To our customers,

Old Company Name in Catalogs and Other Documents

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

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

April 1st, 2010
Renesas Electronics Corporation

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

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

RENESAS

Notice

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

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

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

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

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

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

To all our customers

Regarding the change of names mentioned in the document, such as Hitachi Electric and Hitachi XX, to Renesas Technology Corp.

The semiconductor operations of Mitsubishi Electric and Hitachi were transferred to Renesas Technology Corporation on April 1st 2003. These operations include microcomputer, logic, analog and discrete devices, and memory chips other than DRAMs (flash memory, SRAMs etc.) Accordingly, although Hitachi, Hitachi, Ltd., Hitachi Semiconductors, and other Hitachi brand names are mentioned in the document, these names have in fact all been changed to Renesas Technology Corp. Thank you for your understanding. Except for our corporate trademark, logo and corporate statement, no changes whatsoever have been made to the contents of the document, and these changes do not constitute any alteration to the contents of the document itself.

Renesas Technology Home Page: http://www.renesas.com

Renesas Technology Corp.
Customer Support Dept.
April 1, 2003

Cautions

Keep safety first in your circuit designs!

1. Renesas Technology Corporation puts the maximum effort into making semiconductor products better and more reliable, but there is always the possibility that trouble may occur with them. Trouble with semiconductors may lead to personal injury, fire or property damage.
   Remember to give due consideration to safety when making your circuit designs, with appropriate measures such as (i) placement of substitutive, auxiliary circuits, (ii) use of nonflammable material or (iii) prevention against any malfunction or mishap.

Notes regarding these materials

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

HD404449 Series

Rev. 6.0
Sept. 1998

Description

The HD404449 Series is a HMCS400-series microcomputer designed to increase program productivity with large-capacity memory. Each microcomputer has four timers, two serial interfaces, A/D converter, input capture circuit, 32-kHz oscillator for clock, and four low-power dissipation modes.

The HD404449 Series includes three chips: the HD404448 with 8-kword ROM; the HD404449 with 16-kword ROM; and HD4074449 with 16-kword PROM (ZTAT™ version).

The HD4074449 is a PROM version (ZTAT™ microcomputer). A program can be written to the PROM by a PROM writer, which can dramatically shorten system development periods and smooth the process from debugging to mass production. (The ZTAT™ version is 27256-compatible.)

ZTAT™: Zero Turn Around Time ZTAT is a trademark of Hitachi Ltd.

Features

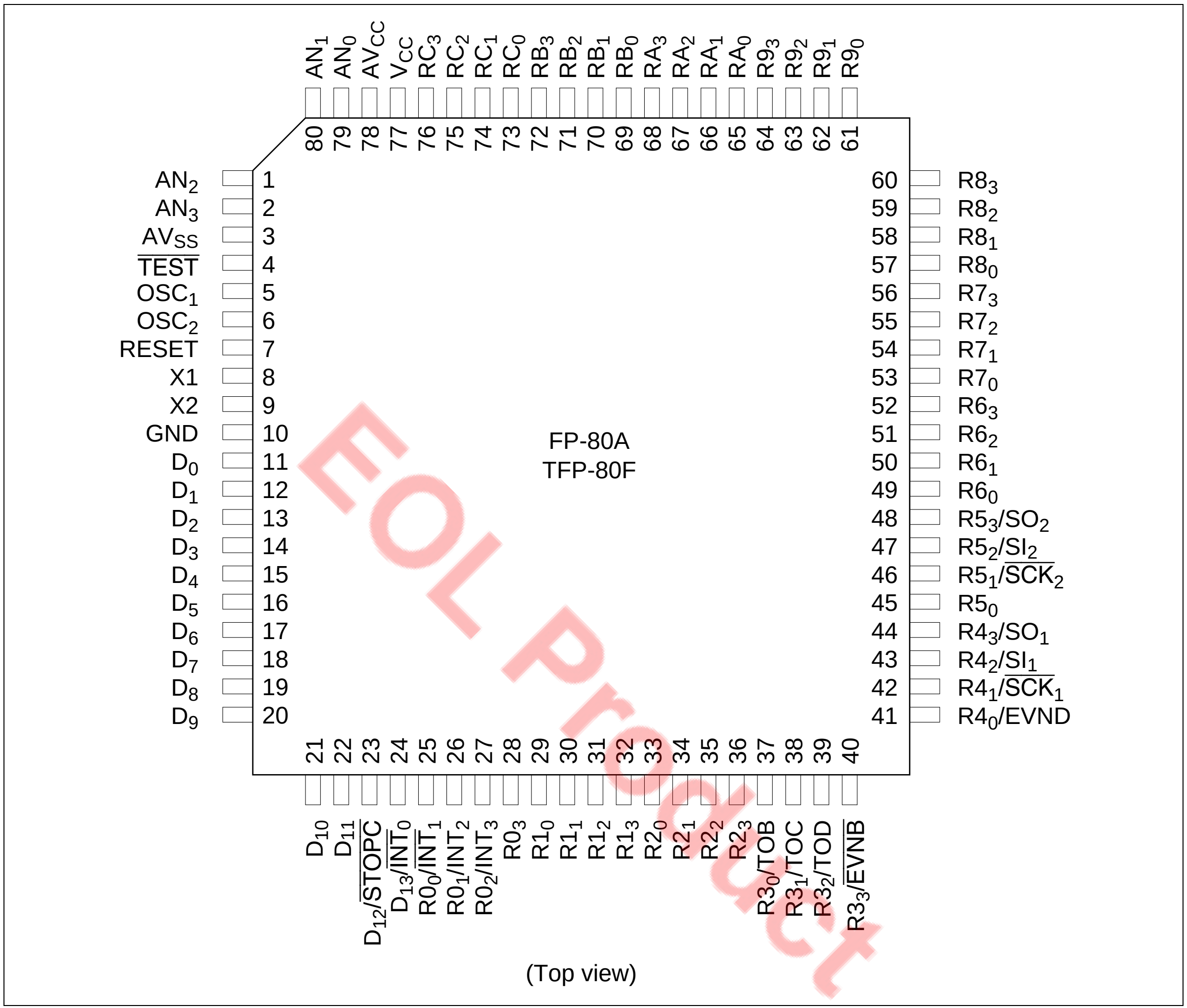
- 8,192-word × 10-bit ROM (HD404448)
  16,384-word × 10-bit ROM (HD404449 and HD4074449)
- 1,152-digit × 4-bit RAM
- 64 I/O pins, including 10 high-current pins (15 mA, max)
- Four timer/counters
- Eight-bit input capture circuit
- Three timer outputs (including two PWM outputs)
- Two event counter inputs (including one double-edge function)
- Two clock-synchronous 8-bit serial interfaces
- A/D converter (4-channel × 8-bit)
- Built-in oscillators
  — Main clock: 4-MHz ceramic oscillator or crystal (an external clock is also possible)
  — Subclock: 32.768-kHz crystal
- Eleven interrupt sources
  — Four by external sources, including two double-edge function
  — Seven by internal sources

- Subroutine stack up to 16 levels, including interrupts
- Four low-power dissipation modes
  - Subactive mode
  - Standby mode
  - Watch mode
  - Stop mode
- One external input for transition from stop mode to active mode
- Instruction cycle time: 1 µs (f_{OSC} = 4 MHz)
- Two operating modes
  - MCU mode (HD404448, HD404449)
  - MCU/PROM mode (HD4074449)

Ordering Information

Type	Product Name	Model Name	ROM (Words)	Package
Mask ROM	HD404448	HD404448H	8,192	80-pin plastic QFP (FP-80A)
		HD404448TF		80-pin plastic QFP (TFP-80F)
	HD404449	HD404449H	16,384	80-pin plastic QFP (FP-80A)
		HD404449TF		80-pin plastic QFP (TFP-80F)
ZTAT™	HD4074449	HD4074449H	16,384	80-pin plastic QFP (FP-80A)
		HD4074449TF		80-pin plastic QFP (TFP-80F)

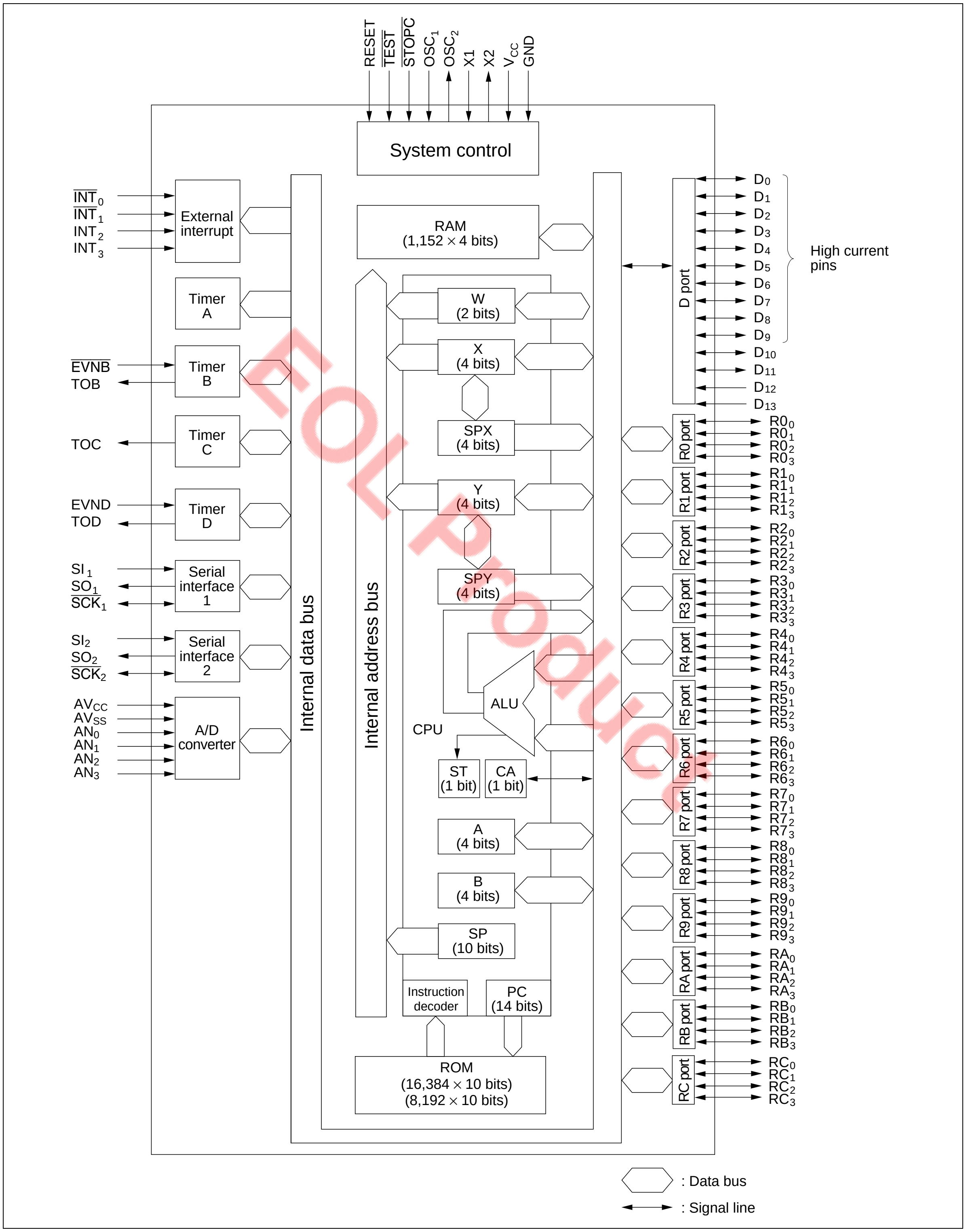
Pin Arrangement



Pin Description

Item	Symbol	Pin Number	I/O	Function
Power supply	V_{CC}	77		Applies power voltage
	GND	10		Connected to ground
Test	$\overline{TEST}$	4	I	Used for factory testing only: Connect this pin to V_{CC}
Reset	RESET	7	I	Resets the MCU
Oscillator	OSC_1	5	I	Input/output pins for the internal oscillator circuit: Connect them to a ceramic oscillator, crystal, or connect OSC_1 to an external oscillator circuit
	OSC_2	6	O	
	X1	8	I	Used for a 32.768-kHz crystal for clock purposes. If not to be used, fix the X1 pin to V_{CC} and leave the X2 pin open.
	X2	9	O	
Port	D_0–D_{11}	11–22	I/O	Input/output pins addressed by individual bits; pins D_0–D_9 are high-current pins that can each supply up to 15 mA
	D_{12}, D_{13}	23, 24	I	Input pins addressable by individual bits
	$R0_0$–RC_3	25–76	I/O	Input/output pins addressable in 4-bit units
Interrupt	$\overline{INT_0}$, $\overline{INT_1}$, INT_2, INT_3	24–27	I	Input pins for external interrupts
Stop clear	$\overline{STOPC}$	23	I	Input pin for transition from stop mode to active mode
Serial	$\overline{SCK_1}$, $\overline{SCK_2}$	42, 46	I/O	Serial clock input/output pin
interface	SI_1, SI_2	43, 47	I	Serial receive data input pin
	SO_1, SO_2	44, 48	O	Serial transmit data output pin
Timer	TOB, TOC, TOD	37–39	O	Timer output pins
	$\overline{EVNB}$, EVND	40, 41	I	Event count input pins
A/D converter	AV_{CC}	78		Power pin for A/D converter: Connect it to the same potential as V_{CC}, as physically close to the V_{CC} pin as possible
	AV_{SS}	3		Ground for AV_{CC}: Connect it to the same potential as GND, as physically close to the GND pin as possible
	AN_0–AN_3	79, 80, 1, 2	I	Analog input pins for A/D converter

Block Diagram



Memory Map

ROM Memory Map

The ROM memory map is shown in figure 1 and described below.

