transfer must be stopped first.

Remarks 1. f_X: Main system clock oscillation frequency

2. Values in parentheses apply to operation with f_X = 10.0 MHz

(2) Serial operating mode register 1 (CSIM1)

This register sets serial interface channel 1 serial clock, operating mode, operation enable/stop and automatic transmit/receive operation enable/stop.

CSIM1 is set with a 1-bit or 8-bit memory manipulation instruction.

$\overline{\text{RESET}}$ input sets CSIM1 to 00H.

Figure 17-3. Serial Operating Mode Register 1 Format

Symbol	<7>	6	<5>	4	3	2	1	0	Address	When Reset	R/W
CSIM1	CSIE1	DIR	ATE	0	0	0	CSIM11	CSIM10	FF68H	00H	R/W

CSIM11	CSIM10	Serial Interface Channel 1 Clock Selection
0	x	Clock externally input to $\overline{\text{SCK1}}$ pin[Note 1]
1	0	8-bit timer register 2 (TM2) output
1	1	Clock specified with bits 4 to 7 of timer clock select register 3 (TCL3)

ATE	Serial Interface Channel 1 Operating Mode Selection
0	3-wire serial I/O mode
1	3-wire serial I/O mode with automatic transmit/receive function

DIR	Start Bit	SI1 Pin Function	SO1 Pin Function
0	MSB	SI1/P20 (Input)	SO1 (CMOS output)
1	LSB		

<table>
<tr><th>CSIE1</th><th>CSIM11</th><th>PM20</th><th>P20</th><th>PM21</th><th>P21</th><th>PM22</th><th>P22</th><th>Shift Register 1 Operation</th><th>Serial Clock Counter Operation Control</th><th>SI1/P20 Pin Function</th><th>SO1/P21 Pin Function</th><th>$\overline{\text{SCK1}}$/P22 Pin Function</th></tr>
<tr><td>0</td><td>×</td><td>×[Note 2]</td><td>×[Note 2]</td><td>×[Note 2]</td><td>×[Note 2]</td><td>×[Note 2]</td><td>×[Note 2]</td><td>Operation stop</td><td>Clear</td><td>P20 (CMOS input/output)</td><td>P21 (CMOS input/output)</td><td>P22 (CMOS input/output)</td></tr>
<tr><td rowspan="2">1</td><td>0</td><td rowspan="2">1[Note 3]</td><td rowspan="2">×[Note 3]</td><td rowspan="2">0</td><td rowspan="2">0</td><td>1</td><td>×</td><td rowspan="2">Operation enable</td><td rowspan="2">Count operation</td><td rowspan="2">SI1[Note 3] (Input)</td><td rowspan="2">SO1 (CMOS output)</td><td>$\overline{\text{SCK1}}$ (Input)</td></tr>
<tr><td>1</td><td>0</td><td>1</td><td>$\overline{\text{SCK1}}$ (CMOS output)</td></tr>
</table>

Notes **1.** If the external clock input has been selected with CSIM11 set to 0, set bit 1 (BUSY1) and bit 2 (STRB) of the automatic data transmit/receive control register (ADTC) to 0 and 0, respectively.

2. Can be used freely as port function.

3. Can be used as P20 (CMOS input/output) when only transmitter is used (Set bit 7 (RE) of ADTC to 0).

Remark × : don't care

PMxx : Port mode register

Pxx : Output latch of port

(3) Automatic data transmit/receive control register (ADTC)

This register sets automatic receive enable/disable, the operating mode, strobe output enable/disable, busy input enable/disable, error check enable/disable, and displays automatic transmit/receive execution and error detection.

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

$\overline{\text{RESET}}$ input sets ADTC to 00H.

Figure 17-4. Automatic Data Transmit/Receive Control Register Format

Symbol	<7>	<6>	<5>	<4>	<3>	<2>	<1>	<0>	Address	When Reset	R/W
ADTC	RE	ARLD	ERCE	ERR	TRF	STRB	BUSY1	BUSY0	FF69H	00H	R/W[Note 1]

R/W

BUSY1	BUSY0	Busy Input Control
0	×	Not using busy input
1	0	Busy input enable (active high)
1	1	Busy input enable (active low)

R/W

STRB	Strobe Output Control
0	Strobe output disable
1	Strobe output enable

R

TRF	Status of Automatic Transmit/Receive Function[Note 2]
0	Detection of termination of automatic transmission/reception (This bit is set to 0 upon suspension of automatic transmission/reception or when ARLD = 0)
1	During automatic transmission/reception (This bit is set to 1 when data is written to SIO1)

R

ERR	Error Detection of Automatic Transmit/ Receive Function
0	No error in automatic transmission/reception (This bit is set to 0 when data is written to SIO1)
1	Error occurred in automatic transmission/ reception

R/W

ERCE	Error Check Control of Automatic Transmit/ Receive Function
0	Error check disable in automatic transmission/reception
1	Error check enable (only when BUSY1 = 1)

R/W

ARLD	Operating Mode Selection of Automatic Transmit/Receive Function
0	Single operating mode
1	Repetitive operating mode

R/W

RE	Receive Control of Automatic Transmit/ Receive Function
0	Receive disable
1	Receive enable

Notes 1. Bits 3 and 4 (TRF and ERR) are read-only bits.

2. The termination of automatic transmission/reception should be judged by using TRF, not CSIIF1 (interrupt request flag).

Caution When an external clock input is selected with bit 1 (CSIM11) of serial operating mode register 1 (CSIM1) set to 0, set STRB and BUSY1 of ADTC to 0, 0.

Remark ×: don't care

17.4 Serial Interface Channel 1 Operations

The following three operating modes are available to the serial interface channel 1.

- Operation stop mode
- 3-wire serial I/O mode
- 3-wire serial I/O mode with automatic transmit/receive function

17.4.1 Operation stop mode

Serial transfer is not carried out in the operation stop mode. Thus, power consumption can be reduced. The serial I/O shift register 1 (SIO1) does not carry out shift operation either, and thus it can be used as a normal 8-bit register.

In the operation stop mode, the P20/SI1, P21/SO1, P22/$\overline{\text{SCK1}}$, P23/STB and P24/BUSY pins can be used as normal input/output ports.

(1) Register setting

The operation stop mode is set with the serial operating mode register 1 (CSIM1).
CSIM1 is set with a 1-bit or 8-bit memory manipulation instruction.
$\overline{\text{RESET}}$ input sets CSIM1 to 00H.

Symbol	<7>	6	<5>	4	3	2	1	0	Address	When Reset	R/W
CSIM1	CSIE1	DIR	ATE	0	0	0	CSIM11	CSIM12	FF68H	00H	R/W

<table>
<tr><th>CSIE1</th><th>CSIM11</th><th>PM20</th><th>P20</th><th>PM21</th><th>P21</th><th>PM22</th><th>P22</th><th>Shift Register 1 Operation</th><th>Serial Clock Counter Operation Control</th><th>SI1/P20 Pin Function</th><th>SO1/P21 Pin Function</th><th>$\overline{\text{SCK1}}$/P22 Pin Function</th></tr>
<tr><td>0</td><td>×</td><td>× Note 1</td><td>× Note 1</td><td>× Note 1</td><td>× Note 1</td><td>× Note 1</td><td>× Note 1</td><td>Operation stop</td><td>Clear</td><td>P20 (CMOS input/output)</td><td>P21 (CMOS input/output)</td><td>P22 (CMOS input/output)</td></tr>
<tr><td rowspan="2">1</td><td>0</td><td rowspan="2">1 Note 2</td><td rowspan="2">× Note 2</td><td rowspan="2">0</td><td rowspan="2">0</td><td>1</td><td>×</td><td rowspan="2">Operation enable</td><td rowspan="2">Count operation</td><td rowspan="2">SI1 Note 2 (Input)</td><td rowspan="2">SO1 (CMOS output)</td><td>$\overline{\text{SCK1}}$ (Input)</td></tr>
<tr><td>1</td><td>0</td><td>1</td><td>$\overline{\text{SCK1}}$ (CMOS output)</td></tr>
</table>

Notes 1. Can be used freely as port function.

2. Can be used as P20 (CMOS input/output) when only transmitter is used. (Set bit 7 (RE) of the automatic data transmit/receive control register (ADTC) to 0.)

Remark × : don’t care
PMxx : Port mode register
Pxx : Output latch of port

17.4.2 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 serial interface as is the case with the 75X/XL, 78K and 17K series.

Communication is carried out with three lines of serial clock ($\overline{\text{SCK1}}$), serial output (SO1) and serial input (SI1).

(1) Register setting

The 3-wire serial I/O mode is set with the serial operating mode register 1 (CSIM1).

CSIM1 is set with a 1-bit or 8-bit memory manipulation instruction.

$\overline{\text{RESET}}$ input sets CSIM1 to 00H.

Symbol	<7>	6	<5>	4	3	2	1	0	Address	When Reset	R/W
CSIM1	CSIE1	DIR	ATE	0	0	0	CSIM11	CSIM10	FF68H	00H	R/W

CSIM11	CSIM10	Serial Interface Channel 1 Clock Selection
0	x	Clock externally input to $\overline{\text{SCK1}}$ pin[Note 1]
1	0	8-bit timer register 2 (TM2) output
1	1	Clock specified with bits 4 to 7 of timer clock select register 3 (TCL3)

ATE	Serial Interface Channel 1 Operating Mode Selection
0	3-wire serial I/O mode
1	3-wire serial I/O mode with automatic transmit/receive function

DIR	Start Bit	SI1 Pin Function	SO1 Pin Function
0	MSB	SI1/P20 (Input)	SO1 (CMOS output)
1	LSB		

CSIE1	CSIM11	PM20	P20	PM21	P21	PM22	P22	Shift Register 1 Operation	Serial Clock Counter Operation Control	SI1/P20 Pin Function	SO1/P21 Pin Function	$\overline{\text{SCK1}}$/P22 Pin Function
0	×	×[Note 2]	×[Note 2]	×[Note 2]	×[Note 2]	×[Note 2]	×[Note 2]	Operation stop	Clear	P20 (CMOS input/output)	P21 (CMOS input/output)	P22 (CMOS input/output)
1	0	1[Note 3]	×[Note 3]	0	0	1	×	Operation enable	Count operation	SI1[Note 3] (Input)	SO1 (CMOS output)	$\overline{\text{SCK1}}$ (Input)
	1					0	1					$\overline{\text{SCK1}}$ (CMOS output)

Notes **1.** If the external clock input has been selected with CSIM11 set to 0, set bit 1 (BUSY1), or bit 2 (STRB) of the automatic data transmit/receive control register (ADTC) to 0, 0.

2. Can be used freely as port function.

3. Can be used as P20 (CMOS input/output) when only transmitter is used (Set bit 7 (RE) of ADTC to 0).

Remark × : don't care

PMxx : Port mode register

Pxx : Output latch of port

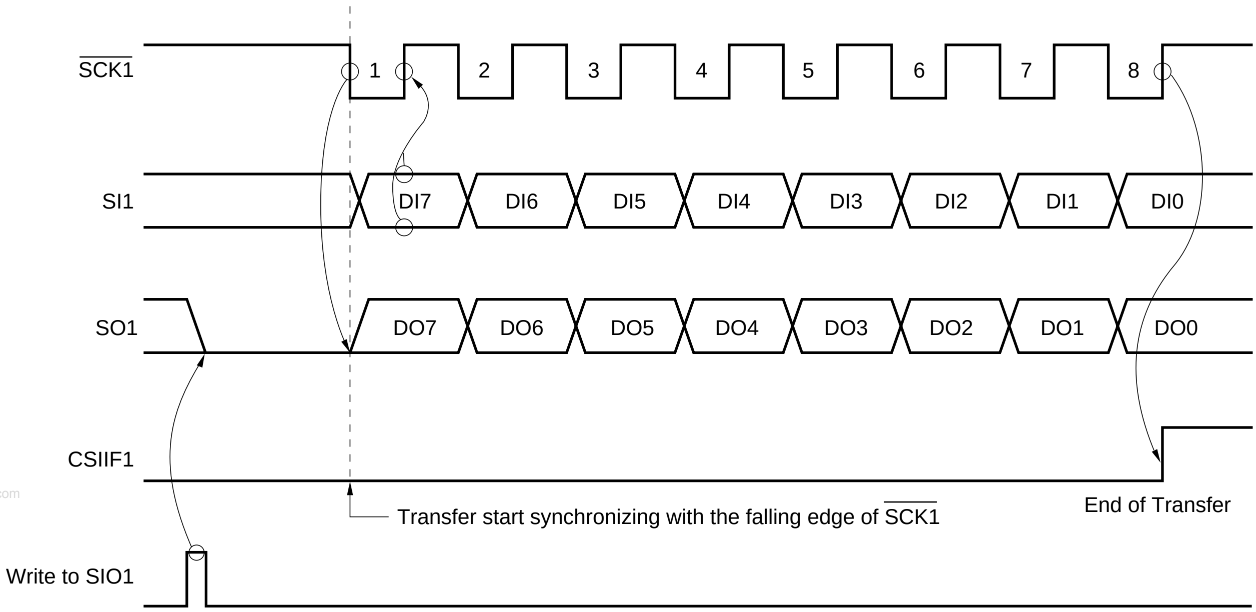
(2) Communication operation

The 3-wire serial I/O mode is used for data transmission/reception in 8-bit units. Data transmission/reception is carried out bit-wise in synchronization with the serial clock.

Shift operation of the serial I/O shift register 1 (SIO1) is carried out at the falling edge of the serial clock ($\overline{SCK1}$). The transmit data is held in the SO1 latch and is output from the SO1 pin. The receive data input to the SI1 pin is latched into SIO1 at the rising edge of $\overline{SCK1}$.

Upon termination of 8-bit transfer, the SIO1 operation stops automatically and the interrupt request flag (CSIIF1) is set.

Figure 17-5. 3-Wire Serial I/O Mode Timings



Caution SO1 pin will be low by writing to SIO1.

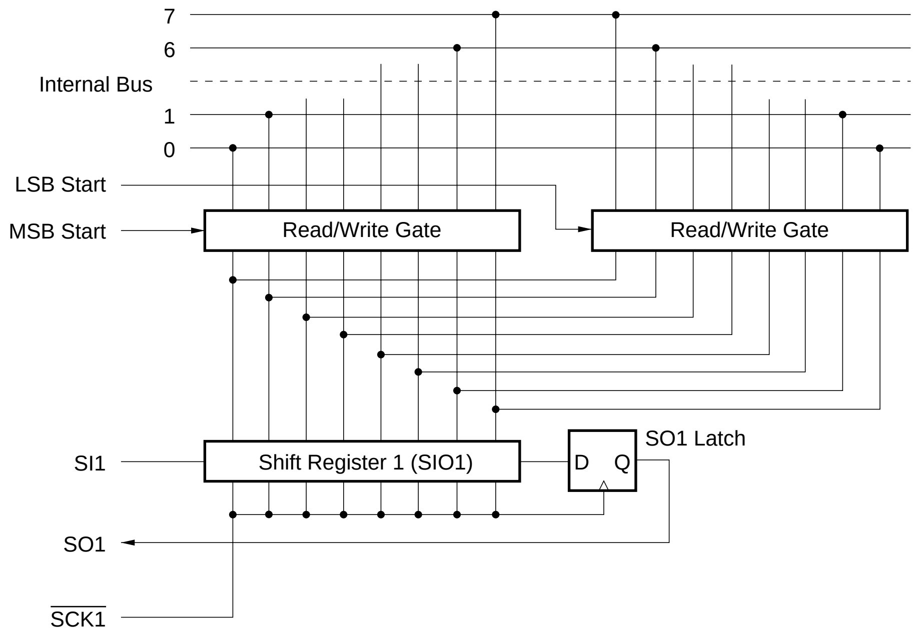
★ (3) MSB/LSB switching as the start bit

The 3-wire serial I/O mode enables to select transfer to start at MSB or LSB.

Figure 17-6 shows the configuration of the serial I/O shift register 1 (SIO1) and internal bus. As shown in the figure, MSB/LSB can be read/written in inverted form.

MSB/LSB switching as the start bit can be specified with bit 6 (DIR) of the serial operating mode register 1 (CSIM1).

Figure 17-6. Circuit of Switching in Transfer Bit Order



Start bit switching is realized by switching the bit order for data write to SIO1. The SIO1 shift order remains unchanged.

Thus, switch the MSB/LSB start bit before writing data to the shift register.

(4) Start of transfer

A serial transfer is started by setting transfer data in the serial I/O shift register 1 (SIO1) if the following two conditions have been satisfied:

- The serial interface channel 1 operation control bit (CSIE1) = 1.
- After an 8-bit serial transfer, the internal serial clock is stopped or $\overline{SCK1}$ is high.

Caution Setting CSIE1 to 1 after writing data in SIO1 does not initiate transfer operation.

When an 8-bit data transfer ends, serial transfer is stopped automatically and the interrupt request flag (CSIIF1) is set.

17.4.3 3-wire serial I/O mode operation with automatic transmit/receive function

This 3-wire serial I/O mode is used for transmission/reception of a maximum of 32-byte data without the use of software. Once transfer is started, the data prestored in the RAM can be transmitted by the set number of bytes, and data can be received and stored in the RAM by the set number of bytes.

Handshake signals (STB and BUSY) are supported by hardware to transmit/receive data continuously. OSD (On Screen Display) LSI and peripheral LSI including LCD controller/driver can be connected without difficulty.

(1) Register setting

The 3-wire serial I/O mode with automatic transmit/receive function is set with the serial operating mode register 1 (CSIM1) and the automatic data transmit/receive control register (ADTC).

(a) Serial operating mode register 1 (CSIM1)

CSIM1 is set with a 1-bit or 8-bit memory manipulation instruction.

$\overline{\text{RESET}}$ input sets CSIM1 to 00H.

Symbol	<7>	6	<5>	4	3	2	1	0	Address	When Reset	R/W
CSIM1	CSIE1	DIR	ATE	0	0	0	CSIM11	CSIM10	FF68H	00H	R/W

CSIM11	CSIM10	Serial Interface Channel 1 Clock Selection
0	×	Clock externally input to $\overline{SCK1}$ pin[Note 1]
1	0	8-bit timer register 2 (TM2) output
1	1	Clock specified with bits 4 to 7 of timer clock select register 3 (TCL3)

ATE	Serial Interface Channel 1 Operating Mode Selection
0	3-wire serial I/O mode
1	3-wire serial I/O mode with automatic transmit/receive function

<table>
<tr><th>DIR</th><th>Start Bit</th><th>SI1 Pin Function</th><th>SO1 Pin Function</th></tr>
<tr><td>0</td><td>MSB</td><td rowspan="2">SI1/P20 (Input)</td><td rowspan="2">SO1 (CMOS output)</td></tr>
<tr><td>1</td><td>LSB</td></tr>
</table>

<table>
<tr><th>CSIE1</th><th>CSIM11</th><th>PM20</th><th>P20</th><th>PM21</th><th>P21</th><th>PM22</th><th>P22</th><th>Shift Register 1 Operation</th><th>Serial Clock Counter Operation Control</th><th>SI1/P20 Pin Function</th><th>SO1/P21 Pin Function</th><th>$\overline{SCK1}$/P22 Pin Function</th></tr>
<tr><td>0</td><td>×</td><td>Note 2
×</td><td>Note 2
×</td><td>Note 2
×</td><td>Note 2
×</td><td>Note 2
×</td><td>Note 2
×</td><td>Operation stop</td><td>Clear</td><td>P20
(CMOS input/output)</td><td>P21
(CMOS input/output)</td><td>P22
(CMOS input/output)</td></tr>
<tr><td rowspan="2">1</td><td>0</td><td rowspan="2">Note 3
1</td><td rowspan="2">Note 3
×</td><td rowspan="2">0</td><td rowspan="2">0</td><td>1</td><td>×</td><td rowspan="2">Operation enable</td><td rowspan="2">Count operation</td><td rowspan="2">SI1[Note 3]
(Input)</td><td rowspan="2">SO1
(CMOS output)</td><td>$\overline{SCK1}$
(Input)</td></tr>
<tr><td>1</td><td>0</td><td>1</td><td>$\overline{SCK1}$
(CMOS output)</td></tr>
</table>

Notes
1. If the external clock input has been selected with CSIM11 set to 0, set bit 1 (BUSY1) and bit 2 (STRB) of the automatic data transmit/receive control register (ADTC) to 0 and 0, respectively.
2. Can be used freely as port function.
3. Can be used as P20 (CMOS input/output) when only transmitter is used (Set bit 7 (RE) of ADTC to 0).

Remark × : don't care
PMxx : Port mode register
Pxx : Output latch of port

(b) Automatic data transmit/receive control register (ADTC)
ADTC is set with a 1-bit or 8-bit memory manipulation instruction.
$\overline{RESET}$ input sets ADTC to 00H.

Symbol	<7>	<6>	<5>	<4>	<3>	<2>	<1>	<0>	Address	When Reset	R/W
ADTC	RE	ARLD	ERCE	ERR	TRF	STRB	BUSY1	BUSY0	FF69H	00H	R/W[Note 1]

R/W

BUSY1	BUSY0	Busy Input Control
0	×	Not using busy input
1	0	Busy input enable (active high)
1	1	Busy input enable (active low)

R/W

STRB	Strobe Output Control
0	Strobe output disable
1	Strobe output enable

R

TRF	Status of Automatic Transmit/Receive Function[Note 2]
0	Detection of termination of automatic transmission/reception (This bit is set to 0 upon suspension of automatic transmission/reception or when ARLD = 0)
1	During automatic transmission/reception (This bit is set to 1 when data is written to SIO1)

R

ERR	Error Detection of Automatic Transmit/Receive Function
0	No error in automatic transmission/reception (This bit is set to 0 when data is written to SIO1)
1	Error occurred in automatic transmission/reception

R/W

ERCE	Error Check Control of Automatic Transmit/Receive Function
0	Error check disable in automatic transmission/reception
1	Error check enable (only when BUSY1 = 1)

R/W

ARLD	Operating Mode Selection of Automatic Transmit/Receive Function
0	Single operating mode
1	Repetitive operating mode

R/W

RE	Receive Control of Automatic Transmit/Receive Function
0	Receive disable
1	Receive enable

Notes 1. Bits 3 and 4 (TRF and ERR) are read-only bits.

2. The termination of automatic transmission/reception should be judged by using TRF, not CSIIF1 (interrupt request flag).

Caution When an external clock input is selected with bit 1 (CSIM11) of the serial operating mode register 1 (CSIM1) set to 0, set STRB and BUSY1 of ADTC to 0, 0 (handshake control cannot be performed when an external clock is input).

Remark ×: don't care

(2) Automatic transmit/receive data setting

(a) Transmit data setting

<1> Write transmit data from the least significant address FAC0H of buffer RAM (up to FADFH at maximum). However, the transmit data should be in the order from high-order address to low-order address.
<2> Set to the automatic data transmit/receive address pointer (ADTP) the value obtained by subtracting 1 from the number of transmit data bytes.

(b) Automatic transmit/receive mode setting

<1> Set CSIE1 and ATE of the serial operating mode register 1 (CSIM1) to 1, 1.
<2> Set RE of the automatic data transmit/receive control register (ADTC) to 1.
<3> Write any value to the serial I/O shift register 1 (SIO1) (transfer start trigger).

Caution Writing any value to SIO1 orders the start of automatic transmit/receive operation and the written value has no meaning.

The following operations are automatically carried out when (a) and (b) are carried out.

- After the buffer RAM data specified with ADTP is transferred to SIO1, transmission is started (start of automatic transmit/receive operation).
- The received data is written to the buffer RAM address specified with ADTP.
- ADTP is decremented and the next data transmission/reception is carried out. Data transmission/reception continues until the ADTP decremental output becomes 00H and address FAC0H data is output (end of automatic transmit/receive operation).
- When automatic transmit/receive operation is terminated, TRF is cleared to 0.

(3) Communication operation

(a) Basic transmit/receive mode

This transmit/receive mode is the same as the 3-wire serial I/O mode in which specified number of data are transmitted/received in 8-bit units.

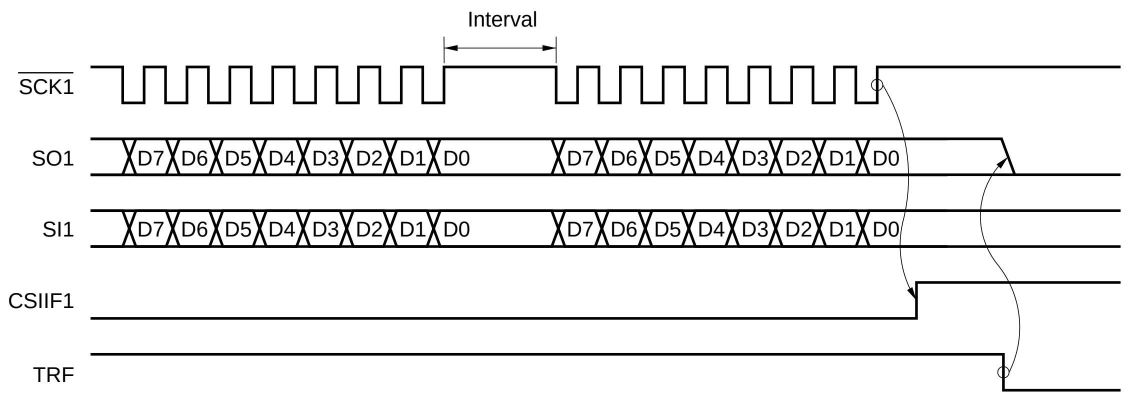
Serial transfer starts when any data is written to the serial I/O shift register 1 (SIO1) while the serial operating mode register 1 (CSIM1) bit 7 (CSIE1) is set to 1.

Upon completion of transmission of the last byte, the interrupt request flag (CSIIF1) is set. However, the termination of automatic transmission/reception should be judged by using bit 3 (TRF) of the automatic data transmit/receive control register (ADTC), not CSIIF1.

If busy control and strobe control are not executed, the P23/STB and P24/BUSY pins can be used as normal input/output ports.

Figure 17-7 shows the basic transmit/receive mode operation timings, and Figure 17-8 shows the operation flowchart. In addition, Figure 17-9 shows the buffer RAM operation in 6-byte transmission/reception.

Figure 17-7. Basic Transmit/Receive Mode Operation Timings



Cautions 1. Because, in the basic transmit/receive mode, the automatic transmit/receive function writes/reads data to/from the buffer RAM after 1-byte transmission/reception, an interval is inserted until the next transmission/reception.

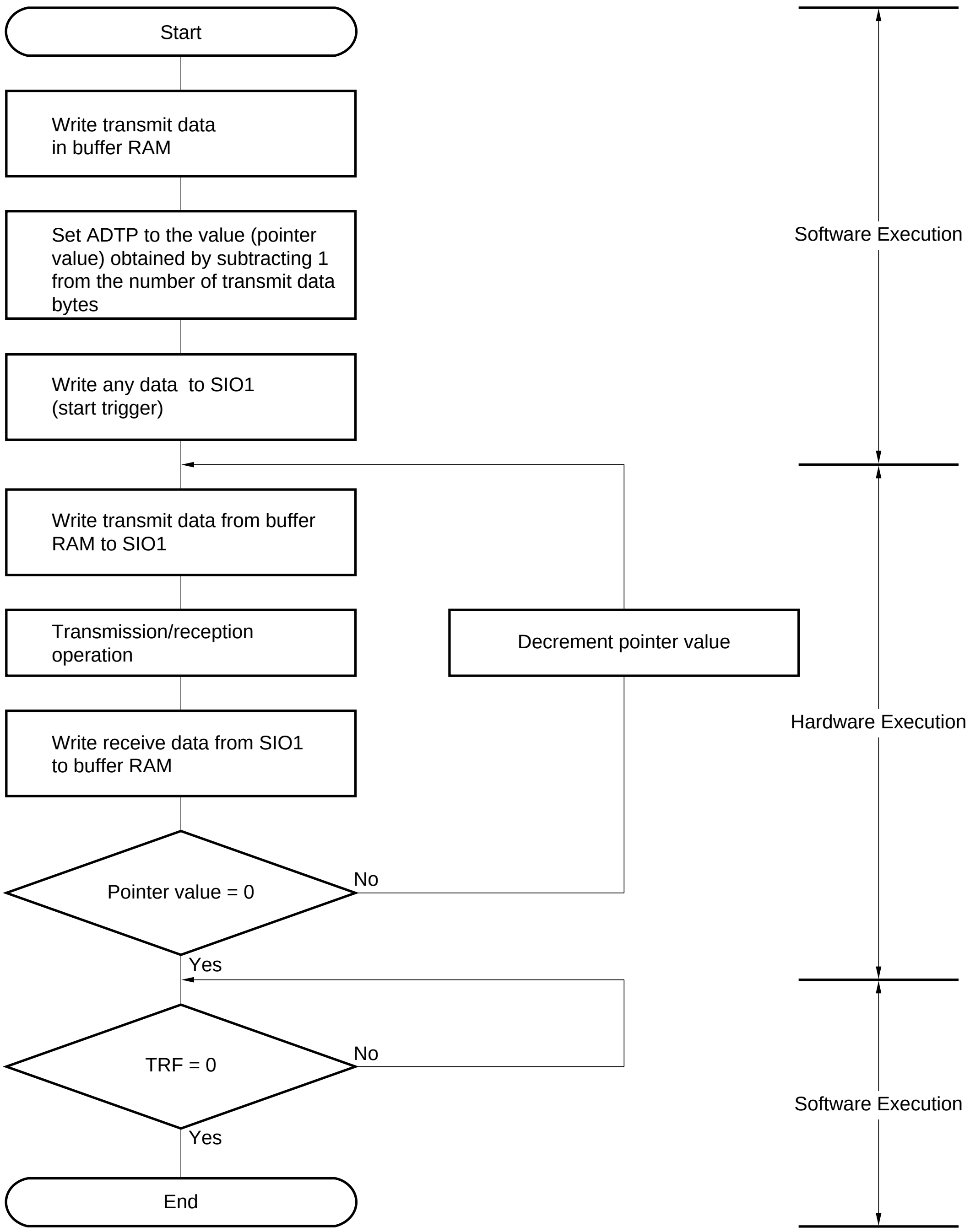
As the buffer RAM write/read is performed at the same time as CPU processing, the maximum interval is dependent upon CPU processing (see (5) Automatic data transmit/receive interval).

2. When TRF is cleared, the SO1 pin becomes low.

Remark CSIIF1 : Interrupt request flag

TRF : Bit 3 of the automatic data transmit/receive control register (ADTC)

Figure 17-8. Basic Transmit/Receive Mode Flowchart



ADTP : Automatic data transmit/receive address pointer
SIO1 : Serial I/O shift register 1
TRF : Bit 3 of automatic data transmit/receive control register (ADTC)

In 6-byte transmission/reception (ARLD = 0, RE = 1) in basic transmit/receive mode, buffer RAM operates as follows.

(i) Before transmission/reception (Refer to **Figure 17-9 (a)**)
After any data has been written to serial I/O shift register 1 (SIO1) (start trigger: this data is not transferred), transmit data 1 (T1) is transferred from the buffer RAM to SIO1. When transmission of the first byte is completed, the receive data 1 (R1) is transferred from SIO1 to the buffer RAM, and automatic data transmit/receive address pointer (ADTP) is decremented. Then transmit data 2 (T2) is transferred from the buffer RAM to SIO1.

(ii) 4th byte transmission/reception point (Refer to **Figure 17-9 (b)**)
Transmission/reception of the third byte is completed, and transmit data 4 (T4) is transferred from the buffer RAM to SIO1. When transmission of the fourth byte is completed, the receive data 4 (R4) is transferred from SIO1 to the buffer RAM, and ADTP is decremented.

(iii) Completion of transmission/reception (Refer to **Figure 17-9 (c)**)
When transmission of the sixth byte is completed, the receive data 6 (R6) is transferred from SIO1 to the buffer RAM, and the interrupt request flag (CSIIF1) is set (INTCSI1 generation).