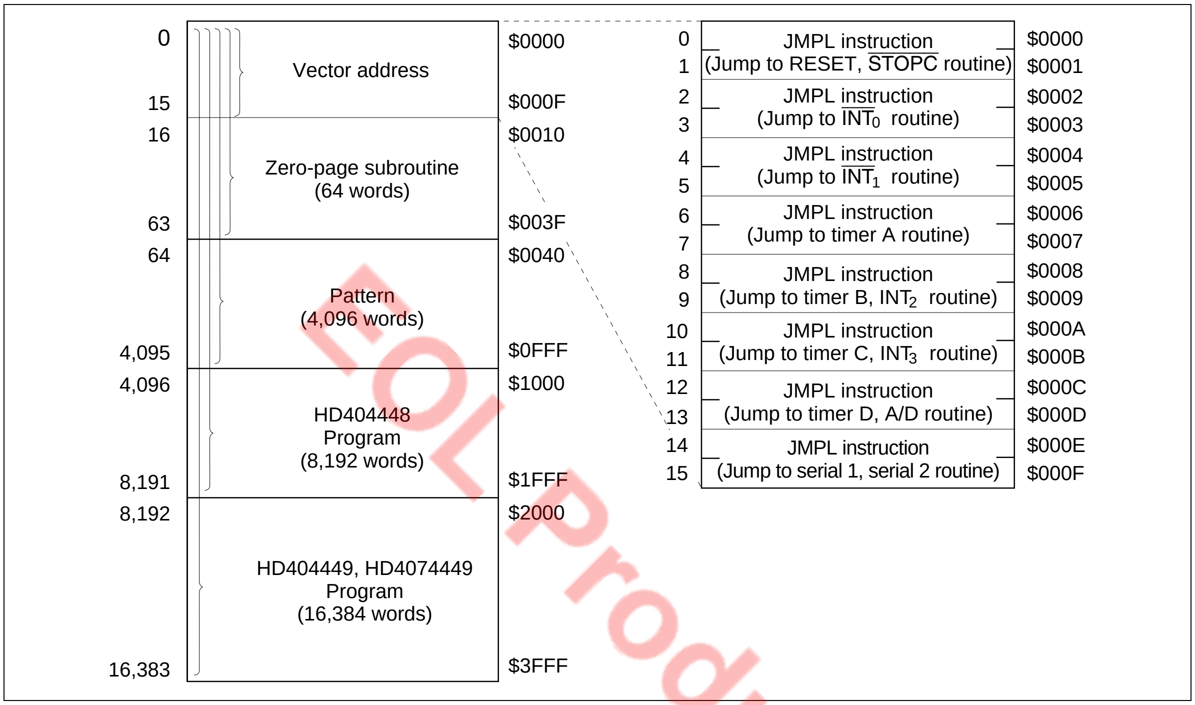


Figure 1 ROM Memory Map

Vector Address Area ($0000–$000F): Reserved for JMPL instructions that branch to the start addresses of the reset and interrupt routines. After MCU reset or an interrupt, program execution continues from the vector address.

Zero-Page Subroutine Area ($0000–$003F): Reserved for subroutines. The program branches to a subroutine in this area in response to the CAL instruction.

Pattern Area ($0000–$0FFF): Contains ROM data that can be referenced with the P instruction.

Program Area ($0000–$1FFF (HD404448), $0000–$3FFF (HD404449, HD4074449)): Used for program coding.

RAM Memory Map

The MCU contains a 1,152-digit × 4-bit RAM area consisting of a memory register area, a data area, and a stack area. In addition, an interrupt control bits area, special register area, and register flag area are mapped onto the same RAM memory space as a RAM-mapped register area outside the above areas. The RAM memory map is shown in figure 2 and described as follows.

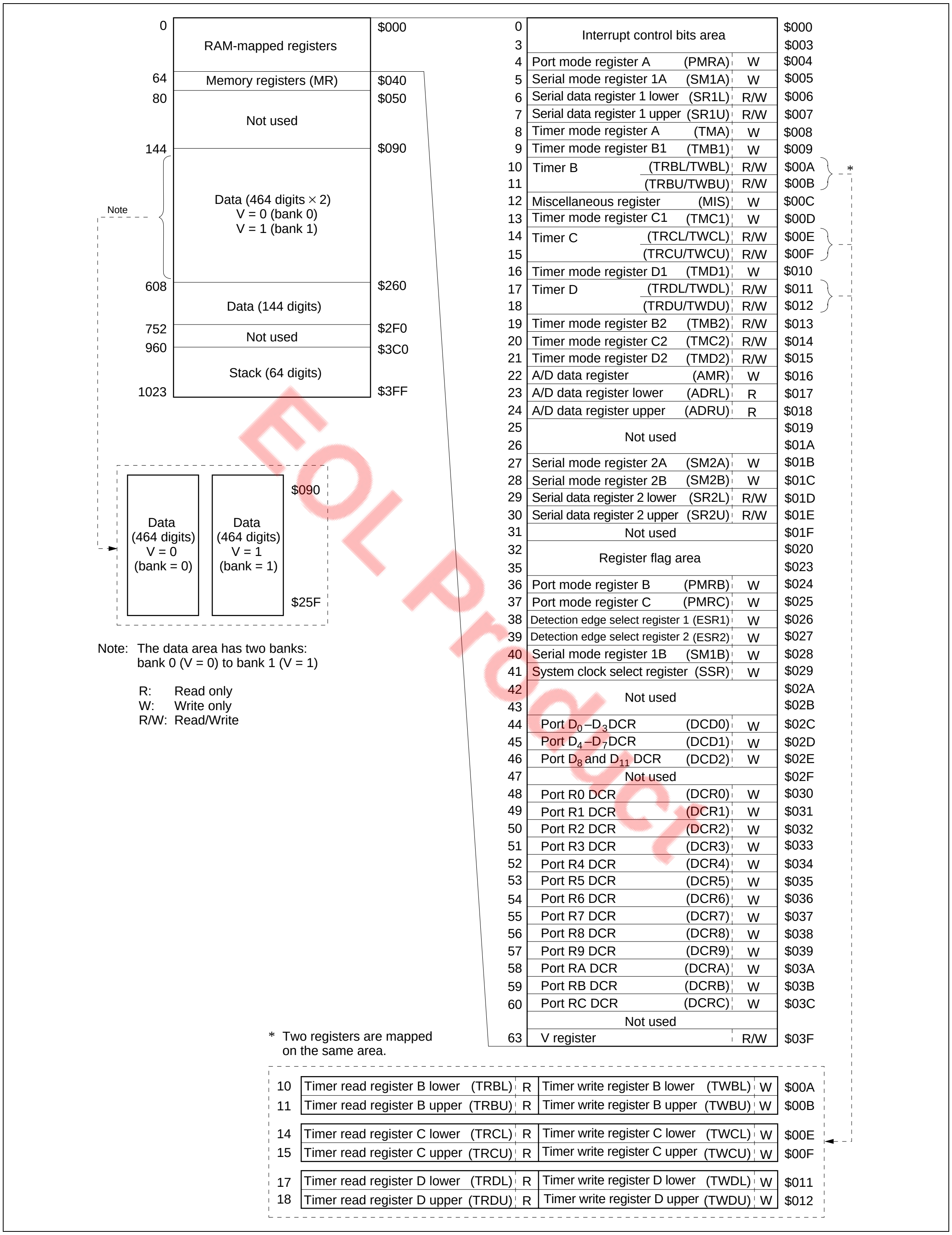


Figure 2 RAM Memory Map

RAM-Mapped Register Area ($000–$03F):

- Interrupt Control Bits Area ($000–$003)

  This area is used for interrupt control bits (figure 3). These bits can be accessed only by RAM bit manipulation instructions (SEM/SEMD, REM/REMD, and TM/TMD). However, note that not all the instructions can be used for each bit. Limitations on using the instructions are shown in figure 4.

- Special Function Register Area ($004–$01F, $024–$03F)

  This area is used as mode registers and data registers for external interrupts, serial interface 1, serial interface 2, timer/counters, A/D converter, and as data control registers for I/O ports. The structure is shown in figures 2 and 5. These registers can be classified into three types: write-only (W), read-only (R), and read/write (R/W). RAM bit manipulation instructions cannot be used for these registers.

- Register Flag Area ($020–$023)

  This area is used for the DTON, WDON, and other register flags and interrupt control bits (figure 3). These bits can be accessed only by RAM bit manipulation instructions (SEM/SEMD, REM/REMD, and TM/TMD). However, note that not all the instructions can be used for each bit. Limitations on using the instructions are shown in figure 4.

Memory Register (MR) Area ($040–$04F): Consisting of 16 addresses, this area (MR0–MR15) can be accessed by register-register instructions (LAMR and XMRA). The structure is shown in figure 6.

Data Area ($090–$2EF): 464 digits from $090 to $25F have two banks, which can be selected by setting the bank register (V: $03F). Before accessing this area, set the bank register to the required value (figure 7). The area from $260 to $2EF is accessed without setting the bank register.

Stack Area ($3C0–$3FF): Used for saving the contents of the program counter (PC), status flag (ST), and carry flag (CA) at subroutine call (CAL or CALL instruction) and for interrupts. This area can be used as a 16-level nesting subroutine stack in which one level requires four digits. The data to be saved and the save conditions are shown in figure 6.

The program counter is restored by either the RTN or RTNI instruction, but the status and carry flags can only be restored by the RTNI instruction. Any unused space in this area is used for data storage.

	Bit 3	Bit 2	Bit 1	Bit 0	
0	IM0 (IM of $\overline{INT_0}$)	IF0 (IF of $\overline{INT_0}$)	RSP (Reset SP bit)	IE (Interrupt enable flag)	\$000
1	IMTA (IM of timer A)	IFTA (IF of timer A)	IM1 (IM of $\overline{INT_1}$)	IF1 (IF of $\overline{INT_1}$)	\$001
2	IMTC (IM of timer C)	IFTC (IF of timer C)	IMTB (IM of timer B)	IFTB (IF of timer B)	\$002
3	IMS1 (IM of serial interface 1)	IFS1 (IF of serial interface 1)	IMTD (IM of timer D)	IFTD (IF of timer D)	\$003

Interrupt control bits area

	Bit 3	Bit 2	Bit 1	Bit 0	
32	DTON (Direct transfer on flag)	ADSF (A/D start flag)	WDON (Watchdog on flag)	LSON (Low speed on flag)	\$020
33	RAME (RAM enable flag)	Not used	ICEF (Input capture error flag)	ICSF (Input capture status flag)	\$021
34	IM3 (IM of INT_3)	IF3 (IF of INT_3)	IM2 (IM of INT_2)	IF2 (IF of INT_2)	\$022
35	IMS2 (IM of serial interface 2)	IFS2 (IF of serial interface 2)	IMAD (IM of A/D)	IFAD (IF of A/D)	\$023

Register flag area

IF: Interrupt request flag
IM: Interrupt mask
IE: Interrupt enable flag
SP: Stack pointer

Figure 3 Configuration of Interrupt Control Bits and Register Flag Areas

	SEM/SEMD	REM/REMD	TM/TMD
IE	Allowed	Allowed	Allowed
IM			
LSON			
IF	Not executed	Allowed	Allowed
ICSF			
ICEF			
RAME			
RSP	Not executed	Allowed	Inhibited
WDON	Allowed	Not executed	Inhibited
ADSF	Allowed	Inhibited	Allowed
DTON	Not executed in active mode	Allowed	Allowed
	Used in subactive mode		
Not used	Not executed	Not executed	Inhibited

Note: WDON is reset by MCU reset or by $\overline{\text{STOPC}}$ enable for stop mode cancellation.
The REM or REMD instuction must not be executed for ADSF during A/D conversion.
DTON is always reset in active mode.
If the TM or TMD instruction is executed for the inhibited bits or non-existing bits, the value in ST becomes invalid.

Figure 4 Usage Limitations of RAM Bit Manipulation Instructions

	Bit 3	Bit 2	Bit 1	Bit 0
\$000 – \$003	Interrupt control bits area			
PMRA \$004	$R5_2/SI_2$	$R5_3/SO_2$	$R4_2/SI_1$	$R4_3/SO_1$
SM1A \$005	$R4_1/\overline{SCK}_1$	Serial transmit clock speed selection 1		
SR1L \$006	Serial data register 1 (lower digit)			
SR1U \$007	Serial data register 1 (upper digit)			
TMA \$008	*1	Clock source selection (timer A)		
TMB1 \$009	*2	Clock source selection (timer B)		
TRBL/TWBL \$00A	Timer B register (lower digit)			
TRBU/TWBU \$00B	Timer B register (upper digit)			
MIS \$00C	*3	$R4_3/SO_1$ PMOS control	Interrupt frame period selection	
TMC1 \$00D	*2	Clock source selection (timer C)		
TRCL/TWCL \$00E	Timer C register (lower digit)			
TRCU/TWCU \$00F	Timer C register (upper digit)			
TMD1 \$010	*2	Clock source selection (timer D)		
TRDL/TWDL \$011	Timer D register (lower digit)			
TRDU/TWDU \$012	Timer D register (upper digit)			
TMB2 \$013	Not used	Not used	Timer-B output mode selection	
TMC2 \$014	Not used	Timer-C output mode selection		
TMD2 \$015	*4	Timer-D output mode selection		
AMR \$016	Analog channel selection		Not used	*5
ADRL \$017	A/D data register (lower digit)			
ADRU \$018	A/D data register (upper digit)			
	Not used			
SM2A \$01B	$R5_1/\overline{SCK}_2$	Serial transmit clock speed selection 2		
SM2B \$01C	Not used	$R5_3/SO_2$ PMOS control	*6	*7
SR2L \$01D	Serial data register 2 (lower digit)			
SR2U \$01E	Serial data register 2 (upper digit)			
	Not used			
\$020 – \$023	Register flag area			
PMRB \$024	Not used	$R0_2/INT_3$	$R0_1/INT_2$	$R0_0/\overline{INT}_1$
PMRC \$025	$D_{13}/\overline{INT}_0$	$D_{12}/\overline{STOPC}$	$R4_0$/EVND	$R3_3/\overline{EVNB}$
ESR1 \$026	INT_3 detection edge selection		INT_2 detection edge selection	
ESR2 \$027	EVND detection edge selection		Not used	
SM1B \$028	Not used	Not used	*8	*9
SSR \$029	*10	*11	*12	Not used
	Not used			
DCD0 \$02C	Port D_3 DCR	Port D_2 DCR	Port D_1 DCR	Port D_0 DCR
DCD1 \$02D	Port D_7 DCR	Port D_6 DCR	Port D_5 DCR	Port D_4 DCR
DCD2 \$02E	Port D_{11} DCR	Port D_{10} DCR	Port D_9 DCR	Port D_8 DCR
	Not used			
DCR0 \$030	Port $R0_3$ DCR	Port $R0_2$ DCR	Port $R0_1$ DCR	Port $R0_0$ DCR
DCR1 \$031	Port $R1_3$ DCR	Port $R1_2$ DCR	Port $R1_1$ DCR	Port $R1_0$ DCR
DCR2 \$032	Port $R2_3$ DCR	Port $R2_2$ DCR	Port $R2_1$ DCR	Port $R2_0$ DCR
DCR3 \$033	Port $R3_3$ DCR	Port $R3_2$ DCR	Port $R3_1$ DCR	Port $R3_0$ DCR
DCR4 \$034	Port $R4_3$ DCR	Port $R4_2$ DCR	Port $R4_1$ DCR	Port $R4_0$ DCR
DCR5 \$035	Port $R5_3$ DCR	Port $R5_2$ DCR	Port $R5_1$ DCR	Port $R5_0$ DCR
DCR6 \$036	Port $R6_3$ DCR	Port $R6_2$ DCR	Port $R6_1$ DCR	Port $R6_0$ DCR
DCR7 \$037	Port $R7_3$ DCR	Port $R7_2$ DCR	Port $R7_1$ DCR	Port $R7_0$ DCR
DCR8 \$038	Port $R8_3$ DCR	Port $R8_2$ DCR	Port $R8_1$ DCR	Port $R8_0$ DCR
DCR9 \$039	Port $R9_3$ DCR	Port $R9_2$ DCR	Port $R9_1$ DCR	Port $R9_0$ DCR
DCRA \$03A	Port RA_3 DCR	Port RA_2 DCR	Port RA_1 DCR	Port RA_0 DCR
DCRB \$03B	Port RB_3 DCR	Port RB_2 DCR	Port RB_1 DCR	Port RB_0 DCR
DCRC \$03C	Port RC_3 DCR	Port RC_2 DCR	Port RC_1 DCR	Port RC_0 DCR
	Not used			
V \$03F	Not used	Not used	Not used	*13

Notes:
1. Timer-A/time-base
2. Auto-reload on/off
3. Pull-up MOS control
4. Input capture selection
5. A/D conversion time
6. SO_2 ouput control in idle states
7. Serial clock source selection 2
8. SO_1 output level control in idle states
9. Serial clock source selection 1
10. 32-kHz oscillation stop
11. 32-kHz oscillation division ratio
12. System clock selection
13. Bank 0, 1 selection

Figure 5 Special Function Register Area

Memory registers

64	MR(0)	$040
65	MR(1)	$041
66	MR(2)	$042
67	MR(3)	$043
68	MR(4)	$044
69	MR(5)	$045
70	MR(6)	$046
71	MR(7)	$047
72	MR(8)	$048
73	MR(9)	$049
74	MR(10)	$04A
75	MR(11)	$04B
76	MR(12)	$04C
77	MR(13)	$04D
78	MR(14)	$04E
79	MR(15)	$04F

Stack area

960	Level 16	$3C0
	Level 15	
	Level 14	
	Level 13	
	Level 12	
	Level 11	
	Level 10	
	Level 9	
	Level 8	
	Level 7	
	Level 6	
	Level 5	
	Level 4	
	Level 3	
	Level 2	
1023	Level 1	$3FF

	Bit 3	Bit 2	Bit 1	Bit 0	
1020	ST	$\overline{PC_{13}}$	$\overline{PC_{12}}$	$\overline{PC_{11}}$	$3FC
1021	$\overline{PC_{10}}$	$\overline{PC_{9}}$	$\overline{PC_{8}}$	$\overline{PC_{7}}$	$3FD
1022	CA	$\overline{PC_{6}}$	$\overline{PC_{5}}$	$\overline{PC_{4}}$	$3FE
1023	$\overline{PC_{3}}$	$\overline{PC_{2}}$	$\overline{PC_{1}}$	$\overline{PC_{0}}$	$3FF

PC_{13}–PC_0 : Program counter
ST: Status flag
CA: Carry flag

Figure 6 Configuration of Memory Registers and Stack Area, and Stack Position

Bank register (V: $03F)

Bit	3	2	1	0
Initial value	—	—	—	0
Read/Write	—	—	—	R/W
Bit name	Not used	Not used	Not used	V0

V0	Bank area selection
0	Bank 0 is selected
1	Bank 1 is selected

Note: After reset, the value in the bank register is 0, and therefore bank 0 is selected.

Figure 7 Bank Register (V)

Functional Description

Registers and Flags

The MCU has nine registers and two flags for CPU operations. They are shown in figure 8 and described below.

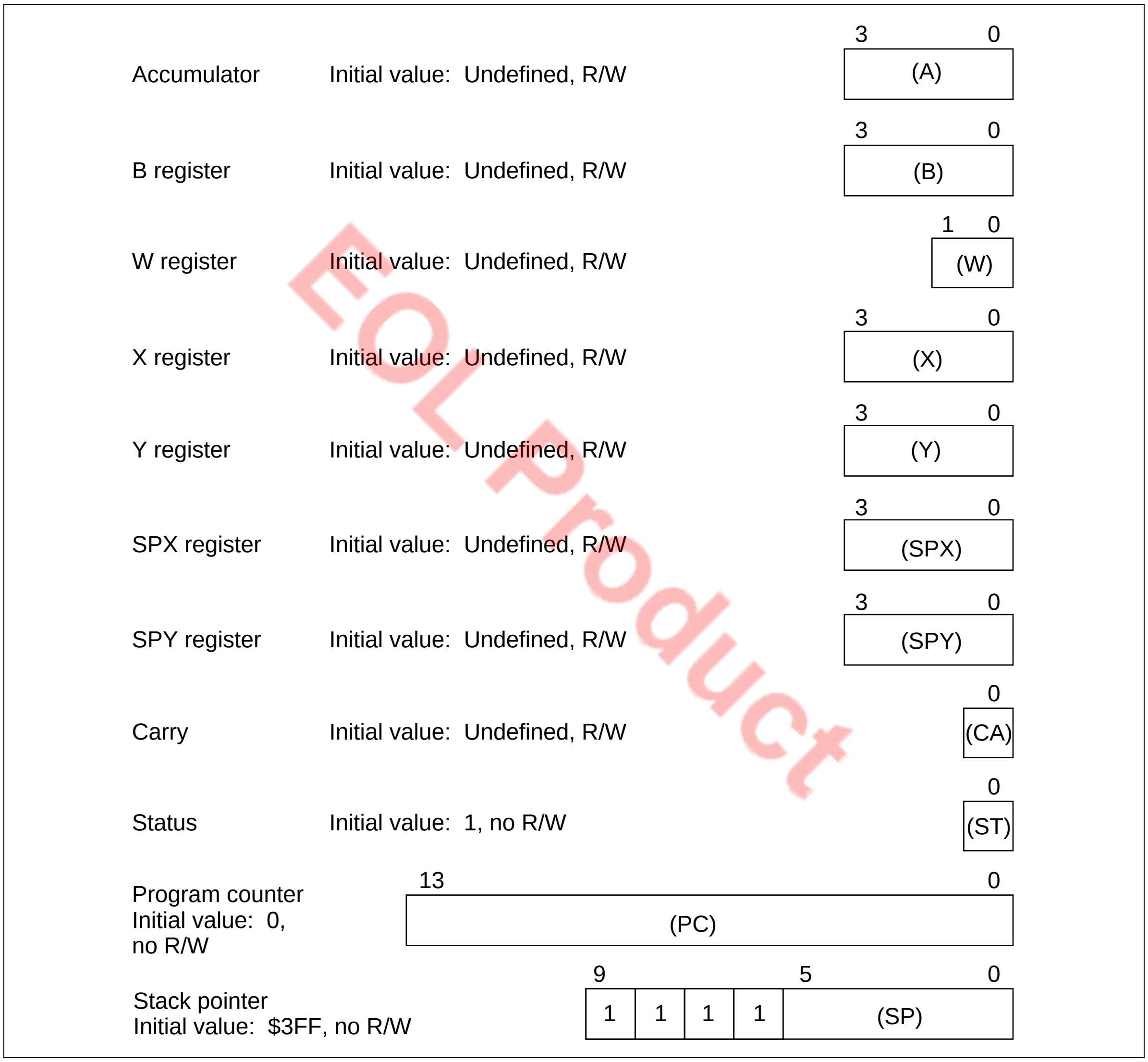


Figure 8 Registers and Flags

Accumulator (A), B Register (B): Four-bit registers used to hold the results from the arithmetic logic unit (ALU) and transfer data between memory, I/O, and other registers.

W Register (W), X Register (X), Y Register (Y): Two-bit (W) and four-bit (X and Y) registers used for indirect RAM addressing. The Y register is also used for D-port addressing.

SPX Register (SPX), SPY Register (SPY): Four-bit registers used to supplement the X and Y registers.

Carry Flag (CA): One-bit flag that stores any ALU overflow generated by an arithmetic operation. CA is affected by the SEC, REC, ROTL, and ROTR instructions. A carry is pushed onto the stack during an interrupt and popped from the stack by the RTNI instruction—but not by the RTN instruction.

Status Flag (ST): One-bit flag that latches any overflow generated by an arithmetic or compare instruction, not-zero decision from the ALU, or result of a bit test. ST is used as a branch condition of the BR, BRL, CAL, and CALL instructions. The contents of ST remain unchanged until the next arithmetic, compare, or bit test instruction is executed, but become 1 after the BR, BRL, CAL, or CALL instruction is read, regardless of whether the instruction is executed or skipped. The contents of ST are pushed onto the stack during an interrupt and popped from the stack by the RTNI instruction—but not by the RTN instruction.

Program Counter (PC): 14-bit binary counter that points to the ROM address of the instruction being executed.

Stack Pointer (SP): Ten-bit pointer that contains the address of the stack area to be used next. The SP is initialized to $3FF by MCU reset. It is decremented by 4 when data is pushed onto the stack, and incremented by 4 when data is popped from the stack. The top four bits of the SP are fixed at 1111, so a stack can be used up to 16 levels.

The SP can be initialized to $3FF in another way: by resetting the RSP bit with the REM or REMD instruction.

Reset

The MCU is reset by inputting a high-level voltage to the RESET pin. At power-on or when stop mode is cancelled, RESET must be high for at least one t_{RC} to enable the oscillator to stabilize. During operation, RESET must be high for at least two instruction cycles.

Initial values after MCU reset are listed in table 1.