Figure 17-9. Buffer RAM Operation in 6-Byte Transmission/Reception (in Basic Transmit/Receive Mode) (1/2)

(a) Before transmission/reception

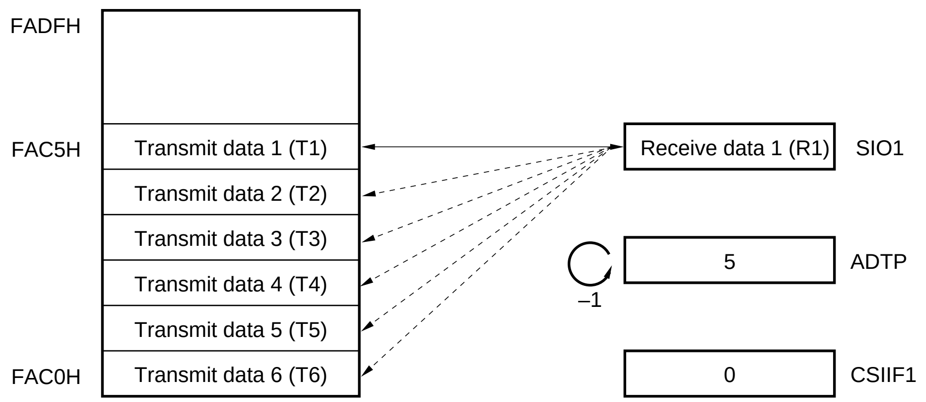
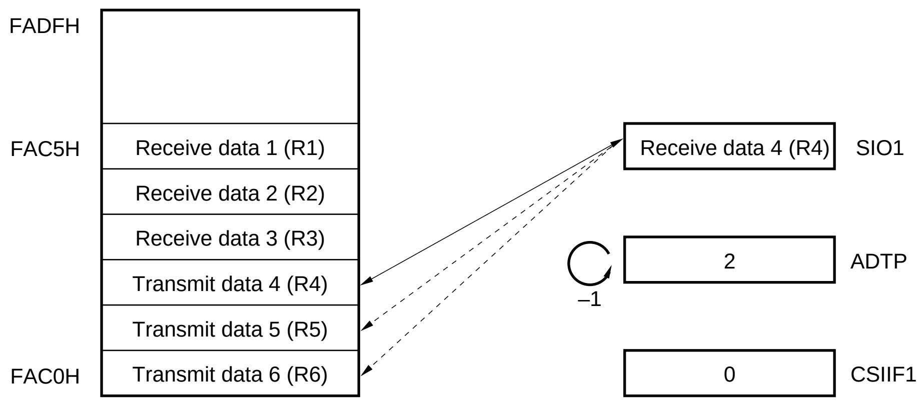
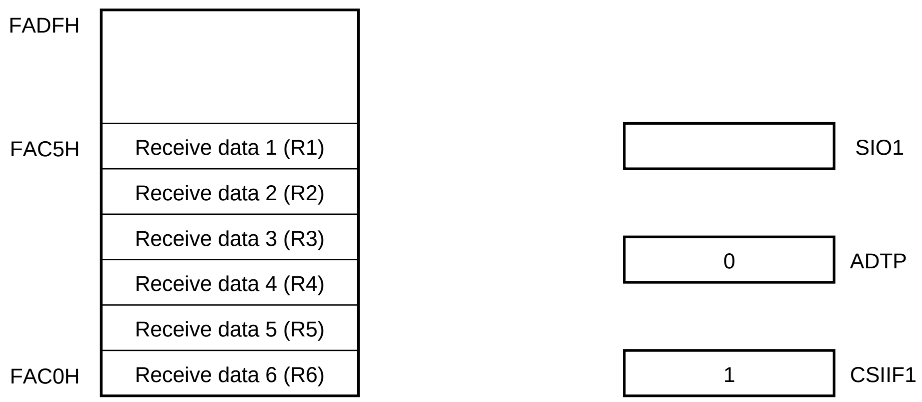


Figure 17-9. Buffer RAM Operation in 6-Byte Transmission/Reception (in Basic Transmit/Receive Mode) (2/2)

(b) 4th byte transmission/reception point



(c) Completion of transmission/reception



(b) Basic transmit mode

In this mode, the specified number of 8-bit unit data are transmitted.

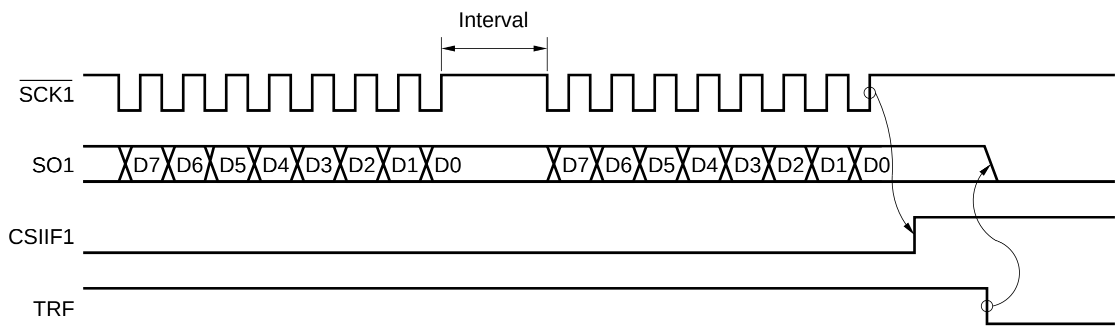
Serial transfer starts when any data is written to the serial I/O shift register 1 (SIO1) while the serial operating mode register 1 (CSIM1) bit 7 (CSIE1) is set to 1.

Upon completion of transmission of the last byte, the interrupt request flag (CSIIF1) is set. However, the termination of automatic transmission/reception should be judged by using bit 3 (TRF) of the automatic data transmit/receive control register (ADTC), not CSIIF1.

If receive operation, busy control and strobe control are not executed, the P20/SI1, P23/STB and P24/BUSY pins can be used as normal input/output ports.

Figure 17-10 shows the basic transmission mode operation timings, and Figure 17-11 shows the operation flowchart. In addition, Figure 17-12 shows the buffer RAM operation in 6-byte transmission.

Figure 17-10. Basic Transmit Mode Operation Timings



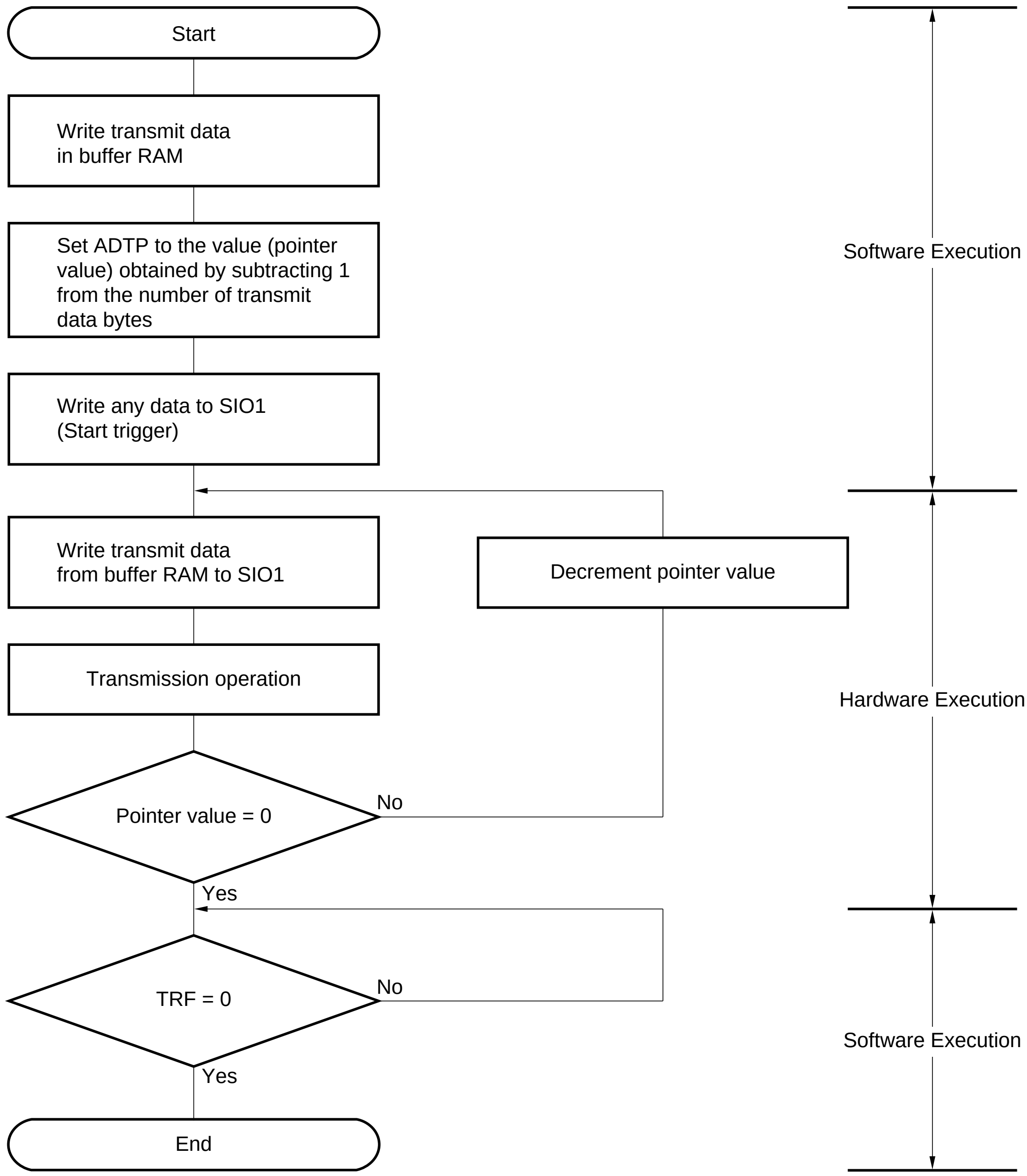
Cautions 1. Because, in the basic transmit mode, the automatic transmit/receive function reads data from the buffer RAM after 1-byte transmission, an interval is inserted until the next transmission. As the buffer RAM read is performed at the same time as CPU processing, the maximum interval is dependent upon CPU processing (see (5) Automatic data transmit/receive interval).

2. When TRF is cleared, the SO1 pin becomes low.

Remark CSIIF1 : Interrupt request flag

TRF : Bit 3 of the automatic data transmit/receive control register (ADTC)

Figure 17-11. Basic Transmit Mode Flowchart



ADTP : Automatic data transmit/receive address pointer
SIO1 : Serial I/O shift register 1
TRF : Bit 3 of automatic data transmit/receive control register (ADTC)

In 6-byte transmission (ARLD = 0, RE = 0) in basic transmit mode, buffer RAM operates as follows.

(i) Before transmission (Refer to **Figure 17-12 (a)**)
After any data has been written to serial I/O shift register 1 (SIO1) (start trigger: this data is not transferred), transmit data 1 (T1) is transferred from the buffer RAM to SIO1. When transmission of the first byte is completed, automatic data transmit/receive address pointer (ADTP) is decremented. Then transmit data 2 (T2) is transferred from the buffer RAM to SIO1.

(ii) 4th byte transmission point (Refer to **Figure 17-12 (b)**)
Transmission of the third byte is completed, and transmit data 4 (T4) is transferred from the buffer RAM to SIO1. When transmission of the fourth byte is completed, ADTP is decremented.

(iii) Completion of transmission (Refer to **Figure 17-12 (c)**)
When transmission of the sixth byte is completed, the interrupt request flag (CSIIF1) is set (INTCSI1 generation).

Figure 17-12. Buffer RAM Operation in 6-Byte Transmission (in Basic Transmit Mode) (1/2)

(a) Before transmission

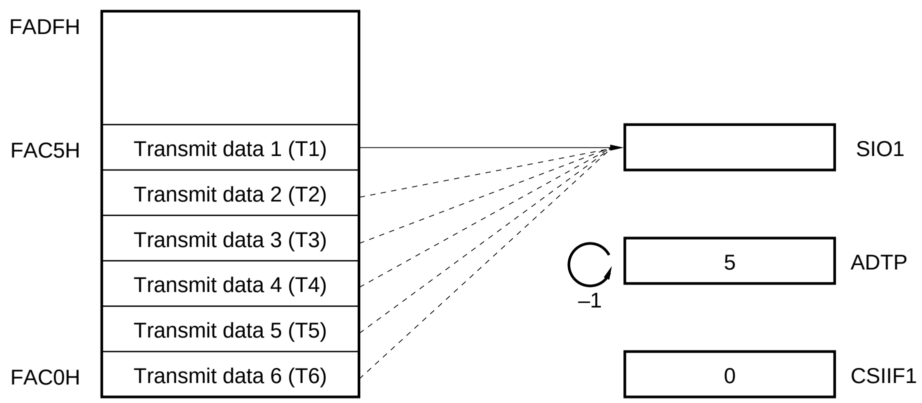
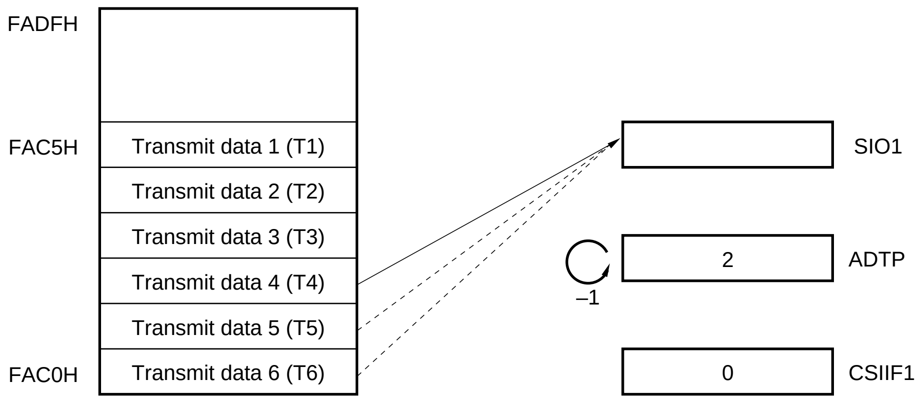
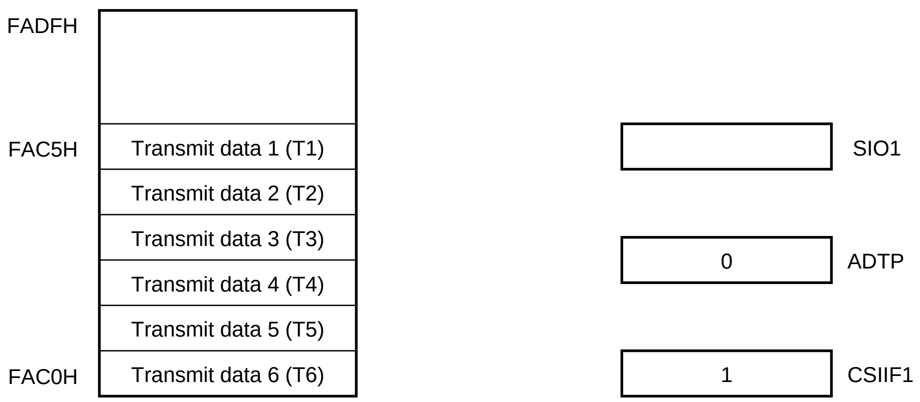


Figure 17-12. Buffer RAM Operation in 6-Byte Transmission (in Basic Transmit Mode) (2/2)

(b) 4th byte transmission point



(c) Completion of transmission



(c) Repeat transmit mode

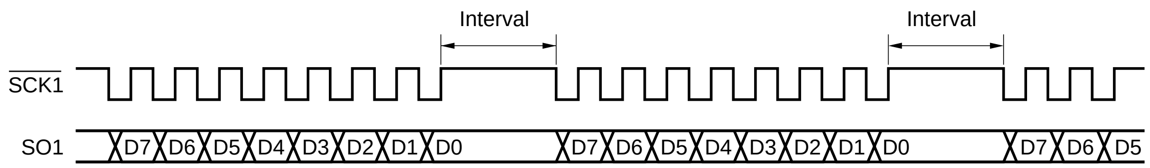
In this mode, data stored in the buffer RAM is transmitted repeatedly.

Serial transfer starts by writing any data to the serial I/O shift register 1 (SIO1) when 1 is set in the serial operating mode register 1 (CSIM1) bit 7 (CSIE1).

Unlike the basic transmit mode, after the last byte (data in address FAC0H) has been transmitted, the interrupt request flag (CSIIF1) is not set, the value at the time when the transmission was started is set in the automatic data transmit/receive address pointer (ADTP) again, and the buffer RAM contents are transmitted again. When a reception operation, busy control and strobe control are not performed, the P20/SI1, P23/STB and P24/BUSY pins can be used as normal input/output ports.

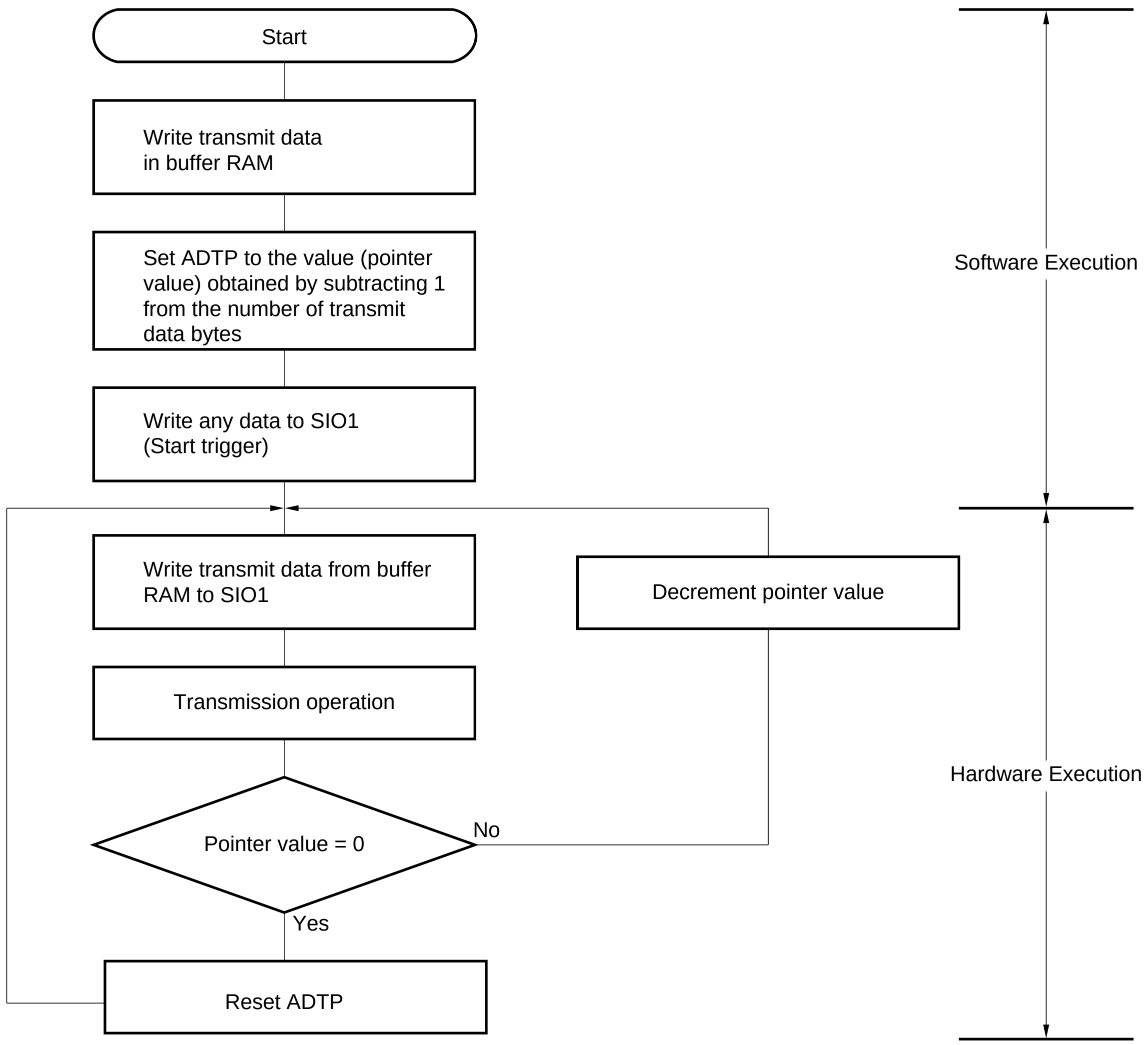
The repeat transmission mode operation timing is shown in Figure 17-13, and the operation flowchart in Figure 17-14. In addition, buffer RAM operation in 6-byte transmission in the repeat transmit mode is shown in Figure 17-15.

Figure 17-13. Repeat Transmit Mode Operation Timings



Caution Because, in the repeat transmission mode, the automatic transmit/receive function reads data from the buffer RAM after 1-byte transmission, an interval is inserted until the next transmission. As the buffer RAM read is performed at the same time as CPU processing, the maximum interval is dependent upon CPU processing (see (5) Automatic data transmit/receive interval).

Figure 17-14. Repeat Transmit Mode Flowchart



ADTP : Automatic data transmit/receive address pointer
SIO1 : Serial I/O shift register 1

In 6-byte transmission (ARLD = 1, RE = 0) in the repeat transmit mode, buffer RAM operates as follows.

(i) Before transmission (Refer to **Figure 17-15 (a)**)
After any data has been written to serial I/O shift register 1 (SIO1) (start trigger: this data is not transferred), transmit data 1 (T1) is transferred from the buffer RAM to SIO1. When transmission of the first byte is completed, automatic data transmit/receive address pointer (ADTP) is decremented. Then transmit data 2 (T2) is transferred from the buffer RAM to SIO1.

(ii) Upon completion of transmission of 6 bytes (Refer to **Figure 17-15 (b)**)
When transmission of the sixth byte is completed, the interrupt request flag (CSIIF1) is not set.
The ADTP is set with the initial pointer value again.

(iii) 7th byte transmission point (Refer to **Figure 17-15 (c)**)
Transmit data 1 (T1) is transferred from the buffer RAM to SIO1 again. When transmission of the first byte is completed, ADTP is decremented. Then transmit data 2 (T2) is transferred from the buffer RAM to SIO1.

Figure 17-15. Buffer RAM Operation in 6-Byte Transmission (in Repeat Transmit Mode) (1/2)

(a) Before transmission

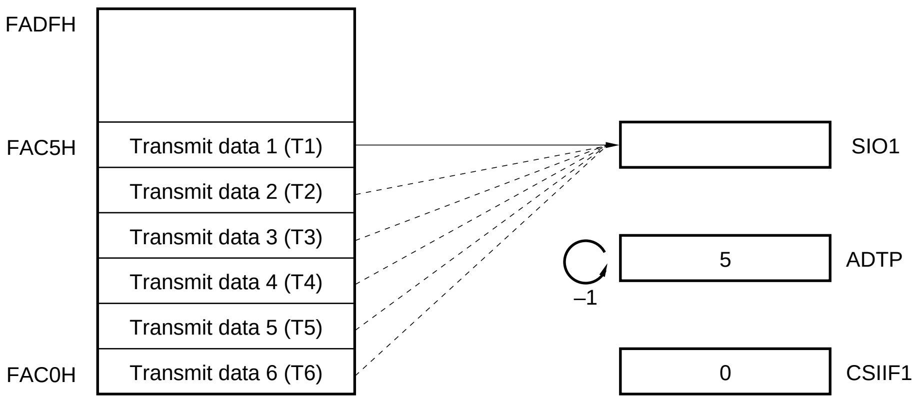
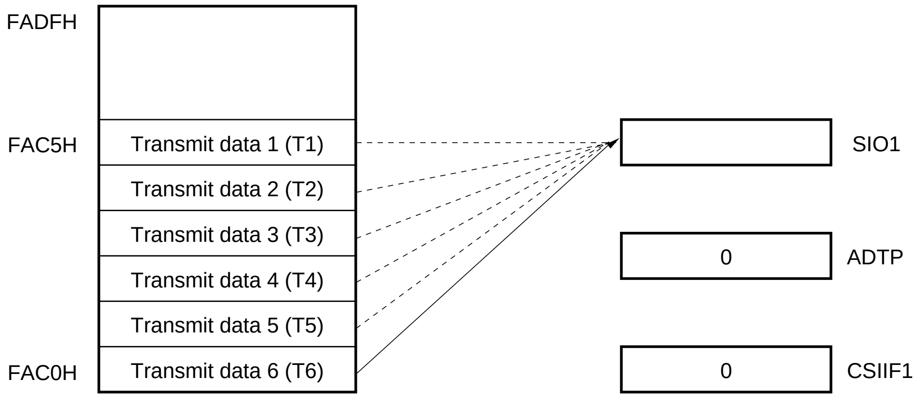
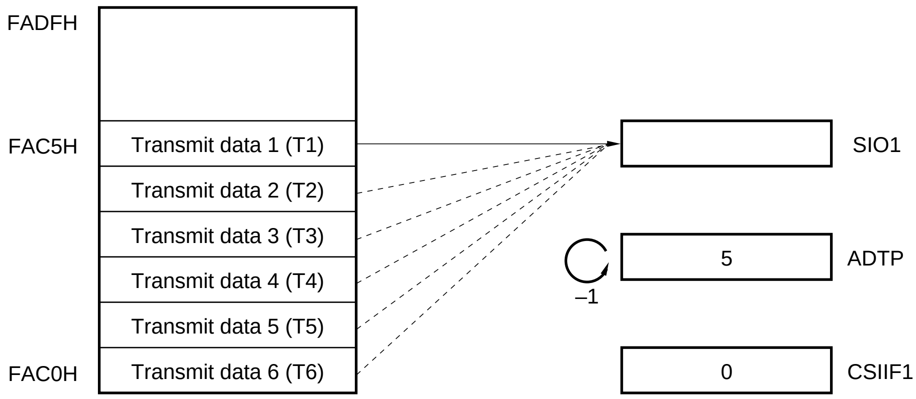


Figure 17-15. Buffer RAM Operation in 6-Byte Transmission (in Repeat Transmit Mode) (2/2)

(b) Upon completion of transmission of 6 bytes



(c) 7th byte transmission point



(d) Automatic transmission/reception suspending and restart

Automatic transmission/reception can be temporarily suspended by setting bit 7 (CSIE1) of the serial operating mode register 1 (CSIM1) to 0.

During 8-bit data transfer, the transmission/reception is not suspended. It is suspended upon completion of 8-bit data transfer.

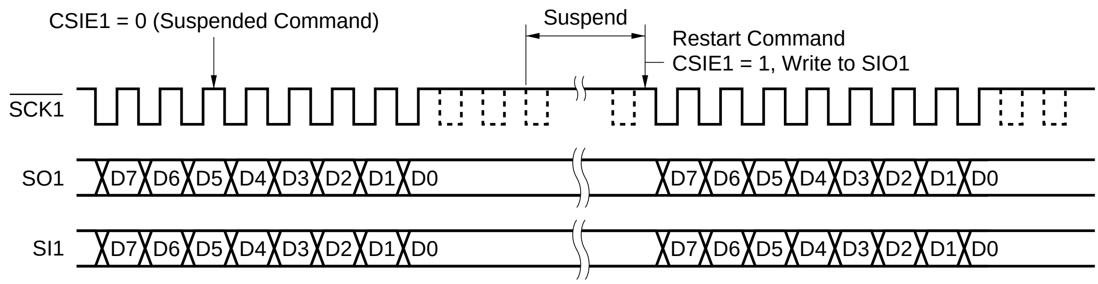
When suspended, bit 3 (TRF) of the automatic data transmit/receive control register (ADTC) is set to 0 after transfer of the 8th bit, and all the port pins used with the serial interface pins for dual function (P20/SI1, P21/SO1, P22/$\overline{SCK1}$, P23/STB and P24/BUSY) are set to the port mode.

Automatic transmission/reception can be restarted and the remaining data can be transferred by setting CSIE1 to 1 and writing any data to the serial I/O shift register 1 (SIO1).

Cautions 1. If the HALT instruction is executed during automatic transmission/reception, transfer is suspended and the HALT mode is set even during 8-bit data transfer. When the HALT mode is cleared, automatic transmission/reception is restarted at the suspended point.

2. When the automatic transmit/receive operation is suspended, do not change the operation mode to the 3-wire serial I/O mode while TRF = 1.

Figure 17-16. Automatic Transmission/Reception Suspension and Restart



CSIE1: Bit 7 of the serial operating mode register 1 (CSIM1)

★ (4) Synchronization control

Busy control and strobe control are functions to synchronize transmission/reception between the master device and a slave device.

By using these functions, a shift in bits being transmitted or received can be detected.

(a) Busy control option

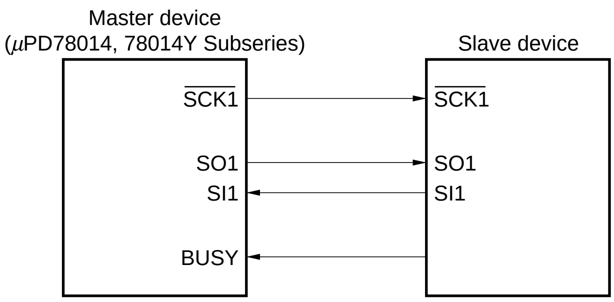
Busy control is a function to keep the serial transmission/reception by the master device waiting while the busy signal output by a slave device to the master is active.

When using this busy control option, the following conditions must be satisfied.

- Bit 5 (ATE) of the serial operation mode register 1 (CSIM1) is set to 1.
- Bit 1 (BUSY1) of the automatic data transmission/reception control register (ADTC) is set to 1.

Figure 17-17 shows the system configuration of the master device and a slave device when the busy control option is used.

Figure 17-17. System Configuration when Busy Control Option Is Used



The master device inputs the busy signal output by the slave device to the BUSY/P24 pin. The master device samples the input busy signal in synchronization with the falling of the serial clock. Even if the busy signal becomes active while 8-bit data is being transmitted or received, transmission/reception by the master is not kept waiting. If the busy signal is active at the rising edge of the serial clock 2 clocks after completion of transmission/reception of the 8-bit data, the busy input becomes valid. After that, the master transmission/reception is kept waiting while the busy signal is active.

The active level of the busy signal is set by bit 0 (BUSY0) of ADTC.

BUSY0 = 0: Active high

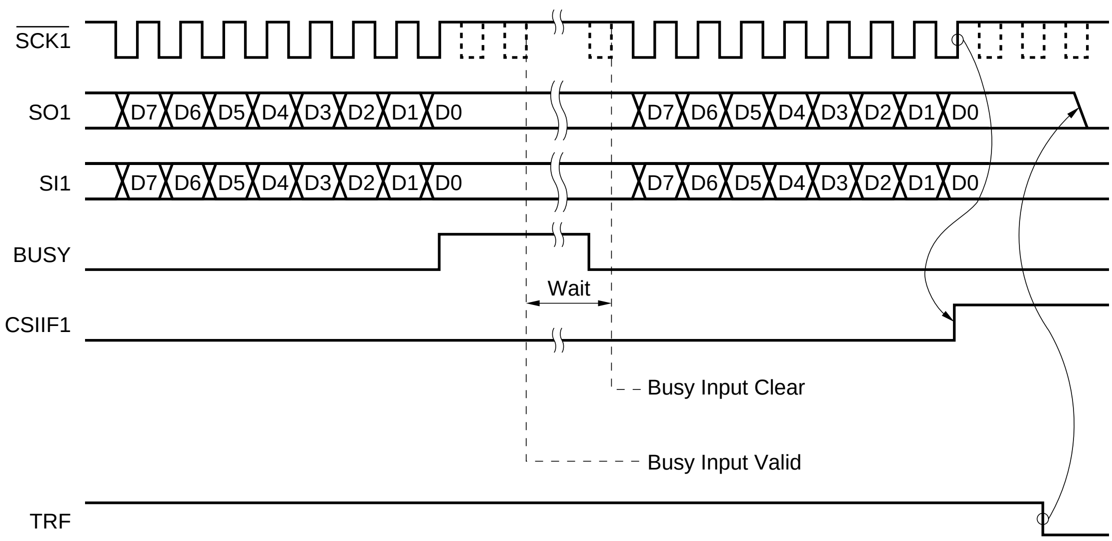
BUSY0 = 1: Active low

When using the busy control option, select the internal clock as the serial clock. Control with the busy signal cannot be implemented with the external clock.

Figure 17-18 shows the operation timing when the busy control option is used.

Caution Busy control cannot be used simultaneously with the interval time control function of the automatic data transmission/reception interval specification register (ADTI). If used, busy control is invalid.

Figure 17-18. Operation Timings when Using Busy Control Option (BUSY0 = 0)



Caution When TRF is cleared, the SO1 pin becomes low.

Remark CSIIF1 : Interrupt request flag
TRF : Bit 3 of the automatic data transmit/receive control register (ADTC)

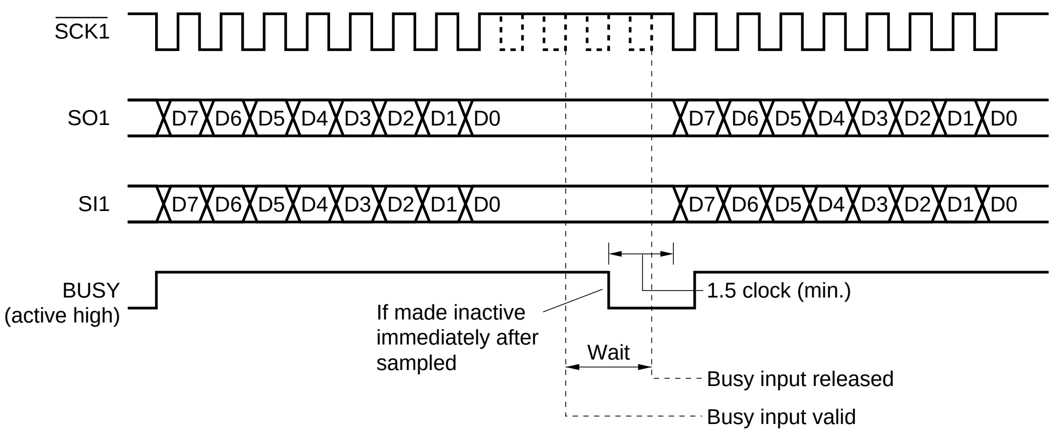
When the busy signal becomes inactive, waiting is released. If the sampled busy signal is inactive, transmission/reception of the next 8-bit data is started at the falling edge of the next clock.

Because the busy signal is asynchronous with the serial clock, it takes up to 1 clock until the busy signal, even if made inactive by the slave, is sampled. It takes 0.5 clock until data transfer is started after the busy signal was sampled.