Table 1 Initial Values After MCU Reset

Item		Abbr.	Initial Value	Contents
Program counter		(PC)	$0000	Indicates program execution point from start address of ROM area
Status flag		(ST)	1	Enables conditional branching
Stack pointer		(SP)	$3FF	Stack level 0
Interrupt flags/mask	Interrupt enable flag	(IE)	0	Inhibits all interrupts
	Interrupt request flag	(IF)	0	Indicates there is no interrupt request
	Interrupt mask	(IM)	1	Prevents (masks) interrupt requests
I/O	Port data register	(PDR)	All bits 1	Enables output at level 1
	Data control register	(DCD0–DCD2)	All bits 0	Turns output buffer off (to high impedance)
		(DCR0–DCRC)	All bits 0	
	Port mode register A	(PMRA)	0000	Refer to description of port mode register A
	Port mode register B	(PMRB)	- 000	Refer to description of port mode register B
	Port mode register C	(PMRC)	0000	Refer to description of port mode register C
	Detection edge select register 1	(ESR1)	0000	Disables edge detection
	Detection edge select register 2	(ESR2)	00 - -	Disables edge detection
Timer/ counters, serial interface	Timer mode register A	(TMA)	0000	Refer to description of timer mode register A
	Timer mode register B1	(TMB1)	0000	Refer to description of timer mode register B1
	Timer mode register B2	(TMB2)	- - 00	Refer to description of timer mode register B2
	Timer mode register C1	(TMC1)	0000	Refer to description of timer mode register C1
	Timer mode register C2	(TMC2)	- 000	Refer to description of timer mode register C2
	Timer mode register D1	(TMD1)	0000	Refer to description of timer mode register D1
	Timer mode register D2	(TMD2)	0000	Refer to description of timer mode register D2
	Serial mode register 1A	(SM1A)	0000	Refer to description of serial mode register 1A
	Serial mode register 1B	(SM1B)	- - 00	Refer to description of serial mode register 1B
	Serial mode register 2A	(SM2A)	0000	Refer to description of serial mode register 2A
	Serial mode register 2B	(SM2B)	- 000	Refer to description of serial mode register 2B
	Prescaler S	(PSS)	$000	—
	Prescaler W	(PSW)	$00	—

Item		Abbr.	Initial Value	Contents
Timer/ counters, serial interface	Timer counter A	(TCA)	$00	—
	Timer counter B	(TCB)	$00	—
	Timer counter C	(TCC)	$00	—
	Timer counter D	(TCD)	$00	—
	Timer write register B	(TWBU, TWBL)	$X0	—
	Timer write register C	(TWCU, TWCL)	$X0	—
	Timer write register D	(TWDU, TWDL)	$X0	—
	Octal counter		000	—
A/D	A/D mode register	(AMR)	00 - 0	Refer to description of A/D mode register
Bit register	Low speed on flag	(LSON)	0	Refer to description of operating modes
	Watchdog timer on flag	(WDON)	0	Refer to description of timer C
	A/D start flag	(ADSF)	0	Refer to description of A/D converter
	Direct transfer on flag	(DTON)	0	Refer to description of operating modes
	Input capture status flag	(ICSF)	0	Refer to description of timer D
	Input capture error flag	(ICEF)	0	Refer to description of timer D
Others	Miscellaneous register	(MIS)	0000	Refer to description of operating modes, and oscillator circuit
	System clock select register bits 2–0	(SSR2–SSR0)	00 -	Refer to description of operating modes, and oscillator circuit
	Bank register	(V)	- - - 0	Refer to description of RAM memory map

Notes: 1. The statuses of other registers and flags after MCU reset are shown in the following table.
2. X indicates invalid value. – indicates that the bit does not exist.

Item	Abbr.	Status After Cancellation of Stop Mode by $\overline{STOPC}$ Input	Status After Cancellation of Stop Mode by MCU Reset	Status After all Other Types of Reset
Carry flag	(CA)	Pre-stop-mode values are not guaranteed; values must be initialized by program		Pre-stop-mode values are not guaranteed; values must be initialized by program
Accumulator	(A)	Pre-stop-mode values are not guaranteed; values must be initialized by program		Pre-stop-mode values are not guaranteed; values must be initialized by program
B register	(B)	Pre-stop-mode values are not guaranteed; values must be initialized by program		Pre-stop-mode values are not guaranteed; values must be initialized by program
W register	(W)	Pre-stop-mode values are not guaranteed; values must be initialized by program		Pre-stop-mode values are not guaranteed; values must be initialized by program
X/SPX register	(X/SPX)	Pre-stop-mode values are not guaranteed; values must be initialized by program		Pre-stop-mode values are not guaranteed; values must be initialized by program
Y/SPY register	(Y/SPY)	Pre-stop-mode values are not guaranteed; values must be initialized by program		Pre-stop-mode values are not guaranteed; values must be initialized by program
Serial data register	(SRL, SRU)	Pre-stop-mode values are not guaranteed; values must be initialized by program		Pre-stop-mode values are not guaranteed; values must be initialized by program
A/D data register	(ADRL, ADRU)	Pre-stop-mode values are not guaranteed; values must be initialized by program		Pre-stop-mode values are not guaranteed; values must be initialized by program
RAM		Pre-stop-mode values are retained		
RAM enable flag	(RAME)	1	0	0
Port mode register 1 bit 2	(PMRC12)	Pre-stop-mode values are retained	0	0
System clock select register bit 3	(SSR3)	Pre-stop-mode values are retained	0	0

Interrupts

The MCU has 11 interrupt sources: four external signals ($\overline{INT_0}$, $\overline{INT_1}$, INT_2, INT_3), four timer/counters (timers A, B, C, and D), two serial interfaces (serial 1, serial 2), and A/D converter.

An interrupt request flag (IF), interrupt mask (IM), and vector address are provided for each interrupt source, and an interrupt enable flag (IE) controls the entire interrupt process.

Some vector addresses are shared by two different interrupts. They are timer B and INT_2, timer C and INT_3, timer D and A/D converter, and serial interface 1 and serial interface 2. So the type of request that has occurred must be checked at the beginning of interrupt processing.

Interrupt Control Bits and Interrupt Processing: Locations $000 to $003 and $022 to $023 in RAM are reserved for the interrupt control bits which can be accessed by RAM bit manipulation instructions.

The interrupt request flag (IF) cannot be set by software. MCU reset initializes the interrupt enable flag (IE) and the IF to 0 and the interrupt mask (IM) to 1.

A block diagram of the interrupt control circuit is shown in figure 9, interrupt priorities and vector addresses are listed in table 2, and interrupt processing conditions for the 11 interrupt sources are listed in table 3.

An interrupt request occurs when the IF is set to 1 and the IM is set to 0. If the IE is 1 at that point, the interrupt is processed. A priority programmable logic array (PLA) generates the vector address assigned to that interrupt source.

The interrupt processing sequence is shown in figure 10 and an interrupt processing flowchart is shown in figure 11. After an interrupt is acknowledged, the previous instruction is completed in the first cycle. The IE is reset in the second cycle, the carry, status, and program counter values are pushed onto the stack during the second and third cycles, and the program jumps to the vector address to execute the instruction in the third cycle.

Program the JMPL instruction at each vector address, to branch the program to the start address of the interrupt program, and reset the IF by a software instruction within the interrupt program.

Table 2 Vector Addresses and Interrupt Priorities

Reset/Interrupt	Priority	Vector Address
RESET, $\overline{\text{STOPC}}$*	—	$0000
$\overline{\text{INT}_0}$	1	$0002
$\overline{\text{INT}_1}$	2	$0004
Timer A	3	$0006
Timer B, INT_2	4	$0008
Timer C, INT_3	5	$000A
Timer D, A/D	6	$000C
Serial 1, Serial 2	7	$000E

Note: * The $\overline{\text{STOPC}}$ interrupt request is valid only in stop mode

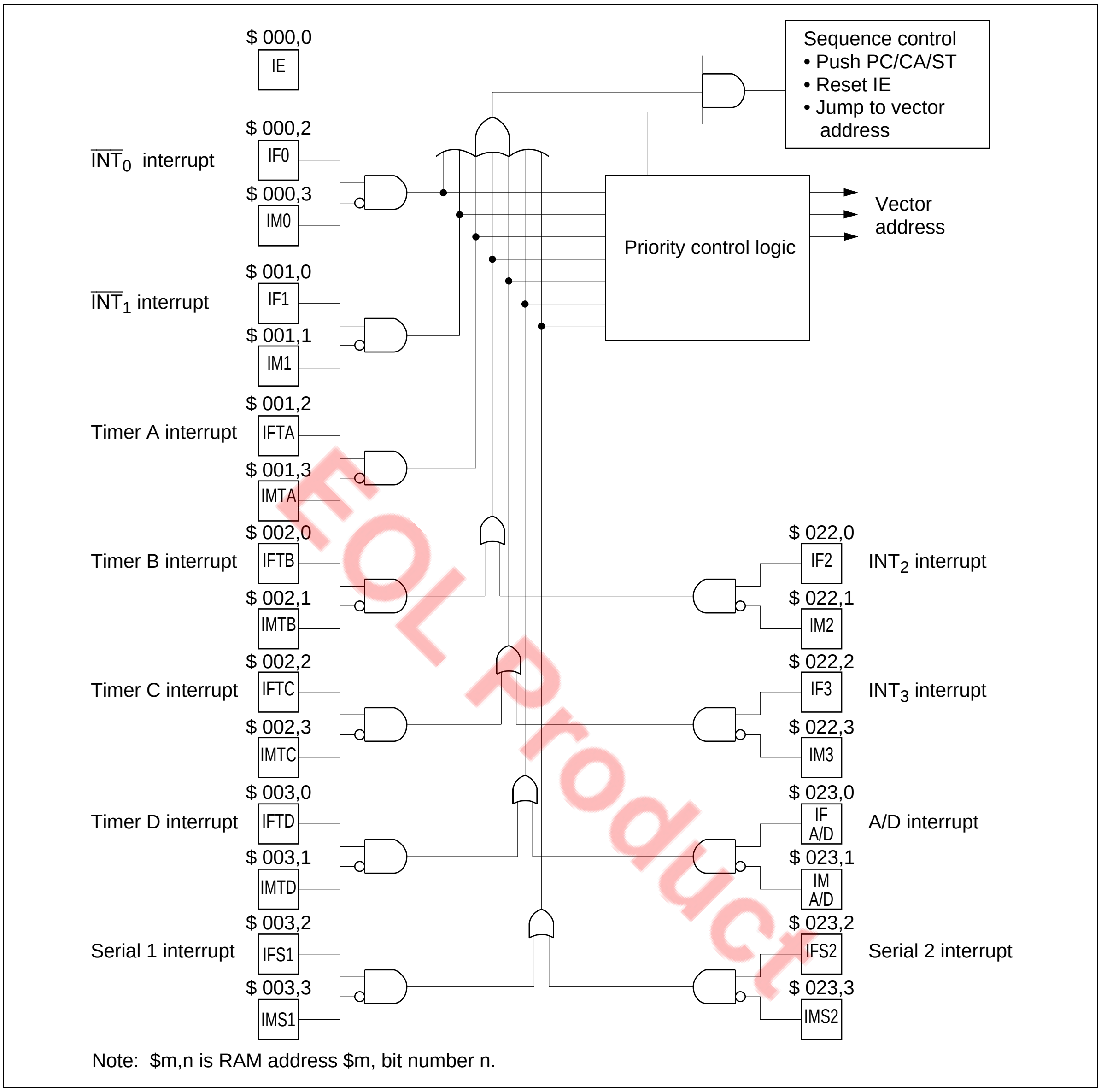


Figure 9 Interrupt Control Circuit

Table 3 Interrupt Processing and Activation Conditions

Interrupt Control Bit	Interrupt Source $\overline{INT_0}$	$\overline{INT_1}$	Timer A	Timer B or INT_2	Timer C or INT_3	Timer D or A/D	Serial 1 or Serial 2
IE	1	1	1	1	1	1	1
IF0 · $\overline{IM0}$	1	0	0	0	0	0	0
IF1 · $\overline{IM1}$	*	1	0	0	0	0	0
IFTA · $\overline{IMTA}$	*	*	1	0	0	0	0
IFTB · $\overline{IMTB}$ + IF2 · $\overline{IM2}$	*	*	*	1	0	0	0
IFTC · $\overline{IMTC}$ + IF3 · $\overline{IM3}$	*	*	*	*	1	0	0
IFTD · $\overline{IMTD}$ + IFAD · $\overline{IMAD}$	*	*	*	*	*	1	0
IFS1 · $\overline{IMS1}$ + IFS2 · $\overline{IMS2}$	*	*	*	*	*	*	1

Note: Bits marked * can be either 0 or 1. Their values have no effect on operation.