To accurately release waiting, the slave must keep the busy signal inactive at least for the duration of 1.5 clock.

Figure 17-19 shows the timing of the busy signal and releasing the waiting. This figure shows an example where the busy signal is active as soon as transmission/reception has been started.

Figure 17-19. Busy Signal and Wait Release (when BUSY0 = 0)



(b) Busy & strobe control option

Strobe control is a function to synchronize data transmission/reception between the master and slave devices. The master device outputs the strobe signal from the STB/P23 pin when 8-bit transmission/ receptixon has been completed. By this signal, the slave device can determine the timing of the end of data transmission. Therefore, synchronization is established even if a bit shift occurs because noise is superimposed on the serial clock, and transmission of the next byte is not affected by the bit shift.

To use the strobe control option, the following conditions must be satisfied:

- Bit 5 (ATE) of the serial operation mode register 1 (CSIM1) is set to 1.
- Bit 2 (STRB) of the automatic data transmission/reception control register (ADTC) is set to 1.

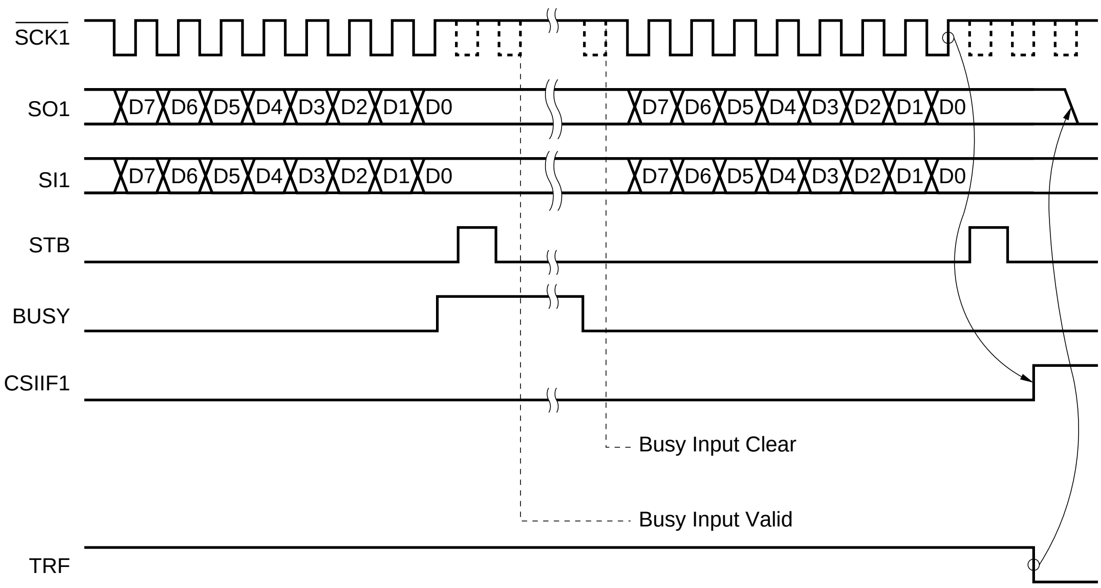
Usually, the busy control and strobe control options are simultaneously used as handshake signals. In this case, the strobe signal is output from the STB/P23 pin, and the BUSY/P24 pin is sampled, and transmission/ reception can be kept waiting while the busy signal is input.

When the strobe control option is not used, the P23/STB pin can be used as a normal I/O port pin.

Figure 17-20 shows the operation timing when the busy & strobe control options are used.

When the strobe control option is used, the interrupt request flag (CSIIF1) that is set on completion of transmission/reception is set after the strobe signal is output.

Figure 17-20. Operation Timings when Using Busy & Strobe Control Option (BUSY0 = 0)



Caution When TRF is cleared, the SO1 pin becomes low.

Remark CSIIF1 : Interrupt request flag

TRF : Bit 3 of the automatic data transmit/receive control register (ADTC)

(c) Bit shift detection by busy signal

During automatic transmission/reception, a bit shift of the serial clock of the slave device may occur because noise is superimposed on the serial clock signal output by the master device. Unless the strobe control option is used at this time, the bit shift affects transmission of the next byte. In this case, the master can detect the bit shift by checking the busy signal during transmission by using the busy control option.

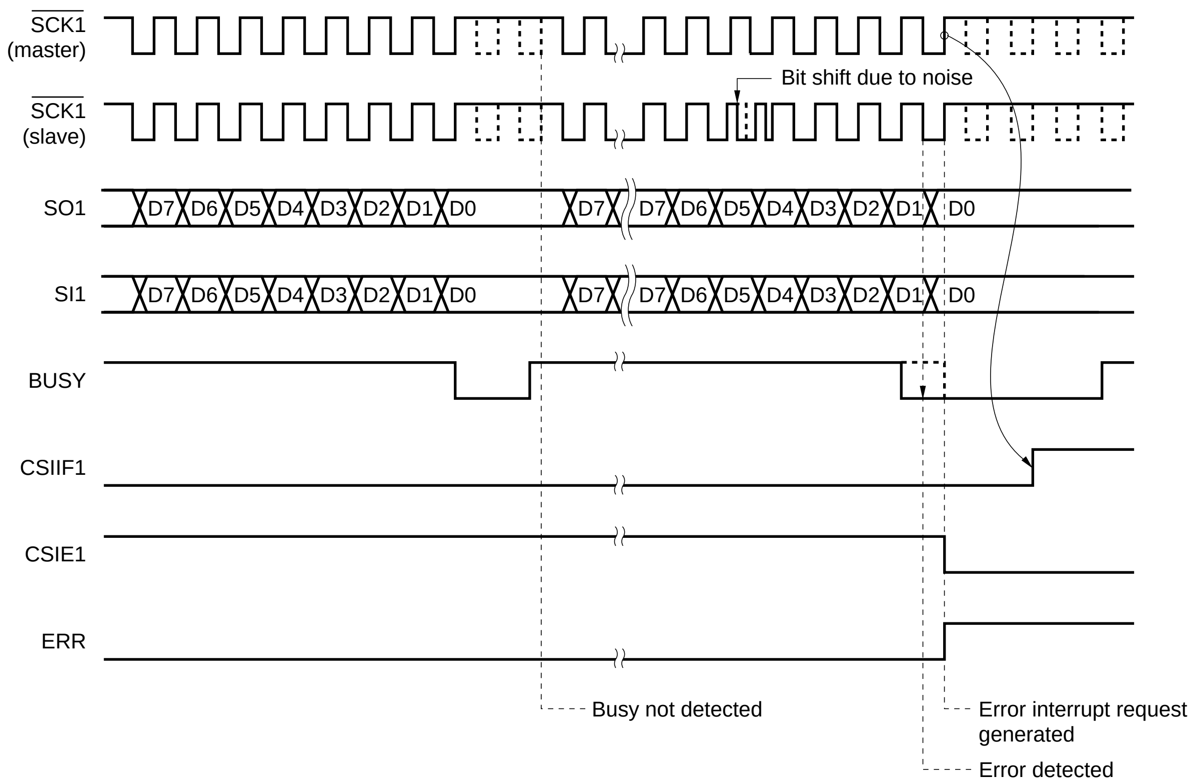
A bit shift is detected by using the busy signal as follows:

The slave outputs the busy signal after the rising of the eighth serial clock during data transmission/reception (to not keep transmission/reception waiting by the busy signal at this time, make the busy signal inactive within 2 clocks).

The master samples the busy signal in synchronization of the falling of the leading side of the serial clock. If a bit shift does not occur, all the eight serial clocks that have been sampled are inactive. If the sampled serial clocks are active, it is assumed that a bit shift has occurred, and error processing is executed (by setting bit 4 (ERR) of the automatic transmission/reception control register (ADTC) to 1).

Figure 17-21 shows the operation timing of the bit shift detection function by the busy signal.

Figure 17-21. Operation Timing of Bit Shift Detection Function by Busy Signal (when BUSY0 = 1)



CSIIF1 : Interrupt request flag

CSIE1 : Bit 7 of serial operation mode register1 (CSIM1)

ERR : Bit 4 of automatic data transmission/reception control register (ADTC)

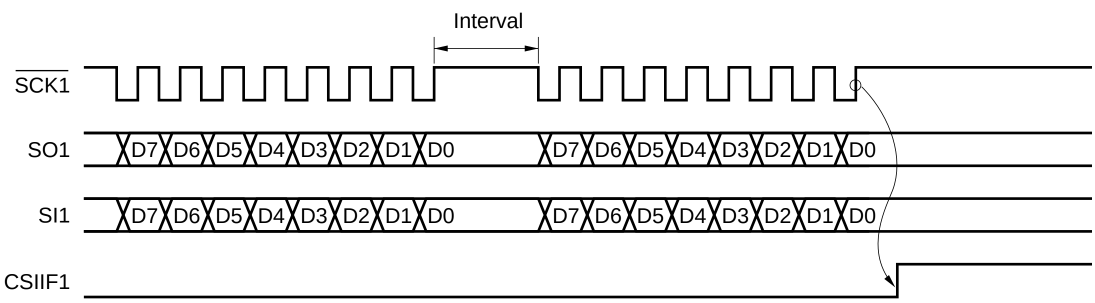
(5) Automatic data transmit/receive interval

When the automatic data transmit/receive function is used, one byte is transmitted/received and then the read/write operations from/to the buffer RAM are performed, therefore an interval is inserted before the next data transmission/reception.

When the automatic data transmit/receive function is performed by an internal clock, since the read/write operations from/to the buffer RAM are done in parallel with CPU processing, the interval depends on the CPU processing at the moment of serial clock's eighth rising-edge timing.

When the automatic data transmit/receive function is performed by an external clock, it must be chosen so that the interval may be longer than the value shown in (b).

Figure 17-22. Automatic Data Transmit/Receive Interval



CSIIF1: Interrupt request flag

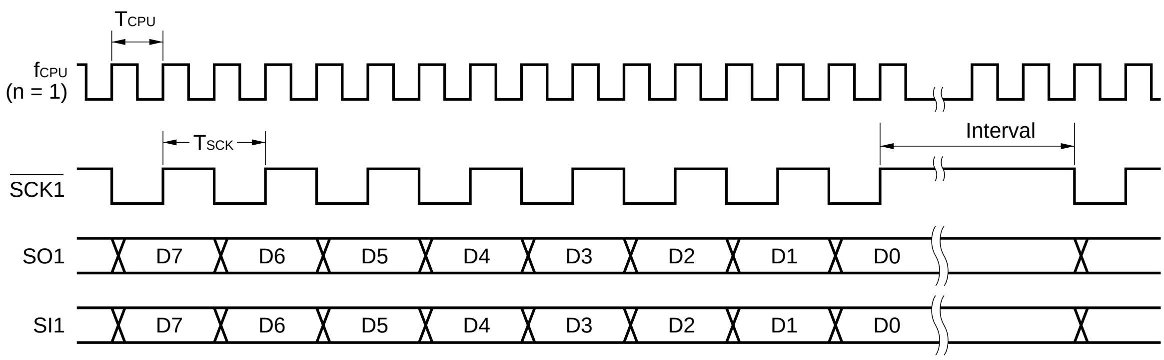
(a) In case the automatic data transmit/receive function is performed by an internal clock
When bit 1 (CSIM11) of the serial operation mode register 1 (CSIM1) is set to 1, the internal clock performs. In this case, the interval is determined as follows by CPU processing.

Table 17-2. Interval by CPU Processing (in Internal Clock Operation)

CPU Processing	Interval
When using multiply instruction	MAX. (2.5 T_{SCK}, 26 T_{CPU})
When using divide instruction	MAX. (2.5 T_{SCK}, 40 T_{CPU})
External access 1 wait mode	MAX. (2.5 T_{SCK}, 18 T_{CPU})
Other than above	MAX. (2.5 T_{SCK}, 14 T_{CPU})

T_{SCK} : 1/f_{SCK}
f_{SCK} : Serial clock frequency
T_{CPU} : 1/f_{CPU}
f_{CPU} : CPU clock (set by bit 0 to bit 2 (PCC0 to PCC2) of processor clock control register)
MAX. (a, b) : a or b, whichever greater

Figure 17-23. Operating Timing in Operating Automatic Transmission/Reception with Internal Clock



f_{CPU} : CPU clock (set by bit 0 to bit 2 (PCC0 to PCC2) of processor clock control register (PCC))
T_{CPU} : 1/f_{CPU}
T_{SCK} : 1/f_{SCK}
f_{SCK} : Serial clock frequency

(b) In case the automatic data transmit/receive function is performed by an external clock

When bit 1 (CSIM11) of the serial operation mode register 1 (CSIM1) is cleared to 0, the external clock performs.

When the automatic data transmit/receive function is performed by an external clock, it must be chosen so that the interval may be longer than the value shown below.

Table 17-3. Interval by CPU Processing (in External Clock Operation)

CPU Processing	Interval
When using multiply instruction	26 T_{CPU} or more
When using divide instruction	40 T_{CPU} or more
External access 1 wait mode	18 T_{CPU} or more
Other than above	14 T_{CPU} or more

T_{CPU}: 1/f_{CPU}

f_{CPU} : CPU clock (set by bit 0 to bit 2 (PCC0 to PCC2) of processor clock control register)

[MEMO]

CHAPTER 18   INTERRUPT FUNCTIONS AND TEST FUNCTION

18.1 Interrupt Function Types

The following three types of interrupt functions are used.

(1) Non-maskable interrupt
This interrupt is acknowledged unconditionally even in a 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.
The non-maskable interrupt has one source of interrupt request from the watchdog timer.

(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). 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 18-1**).
A standby release signal is generated.
The maskable interrupt has four sources of external interrupt requests and eight sources of internal interrupt requests.

(3) Software interrupt
This is a vectored interrupt to be generated by executing the BRK instruction. It is acknowledged even in a disabled state. The software interrupt does not undergo interrupt priority control.

18.2 Interrupt Sources and Configuration

There are total of 14 non-maskable, maskable and software interrupts in the interrupt sources (see **Table 18-1**).

Table 18-1. Interrupt Source List

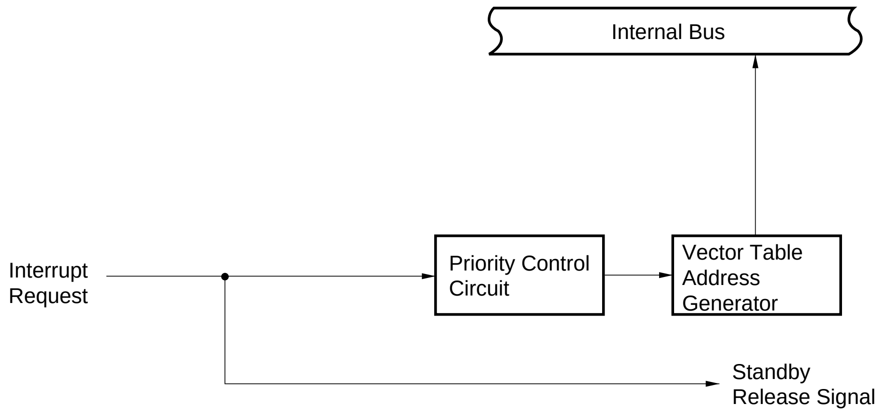
Interrupt Type	Default Priority[Note 1]	Interrupt Source		Internal/ External	Vector Table Address	Basic Configuration Type[Note 2]
		Name	Trigger			
Non-maskable	—	INTWDT	Watchdog timer overflow (with watchdog timer mode 1 selected)	Internal	0004H	(A)
Maskable	0	INTWDT	Watchdog timer overflow (with interval timer mode selected)			(B)
	1	INTP0	Pin input edge detection	External	0006H	(C)
	2	INTP1			0008H	(D)
	3	INTP2			000AH	
	4	INTP3			000CH	
	5	INTCSI0	End of serial interface channel 0 transfer	Internal	000EH	(B)
	6	INTCSI1	End of serial interface channel 1 transfer		0010H	
	7	INTTM3	Reference time interval signal from watch timer		0012H	
	8	INTTM0	16-bit timer/event counter match signal generation		0014H	
	9	INTTM1	8-bit timer/event counter 1 match signal generation		0016H	
	10	INTTM2	8-bit timer/event counter 2 match signal generation		0018H	
	11	INTAD	End of A/D converter conversion		001AH	
Software	—	BRK	Execution of BRK instruction	—	003EH	(E)

Notes 1. Default priorities are intended for two or more simultaneously generated maskable interrupt requests. 0 is the highest priority and 11 is the lowest priority.

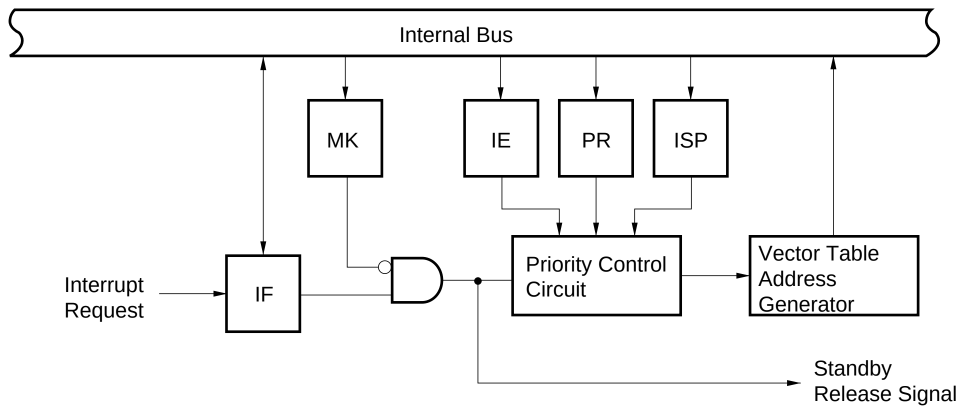
2. Basic configuration types (A) to (E) correspond to A to E in Figure 18-1.

Figure 18-1. Basic Configuration of Interrupt Function (1/2)

(A) Internal non-maskable interrupt



(B) Internal maskable interrupt



(C) External maskable interrupt (INTP0)

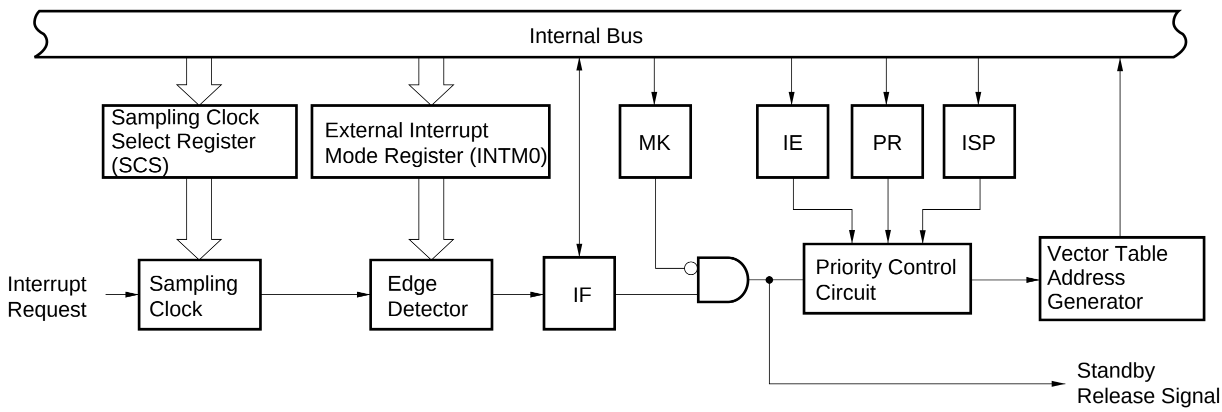
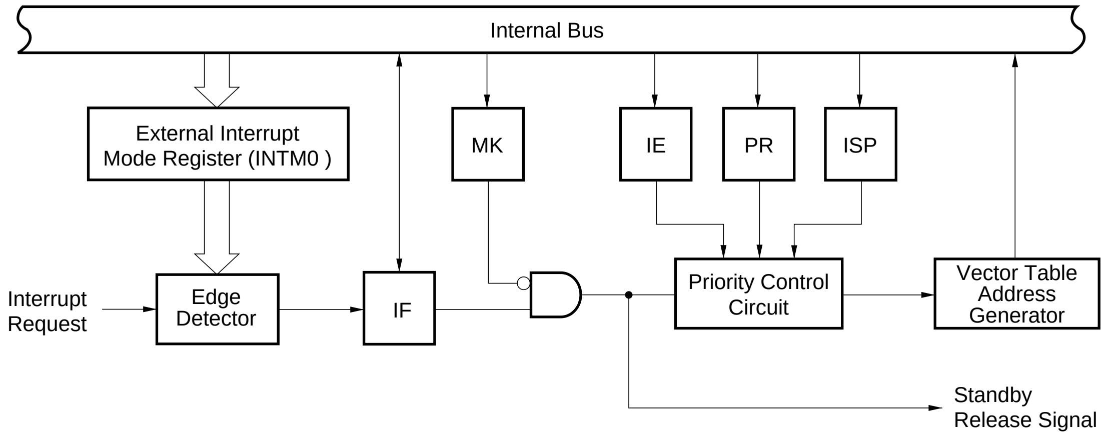
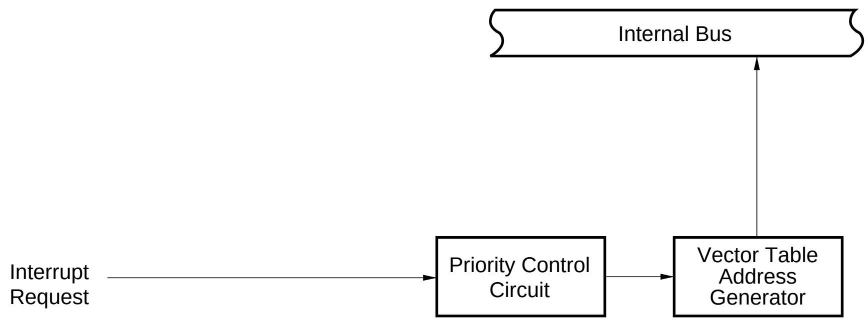


Figure 18-1. Basic Configuration of Interrupt Function (2/2)

(D) External maskable interrupt (except INTP0)



(E) Software interrupt



IF : Interrupt request flag
IE : Interrupt enabled flag
ISP : Inservice priority flag
MK : Interrupt mask flag
PR : Priority specify flag

18.3 Interrupt Function Control Registers

The following six types of registers are used to control the interrupt functions.

- Interrupt request flag register (IF0L, IF0H)
- Interrupt mask flag register (MK0L, MK0H)
- Priority specify flag register (PR0L, PR0H)
- External interrupt mode register (INTM0)
- Sampling clock select register (SCS)
- Program status word (PSW)

Table 18-2 gives a listing of interrupt request flags, interrupt mask flags and priority specify flag names corresponding to interrupt request sources.

Table 18-2. 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	
INTCSI0	CSIIF0		CSIMK0		CSIPR0	
INTCSI1	CSIIF1		CSIMK1		CSIPR1	
INTTM3	TMIF3		TMMK3		TMPR3	
INTTM0	TMIF0	IF0H	TMMK0	MK0H	TMPR0	PR0H
INTTM1	TMIF1		TMMK1		TMPR1	
INTTM2	TMIF2		TMMK2		TMPR2	
INTAD	ADIF		ADMK		ADPR	

(1) Interrupt request flag registers (IF0L, IF0H)

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 and IF0H are set with a 1-bit or 8-bit memory manipulation instruction. When IF0L and IF0H are used together as a 16-bit register IF0, they are set with a 16-bit memory manipulation instruction.

$\overline{\text{RESET}}$ input sets these registers to 00H.

Figure 18-2. Interrupt Request Flag Register Format

Symbol	<7>	<6>	<5>	<4>	<3>	<2>	<1>	<0>	Address	When Reset	R/W
IF0L	TMIF3	CSIIF1	CSIIF0	PIF3	PIF2	PIF1	PIF0	TMIF4	FFE0H	00H	R/W

	7	6	<5>	4	<3>	<2>	<1>	<0>			
IF0H	0	0	WTIF[Note]	0	ADIF	TMIF2	TMIF1	TMIF0	FFE1H	00H	R/W

××IF	Interrupt Request Flag
0	No interrupt request signal is generated
1	Interrupt request signal is generated and interrupt requested

Note WTIF flag is a 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 mode 1 is used, set TMIF4 flag to 0.

2. Be sure to set bits 4, 6, and 7 of IF0H to 0.

(2) Interrupt mask flag registers (MK0L, MK0H)

The interrupt mask flag is used to enable/disable the corresponding maskable interrupt service and the standby clear.

MK0L and MK0H are set with a 1-bit or 8-bit memory manipulation instruction. When MK0L and MK0H are used together as a 16-bit register MK0, they are set with a 16-bit memory manipulation instruction.

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

Figure 18-3. Interrupt Mask Flag Register Format

Symbol	<7>	<6>	<5>	<4>	<3>	<2>	<1>	<0>	Address	When Reset	R/W
MK0L	TMMK3	CSIMK1	CSIMK0	PMK3	PMK2	PMK1	PMK0	TMMK4	FFE4H	FFH	R/W

Symbol	7	6	<5>	4	<3>	<2>	<1>	<0>	Address	When Reset	R/W
MK0H	1	1	WTMK[Note]	1	ADMK	TMMK2	TMMK1	TMMK0	FFE5H	FFH	R/W

××MK	Interrupt Servicing, Standby Mode Control
0	Interrupt servicing, standby mode clear enabled
1	Interrupt servicing, standby mode clear disabled

Note WTMK flag controls standby mode clear enabled/disabled. The interrupt function does not control.

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 is also used for 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. Be sure to set bits 4, 6, and 7 of MK0H to 1.

(3) Priority specify flag registers (PR0L, PR0H)

The priority specify flag is used to set the corresponding maskable interrupt priority orders.

PR0L and PR0H are set with a 1-bit or 8-bit memory manipulation instruction. When PR0L and PR0H are used together as a 16-bit register PR0, they are set with a 16-bit memory manipulation instruction.

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

Figure 18-4. Priority Specify Flag Register Format

Symbol	<7>	<6>	<5>	<4>	<3>	<2>	<1>	<0>	Address	When Reset	R/W
PR0L	TMPR3	CSIPR1	CSIPR0	PPR3	PPR2	PPR1	PPR0	TMPR4	FFE8H	FFH	R/W

Symbol	7	6	5	4	<3>	<2>	<1>	<0>	Address	When Reset	R/W
PR0H	1	1	1	1	ADPR	TMPR2	TMPR1	TMPR0	FFE9H	FFH	R/W

××PR	Priority Level Selection
0	High priority level
1	Low priority level

Cautions 1. When a watchdog timer is used in the watchdog timer mode 1, set the TMPR4 flag to 1.

2. Be sure to set bits 4 to 7 of PR0H to 1.

(4) External interrupt mode register (INTM0)

This register sets the valid edge for INTP0 to INTP2.

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

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

Remarks **1.** INTP0 is also used for TI0/P00.

2. INTP3 is fixed at falling edge.

Figure 18-5. External Interrupt Mode Register Format

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

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

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

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

Caution **Set the valid edge for INTP0/TI0/P00 after setting bits 1 through 3 (TMC01 to TMC03) of 16-bit timer mode control register to 000 to stop the timer operation.**

(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 removed with sampling clocks.

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

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

Figure 18-6. Sampling Clock Select Register Format

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

SCS1	SCS0	INTP0 Sampling Clock Selection
0	0	$f_X/2^{N+1}$
0	1	Setting prohibited
1	0	$f_X/2^6$ (156 kHz)
1	1	$f_X/2^7$ (78.1 kHz)

Caution **$f_X/2^{N+1}$ is a clock to be supplied to the CPU and $f_X/2^6$ and $f_X/2^7$ are clocks to be supplied to the peripheral hardware. $f_X/2^{N+1}$ 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 (PCC)

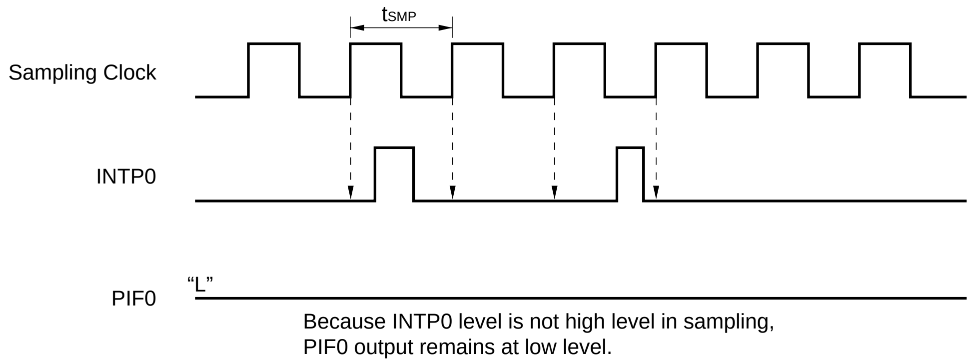
2. f_X: Main system clock oscillation frequency

3. Values in parentheses apply to operation with f_X = 10.0 MHz.

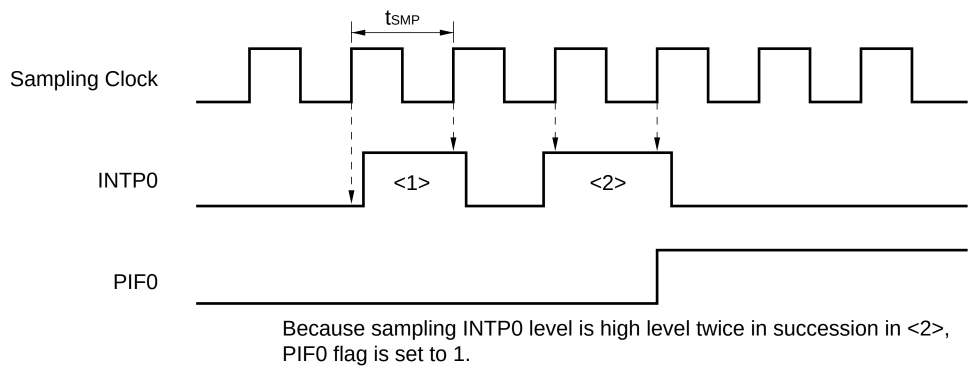
When the sampling INTP0 input level is active twice in succession, the noise eliminator sets the interrupt request flag (PIF0) to 1. Figure18-7 shows noise eliminator input/output timing.

Figure 18-7. Noise Eliminator Input/Output Timing (when rising edge is detected)

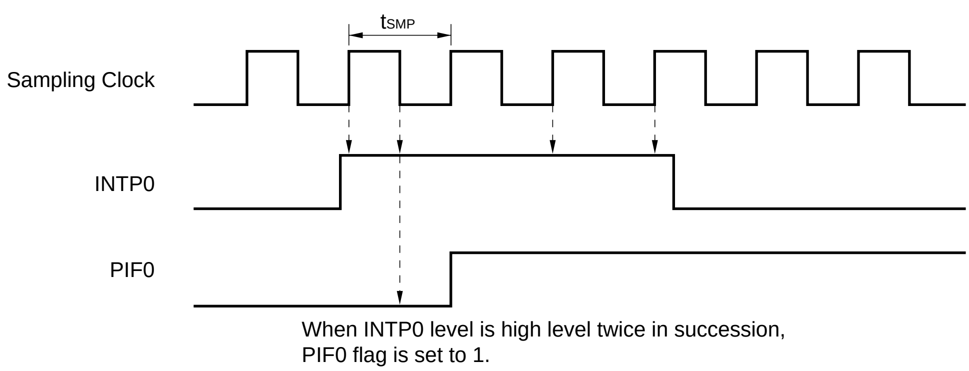
(a) When input is less than the sampling cycle (t_{SMP})



(b) When input is equal to or twice the sampling cycle (t_{SMP})



(c) When input is twice or more than the sampling cycle (t_{SMP})



(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 enable/disable and the ISP flag to control multiple interrupt servicing are mapped.

Besides 8-bit units 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, and when the BRK instruction is executed, the contents of PSW is automatically saved into a stack and the IE flag is reset to (0). If a maskable interrupt request is acknowledged, the contents of the priority specify flag of the acknowledged interrupt are transferred to the ISP flag. The contents of acknowledged interrupt is also saved into the stack with 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 18-8. Program Status Word Configuration

Symbol	7	6	5	4	3	2	1	0	When Reset
PSW	IE	Z	RBS1	AC	RBS0	0	ISP	CY	02H

Z, RBS1, AC, RBS0, 0, CY → Use when normal instruction is executed

ISP	Priority of Interrupt Currently Being Serviced
0	High-priority interrupt servicing (low-priority interrupt disable)
1	Interrupt request not acknowledged or low priority servicing (all maskable interrupts enable)

IE	Interrupt Request Acknowledge Enable/Disable
0	Disable
1	Enable

18.4 Interrupt Servicing Operations

18.4.1 Non-maskable interrupt request acknowledge operation

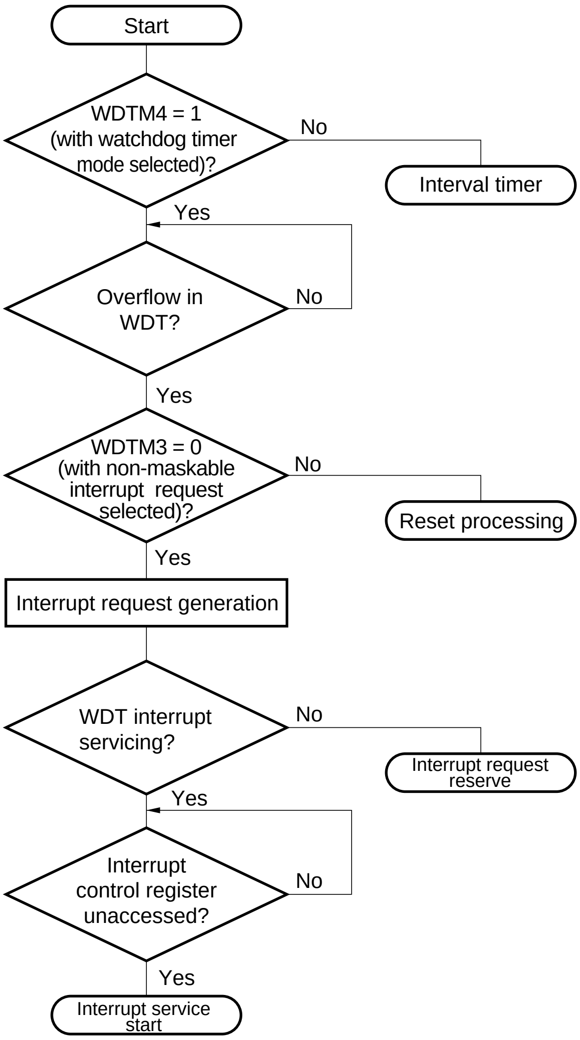
A non-maskable interrupt request is unconditionally acknowledged even if in an interrupt request acknowledge disable state. It does not undergo interrupt priority control and has highest priority over all other interrupts.

If a non-maskable interrupt request is acknowledged, the contents of acknowledged interrupt is saved in the stacks, program status word (PSW) and program counter (PC), in that order, the IE and ISP flags are reset to 0, and the vector table contents are loaded into PC and branched.

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 18-9 shows the flowchart from non-maskable interrupt request generation to acknowledge. Figure 18-10 shows the non-maskable interrupt request acknowledge timing. Figure 18-11 shows the acknowledge operation if multiple non-maskable interrupt requests are generated.

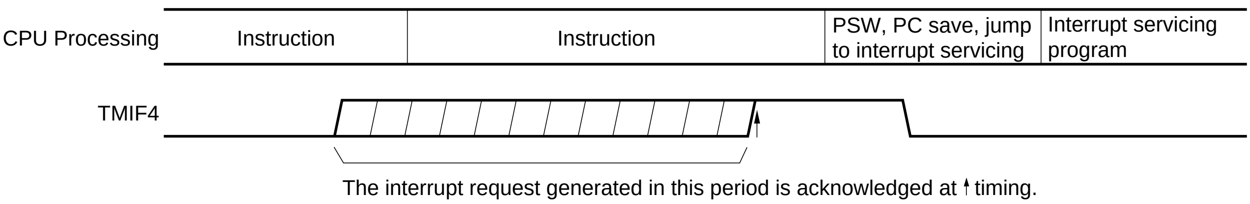
Figure 18-9. Flowchart from Non-Maskable Interrupt Request Generation to Acknowledge



WDTM : Watchdog timer mode register

WDT : Watchdog timer

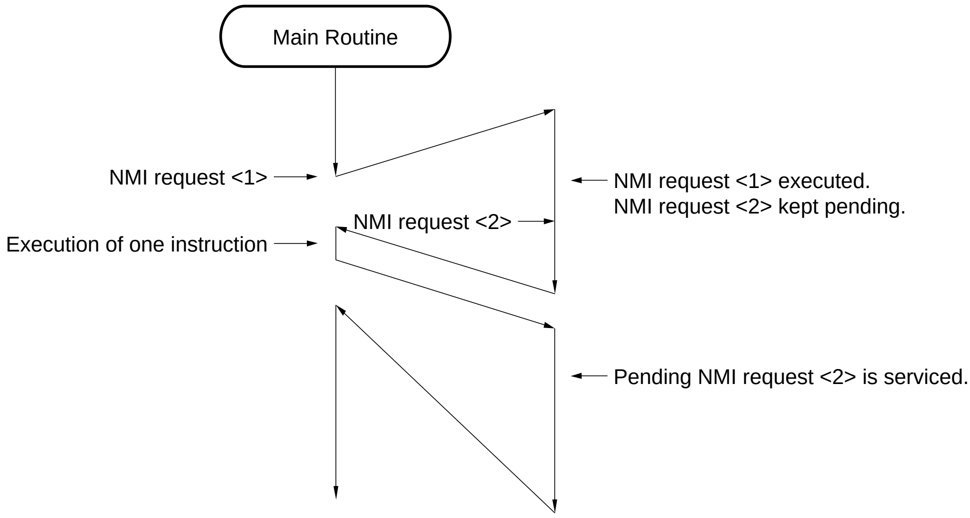
Figure 18-10. Non-Maskable Interrupt Request Acknowledge Timing



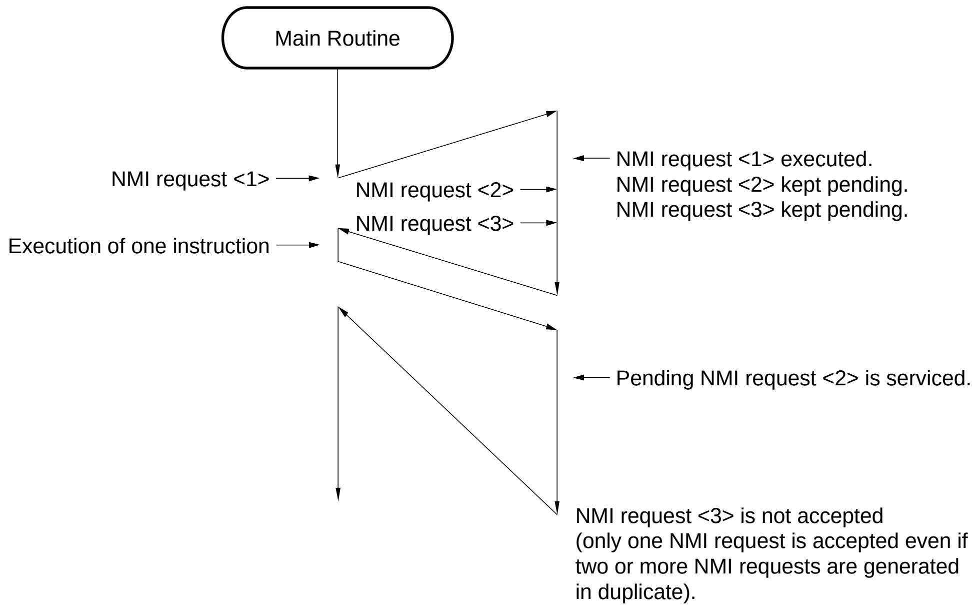
TMIF4 : Watchdog timer interrupt request flag

Figure 18-11. Non-Maskable Interrupt Request Acknowledge Operation

(a) If a new non-maskable interrupt request is generated during non-maskable interrupt servicing program execution



(b) If two non-maskable interrupt requests are newly generated during non-maskable interrupt servicing program execution



18.4.2 Maskable interrupt request acknowledge operation

A maskable interrupt request becomes acknowledgeable when an interrupt request flag is set to 1 and the interrupt mask flag for that 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 request is not acknowledged during high-priority interrupt service (with ISP flag reset to 0).

Wait times from maskable interrupt request generation to interrupt servicing are shown in Table 18-3.

Refer to Figures 18-3 and 18-4 for the interrupt request acknowledge timing.

Table 18-3. Times from Maskable Interrupt Request Generation to Interrupt Service

	Minimum Time	Maximum Time[Note]
When XXPR = 0	13 clocks	63 clocks
When XXPR = 1	15 clocks	65 clocks

Note If an interrupt request is generated just before a divide instruction, the wait time is maximized.

Remark 1 clock: $1/f_{CPU}$ (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 two or more requests are specified for the same priority with the priority specify flag, the interrupt request with the higher default priority is acknowledged first.

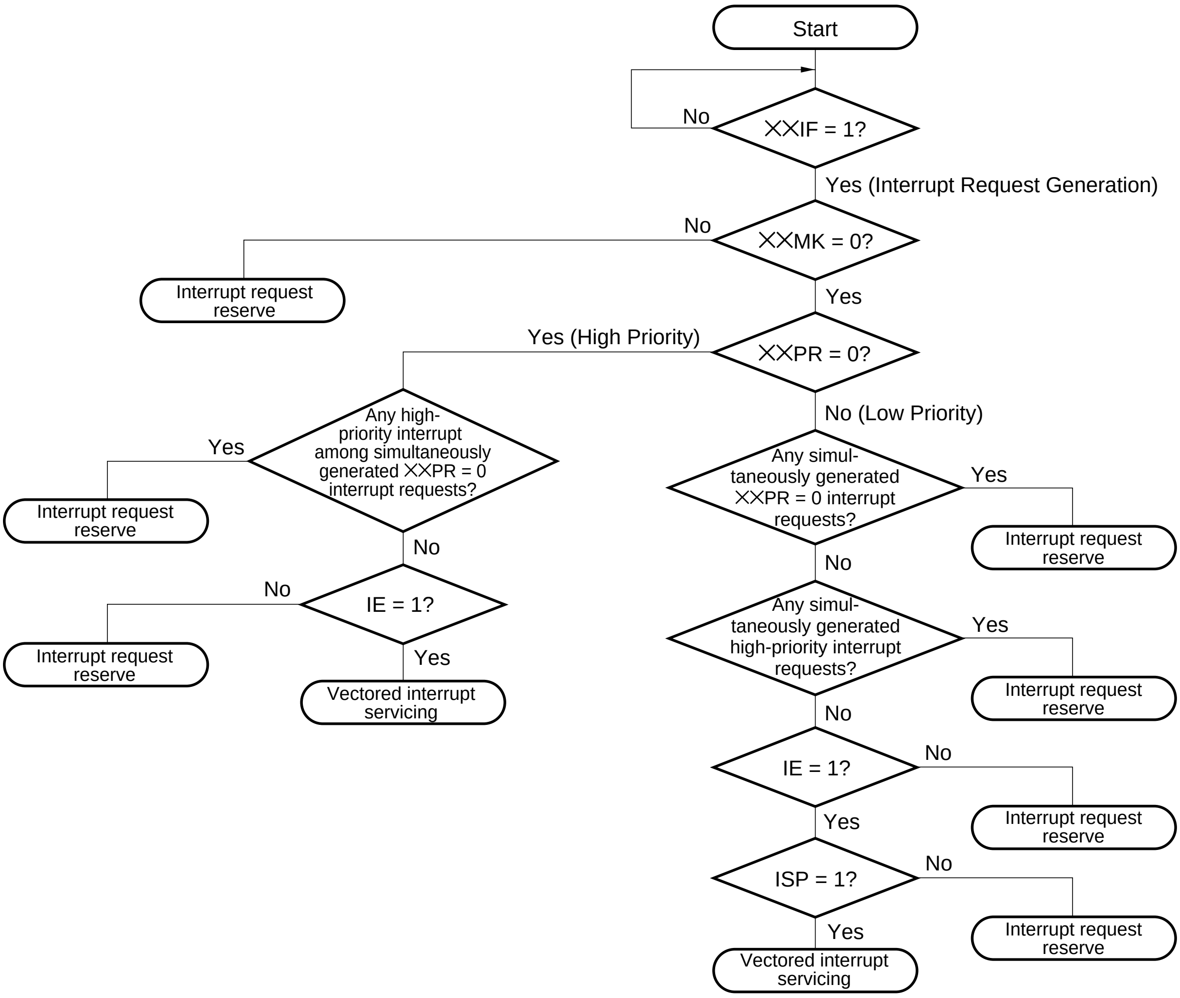
Any reserved interrupt requests are acknowledged when they become acknowledgeable.

Figure 18-12 shows interrupt request acknowledge algorithms.

If a maskable interrupt request is acknowledged, the acknowledged interrupt is 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 acknowledged interrupt request 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 18-12. Interrupt Request Acknowledge Processing Algorithm



××IF : Interrupt request flag

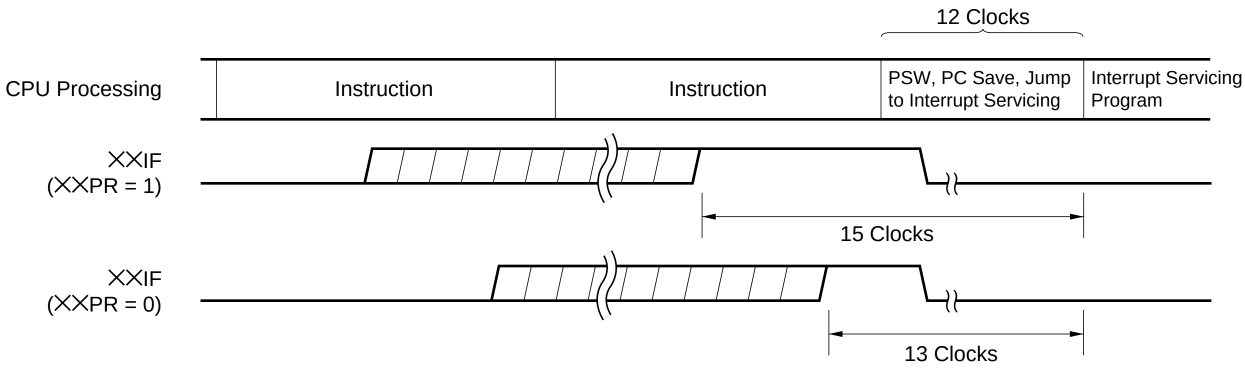
××MK : Interrupt mask flag

××PR : Priority specify flag

IE : Flag to control maskable interrupt request acknowledge (1 = Enable, 0 = Disable)

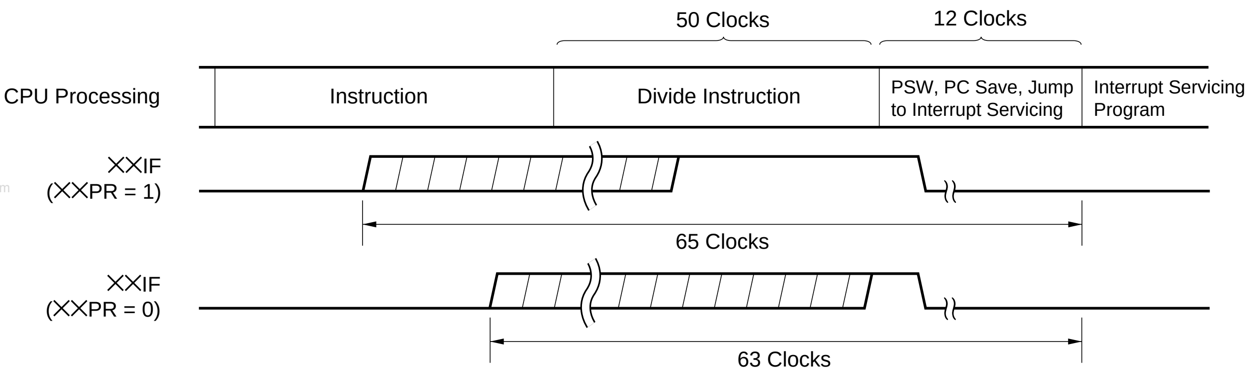
ISP : Flag to indicate the priority of interrupt being serviced (0 = Interrupt with high-priority is being serviced, 1 = Interrupt request is not acknowledged or an Interrupt with low-priority is being serviced)

Figure 18-13. Interrupt Request Acknowledge Timing (Minimum Time)



Remark 1 clock: $1/f_{CPU}$ (f_{CPU}: CPU clock)

Figure 18-14. Interrupt Request Acknowledge Timing (Maximum Time)



Remark 1 clock: $1/f_{CPU}$ (f_{CPU}: CPU clock)

18.4.3 Software interrupt request acknowledge operation

A software interrupt request is acknowledged by BRK instruction execution. Software interrupt request cannot be disabled.

If a software interrupt request is acknowledged, it is 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.

18.4.4 Multiple interrupt servicing

Accepting another interrupt request while an interrupt is being serviced is called nesting interrupts.

Nesting does not take place unless the interrupts (except the non-maskable interrupt) are enabled to be accepted (IE = 1). Accepting another interrupt request is disabled (IE = 0) when one interrupt has been accepted. Therefore, to enable nesting, the EI flag must be set to 1 during interrupt servicing, to enable the another interrupt.

Nesting interrupts may not occur even when the interrupts are enabled. This is controlled by the priorities of the interrupts. Although two types of priorities, default priority and programmable priority, may be assigned to an interrupt, nesting is controlled by using the programmable priority.

If an interrupt with the same level of priority as or the higher priority than the interrupt currently serviced occurs, that interrupt can be accepted and nested. If an interrupt with a priority lower than that of the currently serviced interrupt occurs, that interrupt cannot be accepted and nested.

An interrupt that is not accepted and nested because it is disabled or it has a low priority is kept pending. This interrupt is accepted after servicing of the current interrupt has been completed and one instruction of the main routine has been executed.

Nesting is not enabled while the non-maskable interrupt is being serviced.

Table 18-4 shows the interrupts that can be nested, and Figure 18-15 shows an example of nesting.

Table 18-4. Interrupt Request Enabled for Multiple Interrupt during Interrupt Servicing

Interrupt Servicing \ Multiple Interrupt Request		Non-maskable Interrupt Request	Maskable Interrupt Request			
			×× PR = 0		×× PR = 1	
			IE = 1	IE = 0	IE = 1	IE = 0
Non-maskable interrupt		N/A	N/A	N/A	N/A	N/A
Maskable interrupt	ISP = 0	A	A	N/A	N/A	N/A
	ISP = 1	A	A	N/A	A	N/A
Software interrupt		A	A	N/A	A	N/A

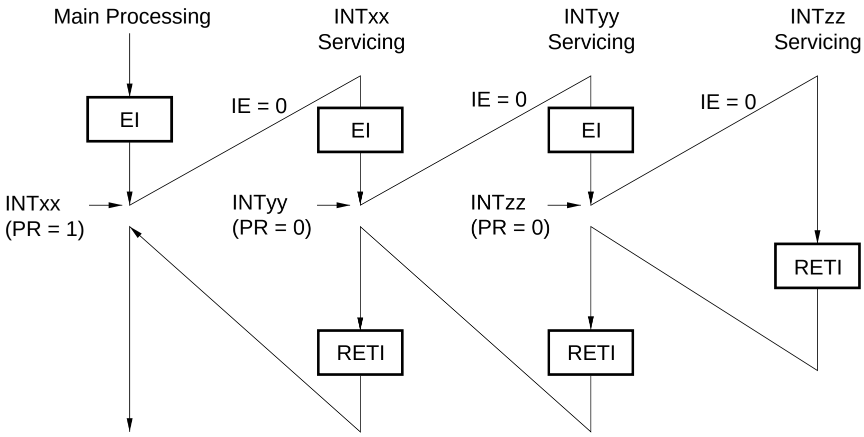
Remarks 1. A : Multiple interrupt enable
N/A : Multiple interrupt disable

2. ISP and IE are flags included in PSW.
ISP = 0 : High-priority interrupt servicing
ISP = 1 : Interrupt request is not acknowledged or low-priority interrupt servicing
IE = 0 : Interrupt request acknowledge disabled
IE = 1 : Interrupt request acknowledge enabled

3. ××PR is a flag included in PR0L, PR0H.
××PR = 0: High-priority level
××PR = 1: Low-priority level

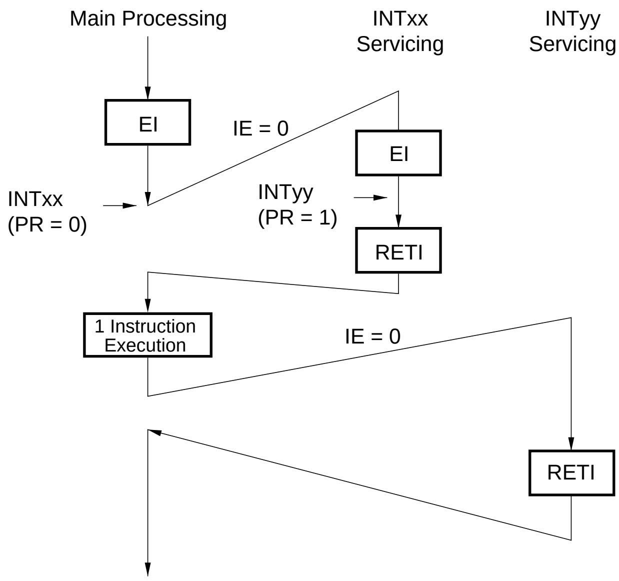
Figure 18-15. Multiple Interrupt Examples (1/2)

Example 1. Two multiple interrupts are 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

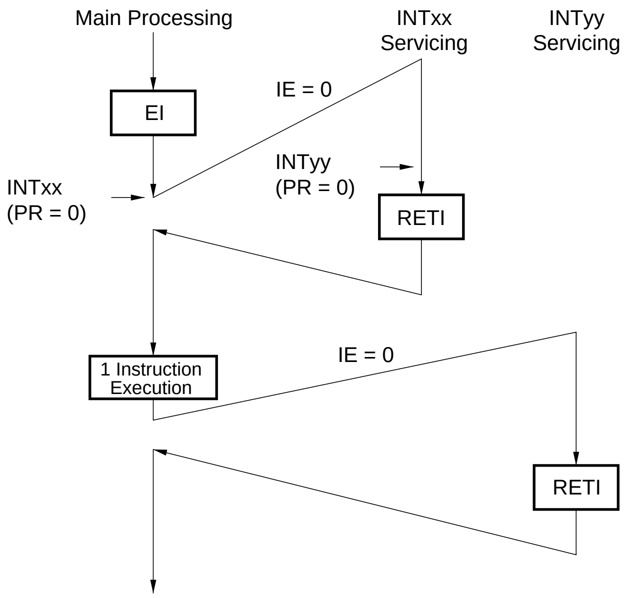


The interrupt request INTyy generated during interrupt INTxx servicing is not acknowledged because the interrupt priority is lower than that of INTxx, and a multiple interrupt is not generated. INTyy request is retained and acknowledged after execution of 1 instruction execution of the main processing.

PR = 0 : High-priority level
PR = 1 : Low-priority level
IE = 0 : Interrupt request acknowledge disabled

Figure 18-15. 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 reserved and acknowledged after 1 instruction execution of the main processing.

PR = 0 : High-priority level
IE = 0 : Interrupt request acknowledge disabled

18.4.5 Interrupt request reserve

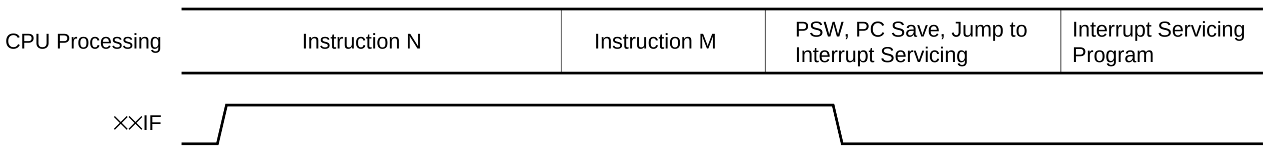
Some instructions may reserve the acknowledge of an instruction request until the completion of the execution of the next instruction even if the interrupt request is generated during the execution. The following shows such instructions (interrupt request reserve instruction).

- 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
- Manipulation instructions for IF0L, IF0H, MK0L, MK0H, PR0L, PR0H and INTM0 registers

Caution BRK instruction is not an interrupt request reserve instruction described above. However, in a software interrupt started by the execution of BRK instruction, the IE flag is cleared to 0. Therefore, interrupt requests are not acknowledged even when a maskable interrupt request is issued during the execution of the BRK instruction. However, non-maskable interrupt requests are acknowledged.

Figure 18-16 shows the interrupt request reserve timing.

Figure 18-16. Interrupt Request Reserve



Remarks
1. Instruction N : Interrupt request reserve instruction
2. Instruction M : Instruction except interrupt reserve instructions
3. Operation of ××IF (interrupt request) is not affected by ××PR (priority level) value.

18.5 Test Function

In this function, when the watch timer overflows and when a rising edge of port 4 is detected, the corresponding test input flag is set (1), and a standby release signal is generated.

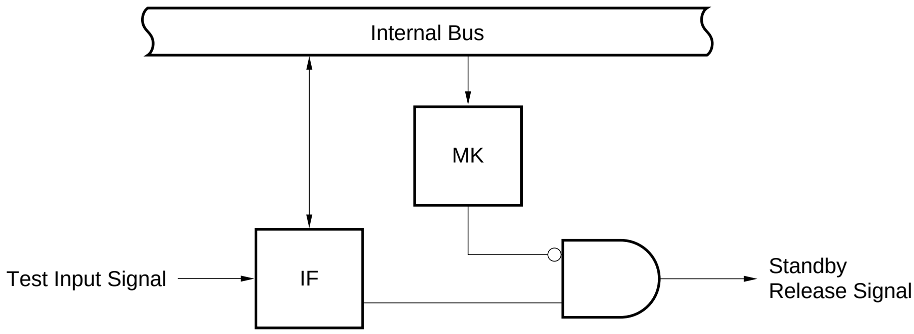
Unlike the interrupt function, vectored processing not performed.

There are two test input sources as listed in Table 18-5. Basic configuration is shown in Figure 18-17.

Table 18-5. Test Input Source

Test Input Source		Internal/External
Name	Trigger	
INTWT	Watch timer overflow	Internal
INTPT4	Falling edge detection of Port 4	External

Figure 18-17. Basic Configuration of Test Function



IF : Test Input Flag
MK : Test Mask Flag

18.5.1 Test function control registers

The following three types of registers are used to control the test functions.

- Interrupt request flag register 0H (IF0H)
- Interrupt mask flag register 0H (MK0H)
- Key return mode register (KRM)

Names of test input flag and test mask flag corresponding to test input signal name are shown in Table 18-6.

Table 18-6. Various Flags Corresponding to Test Input Signal

Test Input Signal Name	Test Input Flag	Test Mask Flag
INTWT	WTIF	WTMK
INTPT4	KRIF	KRMK

(1) Interrupt request flag register 0H (IF0H)
This register displays the watch timer overflow detection/non-detection.
IF0H is set with a 1-bit or 8-bit memory manipulation instruction.
$\overline{\text{RESET}}$ input sets IF0H to 00H.

Figure 18-18. Interrupt Request Flag Register 0H Format

Symbol	7	6	<5>	4	<3>	<2>	<1>	<0>	Address	When Reset	R/W
IF0H	0	0	WTIF	0	ADIF	TMIF2	TMIF1	TMIF0	FFE1H	00H	R/W

WTIF	Watch timer overflow detection flag
0	Non detection
1	Detection

Caution Be sure to set bits 4, 6, and 7 to 0.

(2) Interrupt mask flag register 0H (MK0H)
This register sets standby mode clear enable/disable by watch timer.
MK0H is set with a 1-bit or 8-bit memory manipulation instruction.
$\overline{\text{RESET}}$ input sets MK0H to FFH.

Figure 18-19. Interrupt Mask Flag Register 0H Format

Symbol	7	6	<5>	4	<3>	<2>	<1>	<0>	Address	When Reset	R/W
MK0H	1	1	WTMK	1	ADMK	TMMK2	TMMK1	TMMK0	FFE5H	FFH	R/W

WTMK	Standby mode control by watch timer
0	Standby mode clear enabled
1	Standby mode clear disabled

Caution Be sure to set bits 4, 6, and 7 to 1.

(3) Key return mode register (KRM)

This register set the standby mode clear enable/disable with the key return signal (falling edge detection of Port 4).

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

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

Figure 18-20. Key Return Mode Register Format

Symbol	7	6	5	4	3	2	<1>	<0>	Address	When Reset	R/W
KRM	0	0	0	0	0	0	KRMK	KRIF	FFF6H	02H	R/W

KRIF	Key return signal detection flag
0	Non detection
1	Detection (falling edge detection of Port 4)

KRMK	Standby mode control with key return signal
0	Standby mode clear enabled
1	Standby mode clear disabled

Caution When falling edge detection is used in Port 4, take care to clear KRIF to 0 (KRIF is not automatically cleared to 0).

18.5.2 Test input signal acknowledge operation

(1) Internal test input signal

An internal test input signal (INTWT) is generated when the watch timer overflows and the WTIF flag is set by it. 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, the watch function can be effected.

(2) External test input signal

If a falling edge is input to a pin of port 4 (P40 to P47), an external test input signal (INTP4) is generated, setting the KRIF flag. At this time, the standby release signal is generated if it is not masked by the interrupt mask flag (KRMK). By using port 4 for key return signal input of a key matrix, the presence or absence of a key input can be checked by the status of the KRIF flag.

[MEMO]

CHAPTER 19 EXTERNAL DEVICE EXPANSION FUNCTION

19.1 External Device Expansion Functions

The external device expansion functions are intended to connect external devices to areas other than the internal ROM, RAM and SFR. Connection of external devices uses port 4 to port 6. Port 4 to port 6 control address/data, read/write strobe, wait, address strobe, etc.

Table 19-1. Pin Functions in External Memory Expansion Mode

Pin Function at External Device Connection		Alternate Function
Name	Function	
AD0 to AD7	Multiplexed address/data bus	P40 to P47
A8 to A15	Address bus	P50 to P57
$\overline{RD}$	Read strobe signal	P64
$\overline{WR}$	Write strobe signal	P65
$\overline{WAIT}$	Wait signal	P66
ASTB	Address strobe signal	P67

Table 19-2. State of Port 4 to Port 6 Pins in External Memory Expansion Mode

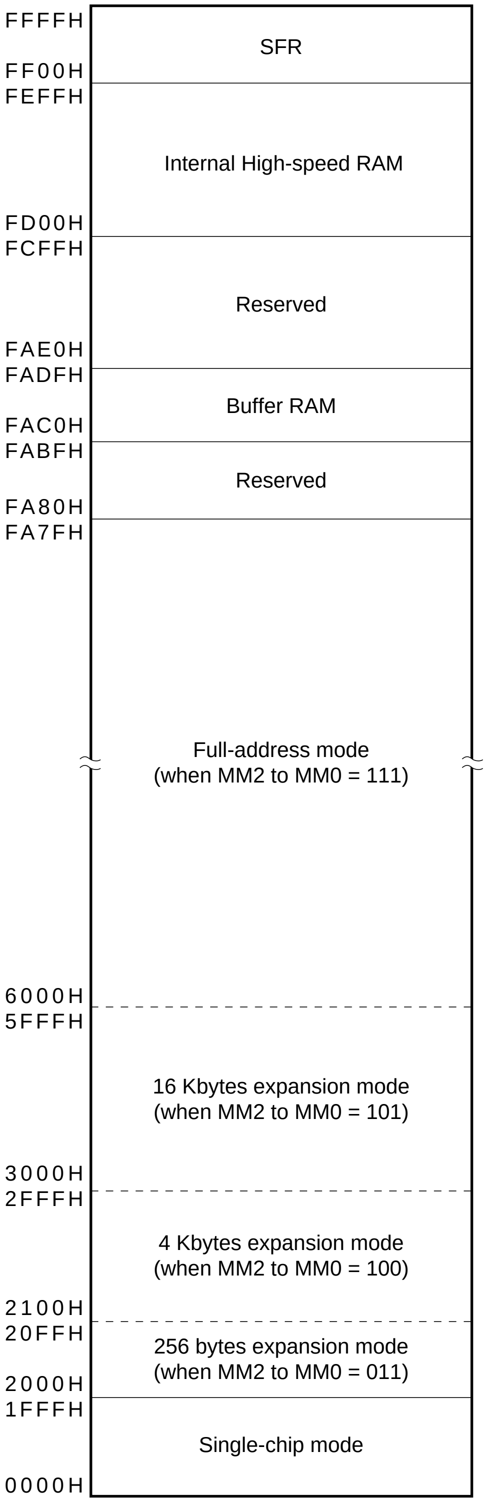
<table>
<tr><td rowspan="2">Port / External Expansion Mode</td><td>Port 4</td><td colspan="8">Port 5</td><td colspan="8">Port 6</td></tr>
<tr><td>0 to 7</td><td>0</td><td>1</td><td>2</td><td>3</td><td>4</td><td>5</td><td>6</td><td>7</td><td>0</td><td>1</td><td>2</td><td>3</td><td>4</td><td>5</td><td>6</td><td>7</td></tr>
<tr><td>Single-chip mode</td><td>Port</td><td colspan="8">Port</td><td colspan="8">Port</td></tr>
<tr><td>256 bytes expansion mode</td><td>Address/Data</td><td colspan="8">Port</td><td colspan="4">Port</td><td colspan="4">$\overline{RD}$, $\overline{WR}$, $\overline{WAIT}$, ASTB</td></tr>
<tr><td>4 Kbytes expansion mode</td><td>Address/Data</td><td colspan="4">Address</td><td colspan="4">Port</td><td colspan="4">Port</td><td colspan="4">$\overline{RD}$, $\overline{WR}$, $\overline{WAIT}$, ASTB</td></tr>
<tr><td>16 Kbytes expansion mode</td><td>Address/Data</td><td colspan="6">Address</td><td colspan="2">Port</td><td colspan="4">Port</td><td colspan="4">$\overline{RD}$, $\overline{WR}$, $\overline{WAIT}$, ASTB</td></tr>
<tr><td>Full-address mode</td><td>Address/Data</td><td colspan="8">Address</td><td colspan="4">Port</td><td colspan="4">$\overline{RD}$, $\overline{WR}$, $\overline{WAIT}$, ASTB</td></tr>
</table>

Caution When the external wait function is not used, the $\overline{WAIT}$ pin can be used as a port in all modes.