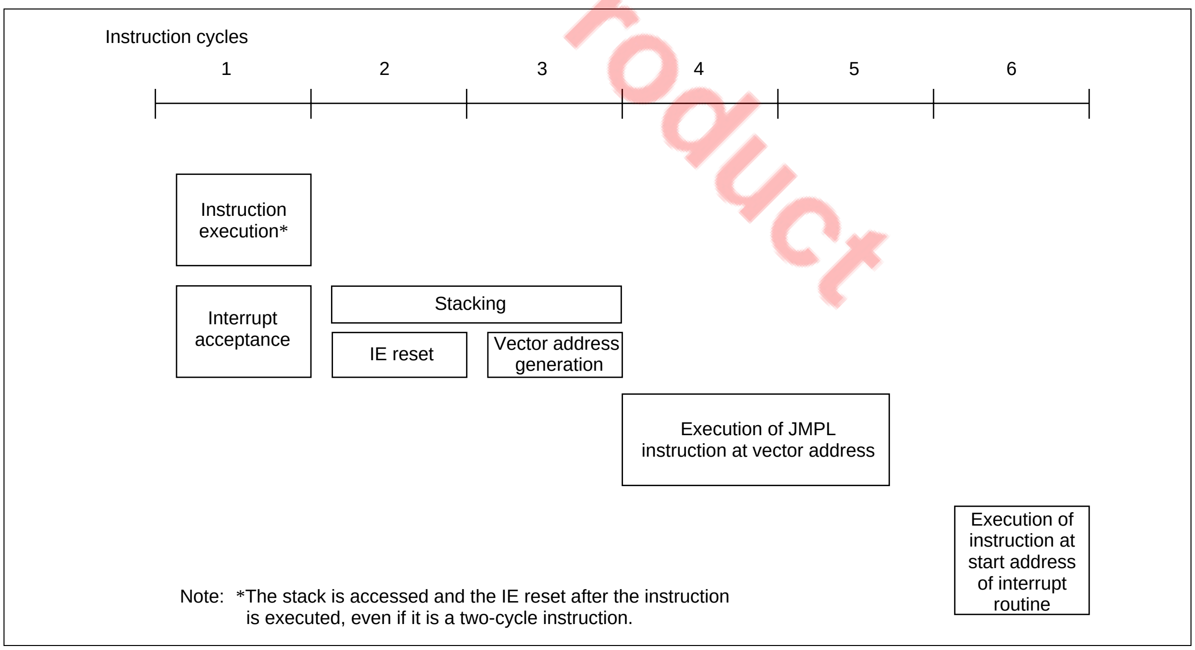


Figure 10 Interrupt Processing Sequence

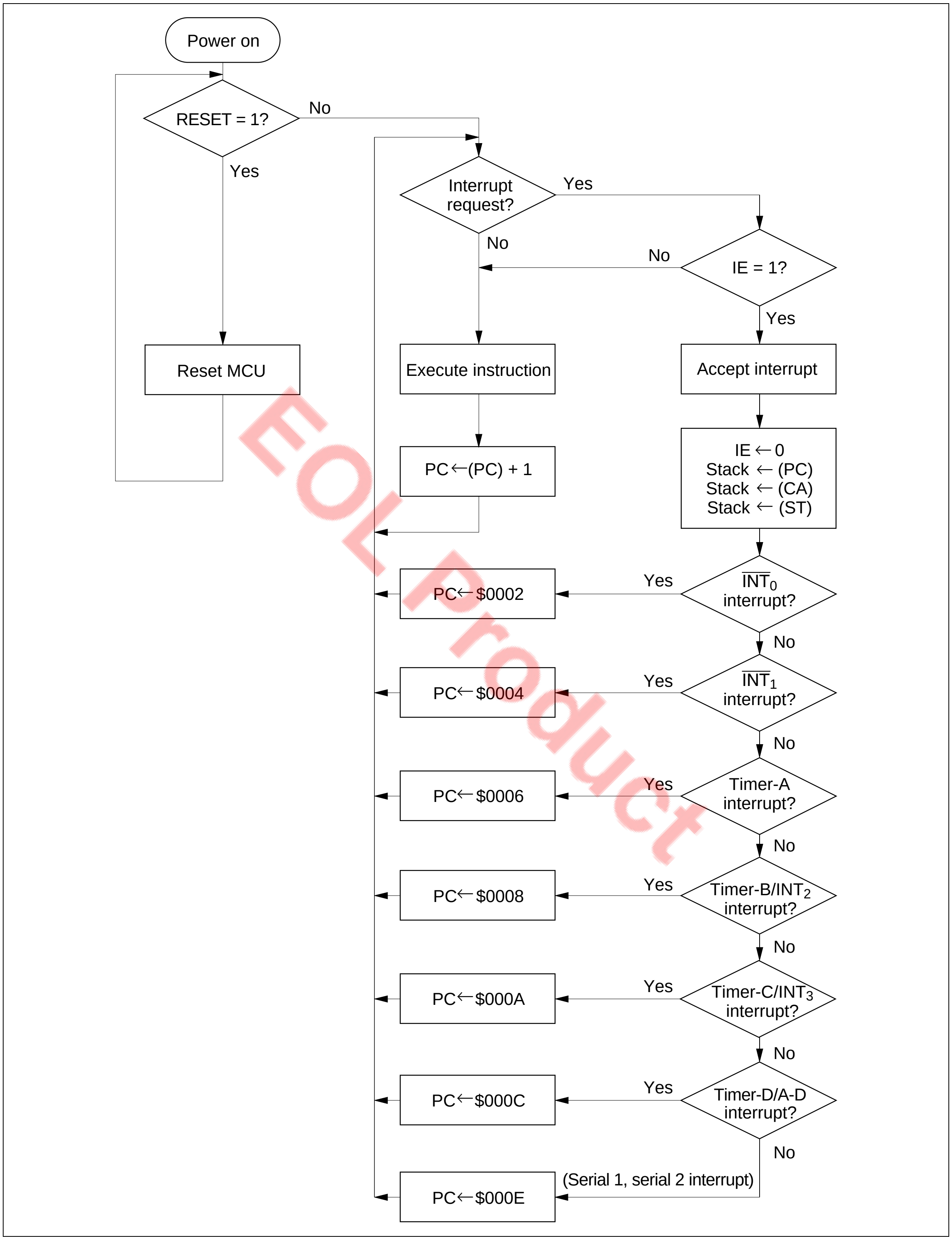


Figure 11 Interrupt Processing Flowchart

Interrupt Enable Flag (IE: $000, Bit 0): Controls the entire interrupt process. It is reset by the interrupt processing and set by the RTNI instruction, as listed in table 4.

Table 4 Interrupt Enable Flag (IE: $000, Bit 0)

IE	Interrupt Enabled/Disabled
0	Disabled
1	Enabled

External Interrupts ($\overline{INT_0}$, $\overline{INT_1}$, INT_2, INT_3): Four external interrupt signals.

External Interrupt Request Flags (IF0, IF1, IF2, IF3: $000, $001, $022): IF0 and IF1 are set at the falling edge of signals input to $\overline{INT_0}$ and $\overline{INT_1}$, and IF2 and IF3 are set at the rising or falling edge of signals input to INT_2 and INT_3, as listed in table 5. The INT_2 and INT_3 interrupt edges are selected by the detection edge select registers (ESR1, ESR2: $026, $027) as shown in figures 12 and 13.

Table 5 External Interrupt Request Flags (IF0–IF3: $000, $001, $022)

IF0–IF3	Interrupt Request
0	No
1	Yes

Detection edge selection register 1 (ESR1: $026)

Bit	3	2	1	0
Initial value	0	0	0	0
Read/Write	W	W	W	W
Bit name	ESR13	ESR12	ESR11	ESR10

ESR13	ESR12	INT_3 detection edge
0	0	No detection
	1	Falling-edge detection
1	0	Rising-edge detection
	1	Double-edge detection*

ESR11	ESR10	INT_2 detection edge
0	0	No detection
	1	Falling-edge detection
1	0	Rising-edge detection
	1	Double-edge detection*

Note: *Both falling and rising edges are detected.

Figure 12 Detection Edge Selection Register 1 (ESR1)

Detection edge selection register 2 (ESR2: $027)

Bit	3	2	1	0
Initial value	0	0	—	—
Read/Write	W	W	—	—
Bit name	ESR23	ESR22	Not used	Not used

ESR23	ESR22	EVND detection edge
0	0	No detection
	1	Falling-edge detection
1	0	Rising-edge detection
	1	Double-edge detection*

Note: *Both falling and rising edges are detected.

Figure 13 Detection Edge Selection Register 2 (ESR2)

External Interrupt Masks (IM0, IM1, IM2, IM3: $000, $001, $022): Prevent (mask) interrupt requests caused by the corresponding external interrupt request flags, as listed in table 6.

Table 6 External Interrupt Masks (IM0–IM3: $000, $001, $022)

IM0–IM3	Interrupt Request
0	Enabled
1	Disabled (Masked)

Timer A Interrupt Request Flag (IFTA: $001, Bit 2): Set by overflow output from timer A, as listed in table 7.

Table 7 Timer A Interrupt Request Flag (IFTA: $001, Bit 2)

IFTA	Interrupt Request
0	No
1	Yes

Timer A Interrupt Mask (IMTA: $001, Bit 3): Prevents (masks) an interrupt request caused by the timer A interrupt request flag, as listed in table 8.

Table 8 Timer A Interrupt Mask (IMTA: $001, Bit 3)

IMTA	Interrupt Request
0	Enabled
1	Disabled (Masked)

Timer B Interrupt Request Flag (IFTB: $002, Bit 0): Set by overflow output from timer B, as listed in table 9.

Table 9 Timer B Interrupt Request Flag (IFTB: $002, Bit 0)

IFTB	Interrupt Request
0	No
1	Yes

Timer B Interrupt Mask (IMTB: $002, Bit 1): Prevents (masks) an interrupt request caused by the timer B interrupt request flag, as listed in table 10.

Table 10 Timer B Interrupt Mask (IMTB: $002, Bit 1)

IMTB	Interrupt Request
0	Enabled
1	Disabled (Masked)

Timer C Interrupt Request Flag (IFTC: $002, Bit 2): Set by overflow output from timer C, as listed in table 11.

Table 11 Timer C Interrupt Request Flag (IFTC: $002, Bit 2)

IFTC	Interrupt Request
0	No
1	Yes

Timer C Interrupt Mask (IMTC: $002, Bit 3): Prevents (masks) an interrupt request caused by the timer C interrupt request flag, as listed in table 12.

Table 12 Timer C Interrupt Mask (IMTC: $002, Bit 3)

IMTC	Interrupt Request
0	Enabled
1	Disabled (Masked)

Timer D Interrupt Request Flag (IFTD: $003, Bit 0): Set by overflow output from timer D, or by the rising or falling edge of signals input to EVND when the input capture function is used, as listed in table 13.

Table 13 Timer D Interrupt Request Flag (IFTD: $003, Bit 0)

IFTD	Interrupt Request
0	No
1	Yes

Timer D Interrupt Mask (IMTD: $003, Bit 1): Prevents (masks) an interrupt request caused by the timer D interrupt request flag, as listed in table 14.

Table 14 Timer D Interrupt Mask (IMTD: $003, Bit 1)

IMTD	Interrupt Request
0	Enabled
1	Disabled (Masked)

Serial Interrupt Request Flags (IFS1: $003, Bit 2; IFS2: $023, Bit 2) Set when data transfer is completed or when data transfer is suspended, as listed in table 15.

Table 15 Serial Interrupt Request Flag (IFS1: $003, Bit 2; IFS2: $023, Bit 2)

IFS1, IFS2	Interrupt Request
0	No
1	Yes

Serial Interrupt Masks (IMS1: $003, Bit 3; IMS2: $023, Bit 3): Prevents (masks) an interrupt request caused by the serial interrupt request flag, as listed in table 16.

Table 16 Serial Interrupt Mask (IMS1: $003, Bit 3; IMS2: $023, Bit 3)

IMS1, IMS2	Interrupt Request
0	Enabled
1	Disabled (Masked)

A/D Interrupt Request Flag (IFAD: $023, Bit 0): Set at the completion of A/D conversion, as listed in table 17.

Table 17 A/D Interrupt Request Flag (IFAD: $023, Bit 0)

IFAD	Interrupt Request
0	No
1	Yes

A/D Interrupt Mask (IMAD: $023, Bit 1): Prevents (masks) an interrupt request caused by the A/D interrupt request flag, as listed in table 18.

Table 18 A/D Interrupt Mask (IMAD: $023, Bit 1)

IMAD	Interrupt Request
0	Enabled
1	Disabled (Masked)

Operating Modes

The MCU has five operating modes as shown in table 19. The operations in each mode are listed in tables 20 and 21. Transitions between operating modes are shown in figure 14.

Active Mode: All MCU functions operate according to the clock generated by the system oscillators OSC_1 and OSC_2.

Table 19 Operating Modes and Clock Status

		Mode Name				
		Active	**Standby**	**Stop**	**Watch**	**Subactive*[2]**
Activation method		RESET cancellation, interrupt request, $\overline{\text{STOPC}}$ cancellation in stop mode, STOP/SBY instruction in subactive mode (when direct transfer is selected)	SBY instruction	STOP instruction when TMA3 = 0	STOP instruction when TMA3 = 1	$\overline{\text{INT}_0}$ or timer A interrupt request from watch mode
Status	System oscillator	OP	OP	Stopped	Stopped	Stopped
	Subsystem oscillator	OP	OP	OP*[1]	OP	OP
Cancellation method		RESET input, STOP/SBY instruction	RESET input, interrupt request	RESET input, $\overline{\text{STOPC}}$ input in stop mode	RESET input, $\overline{\text{INT}_0}$ or timer A interrupt request	RESET input, STOP/SBY instruction

Note: OP implies in operation

1. Operating or stopping the oscillator can be selected by setting bit 3 of the system clock select register (SSR: $029).
2. Subactive mode is an optional function; specify it on the function option list.

Table 20 Operations in Low-Power Dissipation Modes

Function	Stop Mode	Watch Mode	Standby Mode	Subactive Mode*[2]
CPU	Reset	Retained	Retained	OP
RAM	Retained	Retained	Retained	OP
Timer A	Reset	OP	OP	OP
Timer B	Reset	Stopped	OP	OP
Timer C	Reset	Stopped	OP	OP
Timer D	Reset	Stopped	OP	OP
Serial 1, 2	Reset	Stopped*[3]	OP	OP
A/D	Reset	Stopped	OP	Stopped
I/O	Reset*[1]	Retained	Retained	OP

Note: OP implies in operation

1. Output pins are at high impedance.
2. Subactive mode is an optional function specified on the function option list.
3. Transmission/reception is activated if a clock is input in external clock mode. However, all interrupts stop.