Memory maps when using the external device expansion function are as follows.

Figure 19-1. Memory Map when Using External Device Expansion Function (1/2)

(a) μPD78011B, 78011BY Memory Map



(b) μPD78012B, 78012BY Memory Map

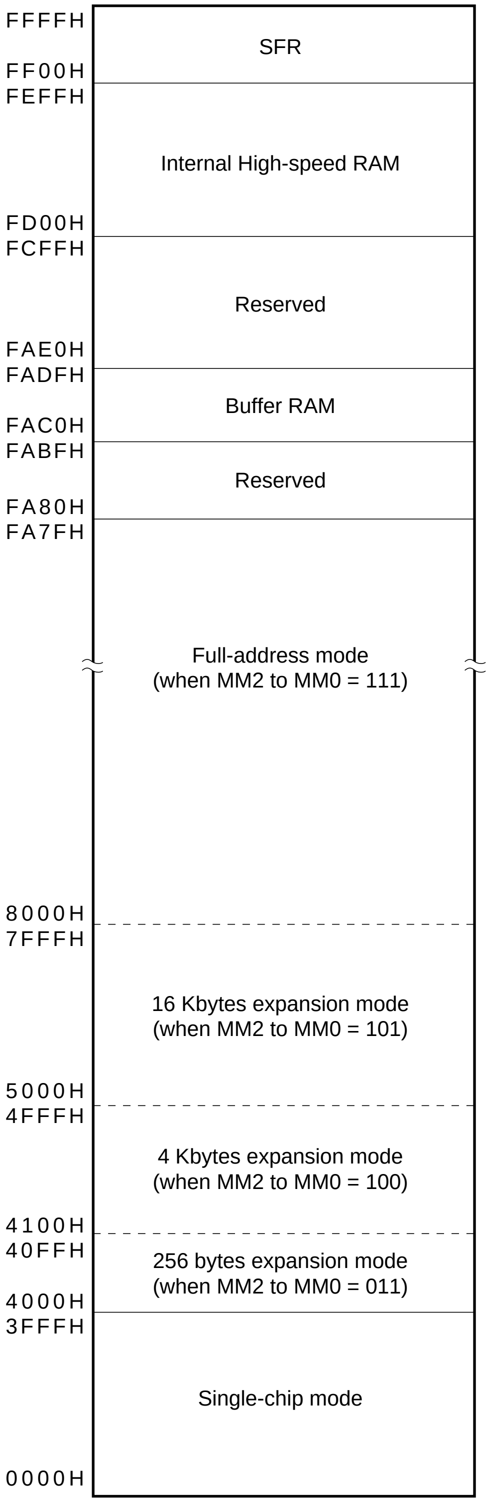
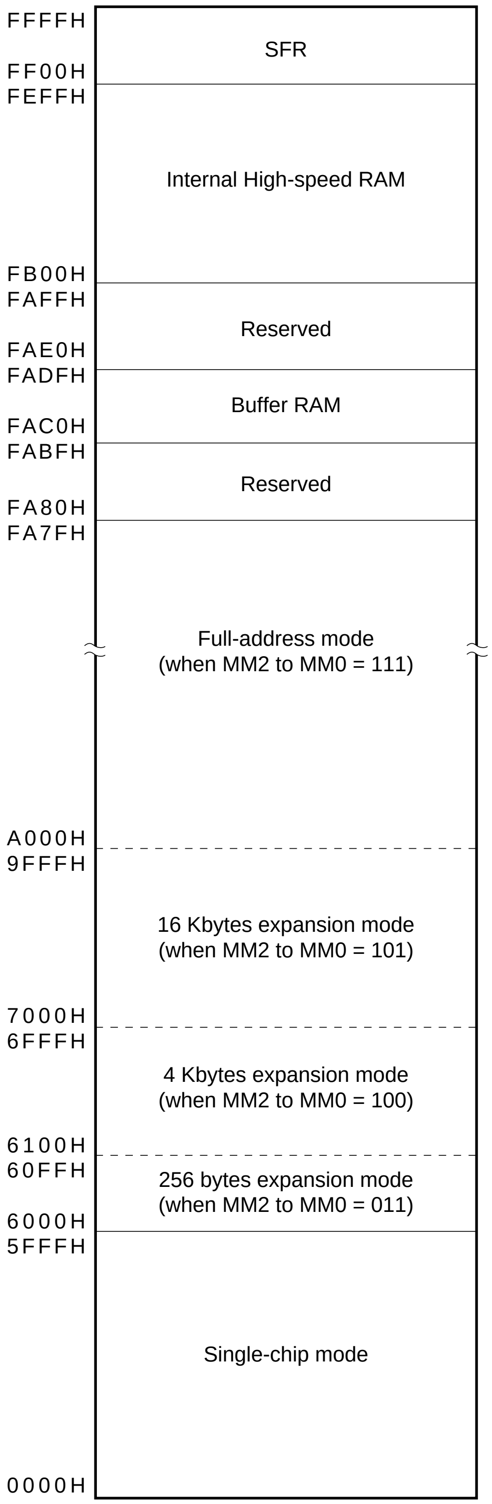
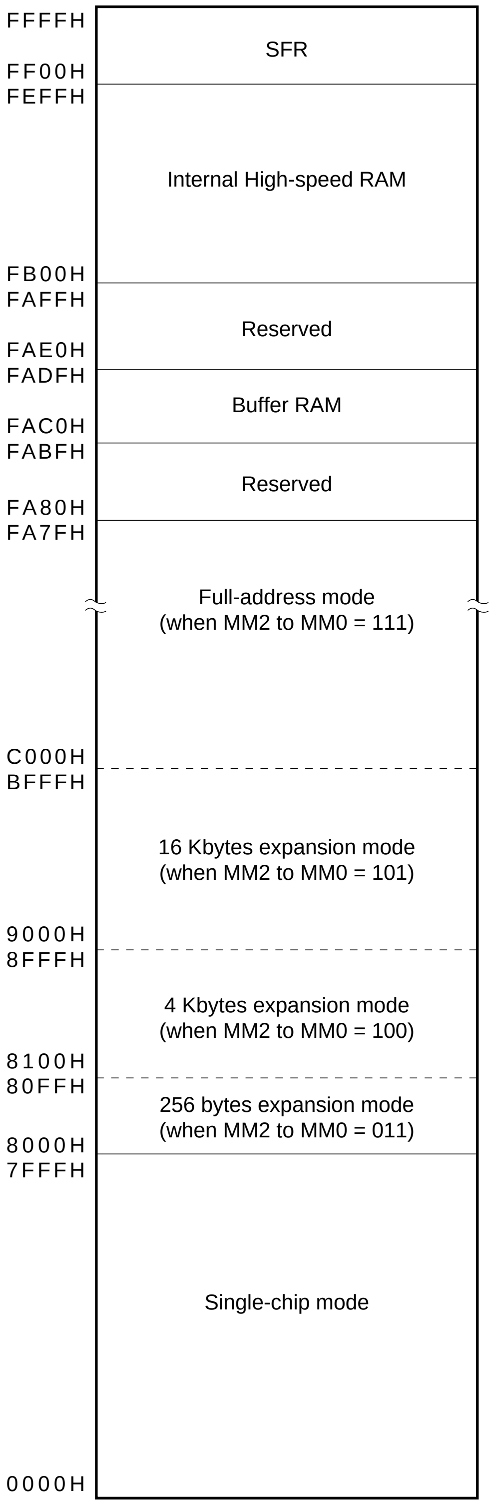


Figure 19-1. Memory Map when Using External Device Expansion Function (2/2)

(a) μPD78013, 78013Y Memory Map



(b) μPD78014, 78014Y, 78P014, 78P014Y Memory Map



19.2 External Device Expansion Control Register

The external device expansion function is controlled by the memory expansion mode register (MM). MM sets the wait count and external expansion area. MM also sets the input/output of port 4.

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

$\overline{\text{RESET}}$ input sets this register to 10H.

Figure 19-2. Memory Expansion Mode Register Format

Symbol	7	6	5	4	3	2	1	0	Address	When Reset	R/W
MM	0	0	PW1	PW0	0	MM2	MM1	MM0	FFF8H	10H	R/W

<table>
<tr><th rowspan="2">MM2</th><th rowspan="2">MM1</th><th rowspan="2">MM0</th><th rowspan="2" colspan="2">Single-chip/Memory Expansion Mode Selection</th><th colspan="6">P40 to P47, P50 to P57, P64 to P67 Pins Condition</th></tr>
<tr><th colspan="2">P40 to P47</th><th>P50 to P53</th><th>P54, P55</th><th>P56, P57</th><th>P64 to P67</th></tr>
<tr><td>0</td><td>0</td><td>0</td><td rowspan="2" colspan="2">Single-chip mode</td><td rowspan="2">Port mode</td><td>Input</td><td rowspan="2" colspan="4">Port mode</td></tr>
<tr><td>0</td><td>0</td><td>1</td><td>Output</td></tr>
<tr><td>0</td><td>1</td><td>1</td><td rowspan="4">Memory expansion mode</td><td>256 bytes mode</td><td rowspan="4" colspan="2">AD0 to AD7</td><td colspan="3">Port mode</td><td rowspan="4">P64 = $\overline{RD}$
P65 = $\overline{WR}$
P66 = $\overline{WAIT}$
P67 = ASTB</td></tr>
<tr><td>1</td><td>0</td><td>0</td><td>4 Kbytes mode</td><td rowspan="3">A8 to A11</td><td colspan="2">Port mode</td></tr>
<tr><td>1</td><td>0</td><td>1</td><td>16 Kbytes mode</td><td rowspan="2">A12, A13</td><td>Port mode</td></tr>
<tr><td>1</td><td>1</td><td>1</td><td>Full-address mode[Note]</td><td>A14, A15</td></tr>
<tr><td colspan="3">Other than above</td><td colspan="8">Setting prohibited</td></tr>
</table>

PW1	PW0	Wait Control
0	0	Without wait
0	1	With wait (1 wait state insertion)
1	0	Setting prohibited
1	1	Wait control with an external wait pin

Note The full-address mode allows external expansion to the entire 64 Kbytes address space except for the internal ROM, RAM and SFR areas and the reserved areas.

Remark P60 to P63 pins enter the port mode regardless of the single-chip mode or memory expansion mode.

19.3 External Device Expansion Function Timing

Timing control signal output pins in the external memory expansion mode are as follows.

(1) $\overline{RD}$ pin (Alternate function: P64)
Read strobe signal output pin. The read strobe signal is output in data access and instruction fetch from external memory.
During internal memory access, the read strobe signal is not output (maintains high level).

(2) $\overline{WR}$ pin (Alternate function: P65)
Write strobe signal output pin. The write strobe signal is output in data access to external memory.
During internal memory access, the write strobe signal is not output (maintains high level).

(3) $\overline{WAIT}$ pin (Alternate function: P66)
External wait signal input pin. When the external wait function is not used, the $\overline{WAIT}$ pin can be used as an input/output port.
During internal memory access, the external wait signal is ignored.

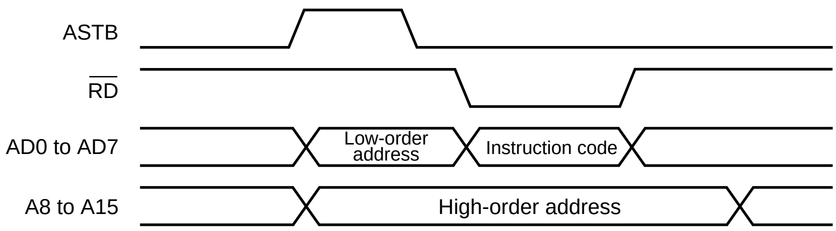
(4) ASTB pin (Alternate function: P67)
Address strobe signal output pin. Timing signal is always output regardless of the data access or instruction fetch from external memory. The address strobe signal is also output when the internal memory is accessed.

(5) AD0 to AD7, A8 to A15 pins (Alternate function: P40 to P47, P50 to P57)
Address/data signal output pins. Valid signals are output or input during instruction fetch and data access from/to external memory.
These signals change when the internal memory is accessed (output values are undefined).

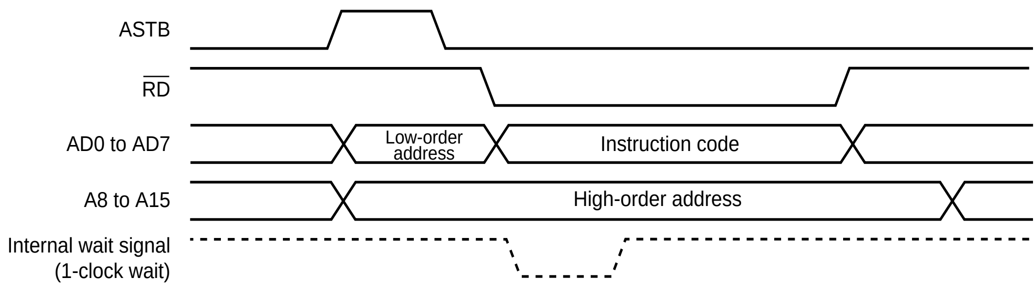
Timing charts are shown in Figures 19-3 to 19-6.

Figure 19-3. Instruction Fetch from External Memory

(a) When without Wait (PW1, PW0 = 0, 0) Setup



(b) When with Wait (PW1, PW0 = 0, 1) Setup



(c) When External Wait (PW1, PW0 = 1, 1) Setup

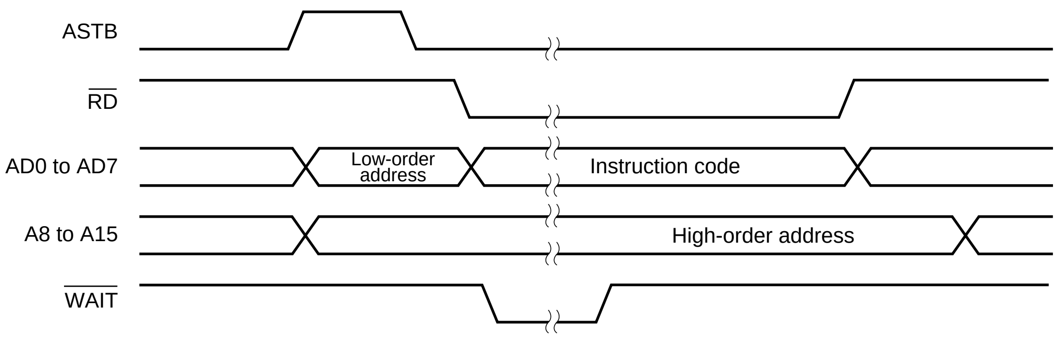
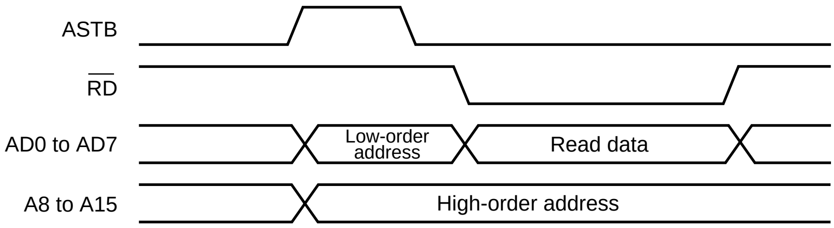
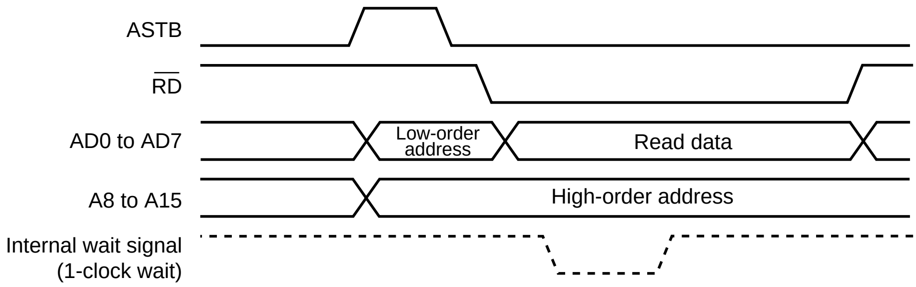


Figure 19-4. External Memory Read Timing

(a) When without Wait (PW1, PW0 = 0, 0) Setup



(b) When with Wait (PW1, PW0 = 0, 1) Setup



(c) When External Wait (PW1, PW0 = 1, 1) Setup

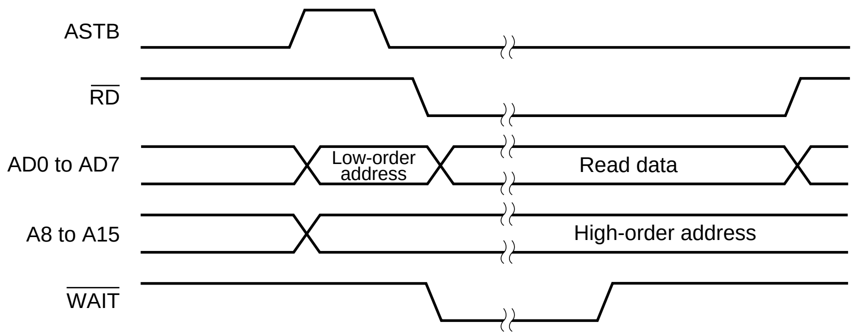
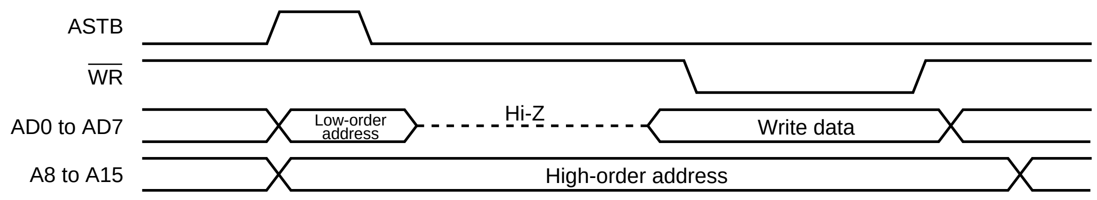
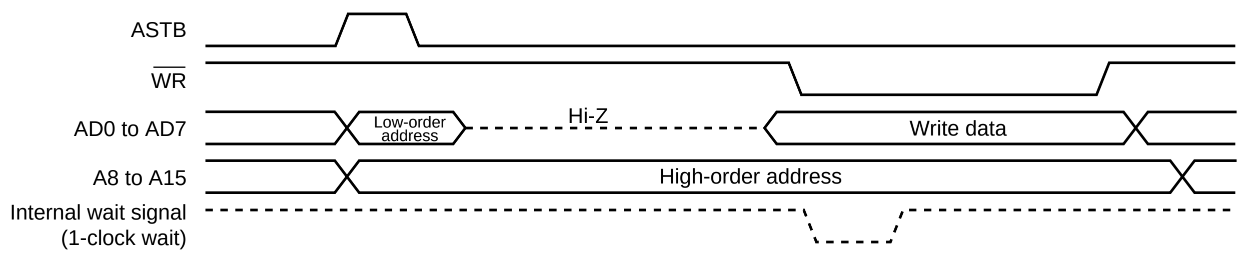


Figure 19-5. External Memory Write Timing

(a) When without Wait (PW1, PW0 = 0, 0) Setup



(b) When with Wait (PW1, PW0 = 0, 1) Setup



(c) When External Wait (PW1, PW0 = 1, 1) Setup

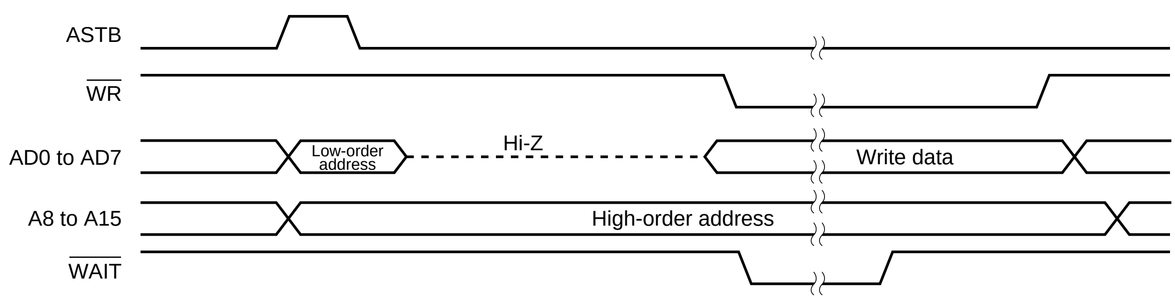
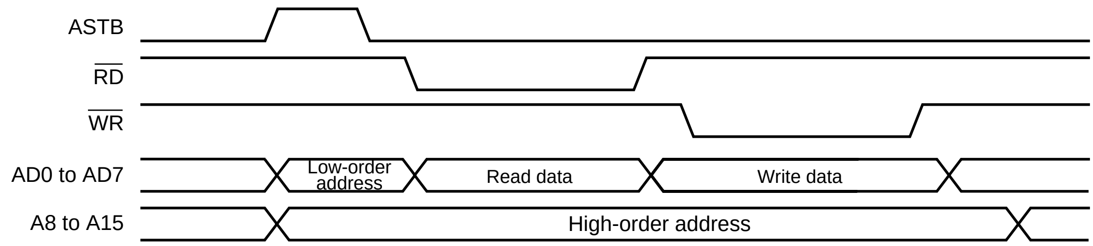
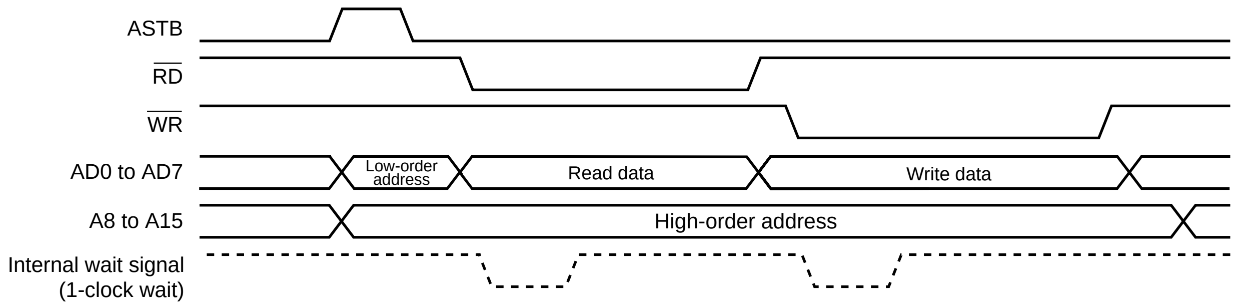


Figure 19-6. External Memory Read Modify Write Timing

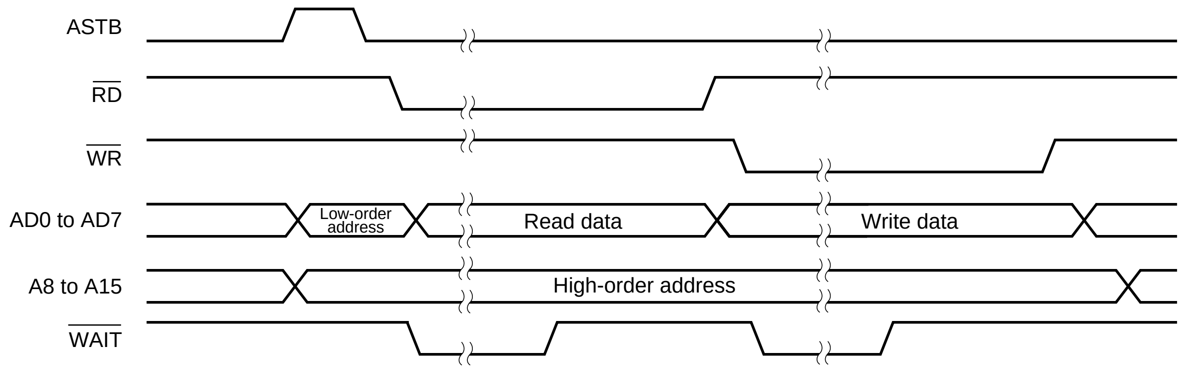
(a) When without Wait (PW1, PW0 = 0, 0) Setup



(b) When with Wait (PW1, PW0 = 0, 1) Setup



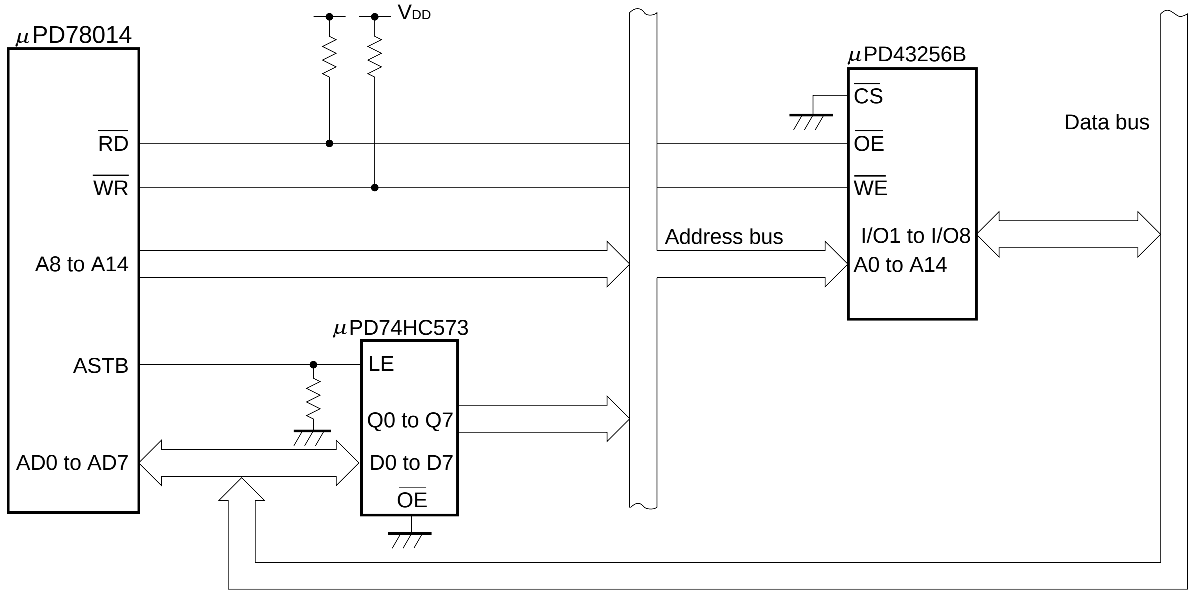
(c) When External Wait (PW1, PW0 = 1, 1) Setup



19.4 Example of Memory Connection

An example of external memory connection with the μPD78014 is shown in Figure 19-7. SRAM is used as the external memory in this application example. In addition, the external device expansion function is used in the full-address mode, and the internal ROM is allocated to addresses 0000H to 7FFFH (32 Kbytes), and SRAM is allocated to addresses above 8000H.

Figure 19-7. Example of Memory Connection with μPD78014



Caution At the external memory read modify write timing, the time from $\overline{RD}$ signal rising to write data output is very short, so that the write data sometimes conflicts with the output data from external memory (SRAM, etc). In this case, it is possible to avoid data conflict by not using the following instructions which generate read modify write timing.

XCH	A, !addr16	XCH	A, [HL + C]
XCH	A, [DE]	MOV1	[HL].bit, CY
XCH	A, [HL]	SET1	[HL].bit
XCH	A, [HL + byte]	CLR1	[HL].bit
XCH	A, [HL + B]	BTCLR	[HL].bit, $addr16

CHAPTER 20 STANDBY FUNCTION

20.1 Standby Function and Configuration

20.1.1 Standby function

The standby function is intended to decrease the 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 like watch operations.

(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} = 2 V) is possible. Thus, the STOP mode is effective to hold data memory contents with ultra-low current consumption.

Because this mode can be cleared 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 cleared, 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 input/output 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. To decrease power dissipation in A/D converter, clear bit 7 (CS) of A/D converter mode register (ADM) to 0 and stop the A/D conversion operation before executing the HALT or STOP instruction.

20.1.2 Standby function control register

A wait time after the STOP mode is cleared 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.

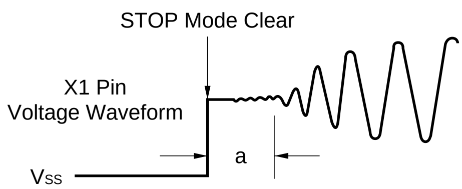
$\overline{\text{RESET}}$ input sets OSTS to 04H. Therefore, when the STOP mode is cleared with $\overline{\text{RESET}}$ input, the time till it is cleared is $2^{18}/f_X$.

Figure 20-1. Oscillation Stabilization Time Select Register Format

Symbol	7	6	5	4	3	2	1	0	Address	When Reset	R/W
OSTS	0	0	0	0	0	OSTS2	OSTS1	OSTS0	FFFAH	04H	R/W

OSTS2	OSTS1	OSTS0	Oscillation Stabilization Time Selection
0	0	0	$2^{13}/f_X$ (819 μs)
0	0	1	$2^{15}/f_X$ (3.28 ms)
0	1	0	$2^{16}/f_X$ (6.55 ms)
0	1	1	$2^{17}/f_X$ (13.1 ms)
1	0	0	$2^{18}/f_X$ (26.2 ms)
Other than above			Setting prohibited

Caution The wait time after the STOP mode is cleared does not include the time from STOP mode clear to clock oscillation start (see "a" below), regardless of clearance by $\overline{\text{RESET}}$ input or by interrupt request generation.



Remarks 1. f_X: Main system clock oscillation frequency

2. Values in parentheses apply to operation with f_X = 10.0 MHz

20.2 Standby Function Operations

20.2.1 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 20-1. HALT Mode Operating Status

Item \ HALT Mode Setting		When HALT instruction is executed during main system clock oscillation		When HALT instruction is executed during subsystem clock oscillation	
		Without subsystem clock[Note 1]	With subsystem clock[Note 2]	When main system clock oscillation continues	When main system clock oscillaton stops
Clock generator		Both main system clock and subsystem clock can be oscillated. Clock supply to the CPU stops.			
CPU		Operation stop.			
Port (output latch)		Status before HALT instruction execution is held.			
16-bit timer/event counter		Operation enabled.			Operation stop.
8-bit timer/event counter		Operation enabled.			Operation enabled when TI1 and TI2 are selected for the count clock.
Watchdog timer		Operation enabled.		Operation stop.	
A/D converter		Operation enabled.			Operation stop.
Watch timer		Operation enabled when $f_X/2^8$ is selected for the count clock.	Operation enabled.		Operation enabled when f_{XT} is selected for the count clock.
Serial Interface	Other than automatic transmit/receive function	Operation enabled.			Operation enabled when external SCK is selected.
	Automatic transmit/receive function	Operation stop.			
External interrupt	INTP0	Operation enabled when the clock ($f_X/2^6$ and $f_X/2^7$) for the peripheral hardware are selected as sampling clock.			Operation stop.
	INTP1 to INTP3	Operation enabled.			
Bus line in External Expansion	AD0 to AD7	High-impedance			
	A8 to A15	Status before HALT instruction execution is held.			
	ASTB	Low level			
	$\overline{WR}$, $\overline{RD}$	High level			
	$\overline{WAIT}$	High-impedance			

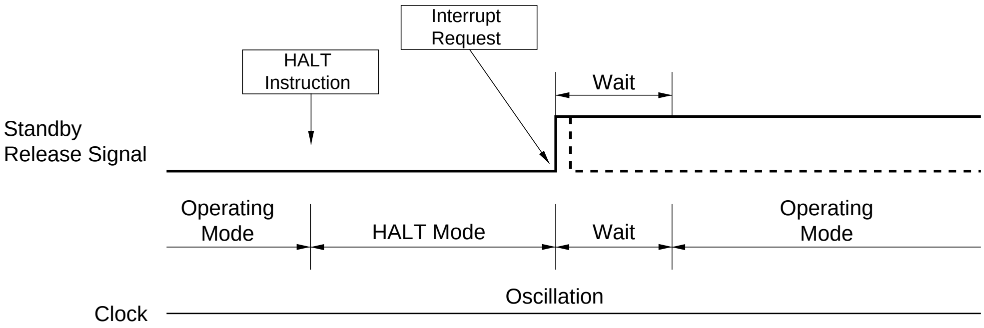
Notes 1. Includes case when an external clock is not supplied in the subsystem clock.

2. Includes case when an external clock is supplied in the subsystem clock.

(2) HALT mode clear
The HALT mode can be cleared with the following four types of sources.

(a) Clear upon unmasked interrupt request
The HALT mode is cleared when the unmasked interrupt request is generated. If interrupt request acknowledge is enabled, vectored interrupt service is carried out. If disabled, the next address instruction is executed.

Figure 20-2. HALT Mode Clear upon Interrupt Request Generation



Remarks 1. The broken line indicates the case when the interrupt request which has cleared the standby status is acknowledged.

2. Wait time will be as follows:
- When branched to the vector : 16.5 to 17.5 clocks
- When not branched to the vector : 4.5 to 5.5 clocks

(b) Clear upon non-maskable interrupt request
The HALT mode is cleared and vectored interrupt service is carried out when the non-maskable interrupt request is generated whether interrupt request acknowledge is enabled or disabled.

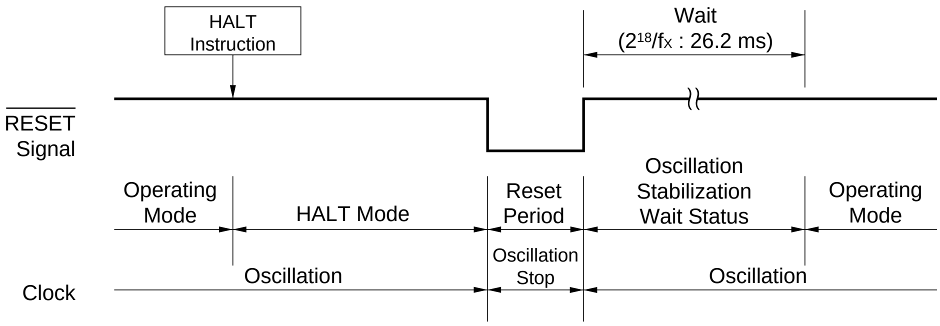
(c) Clear upon unmasked test input
The HALT mode is cleared when the unmasked test signal inputs and the next address instruction of the HALT instruction is executed.

(d) Clear upon $\overline{\text{RESET}}$ input

The HALT mode is cleared when the $\overline{\text{RESET}}$ signal inputs.

As is the case with normal reset operation, a program is executed after branch to the reset vector address.

Figure 20-3. HALT Mode Clear upon $\overline{\text{RESET}}$ Input



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

2. Values in parentheses apply to operation with f_X = 10.0 MHz

Table 20-2. Operation after HALT Mode Clear

Clear 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

Remark ×: don't care

20.2.2 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, X1 input is internally short-circuited to V_{SS} (ground potential) to suppress the leakage 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 clear 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 cleared 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 20-3. STOP Mode Operating Status

Item \ STOP Mode Setting		With subsystem clock	Without subsystem clock
Clock generator		Only main system clock stops oscillation.	
CPU		Operation stop.	
Output port (output latch)		Status before STOP instruction execution is held.	
16-bit timer/event counter		Operation stop.	
8-bit timer/event counter		Operation enabled when TI1 and TI2 are selected for the count clock.	
Watchdog timer		Operation stop.	
A/D converter			
Watch timer		Operation enabled when f_{XT} is selected for the count clock.	Operation stop.
Serial Interface	Other than automatic transmit/receive function	Operation enabled only when external input clock is selected as serial clock.	
	Automatic transmit/receive function	Operation stop.	
External interrupt	INTP0	Operation disabled.	
	INTP1 to INTP3	Operation enabled.	
Bus line in External Expansion	AD0 to AD7	High-impedance	
	A8 to A15	Status before STOP instruction execution is held.	
	ASTB	Low level	
	$\overline{WR}$, $\overline{RD}$	High level	
	$\overline{WAIT}$	High-impedance	

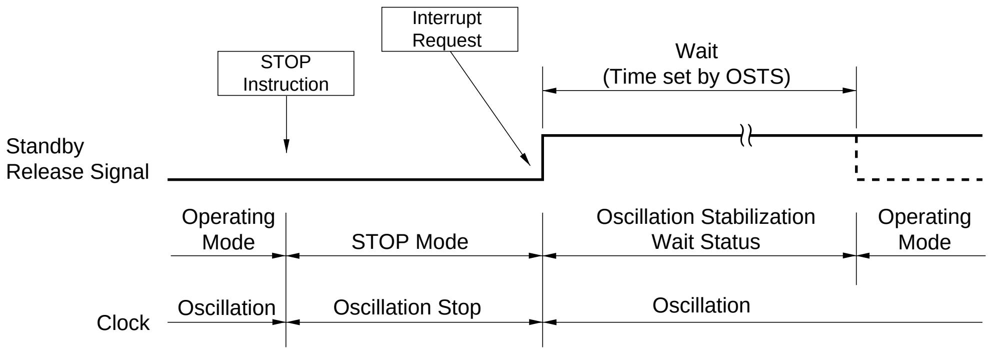
(2) STOP mode clear

The STOP mode can be cleared with the following three types of sources.

(a) Clear upon unmasked interrupt request

The STOP mode is cleared when the unmasked interrupt request is generated. 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 20-4. STOP Mode Clear upon Interrupt Request Generation



Remark The broken line indicates the case when the interrupt request which has cleared the standby status is acknowledged.

(b) Clear upon unmasked test input

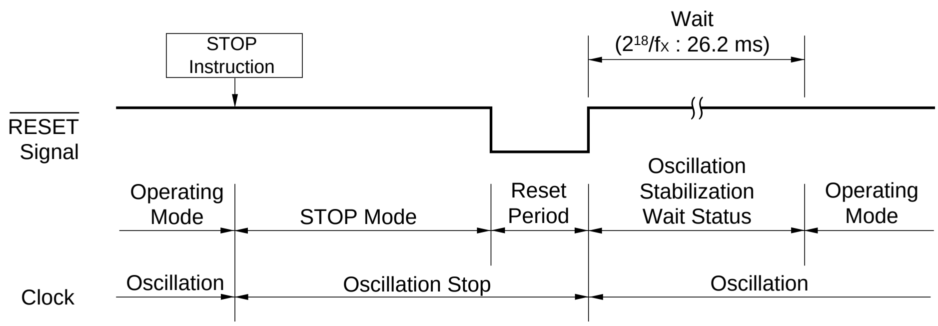
The STOP mode is cleared when the unmasked test signal inputs.

After the lapse of oscillation stabilization time, the instruction at the next address of the STOP instruction is executed.

(c) Clear upon $\overline{\text{RESET}}$ input

The STOP mode is cleared when the $\overline{\text{RESET}}$ signal inputs and after the lapse of oscillation stabilization time, reset operation is carried out.

Figure 20-5. STOP Mode Clear upon $\overline{\text{RESET}}$ Input



Remarks 1. fx: Main system clock oscillation frequency

2. Values in parentheses apply to operation with fx = 10.0 MHz

Table 20-4. Operation after STOP Mode Clear

Clear 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

Remark ×: don't care

CHAPTER 21 RESET FUNCTION

21.1 Reset Function

The following two operations are available to generate the reset function.

(1) External reset input with $\overline{\text{RESET}}$ pin
(2) Internal reset by watchdog timer inadvertent 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 21-1. Each pin has high impedance during reset input or during oscillation stabilization time just after reset clear.

When a high level is input to the $\overline{\text{RESET}}$ pin, the reset is cleared and program execution starts after the lapse of oscillation stabilization time ($2^{18}/f_X$). The reset applied by watchdog timer overflow is automatically cleared after a reset and program execution starts after the lapse of oscillation stabilization time ($2^{18}/f_X$) (see **Figures 21-2** to **21-4**).

Cautions 1. For an external reset, input a low level for 10 *m*s 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 cleared by reset, the STOP mode contents are held during reset input. However, the port pin becomes high-impedance.

Figure 21-1. Block Diagram of Reset Function

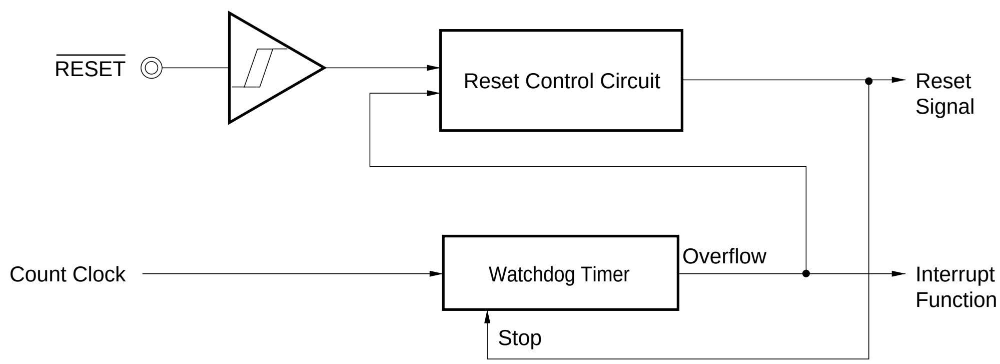


Figure 21-2. Timing of Reset by $\overline{\text{RESET}}$ Input

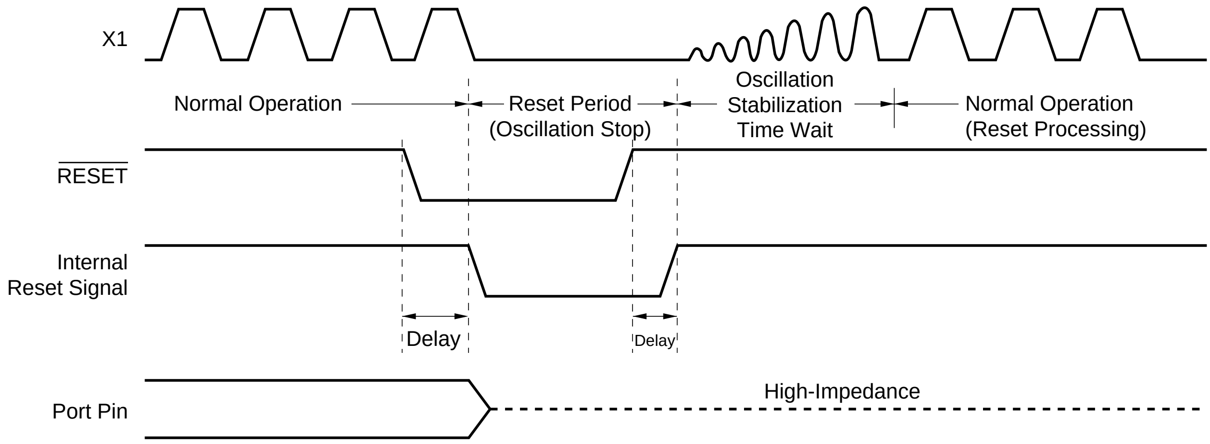


Figure 21-3. Timing of Reset due to Watchdog Timer Overflow

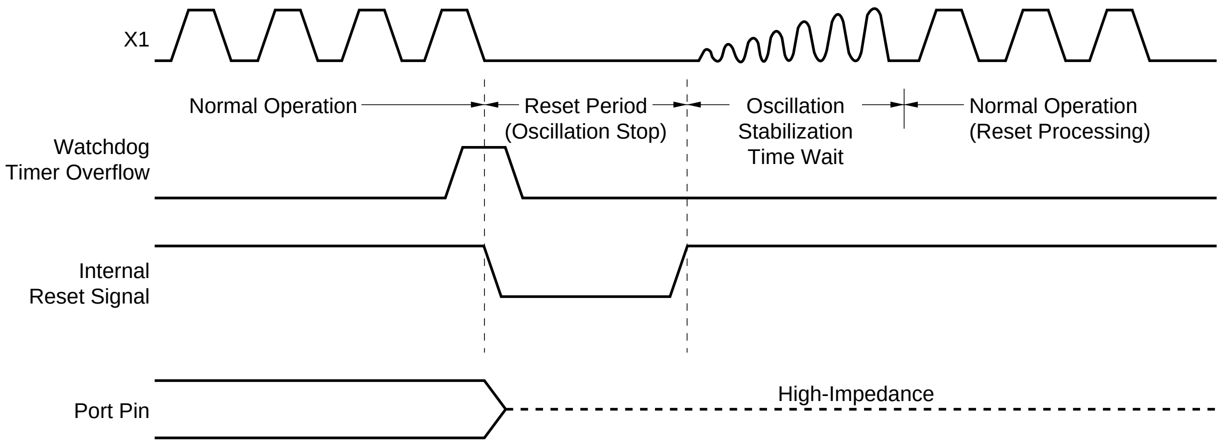


Figure 21-4. Timing of Reset in STOP Mode by $\overline{\text{RESET}}$ Input

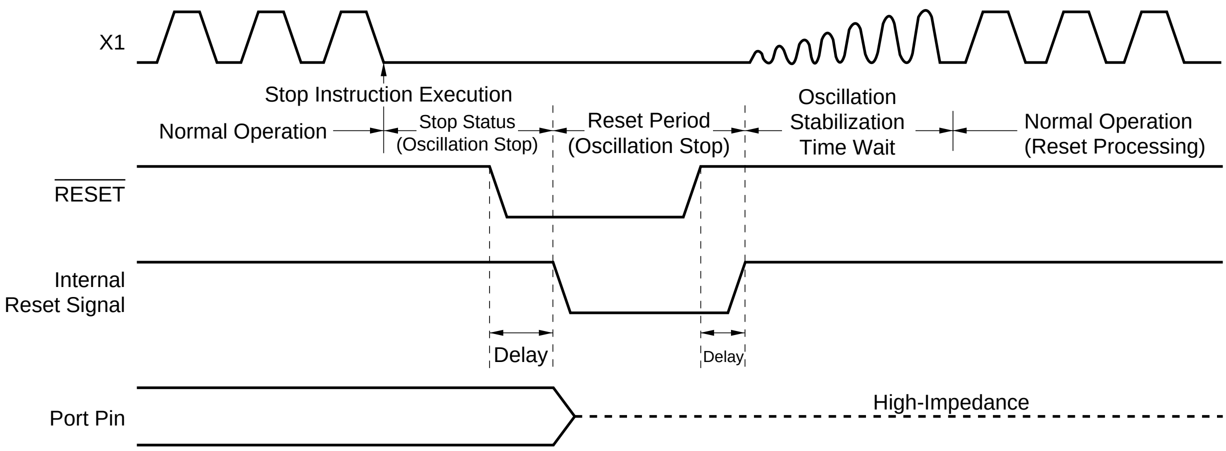


Table 21-1. Hardware Status after Reset (1/2)

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 register	Undefined[Note 2]
Port (Output latch)	Ports 0 to 3 (P0 to P3)	00H
	Ports 4 to 6 (P4 to P6)	Undefined
Port mode register	(PM0)	1FH
	(PM1, PM2, PM3, PM5, PM6)	FFH
Pull-up resistor option register (PUO)		00H
Processor clock control register (PCC)		04H
Memory expansion mode register (MM)		10H
Internal memory size switching register (IMS)		**Note 3**
Oscillation stabilization time select register (OSTS)		04H
16-bit timer/event counter	Timer register (TM0)	0000H
	Compare register (CR00)	Undefined
	Capture register (CR01)	Undefined
	Clock select register (TCL0)	00H
	Mode control register (TMC0)	00H
	Output control register (TOC0)	00H
8-bit timer/event counter	Timer registers (TM1, TM2)	00H
	Compare registers (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 status become undefined. All other hardware status remain unchanged after reset.

2. When reset in the standby mode, pre-reset status is held even after reset.

3. The after-reset values of the memory size switching register (IMS) depend on products.
μPD78011B, 78011BY: 42H, μPD78012B, 78012BY: 44H, μPD78013, 78013Y: C6H,
μPD78014, 78014Y, 78P014, 78P014Y: C8H

Table 21-1. Hardware Status after Reset (2/2)

Hardware		Status after Reset
Watch timer	Mode control register (TMC2)	00H
Watch timer / Watchdog timer	Clock select register (TCL2)	00H
Watchdog timer	Mode register (WDTM)	00H
Serial interface	Clock select register (TCL3)	88H
	Shift registers (SIO0, SIO1)	Undefined
	Mode registers (CSIM0, CSIM1)	00H
	Serial bus interface control register (SBIC)	00H
	Slave address register (SVA)	Undefined
	Automatic data transmit/receive control register (ADTC)	00H
	Automatic data transmit/receive address pointer (ADTP)	00H
	Interrupt timing specify register (SINT)	00H
A/D converter	Mode register (ADM)	01H
	Conversion result register (ADCR)	Undefined
	Input select register (ADIS)	00H
Interrupt	Request flag registers (IF0L, IF0H)	00H
	Mask flag registers (MK0L, MK0H)	FFH
	Priority specify flag registers (PR0L, PR0H)	FFH
	External interrupt mode register (INTM0)	00H
	Key return mode register (KRM)	02H
	Sampling clock select register (SCS)	00H

CHAPTER 22 μPD78P014, 78P014Y

The μPD78P014 and 78P014Y are versions which incorporate a one-time programmable PROM or an EPROM enabled for program write, erase and rewrite. Table 22-1 lists differences between μPD78P014, 78P014Y and mask ROM version.

Table 22-1. Differences between μPD78P014, 78P014Y, and Mask ROM Version

Item	μPD78P014, 78P014Y	Mask ROM Version
Internal ROM configuration	One-time PROM/EPROM	Mask ROM
Internal ROM capacity	32 Kbytes	μPD78011B, 78011BY: 8 Kbytes μPD78012B, 78012BY: 16 Kbytes μPD78013, 78013Y: 24 Kbytes μPD78014, 78014Y: 32 Kbytes
Internal high-speed RAM capacity	1024 bytes	μPD78011B, 78011BY: 512 bytes μPD78012B, 78012BY: 512 bytes μPD78013, 78013Y: 1024 bytes μPD78014, 78014Y: 1024 bytes
Internal ROM and internal high-speed RAM by memory size select register	Enable[Note]	Disable
IC pin	None	Available
V_{PP} pin	Available	None
P60 to P63 pin on-chip pull-up resistor internal mask option	None	Available
Electrical specification	Refer to the data sheet for each part number.	

Note When $\overline{RESET}$ is input, the internal PROM capacity is set to 32 Kbytes, internal high-speed RAM capacity to 1024 bytes.

★ **Caution The noise resistance and noise radiation differs between PROM versions and mask ROM versions. If considering replacing PROM versions with mask ROM versions during the process from trial manufacturing to mass production, evaluate the CS versions (not ES versions) of mask ROM versions carefully.**

22.1 Internal Memory Size Switching Register

The μPD78P014 and 78P014Y can select the internal memory capacity with the internal memory size switching register (IMS). The same memory mapping as that of the mask ROM version with a different internal memory capacity is possible by setting IMS.

In order to make the memory maps of μPD78P014 and 78P014Y identical to a mask ROM version, the value at the time the mask ROM version is reset must be set to IMS.

For the mask ROM version, IMS does not need to be set.

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

The value of IMS becomes the value shown in Table 22-2, at $\overline{RESET}$.

Caution To use a mask ROM version, do not set a value other than those shown in Table 22-2 to IMS.

Figure 22-1. Internal Memory Size Switching Register Format

Symbol	7	6	5	4	3	2	1	0	Address	When Reset	R/W
IMS	RAM2	RAM1	RAM0	0	ROM3	ROM2	ROM1	ROM0	FFF0H	**Note**	W

ROM3	ROM2	ROM1	ROM0	Internal ROM Capacity Selection
0	0	0	1	4 Kbytes
0	0	1	0	8 Kbytes
0	1	0	0	16 Kbytes
0	1	1	0	24 Kbytes
1	0	0	0	32 Kbytes
Other than above				Setting prohibited

RAM2	RAM1	RAM0	Internal High-Speed RAM Capacity Selection
0	0	0	768 bytes
0	0	1	640 bytes
0	1	0	512 bytes
0	1	1	384 bytes
1	0	0	256 bytes
1	0	1	Setting prohibited
1	1	0	1024 bytes
1	1	1	896 bytes

Note The value of the memory size switching register at reset depends on the model (see **Table 22-2**).

Table 22-2. Internal Memory Size Switching Register Value at Reset

Part Number	Value at Reset	Part Number	Value at Reset
μPD78001B, 78001BY	82H	μPD78013, 78013Y	C6H
μPD78002B, 78002BY	64H	μPD78014, 78014Y	C8H
μPD78011B, 78011BY	42H	μPD78P014, 78P014Y	
μPD78012B, 78012BY	44H		

22.2 PROM Programming

The μPD78P014 and 78P014Y incorporate a 32K-byte PROM as program memory. When programming the μPD78P014 and 78P014Y, the PROM programming mode is set by means of the V_{PP} pin and the $\overline{\text{RESET}}$ pin. For the connection of unused pins, see **1.5** or **2.5 Pin Configurations (Top View), (2) PROM programming mode**.

22.2.1 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 μPD78P014 and 78P014Y are set to the PROM programming mode. This is one of the operating modes shown in Table 22-3 below according to the setting of the $\overline{\text{CE}}$ and $\overline{\text{OE}}$ pins.

The PROM contents can be read by setting the read mode.

Table 22-3. PROM Programming Operating Modes

Pin / Operating mode	$\overline{\text{RESET}}$	V_{PP}	V_{DD}	$\overline{\text{CE}}$	$\overline{\text{OE}}$	D0 to D7
Program write	L	+12.5 V	+6 V	L	H	Data input
Program verify				H	L	Data output
Program inhibit				H	H	High-impedance
Read		+5 V	+5 V	L	L	Data output
Output disabled				L	H	High-impedance
Standby				H	×	High-impedance

Remark ×: L or H

22.2.2 PROM write procedure

PROM contents can be written using the following procedure and high-speed writing is enabled.

(1) Fix the $\overline{RESET}$ pin low, and supply +5 V to the V_{PP} pin. Unused pins are handled as shown in **1.5** or **2.5 Pin Configuration, (Top View), (2) PROM programming mode.**
(2) Supply +6 V to the V_{DD} pin and +12.5 V to the V_{PP} pin.
(3) Supply the initial address.
(4) Supply the written data.
(5) Supply the 1 ms program pulse (active low) to the $\overline{CE}$ pin.
(6) Verify mode. If written, proceed to step (8). If not written, repeat steps (4) through (6). If you repeat 25 times and it can't be written, proceed to (7).
(7) Stop the write operation as a defective device.
(8) Supply write data and repeat times from (4) through (6) × 3 ms program pulse (additional write).
(9) Increment the address.
(10) Repeat steps (4) through (9) to the last address.

The timing for steps (2) through (8) above is shown Figure 22-2.

Figure 22-2. PROM Write/Verify Timing

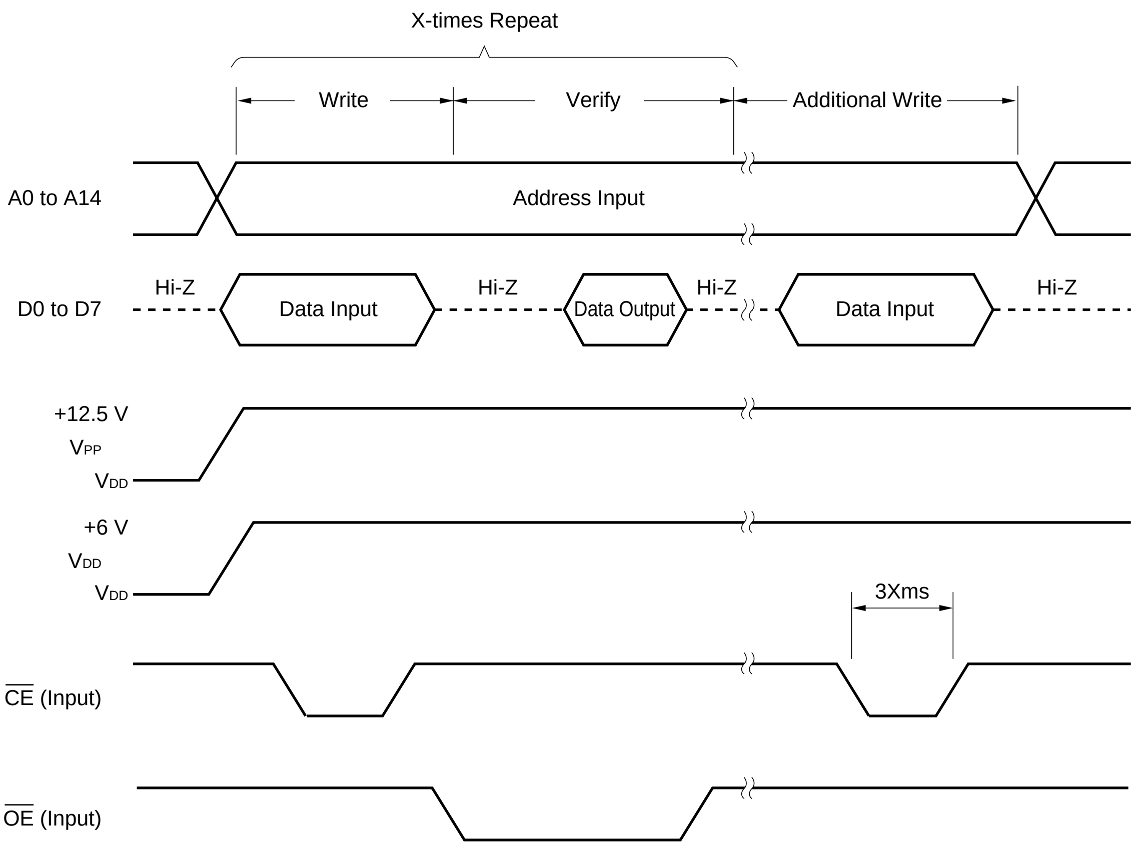
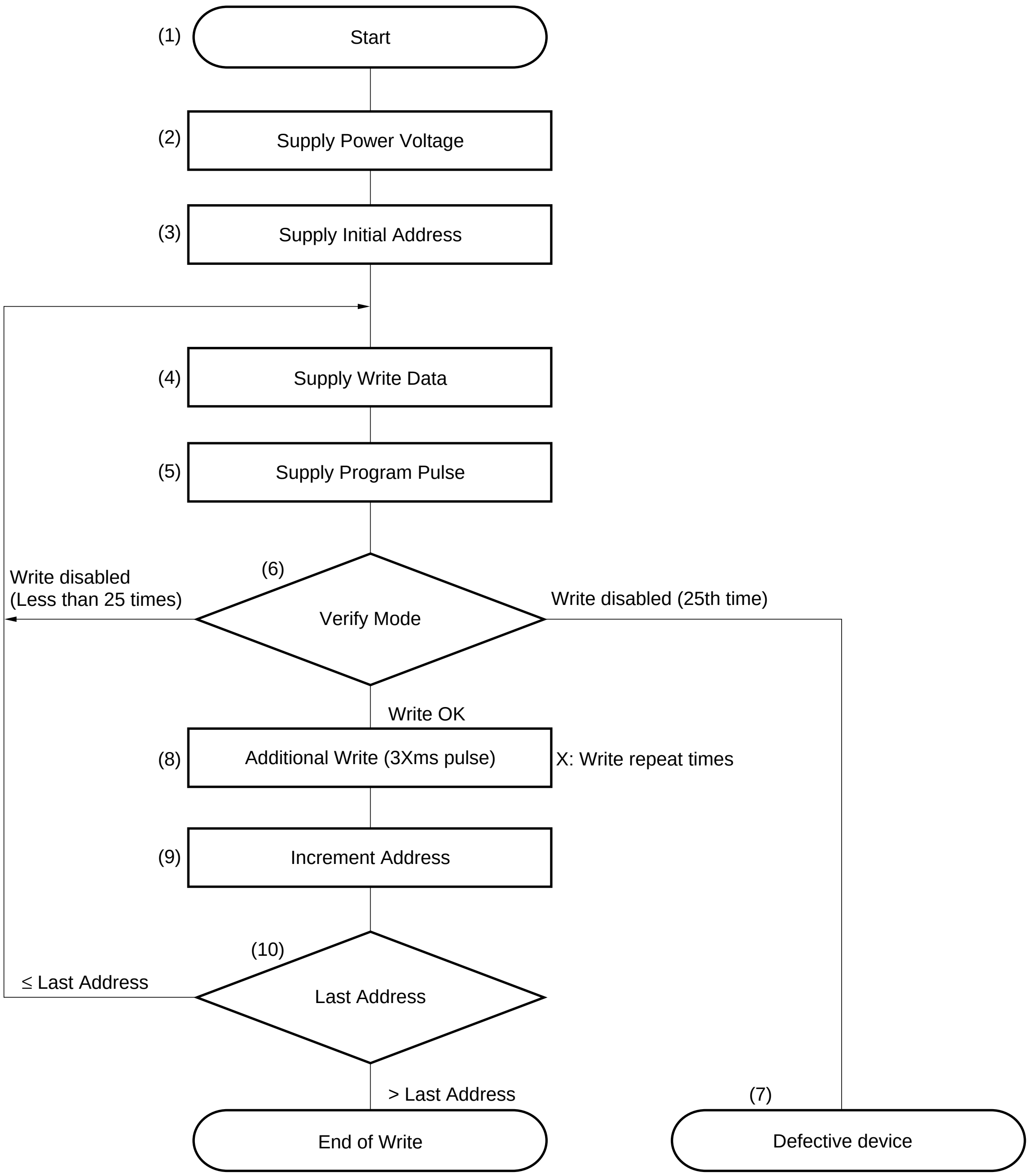


Figure 22-3. Write Procedure Flowchart



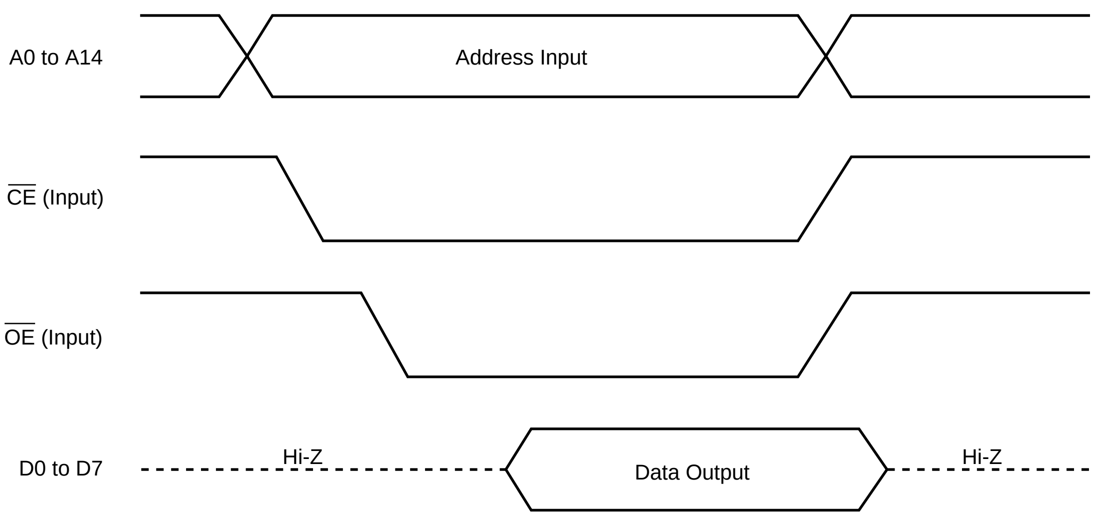
22.2.3 PROM read procedure

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 **1.5** or **2.5 Pin Configurations (Top View), (2) PROM programming mode.**
(2) Supply +5 V to the V_{DD} and V_{PP} pins.
(3) Input address of data to be read to pins A0 to A16.
(4) Read mode.
(5) Output data to pins D0 to D7.

The timing for steps (2) to (5) above is shown in Figure 22 to 4.

Figure 22-4. PROM Read Timing



22.3 Erasure Characteristics (for μPD78P014DW, 78P014YDW)

Through exposure to light having a very short wavelength of less than 400 nm, contents of the programmed data can be erased (FFH).

The μPD78P014DW and 78P014YDW program memory contents are usually erased by ultraviolet rays having a wavelength of 254 nm. The amount of exposure needed to completely erase the μPD78P014DW and 78P014YDW is at least 15 W × s/cm^2 (ultraviolet ray strength × erasure time). The erasure time is about 15 to 20 minutes (when using a 12000 μW/cm^2 ultraviolet ray lamp). However, the required erasure time may be longer in some cases, such as when there has been deterioration of ultraviolet ray lamp performance or when the erasure window is dirty. When erasing, place the μPD78P014DW and 78P014YDW within 2.5 cm of the ultraviolet ray lamp. If a filter has been attached to the ultraviolet ray lamp, remove the filter before erasing.

22.4 Opaque Film on Erasure Window (for μPD78P014DW, 78P014YDW)

When erasing EPROM contents, be sure to cover the erasure window with a shading film to prevent unintentional erasure of EPROM contents by light source other than the ultraviolet ray lamp and to prevent a light source from unintentionally affecting internal circuits other than the EPROM.

22.5 Screening of One-Time PROM Versions

Because of their construction, one-time PROM versions (μPD78P014CW, 78P014YCW, 78P014GC-AB8, 78P014YGC-AB8) cannot be fully tested by NEC before shipment. After the necessary data has been written, it is recommended that screening is implemented in which PROM verification is performed after high-temperature storage under the following conditions.

Storage Temperature	Storage Time
125°C	24 hours

NEC is offering one-time PROM writing to marking, screening and verifying with charge under the name QTOP™ microcontroller. Please contact an NEC sales representative for the details.