Table 21 I/O Status in Low-Power Dissipation Modes

	Output		Input
	Standby mode, watch mode	**Stop mode**	**Active mode, subactive mode**
D_0–D_{11}	Retained	High impedance	Input enabled
D_{12}–D_{13}	—	—	Input enabled
R0–RC	Retained or output of peripheral functions	High impedance	Input enabled

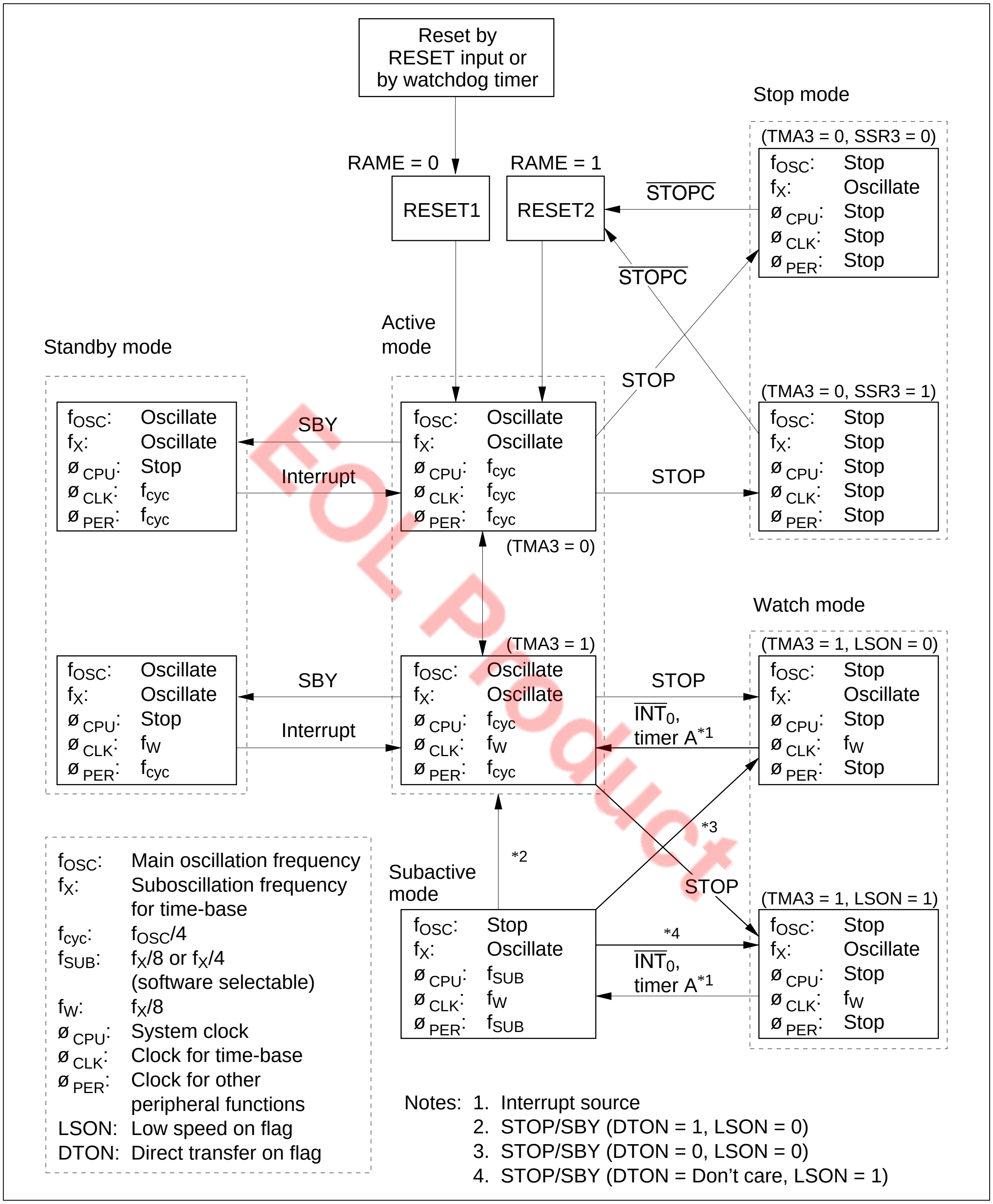


Figure 14 MCU Status Transitions

Standby Mode: In standby mode, the oscillators continue to operate, but the clocks related to instruction execution stop. Therefore, the CPU operation stops, but all RAM and register contents are retained, and the D or R port status, when set to output, is maintained. Peripheral functions such as interrupts, timers, and serial interface continue to operate. The power dissipation in this mode is lower than in active mode because the CPU stops.

The MCU enters standby mode when the SBY instruction is executed in active mode.

Standby mode is terminated by a RESET input or an interrupt request. If it is terminated by RESET input, the MCU is reset as well. After an interrupt request, the MCU enters active mode and executes the next instruction after the SBY instruction. If the interrupt enable flag is 1, the interrupt is then processed; if it is 0, the interrupt request is left pending and normal instruction execution continues. A flowchart of operation in standby mode is shown in figure 15.

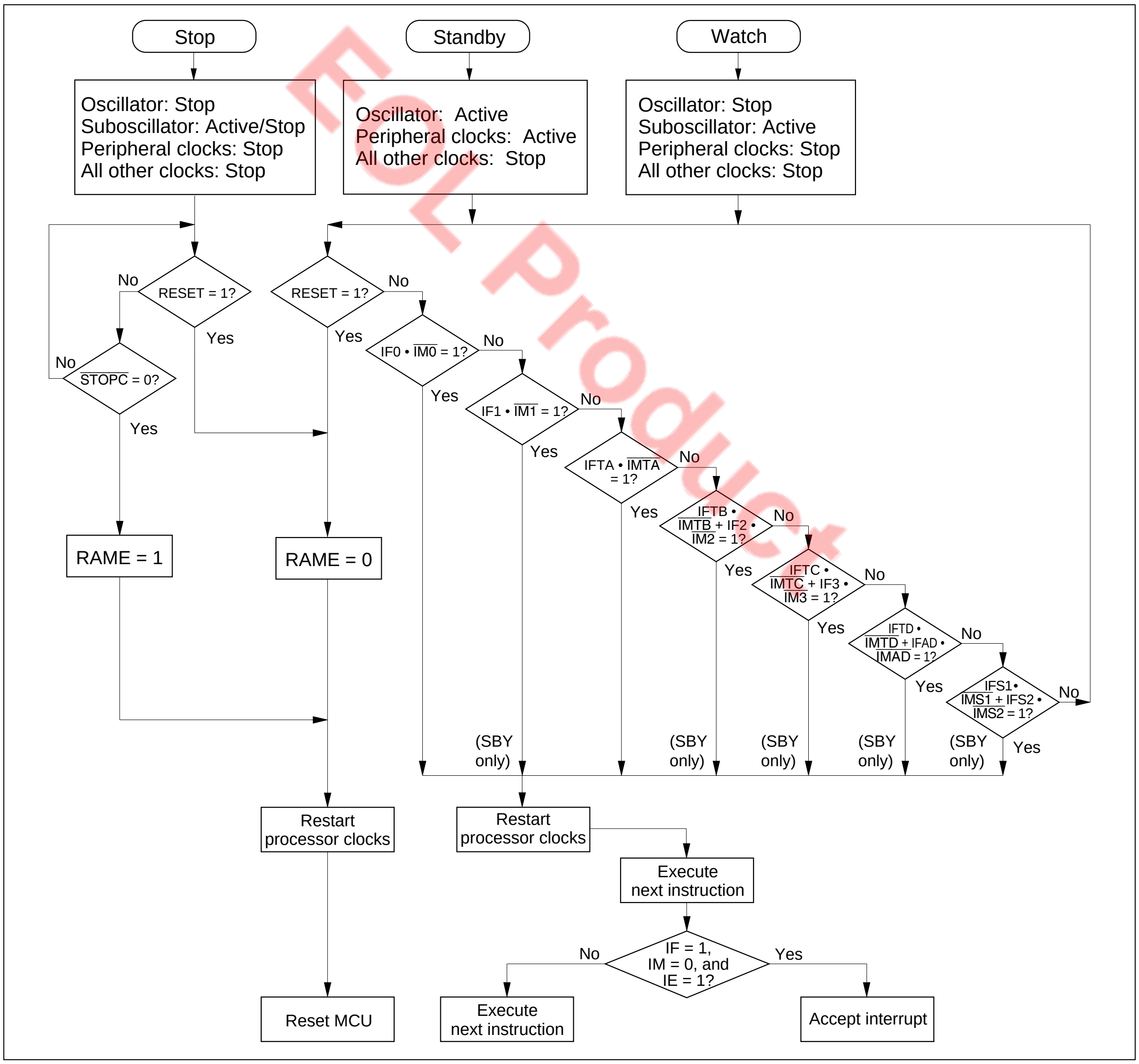


Figure 15 MCU Operation Flowchart

Stop Mode: In stop mode, all MCU operations stop and RAM data is retained. Therefore, the power dissipation in this mode is the least of all modes. The OSC_1 and OSC_2 oscillator stops. Operation of the X1 and X2 oscillator can be selected by setting bit 3 of the system clock select register (SSR: $029; operating: SSR3 = 0, stop: SSR3 = 1) (figure 26). The MCU enters stop mode if the STOP instruction is executed in active mode when bit 3 of timer mode register A (TMA: $008) is set to 0 (TMA3 = 0) (figure 41).

Stop mode is terminated by a RESET input or a $\overline{STOPC}$ input as shown in figure 16. RESET or $\overline{STOPC}$ must be applied for at least one t_{RC} to stabilize oscillation (refer to the AC Characteristics section). When the MCU restarts after stop mode is cancelled, all RAM contents before entering stop mode are retained, but the accuracy of the contents of the accumulator, B register, W register, X/SPX register, Y/SPY register, carry flag, and serial data register cannot be guaranteed.

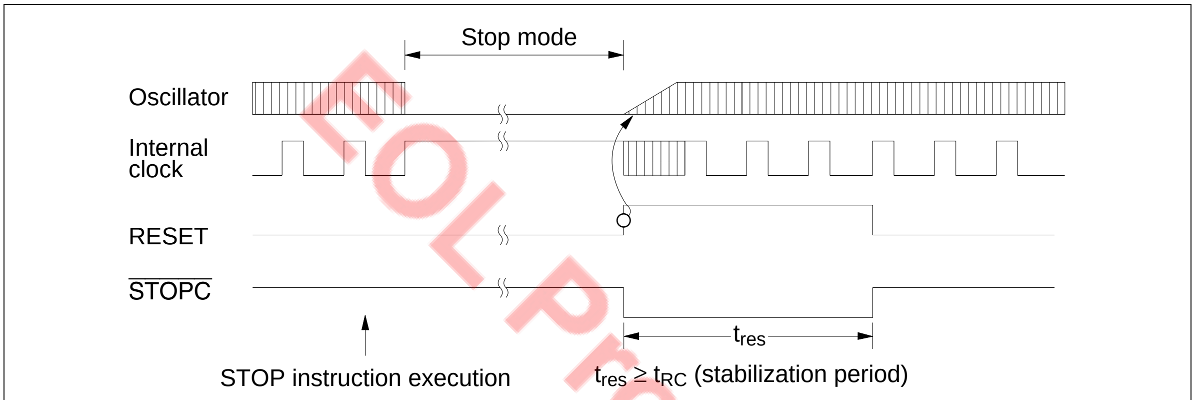


Figure 16 Timing of Stop Mode Cancellation

Watch Mode: In watch mode, the clock function (timer A) using the X1 and X2 oscillator operates but other function operations stop. Therefore, the power dissipation in this mode is the second least to stop mode, and this mode is convenient when only clock display is used. In this mode, the OSC_1 and OSC_2 oscillator stops, but the X1 and X2 oscillator operates. The MCU enters watch mode if the STOP instruction is executed in active mode when TMA3 = 1, or if the STOP or SBY instruction is executed in subactive mode.

Watch mode is terminated by a RESET input or a timer-A/$\overline{INT}_0$ interrupt request. For details of RESET input, refer to the Stop Mode section. When terminated by a timer-A/$\overline{INT}_0$ interrupt request, the MCU enters active mode if LSON is 0, or subactive mode if LSON is 1. After an interrupt request is generated, the time required to enter active mode is t_{RC} for a timer A interrupt, and T_X (where $T + t_{RC} < T_X < 2T + t_{RC}$) for an $\overline{INT}_0$ interrupt, as shown in figure 17.

Operation during mode transition is the same as that at standby mode cancellation (figure 15).