[MEMO]

CHAPTER 23 INSTRUCTION SET

The instruction sets for the μPD78014 and 78014Y Subseries are described in the following pages. For the details of operations and mnemonics (instruction codes) of each instruction, refer to the **78K/0 Series User's Manual, Instructions (U12326E).**

23.1 Legend

23.1.1 Operand identifiers and description methods

Operands are described in the "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 are 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 23-1. Operand Identifiers and Description Methods

Identifier	Description Method
r	X (R0), A (R1), C (R2), B (R3), E (R4), D (R5), L (R6), H (R7)
rp	AX (RP0), BC (RP1), DE (RP2), HL (RP3)
sfr	Special function register symbol[Note]
sfrp	Special function register symbols (16-bit manipulatable register even addresses only)[Note]
saddr	FE20H to FF1FH Immediate data or labels
saddrp	FE20H to FF1FH Immediate data or labels (even addresses only)
addr16	0000H to FFFFH Immediate data or labels
	(Only even addresses for 16-bit data transfer instructions)
addr11	0800H to 0FFFH Immediate data or labels
addr5	0040H to 007FH Immediate data or labels (even addresses only)
word	16-bit immediate data or label
byte	8-bit immediate data or label
bit	3-bit immediate data or label
RBn	RB0 to RB3

Note FFD0H to FFDFH are not addressable.

Remark For special-function register symbols, refer to **Table 5-5 Special Function Register List**.

23.1.2 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
NMIS	:	Non-maskable interrupt servicing flag
( )	:	Memory contents indicated by address or register contents in parentheses
X_H, X_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)

23.1.3 Description of "flag operation" column

(Blank)	:	Unchanged
0	:	Cleared to 0
1	:	Set to 1
×	:	Set/cleared according to the result
R	:	Previously saved value is restored

23.2 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	8	—	r ← byte	
		saddr, #byte	3	12	14	(saddr) ← byte	
		sfr, #byte	3	—	14	sfr ← byte	
		A, r Note 3	1	4	—	A ← r	
		r, A Note 3	1	4	—	r ← A	
		A, saddr	2	8	10	A ← (saddr)	
		saddr, A	2	8	10	(saddr) ← A	
		A, sfr	2	—	10	A ← sfr	
		sfr, A	2	—	10	sfr ← A	
		A, !addr16	3	16	18 + 2n	A ← (addr16)	
		!addr16, A	3	16	18 + 2m	(addr16) ← A	
		PSW, #byte	3	—	14	PSW ← byte	× × ×
		A, PSW	2	—	10	A ← PSW	
		PSW, A	2	—	10	PSW ← A	× × ×
		A, [DE]	1	8	10 + 2n	A ← (DE)	
		[DE], A	1	8	10 + 2m	(DE) ← A	
		A, [HL]	1	8	10 + 2n	A ← (HL)	
		[HL], A	1	8	10 + 2m	(HL) ← A	
		A, [HL+byte]	2	16	18 + 2n	A ← (HL+byte)	
		[HL+byte], A	2	16	18 + 2m	(HL+byte) ← A	
		A, [HL+B]	1	12	14 + 2n	A ← (HL+B)	
		[HL+B], A	1	12	14 + 2m	(HL+B) ← A	
		A, [HL+C]	1	12	14 + 2n	A ← (HL+C)	
		[HL+C], A	1	12	14 + 2m	(HL+C) ← A	

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

Remarks 1. One instruction clock cycle is one cycle of the CPU clock (f_{CPU}) selected by processor clock control register (PCC).

2. Clock indicates when a program is in the internal ROM area.

3. n is the number of waits when the external memory expansion area is read.

4. m is the number of waits when the external memory expansion area is written.

Instruction Group	Mnemonic	Operands	Byte	Clock		Operation	Flag
				Note 1	Note 2		Z AC CY
8-bit data transfer	XCH	A, r Note 3	1	4	—	A ↔ r	
		A, saddr	2	8	12	A ↔ (saddr)	
		A, sfr	2	—	12	A ↔ sfr	
		A, !addr16	3	16	20 + 2n + 2m	A ↔ (addr16)	
		A, [DE]	1	8	12 + 2n + 2m	A ↔ (DE)	
		A, [HL]	1	8	12 + 2n + 2m	A ↔ (HL)	
		A, [HL+byte]	2	16	20 + 2n + 2m	A ↔ (HL+byte)	
		A, [HL+B]	2	16	20 + 2n + 2m	A ↔ (HL+B)	
		A, [HL+C]	2	16	20 + 2n + 2m	A ↔ (HL+C)	
16-bit data transfer	MOVW	rp, #word	3	12	—	rp ← word	
		saddrp, #word	4	16	20	(saddrp) ← word	
		sfrp, #word	4	—	20	sfrp ← word	
		AX, saddrp	2	12	16	AX ← (saddrp)	
		saddrp, AX	2	12	16	(saddrp) ← AX	
		AX, sfrp	2	—	16	AX ← sfrp	
		sfrp, AX	2	—	16	sfrp ← AX	
		AX, rp Note 4	1	8	—	AX ← rp	
		rp, AX Note 4	1	8	—	rp ← AX	
		AX, !addr16	3	20	24 + 4n	AX ← (addr16)	
		!addr16, AX	3	20	24 + 4m	(addr16) ← AX	
	XCHW	AX, rp Note 4	1	8	—	AX ↔ rp	

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
4. Only when rp = BC, DE or HL

Remarks 1. One instruction clock cycle is one cycle of the CPU clock (f_{CPU}) selected by processor clock control register (PCC).
2. Clock indicates when a program is in the internal ROM area.
3. n is the number of waits when the external memory expansion area is read.
4. m is the number of waits when the external memory expansion area is written.

Instruction Group	Mnemonic	Operands	Byte	Clock		Operation	Flag		
				Note 1	**Note 2**		Z	AC	CY
8-bit Operation	ADD	A, #byte	2	8	—	A, CY ← A+byte	×	×	×
		saddr, #byte	3	12	16	(saddr), CY ← (saddr)+byte	×	×	×
		A, r **Note 3**	2	8	—	A, CY ← A+r	×	×	×
		r, A	2	8	—	r, CY ← r+A	×	×	×
		A, saddr	2	8	10	A, CY ← A+(saddr)	×	×	×
		A, !addr16	3	16	18 + 2n	A, CY ← A+(addr16)	×	×	×
		A, [HL]	1	8	10 + 2n	A, CY ← A+(HL)	×	×	×
		A, [HL + byte]	2	16	18 + 2n	A, CY ← A+(HL+byte)	×	×	×
		A, [HL + B]	2	16	18 + 2n	A, CY ← A+(HL+B)	×	×	×
		A, [HL + C]	2	16	18 + 2n	A, CY ← A+(HL+C)	×	×	×
	ADDC	A, #byte	2	8	—	A, CY ← A+byte+CY	×	×	×
		saddr, #byte	3	12	16	(saddr), CY ← (saddr)+byte+CY	×	×	×
		A, r **Note 3**	2	8	—	A, CY ← A+r+CY	×	×	×
		r, A	2	8	—	r, CY ← r+A+CY	×	×	×
		A, saddr	2	8	10	A, CY ← A+(saddr)+CY	×	×	×
		A, !addr16	3	16	18 + 2n	A, CY ← A+(addr16)+CY	×	×	×
		A, [HL]	1	8	10 + 2n	A, CY ← A+(HL)+CY	×	×	×
		A, [HL+byte]	2	16	18 + 2n	A, CY ← A+(HL+byte)+CY	×	×	×
		A, [HL+B]	2	16	18 + 2n	A, CY ← A+(HL+B)+CY	×	×	×
		A, [HL+C]	2	16	18 + 2n	A, CY ← A+(HL+C)+CY	×	×	×
	SUB	A, #byte	2	8	—	A, CY ← A–byte	×	×	×
		saddr, #byte	3	12	16	(saddr), CY ← (saddr)–byte	×	×	×
		A, r **Note 3**	2	8	—	A, CY ← A–r	×	×	×
		r, A	2	8	—	r, CY ← r–A	×	×	×
		A, saddr	2	8	10	A, CY ← A–(saddr)	×	×	×
		A, !addr16	3	16	18 + 2n	A, CY ← A–(addr16)	×	×	×
		A, [HL]	1	8	10 + 2n	A, CY ← A–(HL)	×	×	×
		A, [HL+byte]	2	16	18 + 2n	A, CY ← A–(HL+byte)	×	×	×
		A, [HL+B]	2	16	18 + 2n	A, CY ← A–(HL+B)	×	×	×
		A, [HL+C]	2	16	18 + 2n	A, CY ← 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

Remarks 1. One instruction clock cycle is one cycle of the CPU clock (f_{CPU}) selected by processor clock control register (PCC).

2. Clock indicates when a program is in the internal ROM area.

3. n is the number of waits when the external memory expansion area is read.

Instruction Group	Mnemonic	Operands	Byte	Clock		Operation	Flag		
				Note 1	Note 2		Z	AC	CY
8-bit Operation	SUBC	A, #byte	2	8	—	A, CY ← A–byte–CY	×	×	×
		saddr, #byte	3	12	16	(saddr), CY ← (saddr)–byte–CY	×	×	×
		A, r Note 3	2	8	—	A, CY ← A–r–CY	×	×	×
		r, A	2	8	—	r, CY ← r–A–CY	×	×	×
		A, saddr	2	8	10	A, CY ← A–(saddr)–CY	×	×	×
		A, !addr16	3	16	18 + 2n	A, CY ← A–(addr16)–CY	×	×	×
		A, [HL]	1	8	10 + 2n	A, CY ← A–(HL)–CY	×	×	×
		A, [HL+byte]	2	16	18 + 2n	A, CY ← A–(HL+byte)–CY	×	×	×
		A, [HL+B]	2	16	18 + 2n	A, CY ← A–(HL+B)–CY	×	×	×
		A, [HL+C]	2	16	18 + 2n	A, CY ← A–(HL+C)–CY	×	×	×
	AND	A, #byte	2	8	—	A ← A∧byte	×		
		saddr, #byte	3	12	16	(saddr) ← (saddr) ∧byte	×		
		A, r Note 3	2	8	—	A ← A∧r	×		
		r, A	2	8	—	r ← r∧A	×		
		A, saddr	2	8	10	A ← A∧ (saddr)	×		
		A, !addr16	3	16	18 + 2n	A ← A∧ (addr16)	×		
		A, [HL]	1	8	10 + 2n	A ← A∧ (HL)	×		
		A, [HL+byte]	2	16	18 + 2n	A ← A∧ (HL+byte)	×		
		A, [HL+B]	2	16	18 + 2n	A ← A∧ (HL+B)	×		
		A, [HL+C]	2	16	18 + 2n	A ← A∧ (HL+C)	×		
	OR	A, #byte	2	8	—	A ← A∨byte	×		
		saddr, #byte	3	12	16	(saddr) ← (saddr) ∨byte	×		
		A, r Note 3	2	8	—	A ← A∨r	×		
		r, A	2	8	—	r ← r∨A	×		
		A, saddr	2	8	10	A ← A∨ (saddr)	×		
		A, !addr16	3	16	18 + 2n	A ← A∨ (addr16)	×		
		A, [HL]	1	8	10 + 2n	A ← A∨ (HL)	×		
		A, [HL+byte]	2	16	18 + 2n	A ← A∨ (HL+byte)	×		
		A, [HL+B]	2	16	18 + 2n	A ← A∨ (HL+B)	×		
		A, [HL+C]	2	16	18 + 2n	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

Remarks **1.** One instruction clock cycle is one cycle of the CPU clock (f_{CPU}) selected by processor clock control register (PCC).
2. Clock indicates when a program is in the internal ROM area.
3. n is the number of waits when the external memory expansion area is read.

Instruction Group	Mnemonic	Operands	Byte	Clock		Operation	Flag		
				Note 1	**Note 2**		Z	AC	CY
8-bit Operation	XOR	A, #byte	2	8	—	A ← A $\veebar$ byte	×		
		saddr, #byte	3	12	16	(saddr) ← (saddr) $\veebar$ byte	×		
		A, r **Note 3**	2	8	—	A ← A $\veebar$ r	×		
		r, A	2	8	—	r ← r $\veebar$ A	×		
		A, saddr	2	8	10	A ← A $\veebar$ (saddr)	×		
		A, !addr16	3	16	18 + 2n	A ← A $\veebar$ (addr16)	×		
		A, [HL]	1	8	10 + 2n	A ← A $\veebar$ (HL)	×		
		A, [HL+byte]	2	16	18 + 2n	A ← A $\veebar$ (HL+byte)	×		
		A, [HL+B]	2	16	18 + 2n	A ← A $\veebar$ (HL+B)	×		
		A, [HL+C]	2	16	18 + 2n	A ← A $\veebar$ (HL+C)	×		
	CMP	A, #byte	2	8	—	A–byte	×	×	×
		saddr, #byte	3	12	16	(saddr)–byte	×	×	×
		A, r **Note 3**	2	8	—	A–r	×	×	×
		r, A	2	8	—	r–A	×	×	×
		A, saddr	2	8	10	A–(saddr)	×	×	×
		A, !addr16	3	16	18 + 2n	A–(addr16)	×	×	×
		A, [HL]	1	8	10 + 2n	A–(HL)	×	×	×
		A, [HL+byte]	2	16	18 + 2n	A–(HL+byte)	×	×	×
		A, [HL+B]	2	16	18 + 2n	A–(HL+B)	×	×	×
		A, [HL+C]	2	16	18 + 2n	A–(HL+C)	×	×	×
16-bit Operation	ADDW	AX, #word	3	12	—	AX, CY ← AX+word	×	×	×
	SUBW	AX, #word	3	12	—	AX, CY ← AX–word	×	×	×
	CMPW	AX, #word	3	12	—	AX–word	×	×	×
Multiply/ Divide	MULU	X	2	32	—	AX ← A × X			
	DIVUW	C	2	50	—	AX (Quotient), C (Remainder) ← AX÷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

Remarks 1. One instruction clock cycle is one cycle of the CPU clock (f_{CPU}) selected by processor clock control register (PCC).

2. Clock indicates when a program is in the internal ROM area.

3. n is the number of waits when the external memory expansion area is read.

Instruction Group	Mnemonic	Operands	Byte	Clock Note 1	Clock Note 2	Operation	Flag Z	Flag AC	Flag CY
Increase/ Decrease	INC	r	1	4	—	r ← r+1	×	×	
		saddr	2	8	12	(saddr) ← (saddr)+1	×	×	
	DEC	r	1	4	—	r ← r–1	×	×	
		saddr	2	8	12	(saddr) ← (saddr)–1	×	×	
	INCW	rp	1	8	—	rp ← rp+1			
	DECW	rp	1	8	—	rp ← rp–1			
Rotation	ROR	A, 1	1	4	—	$(CY, A_7 \leftarrow A_0, A_{m-1} \leftarrow A_m) \times 1$			×
	ROL	A, 1	1	4	—	$(CY, A_0 \leftarrow A_7, A_{m+1} \leftarrow A_m) \times 1$			×
	RORC	A, 1	1	4	—	$(CY \leftarrow A_0, A_7 \leftarrow CY, A_{m-1} \leftarrow A_m) \times 1$			×
	ROLC	A, 1	1	4	—	$(CY \leftarrow A_7, A_0 \leftarrow CY, A_{m+1} \leftarrow A_m) \times 1$			×
	ROR4	[HL]	2	20	24 + 2n + 2m	$A_{3-0} \leftarrow (HL)_{3-0}, (HL)_{7-4} \leftarrow A_{3-0}, (HL)_{3-0} \leftarrow (HL)_{7-4}$			
	ROL4	[HL]	2	20	24 + 2n + 2m	$A_{3-0} \leftarrow (HL)_{7-4}, (HL)_{3-0} \leftarrow A_{3-0}, (HL)_{7-4} \leftarrow (HL)_{3-0}$			
BCD Adjust	ADJBA		2	8	—	Decimal Adjust Accumulator after Addition	×	×	×
	ADJBS		2	8	—	Decimal Adjust Accumulator after Subtract	×	×	×
Bit Manipulation	MOV1	CY, saddr.bit	3	12	14	CY ← (saddr.bit)			×
		CY, sfr.bit	3	—	14	CY ← sfr.bit			×
		CY, A.bit	2	8	—	CY ← A.bit			×
		CY, PSW.bit	3	—	14	CY ← PSW.bit			×
		CY, [HL].bit	2	12	14 + 2n	CY ← (HL).bit			×
		saddr.bit, CY	3	12	16	(saddr.bit) ← CY			
		sfr.bit, CY	3	—	16	sfr.bit ← CY			
		A.bit, CY	2	8	—	A.bit ← CY			
		PSW.bit, CY	3	—	16	PSW.bit ← CY	×	×	
		[HL].bit, CY	2	12	16 + 2n + 2m	(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

Remarks 1. One instruction clock cycle is one cycle of the CPU clock (f_{CPU}) selected by processor clock control register (PCC).

2. Clock indicates when a program is in the internal ROM area.

3. n is the number of waits when the external memory expansion area is read.

4. m is the number of waits when the external memory expansion area is written.

Instruction Group	Mnemonic	Operands	Byte	Clock		Operation	Flag		
				Note 1	**Note 2**		Z	AC	CY
Bit Manipulation	AND1	CY, saddr.bit	3	12	14	CY ← CY∧ (saddr.bit)			×
		CY, sfr.bit	3	—	14	CY ← CY∧ sfr.bit			×
		CY, A.bit	2	8	—	CY ← CY∧ A.bit			×
		CY, PSW.bit	3	—	14	CY ← CY∧ PSW.bit			×
		CY, [HL].bit	2	12	14 + 2n	CY ← CY∧ (HL).bit			×
	OR1	CY, saddr.bit	3	12	14	CY ← CY∨ (saddr.bit)			×
		CY, sfr.bit	3	—	14	CY ← CY∨ sfr.bit			×
		CY, A.bit	2	8	—	CY ← CY∨ A.bit			×
		CY, PSW.bit	3	—	14	CY ← CY∨ PSW.bit			×
		CY, [HL].bit	2	12	14 + 2n	CY ← CY∨ (HL).bit			×
	XOR1	CY, saddr.bit	3	12	14	CY ← CY ⊻ (saddr.bit)			×
		CY, sfr.bit	3	—	14	CY ← CY ⊻ sfr.bit			×
		CY, A.bit	2	8	—	CY ← CY ⊻ A.bit			×
		CY, PSW.bit	3	—	14	CY ← CY ⊻ PSW.bit			×
		CY, [HL].bit	2	12	14 + 2n	CY ← CY ⊻ (HL).bit			×
	SET1	saddr.bit	2	8	12	(saddr.bit) ← 1			
		sfr.bit	3	—	16	sfr.bit ← 1			
		A.bit	2	8	—	A.bit ← 1			
		PSW.bit	2	—	12	PSW.bit ← 1	×	×	×
		[HL].bit	2	12	16 + 2n + 2m	(HL).bit ← 1			
	CLR1	saddr.bit	2	8	12	(saddr.bit) ← 0			
		sfr.bit	3	—	16	sfr.bit ← 0			
		A.bit	2	8	—	A.bit ← 0			
		PSW.bit	2	—	12	PSW.bit ← 0	×	×	×
		[HL].bit	2	12	16 + 2n + 2m	(HL).bit ← 0			
	SET1	CY	1	4	—	CY ← 1			1
	CLR1	CY	1	4	—	CY ← 0			0
	NOT1	CY	1	4	—	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

Remarks 1. One instruction clock cycle is one cycle of the CPU clock (f_{CPU}) selected by processor clock control register (PCC).

2. Clock indicates when a program is in the internal ROM area.

3. n is the number of waits when the external memory expansion area is read.

4. m is the number of waits when the external memory expansion area is written.

Instruction Group	Mnemonic	Operands	Byte	Clock		Operation	Flag		
				Note 1	Note 2		Z	AC	CY
Call Return	CALL	!addr16	3	14	—	(SP–1) ← $(PC+3)_H$, (SP–2) ← $(PC+3)_L$, PC ← addr16, SP ← SP–2			
	CALLF	!addr11	2	10	—	(SP–1) ← $(PC+2)_H$, (SP–2) ← $(PC+2)_L$, PC_{15-11} ← 00001, PC_{10-0} ← addr11, SP ← SP–2			
	CALLT	[addr5]	1	12	—	(SP–1) ← $(PC+1)_H$, (SP–2) ← $(PC+1)_L$, PC_H ← (00000000, addr5+1), PC_L ← (00000000, addr5), SP ← SP–2			
	BRK		1	12	—	(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	12	—	PC_H ← (SP+1), PC_L ← (SP), SP ← SP+2			
	RETI		1	12	—	PC_H ← (SP+1), PC_L ← (SP), PSW ← (SP+2), SP ← SP+3, NMIS ← 0	R	R	R
	RETB		1	12	—	PC_H ← (SP+1), PC_L ← (SP), PSW ← (SP+2), SP ← SP+3	R	R	R
Stack Manipulation	PUSH	PSW	1	4	—	(SP–1) ← PSW, SP ← SP–1			
		rp	1	8	—	(SP–1) ← rp_H, (SP–2) ← rp_L, SP ← SP–2			
	POP	PSW	1	4	—	PSW ← (SP), SP ← SP+1	R	R	R
		rp	1	8	—	rp_H ← (SP+1), rp_L ← (SP), SP ← SP+2			
	MOVW	SP, #word	4	—	20	SP ← word			
		SP, AX	2	—	16	SP ← AX			
		AX, SP	2	—	16	AX ← SP			

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

Remarks 1. One instruction clock cycle is one cycle of the CPU clock (f_{CPU}) selected by processor clock control register (PCC).

2. Clock indicates when a program is in the internal ROM area.

Instruction Group	Mnemonic	Operands	Byte	Clock		Operation	Flag		
				Note 1	**Note 2**		Z	AC	CY
Unconditional Branch	BR	!addr16	3	12	—	PC ← addr16			
		$addr16	2	12	—	PC ← PC + 2 + jdisp8			
		AX	2	16	—	PC_H ← A, PC_L ← X			
Conditional Branch	BC	$addr16	2	12	—	PC ← PC + 2 + jdisp8 if CY = 1			
	BNC	$addr16	2	12	—	PC ← PC + 2 + jdisp8 if CY = 0			
	BZ	$addr16	2	12	—	PC ← PC + 2 + jdisp8 if Z = 1			
	BNZ	$addr16	2	12	—	PC ← PC + 2 + jdisp8 if Z = 0			
	BT	saddr.bit, $addr16	3	16	18	PC ← PC + 3 + jdisp8 if (saddr.bit) = 1			
		sfr.bit, $addr16	4	—	22	PC ← PC + 4 + jdisp8 if sfr.bit = 1			
		A.bit, $addr16	3	16	—	PC ← PC + 3 + jdisp8 if A.bit = 1			
		PSW.bit, $addr16	3	—	18	PC ← PC + 3 + jdisp8 if PSW.bit = 1			
		[HL].bit, $addr16	3	20	22 + 2n	PC ← PC + 3 + jdisp8 if (HL).bit = 1			
	BF	saddr.bit, $addr16	4	20	22	PC ← PC + 4 + jdisp8 if (saddr.bit) = 0			
		sfr.bit, $addr16	4	—	22	PC ← PC + 4 + jdisp8 if sfr.bit = 0			
		A.bit, $addr16	3	16	—	PC ← PC + 3 + jdisp8 if A.bit = 0			
		PSW.bit, $addr16	4	—	22	PC ← PC + 4 + jdisp8 if PSW.bit = 0			
		[HL].bit, $addr16	3	20	22 + 2n	PC ← PC + 3 + jdisp8 if (HL).bit = 0			
	BTCLR	saddr.bit, $addr16	4	20	24	PC ← PC + 4 + jdisp8 if (saddr.bit) = 1 then reset (saddr.bit)			
		sfr.bit, $addr16	4	—	24	PC ← PC + 4 + jdisp8 if sfr.bit = 1 then reset sfr.bit			
		A.bit, $addr16	3	16	—	PC ← PC + 3 + jdisp8 if A.bit = 1 then reset A.bit			
		PSW.bit, $addr16	4	—	24	PC ← PC+4 + jdisp8 if PSW.bit = 1 then reset PSW.bit	×	×	×
		[HL].bit, $addr16	3	20	24 + 2n +2m	PC ← PC + 3 + jdisp8 if (HL).bit = 1 then reset (HL).bit			

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

Remarks 1. One instruction clock cycle is one cycle of the CPU clock (f_{CPU}) selected by processor clock control register (PCC).

2. Clock indicates when a program is in the internal ROM area.

3. n is the number of waits when the external memory expansion area is read.

4. m is the number of waits when the external memory expansion area is written.

Instruction Group	Mnemonic	Operands	Byte	Clock		Operation	Flag
				Note 1	Note 2		Z AC CY
Conditional Branch	DBNZ	B, $addr16	2	12	—	B ← B–1, then PC ← PC + 2 + jdisp8 if B ≠ 0	
		C, $addr16	2	12	—	C ← C–1, then PC ← PC + 2 + jdisp8 if C ≠ 0	
		saddr, $addr16	3	16	20	(saddr) ← (saddr) – 1, then PC ← PC + 3 + jdisp8 if (saddr) ≠ 0	
CPU Control	SEL	RBn	2	8	—	RBS1, 0 ← n	
	NOP		1	4	—	No Operation	
	EI		2	—	12	IE ← 1 (Enable Interrupt)	
	DI		2	—	12	IE ← 0 (Disable Interrupt)	
	HALT		2	12	—	Set HALT Mode	
	STOP		2	12	—	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

Remarks **1.** One instruction clock cycle is one cycle of the CPU clock (f_{CPU}) selected by processor clock control register (PCC).
2. Clock indicates when a program is in the internal ROM area.

23.3 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

[MEMO]

★

APPENDIX A DIFFERENCES BETWEEN μPD78014, 78014H, AND 78018F SUBSERIES

Table A-1 shows the major differences between the μPD78014, 78014H, and 78018F Subseries.

Table A-1. Major Differences between μPD78014, 78014H, and 78018F Subseries (1/2)

Item \ Part Number	μPD78014 Subseries	μPD78014H Subseries	μPD78018F Subseries
EMI noise measure	None	Provided	None
Y Subseries	Provided	None	Provided
PROM version	μPD78P014	μPD78P018F	
Supply voltage	V_{DD} = 2.7 to 6.0 V	V_{DD} = 1.8 to 5.5 V	
Internal high-speed RAM size	μPD78011B: 512 bytes μPD78012B: 512 bytes μPD78013: 1024 bytes μPD78014: 1024 bytes μPD78P014: 1024 bytes	μPD78011H: 512 bytes μPD78012H: 512 bytes μPD78013H: 1024 bytes μPD78014H: 1024 bytes	μPD78011F: 512 bytes μPD78012F: 512 bytes μPD78013F: 1024 bytes μPD78014F: 1024 bytes μPD78015F: 1024 bytes μPD78016F: 1024 bytes μPD78018F: 1024 bytes μPD78P018F: 1024 bytes
Internal expansion RAM size	None		μPD78011F: None μPD78012F: None μPD78013F: None μPD78014F: None μPD78015F: 512 bytes μPD78016F: 512 bytes μPD78018F: 1024 bytes μPD78P018F: 1024 bytes
Operation mode of serial interface (Y Subseries)	3-wire/2-wire/SBI/I^2C: 1 ch 3-wire (with automatic transmission/reception): 1 ch	—	3-wire/2-wire/I^2C: 1 ch 3-wire (with automatic transmission/reception): 1 ch
Bit 5 (SIC) of interrupt timing specification register (SINT) in SBI mode (selection of INTCSI0 interrupt source)	When SIC = 1: sets CSIIF0 (interrupt request flag) on detection of bus release	When SIC = 1: Sets CSIIF0 (interrupt request flag) on detection of bus release and at end of transfer	
Bit 5 (SIC) of interrupt timing specification register (SINT) in I^2C bus mode (selection of INTCSI0 interrupt source)	When SIC = 1: Sets CSIIF0 (interrupt request flag) on detection of stop condition	—	When SIC = 1: Sets CSIIF0 (interrupt request flag) on detection of stop condition and at end of transfer

Table A-1. Major Differences between μPD78014, 78014H, and 78018F Subseries (2/2)

Item \ Part Number	μPD78014 Subseries	μPD78014H Subseries	μPD78018F Subseries
Function of bit 7 (BSYE) of serial bus interface control register (SBIC) (Y Subseries)	Control of synchronous bus signal output • When BSYE = 0 Disables output of busy signal in synchronization with falling edge of clock of $\overline{SCK0}$ immediately after instruction that clears this bit to 0 in SBI mode. Make sure that BSYE = 0 in I^2C bus mode. • When BSYE = 1 Outputs busy signal from falling edge of $\overline{SCK0}$ following acknowledge signal in SBI mode.	—	Control of N-ch open-drain output for transmission in I^2C bus mode • When BSYE = 0 Enables output (transmission) • When BSYE = 1 Disables output (reception)
Automatic data transmission/ reception interval specification register (ADTI)	None	Provided	
Package	• 64-pin plastic shrink DIP (750 mil) • 64-pin ceramic shrink DIP (w/window) (750 mil)Note • 64-pin plastic QFP (14 × 14 mm)	• 64-pin plastic shrink DIP (750 mil) • 64-pin plastic QFP (14 × 14 mm) • 64-pin plastic LQFP (12 × 12 mm)	• 64-pin plastic shrink DIP (750 mil) • 64-pin ceramic shrink DIP (w/window) (750 mil)Note • 64-pin plastic QFP (14 × 14 mm) • 64-pin plastic LQFP (12 × 12 mm) • 64-pin ceramic WQFN (14 × 14 mm)Note
Programmer adapter	PA-78P014CW PA-78P014GC	PA-78P018CW PA-78P018GC PA-78P018GK PA-78P018KK-S	
Emulation board	IE-78014-R-EM or IE-78014-R-EM-A	IE-78014-R-EM-A	
Access timing to external memory	Differs between μPD78014 Subseries and other subseries. Refer to individual data sheet		
Electrical characteristics, recommended soldering conditions	Refer to individual data sheet.		

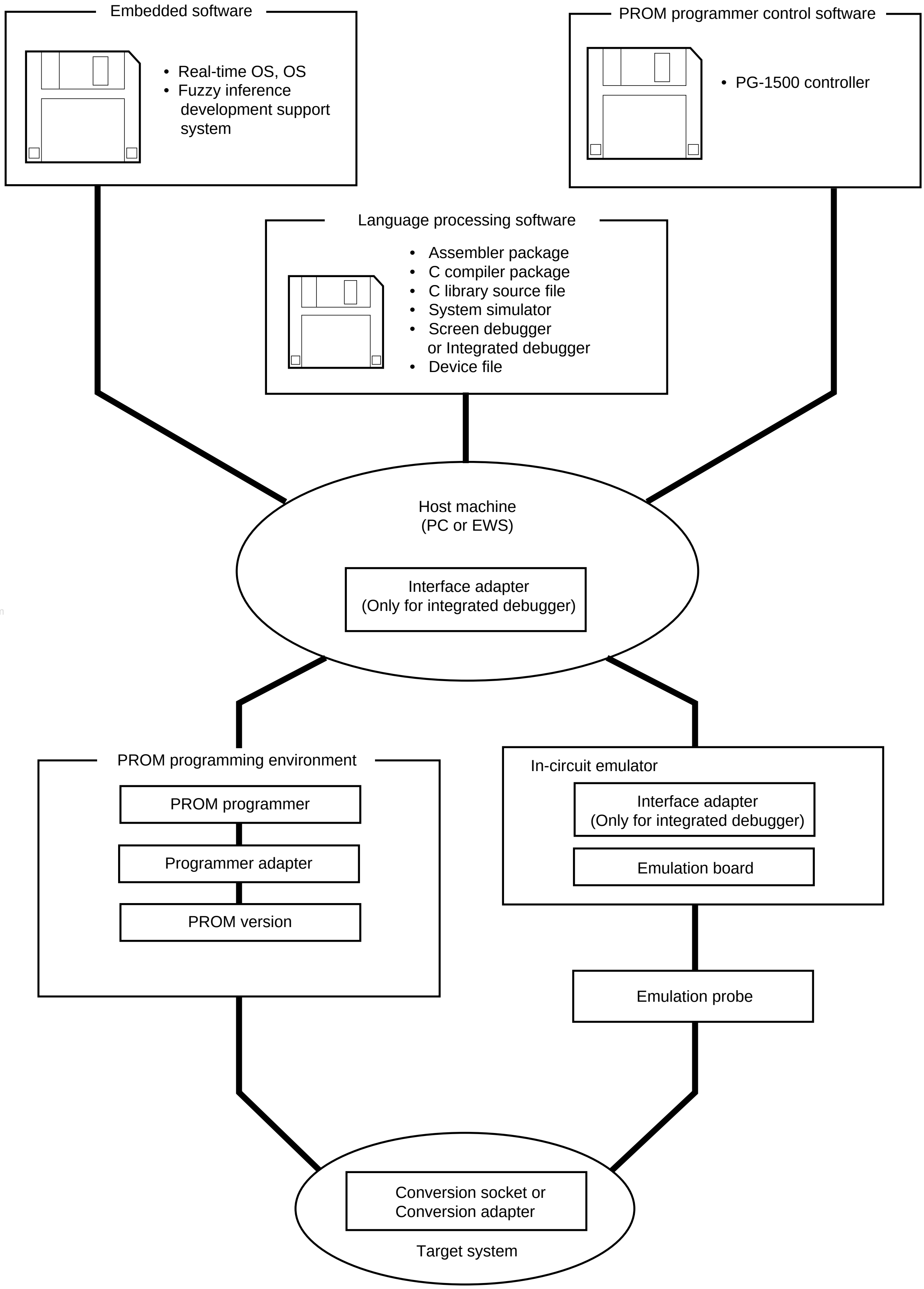
Note PROM version only

APPENDIX B   DEVELOPMENT TOOLS

The following development tools are available for the development of systems which employ the μPD78014 and 78014Y Subseries.