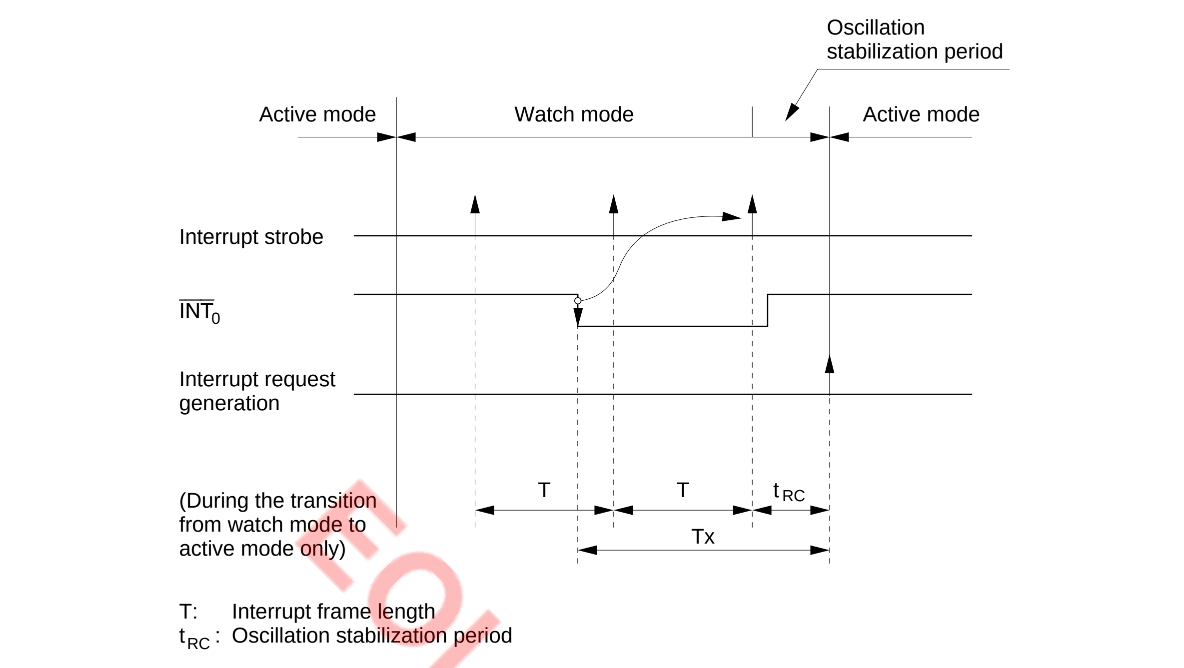


Figure 17 Interrupt Frame

Subactive Mode: The OSC_1 and OSC_2 oscillator stops and the MCU operates with a clock generated by the X1 and X2 oscillator. In this mode, functions other than A/D conversion operate. However, because the operating clock is slow, the power dissipation becomes low, next to watch mode.

The CPU instruction execution speed can be selected as 244 µs or 122 µs by setting bit 2 (SSR2) of the system clock select register (SSR: $029). Note that the SSR2 value must be changed in active mode. If the value is changed in subactive mode, the MCU may malfunction.

When the STOP or SBY instruction is executed in subactive mode, the MCU enters either watch or active mode, depending on the statuses of the low speed on flag (LSON: $020, bit 0) and the direct transfer on flag (DTON: $020, bit 3).

Subactive mode is an optional function that the user must specify on the function option list.

Interrupt Frame: In watch and subactive modes, $ø_{CLK}$ is applied to timer A and the $\overline{INT_0}$ circuit. Prescaler W and timer A operate as the time-base and generate the timing clock for the interrupt frame. Three interrupt frame lengths (T) can be selected by setting the miscellaneous register (MIS: $00C) (figure 18).

In watch and subactive modes, the timer-A/$\overline{INT_0}$ interrupt is generated synchronously with the interrupt frame. The interrupt request is generated synchronously with the interrupt strobe timing except during transition to active mode. The falling edge of the $\overline{INT_0}$ signal is input asynchronously with the interrupt frame timing, but it is regarded as input synchronously with the second interrupt strobe clock after the falling edge. An overflow and interrupt request in timer A is generated synchronously with the interrupt strobe timing.

Miscellaneous register (MIS: $00C)

Bit	3	2	1	0
Initial value	0	0	0	0
Read/Write	W	W	W	W
Bit name	MIS3	MIS2	MIS1	MIS0

MIS3	MIS2
Buffer control. Refer to figure 38.	

MIS1	MIS0	T[*1]	t_{RC}[*1]	Oscillation circuit conditions
0	0	0.24414 ms	0.12207 ms	External clock input
			0.24414 ms[*2]	
0	1	15.625 ms	7.8125 ms	Ceramic oscillator or crystal
1	0	125 ms	62.5 ms	
1	1	Not used		—

Notes: 1. The values of T and t_{RC} are applied when a 32.768-kHz crystal oscillator is used.
2. The value is applied only when direct transfer operation is used.

Figure 18 Miscellaneous Register (MIS)

Direct Transition from Subactive Mode to Active Mode: Available by controlling the direct transfer on flag (DTON: $020, bit 3) and the low speed on flag (LSON: $020, bit 0). The procedures are described below:

- Set LSON to 0 and DTON to 1 in subactive mode.
- Execute the STOP or SBY instruction.
- The MCU automatically enters active mode from subactive mode after waiting for the MCU internal processing time and oscillation stabilization time (Figure 19).

Notes: 1. The DTON flag ($020, bit 3) can be set only in subactive mode. It is always reset in active mode.

2. The transition time (T_D) from subactive mode to active mode:

$t_{RC} < T_D < T + t_{RC}$

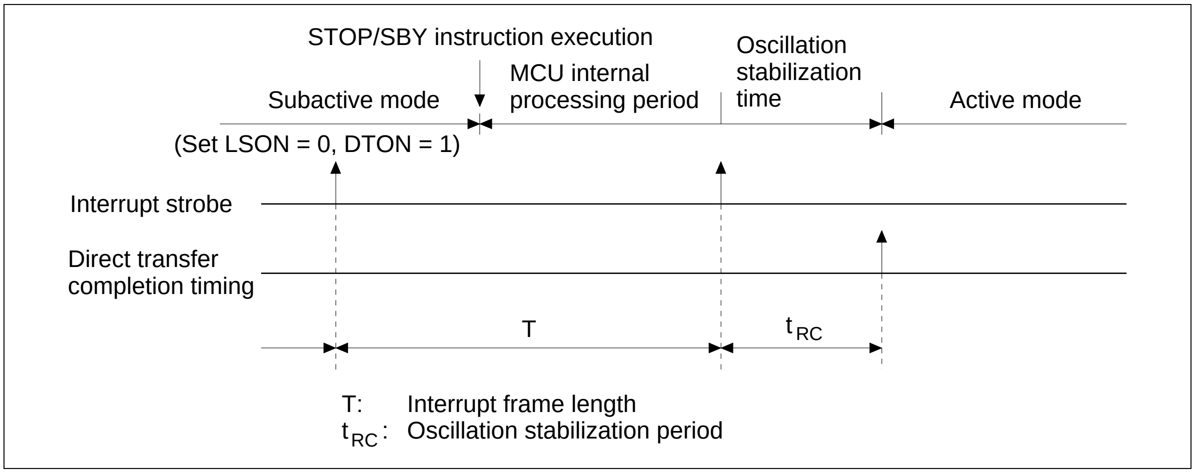


Figure 19 Direct Transition Timing

Stop Mode Cancellation by $\overline{\text{STOPC}}$: The MCU enters active mode from stop mode by a $\overline{\text{STOPC}}$ input as well as by RESET. In either case, the MCU starts instruction execution from the starting address (address 0) of the program. However, the value of the RAM enable flag (RAME: $021, bit 3) differs between cancellation by $\overline{\text{STOPC}}$ and by RESET. When stop mode is cancelled by RESET, RAME = 0; when cancelled by $\overline{\text{STOPC}}$, RAME = 1. RESET can cancel all modes, but $\overline{\text{STOPC}}$ is valid only in stop mode; $\overline{\text{STOPC}}$ input is ignored in other modes. Therefore, when the program requires to confirm that stop mode has been cancelled by $\overline{\text{STOPC}}$ (for example, when the RAM contents before entering stop mode are used after transition to active mode), execute the TEST instruction on the RAM enable flag (RAME) at the beginning of the program.

MCU Operation Sequence: The MCU operates in the sequences shown in figures 20 to 22. It is reset by an asynchronous RESET input, regardless of its status.

The low-power mode operation sequence is shown in figure 22. With the IE flag cleared and an interrupt flag set together with its interrupt mask cleared, if a STOP/SBY instruction is executed, the instruction is cancelled (regarded as an NOP) and the following instruction is executed. Before executing a STOP/SBY instruction, make sure all interrupt flags are cleared or all interrupts are masked.

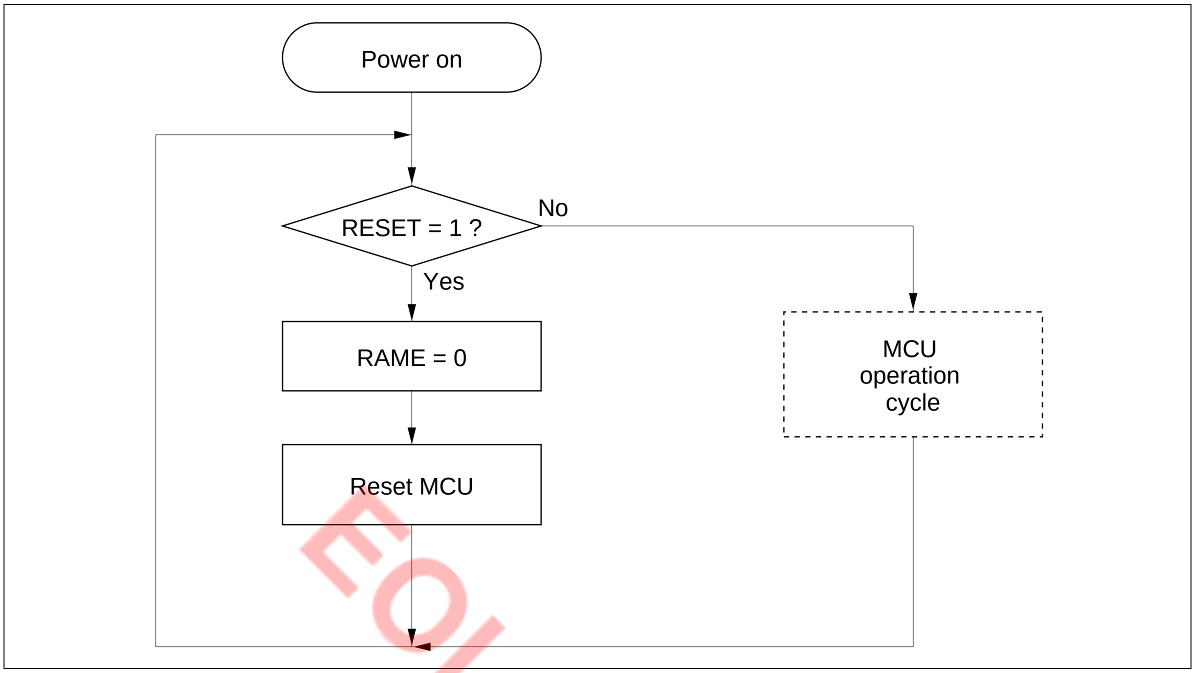
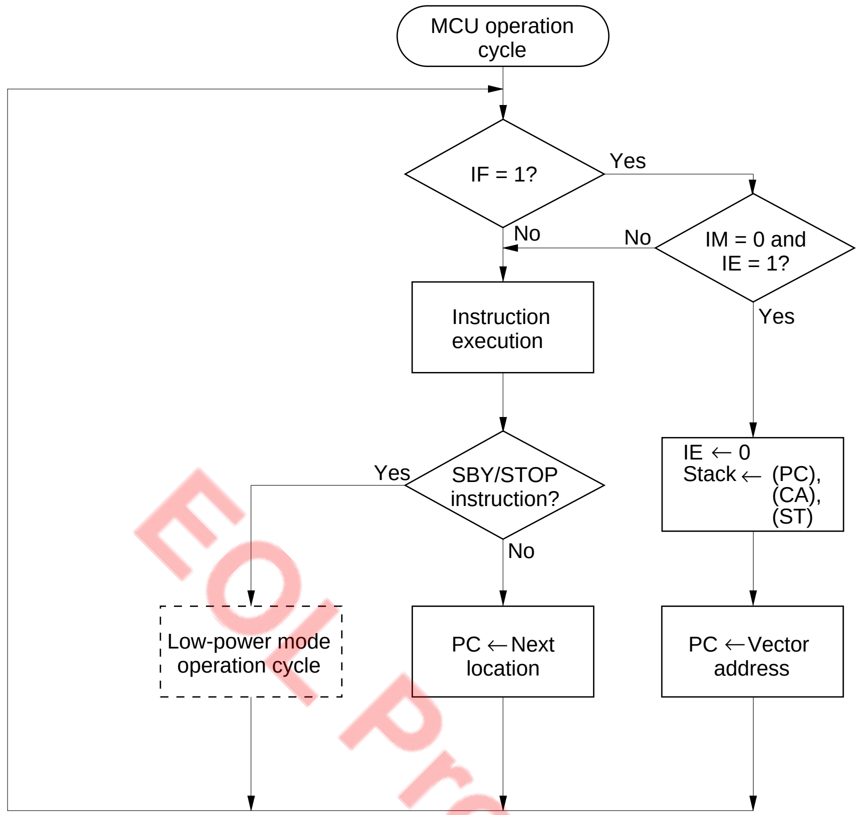