Figure B-1 shows the development tools configuration.

Figure B-1. Development Tools Configuration



B.1 Language Processing Software

RA78K/0 Assembler Package	This is a program to convert a program written in mnemonics into an object code executable with a microcontroller. Further, this assembler is provided with functions capable of automatically creating symbol tables and branch instruction optimization. Use RA78K/0 assembler package in combination with DF78014 device file (option).
	Part Number: μS××××RA78K0
CC78K/0 C Compiler Package	This is a program to convert a program written in C language into an object code executable with a microcontroller. Use CC78K/0 C compiler package in combination with RA78K/0 assembler package and DF78014 device file (option).
	Part Number: μS××××CC78K0
DF78014[Note] Device file	This file contains device-specific information. Use DF78014 device file in combination with RA78K/0, CC78K/0, SM78K0, ID78K0, and SD78K/0 (option).
	Part Number: μS××××DF78014
CC78K/0-L C Library Source File	A function source program configurating object library included in CC78K/0 C compiler package. This is needed when modifying the object library included in the C compiler package to customer's specification.
	Part Number: μS××××CC78K0-L

Note DF78014 can be used in common with any of the RA78K/0, CC78K/0, SM78K0, ID78K0, and SD78K/0 products.

Remark ×××× of the part number differs depending on the host machine or OS used. Refer to the table below.

μS××××RA78K0
μS××××CC78K0
μS××××DF78014
μS××××CC78K0-L

××××	Host Machine	Operating System	Supply Medium
5A13	PC-9800 series	MS-DOS (Ver.3.30 to Ver.6.2[Note])	3.5-inch 2HD
5A10			5-inch 2HD
7B13	IBM PC/AT and compatible machine	See **B.4**	3.5-inch 2HC
7B10			5-inch 2HC
3H15	HP9000 series 300™	HP-UX™(Rel7.05B)	Cartridge tape (QIC-24)
3P16	HP9000 series 700™	HP-UX(Rel9.01)	Digital audio tape (DAT)
3K15	SPARCstation™	SunOS™(Rel4.1.1)	Cartridge tape (QIC-24)
3M15	EWS4800 series (RISC)	EWS-UX/V(Rel4.0)	

Note MS-DOS Ver. 5.0 or later have a task swap function, but this task swap function is not used in above software.

B.2 PROM Programming Tools

B.2.1 Hardware

PG-1500 PROM programmer	This is a PROM programmer capable of programming the single-chip microcontroller's on-chip PROM by manipulating from the stand-alone or host machine through connection of a programmer adapter separately purchasable and the supplied board. It can also program typical PROMs the capacities of which range from 256 Kbits to 4 Mbits.
PA-78P014CW PA-78P014GC PROM programmer adapter	This PROM programmer adapter is for the μPD78P014 and 78P014Y and is connected to the PG-1500. PA-78P014CW: 64-pin plastic shrink DIP (CW type) PA-78P014GC: 64-pin plastic QFP (GC-AB8 type)

B.2.2 Software

PG-1500 Controller	The PG-1500 is controlled from the host machine through connection with the host machine and PG-1500 via serial and/or parallel interface(s).
	Part number: μSxxxxPG1500

Remark Part number ×××× changes by the host machine or OS to be used.

μS××××PG1500

××××	Host Machine	Operating System	Supply Medium
5A13	PC-9800 series	MS-DOS	3.5-inch 2HD
5A10		(Ver.3.30 to Ver.6.2^{Note})	5-inch 2HD
7B13	IBM PC/AT and compatible machine	See **B.4**	3.5-inch 2HD
7B10			5-inch 2HC

Note MS-DOS Ver. 5.0 or later have a task swap function, but this task swap function cannot be used in the above software.

B.3 Debugging Tools

B.3.1 Hardware

★

IE-78000-R-A In-circuit emulator (For integrated debugger)	This is the in-circuit emulator for debugging hardware and/or software when a system is developed with 78K/0 Series devices. This emulator is designed to be used with the integrated debugger (ID78K0). This is used together with interface adapter to connect an emulation probe or host machine.
IE-70000-98-IF-B Interface adapter	This adapter is needed when PC-9800 (excluding notebook models) is used as a host machine of IE-78000-R-A.
IE-70000-98N-IF Interface adapter	This adapter and cable are needed when a PC-9800 notebook-type personal computer is used as a host machine of IE-78000-R-A.
IE-70000-PC-IF-B Interface adapter	This adapter is needed when IBM PC/AT is used as a host machine of IE-78000-R-A.
IE-78000-R-SV3 Interface adapter	This adapter and cable are needed when an EWS is used as a host machine of IE-78000-R-A. This adaptor should be connected to the internal board of IE-78000-R-A. 10Base-5 is supported for Ethernet™ connection. It needs a commercially available adapter for other connection.
IE-78000-R In-circuit emulator (For screen debugger)	This is the in-circuit emulator for debugging hardware and/or software when a system is developed with 78K/0 Series devices. This emulator is designed to be used with the screen debugger (SD78K/0). This is used together with emulation probe. This emulator provides an efficient debugging environment by connecting it with a host computer and a PROM programmer.
IE-78014-R-EM Emulation board	This is the board which emulates the peripheral hardware operations specific for the device (For 5 V). This is used together with in-circuit emulator.
IE-78014-R-EM-A Emulation board	This is the board which emulates hardware operations specific for the device (For 3 to 5.5 V). This is used together with in-circuit emulator.
EP-78240CW-R Emulation probe	This is the probe to connect the in-circuit emulator with a target system. This is for 64-pin plastic shrink DIP (CW type).
EP-78240GC-R Emulation probe	This is the probe to connect the in-circuit emulator with a target system. This is for 64-pin plastic QFP (GC-AB8 type). One 64-pin socket EV-9200GC-64 is included to facilitate development of target system.
EV-9200GC-64 Conversion socket	This socket connects EP-78240GC-R to a target system board designed for 64-pin plastic QFP (GC-AB8 type).

Remark The EV-9200GC-64 is sold in five units as a set.

B.3.2 Software (1/3)

SM78K0 System simulator	It is possible to debug at C source level or assembler level while simulating target system on the host machine. SM78K0 runs on Windows. By using SM78K0, logic and performance verification of application without in-circuit emulator is possible independently of hardware development, and development efficiency and software quality will thus be improved. This is used together with the separately sold device file (DF78014).
	Part Number: μS××××SM78K0

Remark Part number ×××× changes by the host machine or OS to be used.

★ μS<u>××××</u>SM78K0

××××	Host Machine	Operating System	Supply Medium
AA13	PC-9800 series	MS-DOS [Ver.3.30 to Ver.6.2[Note]] + Windows [Ver. 3.0 to Ver. 3.1]	3.5-inch 2HD
AB13	IBM PC/AT and compatible machine (Japanese Windows)	Refer to **B.4**	3.5-inch 2HC
BB13	IBM PC/AT and compatible machine (English Windows)		

Note MS-DOS Ver. 5.0 or later have a task swap function, but this task swap function cannot be used in the above software.

B.3.2 Software (2/3)

★

ID78K0 Integrated debugger	This debugger is a control program used to debug applications of the 78K/0 Series devices. This software has a Windows-based (PC version) or OSF/Motif™-based (EWS version) graphical user interface to provide comfortable operation environments. It also has an enhanced debugging function for C language support, and it is possible to display trace results at the C-language level by using its window integration feature which links source programs, disassembled display, and memory content display to their trace results. In addition, the efficiency when a program is debugged on a real-time operating system can be improved by incorporating function extension modules such as task debugger and system performance analyzer. This is used together with the separately sold device file (DF78014)
	Part Number: μS××××ID78K0

Remark Part number ×××× changes by the host machine or OS to be used.

★ μS××××ID78K0

××××	Host Machine	Operating System	Supply Medium
AA13	PC-9800 series	MS-DOS (Ver.3.30 to Ver.6.2 Note) + Windows (Ver. 3.1)	3.5-inch 2HD
AB13	IBM PC/AT and compatible machine (Japanese Windows)	See **B.4**	3.5-inch 2HC
BB13	IBM PC/AT and compatible machine (English Windows)		
3P16	HP9000 series 700	HP-UX(Rel9.0.1)	Digital audio tape (DAT)
3K15	SPARCstation	SunOS(Rel4.1.1)	Cartridge tape (QIC-24)
3K13			3.5-inch 2HC
3R16	NEWS™ (RISC)	NEWS-OS™(6.1x)	1/4-inch CGMT
3R13			3.5-inch 2HC
3M15	EWS4800 series (RISC)	EWS-UX/V(Rel4.0)	Cartridge tape (QIC-24)

Note MS-DOS Ver. 5.0 or later have a task swap function, but this task swap function cannot be used in the above software.

B.3.2 Software (3/3)

SD78K/0 Screen debugger	The IE-78000-R is controlled in the host machine through connection with the host machine and IE-78000-R via serial interface (RS-232-C). This is used together with the separately sold device file (DF78014).
	Part Number: μS××××SD78K0
DF78014[Note] Device file	This file has device-specific information. This is used together with the separately sold device file (RA78K/0, CC78K/0, SM78K0, ID78K0, SD78K/0)
	Part Number: μS××××DF78014

Note DF78014 can be used in RA78K/0, CC78K/0, SM78K0, and SD78K/0 all in common.

Remark Part number ×××× changes by the host machine or OS to be used.

μS××××SD78K0
μS××××DF78014

××××	Host Machine	Operating System	Supply Medium
5A13	PC-9800 series	MS-DOS (Ver.3.30 to Ver.6.2[Note])	3.5-inch 2HD
5A10			5-inch 2HD
7B13	IBM PC/AT and compatible machine	Refer to **B.4**	3.5-inch 2HC
7B10			5-inch 2HC

Note MS-DOS Ver. 5.0 or later have a task swap function, but this task swap function cannot be used in the above software.

B.4 OS for IBM PC

The following OSs for IBM PC are supported.

When operating SM78K0, ID78K0, FE9200 (See **C.2 Fuzzy Inference Development Support System**), Windows (Ver. 3.0 to Ver. 3.1) is necessary.

OS	Version
PC DOS	Ver. 5.02 to Ver. 6.3
	J6.1/V[Note] to J6.3/V[Note]
IBM DOS™	J5.02/V[Note]
MS-DOS	Ver. 5.0 to Ver. 6.22
	5.0/V[Note] to 6.2/V[Note]

Note Only English mode is supported.

Caution MS-DOS Ver. 5.0 or later have a task swap function, but this task swap function cannot be used in this software.

B.5 System-up Method from Other In-Circuit Emulator to In-Circuit Emulator for the 78K/0 Series

When you already have an in-circuit emulator for the 78K Series or the 75X/XL Series, you can use that in-circuit emulator as the equivalent of a 78K/0 Series in-circuit emulator IE-78000-R or IE-78000-R-A by replacing the internal break board with the IE-78000-R-BK.

Table B-1. System-up Method from Other In-Circuit Emulator to IE-78000-R

Series Name	In-circuit Emulator Owned	Board to be Purchased
75X/XL Series	IE-75000-R[Note], IE-75001-R	IE-78000-R-BK
78K/I Series	IE-78130-R, IE-78140-R	
78K/II Series	IE-78230-R[Note], IE-78230-R-A IE-78240-R[Note], IE-78240-R-A	
78K/III Series	IE-78320-R[Note], IE-78327-R IE-78330-R, IE-78350-R	

Note Available for maintenance purposes only.

Table B-2. System-up Method from Other In-Circuit Emulator to IE-78000-R-A

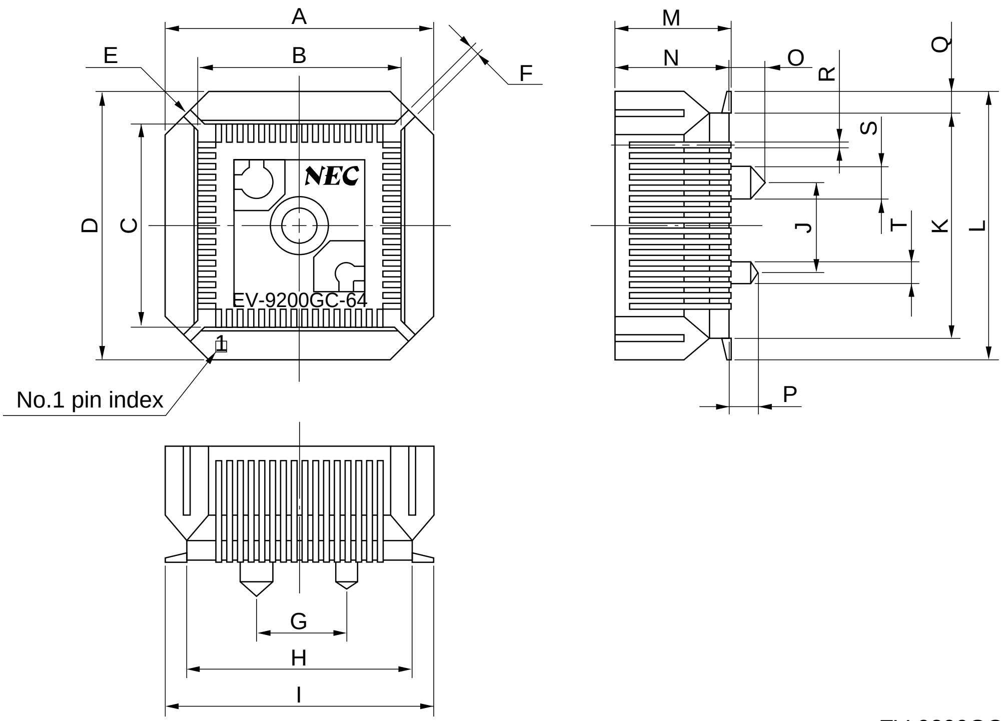
Series Name	In-circuit Emulator Owned	Board to be Purchased
75X/XL Series	IE-75000-R[Note 1], IE-75001-R	IE-78000-R-BK[Note 2]
78K/I Series	IE-78130-R, IE-78140-R	
78K/II Series	IE-78230-R[Note 1], IE-78230-R-A, IE-78240-R[Note 1], IE-78240-R-A	
78K/III Series	IE-78320-R[Note 1], IE-78327-R, IE-78330-R, IE-78350-R	
78K/0 Series	IE-78000-R	—[Note 2]

Notes 1. Available for maintenance purposes only.

2. It is needed to take out to NEC to change a part of in-circuit emulator and replace the control/trace board with the supervisor board

Package Drawing and Footprint of Conversion Socket (EV-9200GC-64)

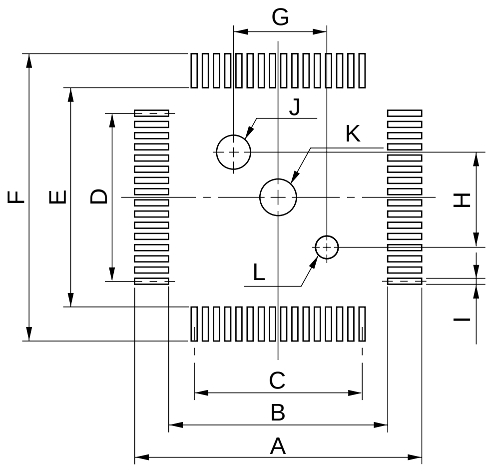
Figure B-2. EV-9200GC-64 Package Drawing (for reference only)



EV-9200GC-64-G0

ITEM	MILLIMETERS	INCHES
A	18.8	0.74
B	14.1	0.555
C	14.1	0.555
D	18.8	0.74
E	4-C 3.0	4-C 0.118
F	0.8	0.031
G	6.0	0.236
H	15.8	0.622
I	18.5	0.728
J	6.0	0.236
K	15.8	0.622
L	18.5	0.728
M	8.0	0.315
N	7.8	0.307
O	2.5	0.098
P	2.0	0.079
Q	1.35	0.053
R	0.35 ± 0.1	$0.014^{+0.004}_{-0.005}$
S	ϕ2.3	ϕ0.091
T	ϕ1.5	ϕ0.059

Figure B-3. EV-9200GC-64 Footprint (for reference only)



EV-9200GC-64-P1

ITEM	MILLIMETERS	INCHES
A	19.5	0.768
B	14.8	0.583
C	$0.8 \pm 0.02 \times 15 = 12.0 \pm 0.05$	$0.031^{+0.002}_{-0.001} \times 0.591 = 0.472^{+0.003}_{-0.002}$
D	$0.8 \pm 0.02 \times 15 = 12.0 \pm 0.05$	$0.031^{+0.002}_{-0.001} \times 0.591 = 0.472^{+0.003}_{-0.002}$
E	14.8	0.583
F	19.5	0.768
G	6.00 ± 0.08	$0.236^{+0.004}_{-0.003}$
H	6.00 ± 0.08	$0.236^{+0.004}_{-0.003}$
I	0.5 ± 0.02	$0.197^{+0.001}_{-0.002}$
J	$\phi 2.36 \pm 0.03$	$\phi 0.093^{+0.001}_{-0.002}$
K	$\phi 2.2 \pm 0.1$	$\phi 0.087^{+0.004}_{-0.005}$
L	$\phi 1.57 \pm 0.03$	$\phi 0.062^{+0.001}_{-0.002}$

Caution Dimensions of 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 “SEMICONDUCTOR DEVICE MOUNTING TECHNOLOGY MANUAL” (C10535E).

APPENDIX C   EMBEDDED SOFTWARE

The following embedded software are available for efficient program development and maintenance of the μPD78014 and 78014Y Subseries.

C.1 Real-time OS (1/2)

RX78K/0 Real-time OS	Real-time OS which is based on the μITRON specification. Supplied with the RX78K/0 nucleus and a tool to prepare multiple information tables (configurator). Used in combination with RA78K/0 assembler package (option) and device file (DF78014) (option).
	Part Number: μS××××RX78013-ΔΔΔΔ

Caution When purchasing the RX78K/0, fill in the purchase application form in advance, and sign the User Agreement.

Remark ×××× and ΔΔΔΔ of the part number differs depending on host machine and OS, etc.

μS××××RX78013-ΔΔΔΔ

ΔΔΔΔ	Product Outline	Maximum number for use in mass production
001	Evaluation object	Do not use for mass-producing product.
100K	Mass production object	100,000
001M		1,000,000
010M		10,000,000
S01	Source program	Source program of mass production object

××××	Host Machine	Operating System	Supply Medium
5A13	PC-9800 series	MS-DOS	3.5-inch 2HD
5A10		(Ver.3.30 to Ver.6.2 Note)	5-inch 2HD
7B13	IBM PC/AT and compatible machine	See **B.4**	3.5-inch 2HC
7B10			5-inch 2HC
3H15	HP9000 series 300	HP-UX(Rel7.05B)	Cartridge tape (QIC-24)
3P16	HP9000 series 700	HP-UX(Rel9.01)	Digital audio tape (DAT)
3K15	SPARCstation	SunOS(Rel4.1.1)	Cartridge tape (QIC-24)
3M15	EWS4800 series (RISC)	EWS-UX/V(Rel4.0)	

Note MS-DOS Ver. 5.0 or later have a task swap function, but this task swap function cannot be used in this software.

C.1 Real-time OS (2/2)

MX78K0 OS	MX78K0 is a subset OS which is based on the μITRON specification. Supplied with the MX78K0 nucleus. MX78K0 OS controls tasks, events, and time. In task control, MX78K0 OS controls task execution order, and then perform the switching process to a task to be executed.
	Part Number: μS××××MX78K0-∆∆∆

Remark ×××× and ∆∆∆ of the part number differs depending on host machine and OS, etc. Refer to the table below.

μS××××MX78K0-∆∆∆

∆∆∆	Product Outline	Cautions
001	Evaluation object	Use for experimental producing
××	Mass production object	Use for mass production
S01	Source program	Purchasable only when purchasing mass production object

××××	Host Machine	Operating System	Supply Medium
5A13	PC-9800 series	MS-DOS (Ver.3.30 to Ver.6.2 Note)	3.5-inch 2HD
5A10			5-inch 2HD
7B13	IBM PC/AT and compatible machine	See **B.4**	3.5-inch 2HC
7B10			5-inch 2HC
3H15	HP9000 series 300	HP-UX (Rel7.05B)	Cartridge tape (QIC-24)
3P16	HP9000 series 700	HP-UX (Rel9.01)	Digital audio tape (DAT)
3K15	SPARCstation	SunOS (Rel4.1.1)	Cartridge tape (QIC-24)
3M15	EWS4800 series (RISC)	EWS-UX/V(Rel4.0)	

Note MS-DOS Ver. 5.0 or later have a task swap function, but this task swap function cannot be used in this software.

C.2 Fuzzy Inference Development Support System

FE9000/FE9200 Fuzzy Knowledge Data Preparation Tool	Program supporting input of fuzzy knowledge data (fuzzy rule and membership function), editing (edit), and evaluation (simulation). FE9200 operates on Windows.
	Part Number: μS××××FE9000 (PC-9800 series) μS××××FE9200 (IBM PC/AT and compatible machine)
FT9080/FT9085 Translator	Program converting fuzzy knowledge data obtained by using fuzzy knowledge data preparation tool to the assembler source program for the RA78K/0.
	Part Number: μS××××FT9080 (PC-9800 series) μS××××FT9085 (IBM PC/AT and compatible machine)
FI78K0 Fuzzy Inference Module	Program executing fuzzy inference. Fuzzy inference is executed by linking fuzzy knowledge data converted by translator.
	Part Number: μS××××FI78K0 (PC-9800 series, IBM PC/AT and compatible machine)
FD78K0 Fuzzy Inference Debugger	Support software evaluating and adjusting fuzzy knowledge data at hardware level by using in-circuit emulator.
	Part Number: μS××××FD78K0 (PC-9800 series, IBM PC/AT and compatible machine)

Remark ×××× of the part number differs depending on the host machine or OS used.

μS××××FE9000
μS××××FT9080
μS××××FI78K0
μS××××FD78K0

××××	Host Machine	Operating System	Supply Medium
5A13	PC-9800 series	MS-DOS (Ver.3.30 to Ver.6.2Note)	3.5-inch 2HD
5A10			5-inch 2HD

Note MS-DOS Ver. 5.0 or later have a task swap function, but this task swap function cannot be used in this software.

μS××××FE9200
μS××××FT9085
μS××××FI78K0
μS××××FD78K0

××××	Host Machine	Operating System	Supply Medium
7B13	IBM PC/AT and compatible machine	See **B.4**	3.5-inch 2HC
7B10			5-inch 2HC

APPENDIX D   REGISTER INDEX

D.1 Register Index (In Alphabetical Order with Respect to the Register Name)

D.2 Register Index (In Alphabetical Order with Respect to the Register Symbol)

APPENDIX E REVISION HISTORY

Major revisions by edition and revised chapters are shown below.

(1/4)

Edition	Major revisions from previous edition	Revised Chapters
4th	μPD78014Y subseries were added as applied devices.	General
	Frequency of main system clock oscillator is changed from 8.38 MHz to 10.0 MHz.	
	Item “After Reset” was added in section 2.1.1 “Normal operating mode pins”.	CHAPTER 2 PIN FUNCTION
	Cautions on pull-up resistors disconnection for P60 to P63 pins by mask option was added.	
	Pin input/output circuit types were changed as follows. Change: Type 5-B to Type 5-E, Type 9-B to Type 11 Addition: Type 16	
	Memory size switching register is incorporated in all applied devices, not only in the μPD78P014.	CHAPTER 3 CPU ARCHITECTURE
	Cautions on using port 1 as A/D converter input was added.	CHAPTER 4 PORT FUNCTIONS
	Cautions on port mode register setting when port 2 is used in the SBI mode was added.	
	Cautions on port mode register setting was added.	
	Cautions on pull-up resistor option register when port is used as dual-function pin was added.	
	Cautions on CR00 setting was added.	CHAPTER 6 16-BIT TIMER/EVENT COUNTER
	Timing chart for square-wave output operation was added.	
	Cautions on using 8-bit timer registers 1 and 2 as a 16-bit timer register was added.	CHAPTER 7 8-BIT TIMER/EVENT COUNTER
	Interval times of interval timer were changed.	CHAPTER 8 WATCH TIMER
	Block diagram for watchdog timer was corrected.	CHAPTER 9 WATCHDOG TIMER
	Cautions on overflow time when the watchdog timer is cleared by setting bit 7 (RUN) of WDTM to 1 was added.	
	Condition under which clock output cannot be used was added.	CHAPTER 10 CLOCK OUTPUT CONTROL CIRCUIT
	Condition under which buzzer output cannot be used was added.	CHAPTER 11 BUZZER OUTPUT CONTROL CIRCUIT

(2/4)

Edition	Major revisions from previous edition	Revised Chapters
4th	Format of the A/D converter mode register was changed.	CHAPTER 12 A/D CONVERTER
	Section 12.4.2 “Input voltage and conversion results” was added.	
	Following items were added in section 12.5 “Cautions on A/D Converter”. (5) AV_{REF} pin input impedance (6) Interrupt request flag of A/D conversion end (INTAD) (7) AV_{DD} pin (8) Interrupt request flag of A/D conversion (ADIF)	
	Block diagram for serial interface channel 0 was changed.	CHAPTER 13 SERIAL INTERFACE CHANNEL 0
	Format of the serial operating mode register 0 was changed.	
	Serial bus interface control register format was changed.	
	Timing charts for various signals in the SBI mode and flag operations in SBIC register were changed.	
	Item (e) Procedure to judge whether slave device is in the busy state or not when device is in the master mode was added in section 13.4.2 “Cautions on SBI mode”.	
	Section 13.4.4 “I^2C bus mode operation” was added.	
	Block diagram for serial interface channel 1 was changed.	CHAPTER 14 SERIAL INTERFACE CHANNEL 1
	Format of the serial operating mode register 1 was changed.	
	Timing chart for 3-wire serial I/O mode was corrected.	
	Timing chart and flowchart for basic transmit/receive mode were corrected.	
	Flowchart for repeat transmission mode was corrected.	
	Timing chart when using busy control option was corrected.	
	Timing chart when using busy & strobe control option was corrected.	
	Cautions on TMIF4 flag of IF0L register was added.	CHAPTER 15 INTERRUPT FUNCTION
	Cautions on using port 0 as output port was added.	
	Item “Watch timer” was added in Table 17-1 “HALT Mode Operating Status”.	CHAPTER 17 STANDBY FUNCTION
	Description of NMIS was added in description of RETI instruction.	CHAPTER 20 INSTRUCTION SET
	APPENDIX B, “EMBEDDED SOFTWARE” was added.	APPENDIX B EMBEDDED SOFTWARE

(3/4)

Edition	Major revisions from previous edition	Revised Chapters
5th	μPD78011B(A), 78012B(A), 78013(A), 78014(A) were added as applied devices.	General
	Cautions on rewriting the timer clock select registers 0 to 2 (TCL0 to TCL2) to other data was added.	
	Watchdog timer count clocks (Inadvertent program loop detection period) selected by TCL20 to TCL22 of the timer clock select register 2 (TCL2) were changed.	
	Recommended connection of unused pins P04, P40 to P47 and P60 to P63 were changed as follows. P04: Connect to V_{SS} → Connect to V_{DD} or V_{SS} P40 to 47, P60 to 63: Connect to V_{DD} or V_{SS} → Connect to V_{DD}	CHAPTER 3 PIN FUNCTION (μPD78014 Subseries) CHAPTER 4 PIN FUNCTION (μPD78014Y Subseries)
	List of maximum time required for CPU clock switchover were corrected.	CHAPTER 7 CLOCK GENERATOR
	Descriptions in section 16.4.5 "I^{2}C bus mode operation" were changed.	CHAPTER 16 SERIAL INTERFACE CHANNEL 0 (μPD78014Y Subseries)
	Following subsections were added in section 16.4.6 "Cautions on Use of I^{2}C Bus Mode". (3) Slave wait release (slave reception), (4) Reception completion processing by a slave	
	Section 16.4.7 "Restrictions on Use of I^{2}C Bus Mode" was added.	
	Sections 20.2 "Operation Codes" and 20.3 "Descriptions of Instructions" in previous edition were deleted.	CHAPTER 23 INSTRUCTION SET
6th	Figure 11-3, "Watchdog Timer Mode Register Format" was changed and cautions for the figure were added.	CHAPTER 11 WATCHDOG TIMER
	Cautions on serial I/O shift register 0 (SIO0) of the μPD78014 subseries were added.	CHAPTER 16 SERIAL INTERFACE CHANNEL 0 (μPD78014Y Subseries)
	Figure 16-45, "Data Transmission from Master to Slave (Both Master and Slave Selected 9-Clock Wait)" was corrected.	
	Figure 16-46, "Data Transmission from Slave to Master (Both Master and Slave Selected 9-Clock Wait)" was corrected.	
	The following products were added. IE-78000-R-A, IE-70000-98-IF-B, IE-70000-98N-IF, IE-70000-PC-IF-B, IE-78000-R-SV3, ID78K0 The development of the following product was completed. IE-78014-R-EM-A	APPENDIX A DEVELOPMENT TOOLS
	An explanation for how to upgrade in-circuit emulator to IE-78000-R-A was added.	
	The version numbers of the supported operating systems were renewed.	APPENDIX A DEVELOPMENT TOOLS and APPENDIX B EMBEDDED SOFTWARE

(4/4)

Edition	Major revisions from previous edition	Revised Chapters
7th	P20, P21, P23 to P26 Block Diagrams, P22 and P27 Block Diagrams, and P30 to P37 Block Diagrams were corrected.	CHAPTER 6 PORT FUNCTIONS
	Figure 9-10 and 9-13, “Square Wave Output Operation Timings” were added.	CHAPTER 9 8-BIT TIMER/EVENT COUNTER
	Caution was added in section 15.1 “Serial Interface Channel 0 Functions”.	CHAPTER 15 SERIAL INTERFACE CHANNEL 0 (μPD78014 Subseries)
	Caution was added in section 15.3 “Serial Interface Channel 0 Control Register (2) Serial operating mode register 0 (CSIM0)”.	
	Cautions were added in section 15.4.3 “(2) (a) Bus release signal (REL), (b) Command signal (CMD), (11) Cautions on SBI mode”.	
	Caution was added in section 16.1 “Serial Interface Channel 0 Functions”.	CHAPTER 16 SERIAL INTERFACE CHANNEL 0 (μPD78014Y Subseries)
	Caution was added in section 16.3 “Serial Interface Channel 0 Control Register (2) Serial operating mode register 0 (CSIM0)”.	
	Cautions were added in section 16.4.3 “(2) (a) Bus release signal (REL), (b) Command signal (CMD), (11) Cautions on SBI mode”.	
	Item “(3) MSB/LSB switching as the start bit” was added in section 17.4.2 “3-wire serial I/O mode operation”.	CHAPTER 17 SERIAL INTERFACE CHANNEL 1
	(3) (d) Busy control option, (e) Busy & strobe control option, and (f) Bit slippage detection function in section 17.4.3 of the former edition were changed to (4) Synchronization control and the description was improved.	
	Caution was added in Table 22-1, “Differences between μPD78P014, 78P014Y, and Mask ROM Version”.	CHAPTER 22 μPD78P014, 78P014Y
	APPENDIX A, “DIFFERENCES BETWEEN μPD78014, 78014H, AND 78018F SUBSERIES” was added.	APPENDIX A DIFFERENCES BETWEEN μPD78014, 78014H, AND 78018F SUBSERIES
	Windows compatible 5-inch FD products was erased in APPENDIX B DEVELOPMENT TOOLS.	APPENDIX B DEVELOPMENT TOOLS
	The following products were changed from “Under development” to “Developed”. • IE-78000-R-A • ID78K0	

NEC

Facsimile Message

From:

Name

Company

Tel. FAX

Address

Although NEC has taken all possible steps to ensure that the documentation supplied to our customers is complete, bug free and up-to-date, we readily accept that errors may occur. Despite all the care and precautions we've taken, you may encounter problems in the documentation. Please complete this form whenever you'd like to report errors or suggest improvements to us.

Thank you for your kind support.

North America
NEC Electronics Inc.
Corporate Communications Dept.
Fax: 1-800-729-9288
1-408-588-6130

Europe
NEC Electronics (Europe) GmbH
Technical Documentation Dept.
Fax: +49-211-6503-274

South America
NEC do Brasil S.A.
Fax: +55-11-6465-6829

Hong Kong, Philippines, Oceania
NEC Electronics Hong Kong Ltd.
Fax: +852-2886-9022/9044

Korea
NEC Electronics Hong Kong Ltd.
Seoul Branch
Fax: 02-528-4411

Taiwan
NEC Electronics Taiwan Ltd.
Fax: 02-719-5951

Asian Nations except Philippines
NEC Electronics Singapore Pte. Ltd.
Fax: +65-250-3583

Japan
NEC Corporation
Semiconductor Solution Engineering Division
Technical Information Support Dept.
Fax: 044-548-7900

I would like to report the following error/make the following suggestion:

Document title: ______________________________

Document number: ____________________ Page number: __________

If possible, please fax the referenced page or drawing.

Document Rating	Excellent	Good	Acceptable	Poor
Clarity	❏	❏	❏	❏
Technical Accuracy	❏	❏	❏	❏
Organization	❏	❏	❏	❏

CS 97.8