Figure 20 MCU Operating Sequence (Power On)



IF: Interrupt request flag
IM: Interrupt mask
IE: Interrupt enable flag
PC: Program counter
CA: Carry flag
ST: Status flag

Figure 21 MCU Operating Sequence (MCU Operation Cycle)

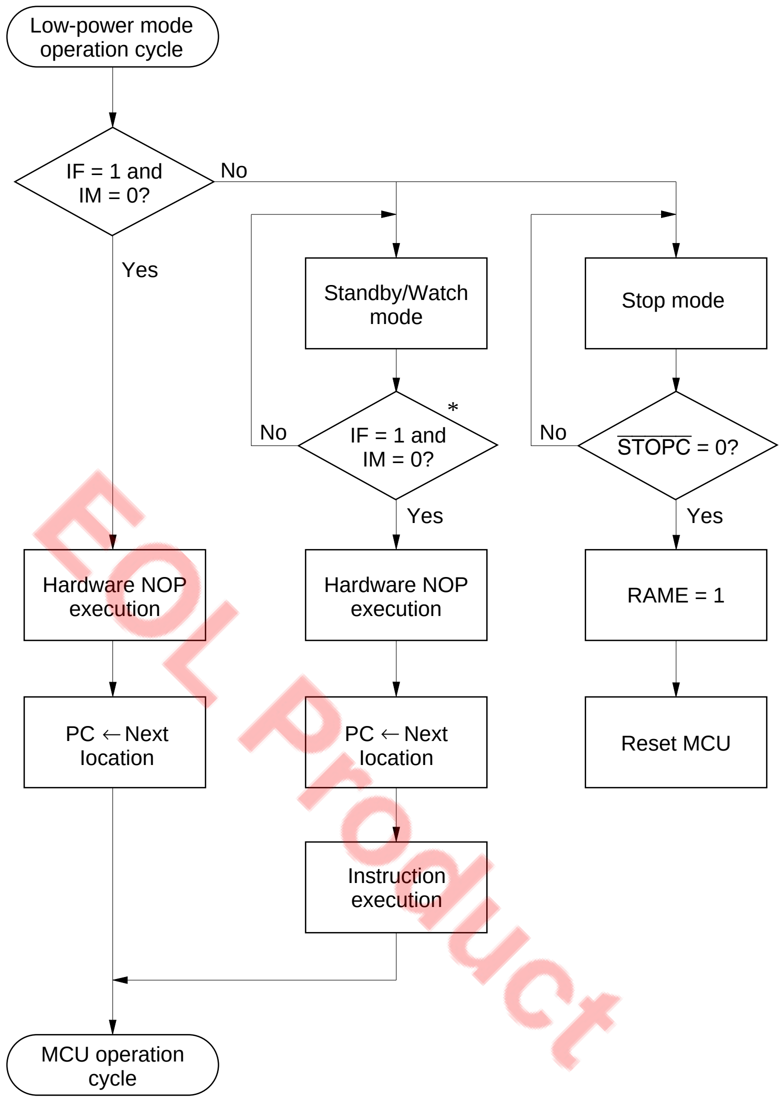


Figure 22 MCU Operating Sequence (Low-Power Mode Operation)

Note: When the MCU is in watch mode or subactive mode, if the high level period before the falling edge of $\overline{INT}_0$ is shorter than the interrupt frame, $\overline{INT}_0$ is not detected. Also, if the low level period after the falling edge of $\overline{INT}_0$ is shorter than the interrupt frame, $\overline{INT}_0$ is not detected.
Edge detection is shown in figure 23. The level of the $\overline{INT}_0$ signal is sampled by a sampling clock. When this sampled value changes to low from high, a falling edge is detected.
In figure 24, the level of the $\overline{INT}_0$ signal is sampled by an interrupt frame. In (a) the sampled value is low at point A, and also low at point B. Therefore, a falling edge is not detected. In (b), the sampled value is high at point A, and also high at point B. A falling edge is not detected in this case either.

When the MCU is in watch mode or subactive mode, keep the high level and low level period of $\overline{INT}_0$ longer than interrupt frame.

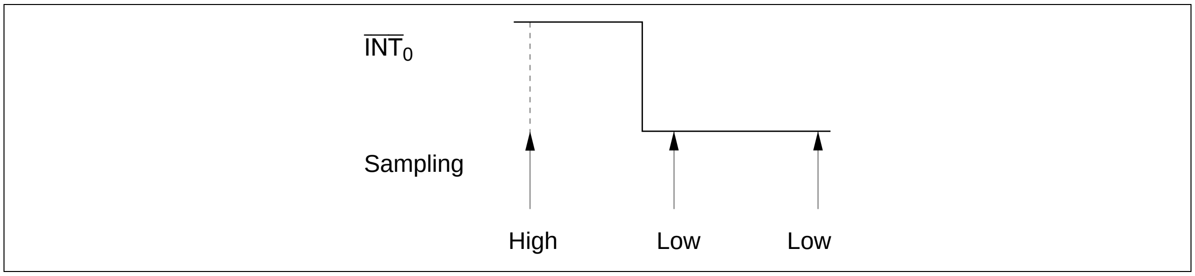


Figure 23 Edge Detection

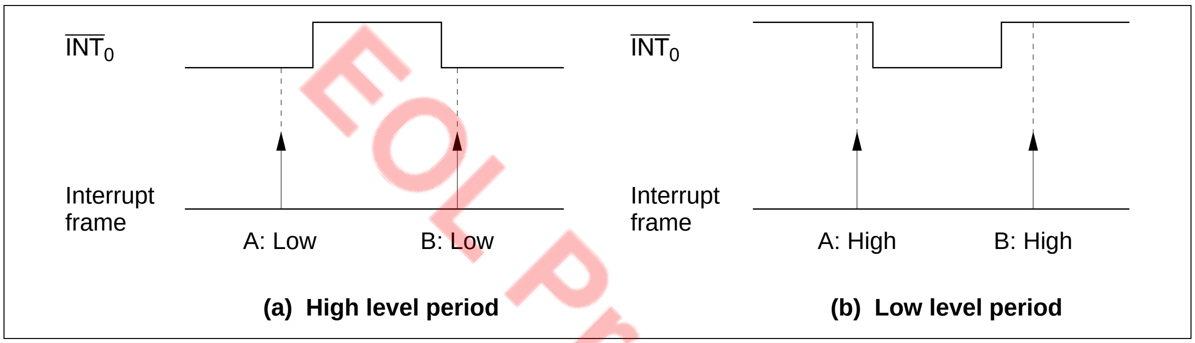


Figure 24 Sampling Example

Internal Oscillator Circuit

A block diagram of the clock generation circuit is shown in figure 25. As shown in table 22, a ceramic oscillator or crystal oscillator can be connected to OSC_1 and OSC_2, and a 32.768-kHz oscillator can be connected to X1 and X2. The system oscillator can also be operated by an external clock. Bit 1 (SSR1) of the system clock select register (SSR: $029) must be selected according to the frequency of the oscillator connected to OSC_1 and OSC_2 (figure 26).

Note: If the system clock select register (SSR: $029) setting does not match the oscillator frequency, subsystems using the 32.768-kHz oscillation will malfunction.

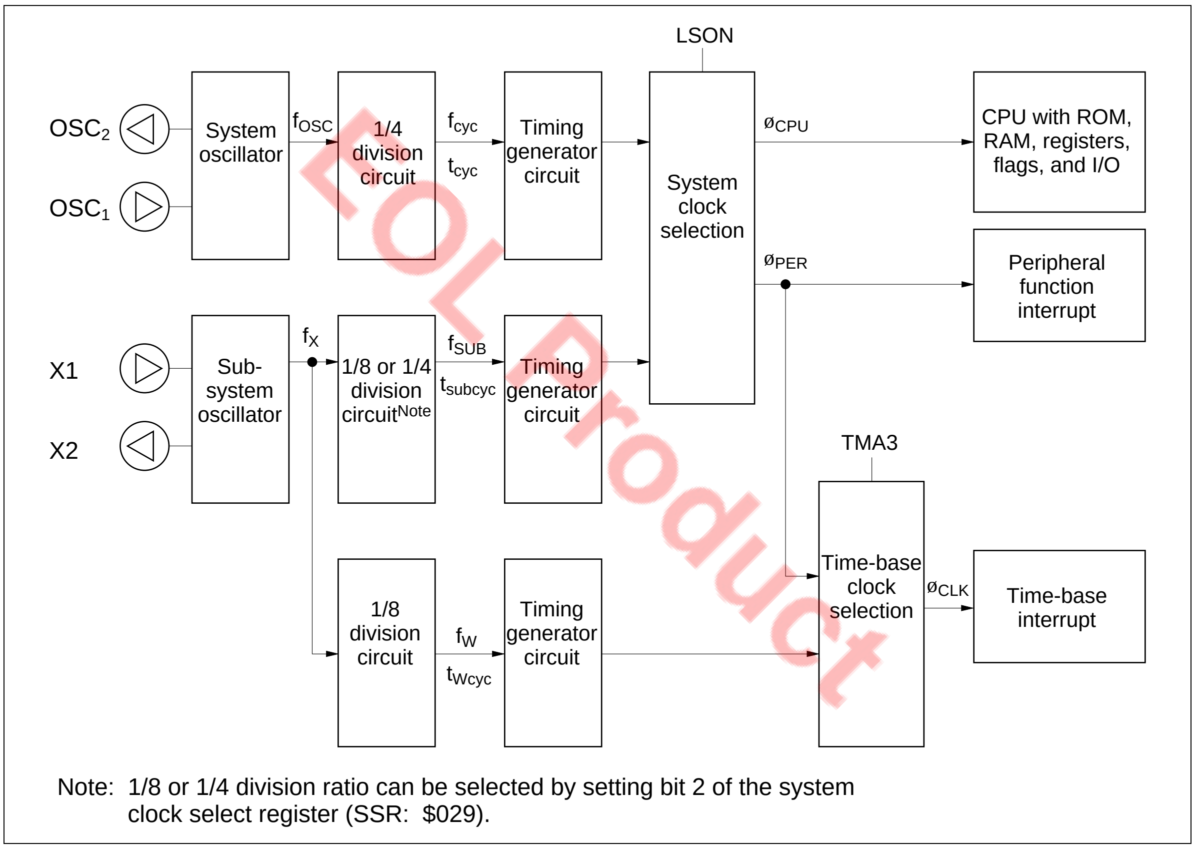


Figure 25 Clock Generation Circuit

System clock select register (SSR: $029)

Bit	3	2	1	0
Initial value	0	0	0	—
Read/Write	W	W	W	—
Bit name	SSR3	SSR2	SSR1	Not used

SSR2	32-kHz oscillation division ratio selection
0	$f_{SUB} = f_X/8$
1	$f_{SUB} = f_X/4$

SSR1	System clock selection
0	0.4 to 1.0 MHz
1	1.6 to 4.0 MHz

SSR3	32-kHz oscillation stop
0	Oscillation operates in stop mode
1	Oscillation stops in stop mode

Figure 26 System Clock Select Register

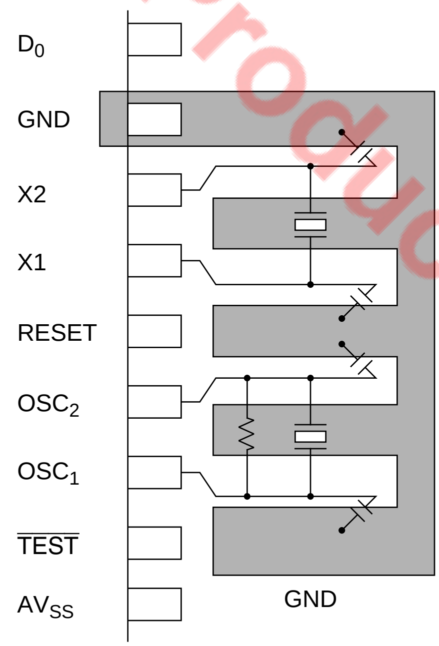
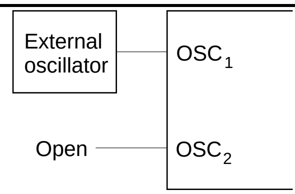
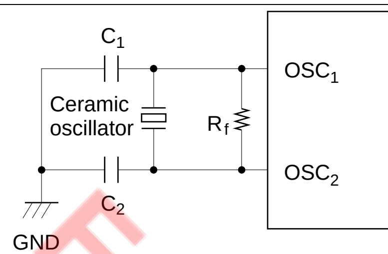
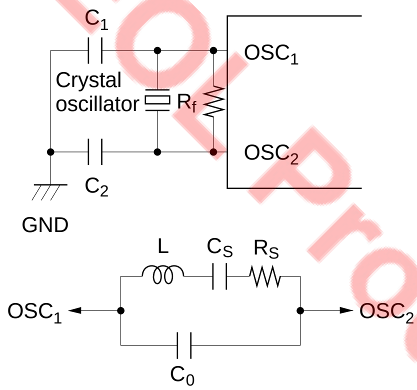
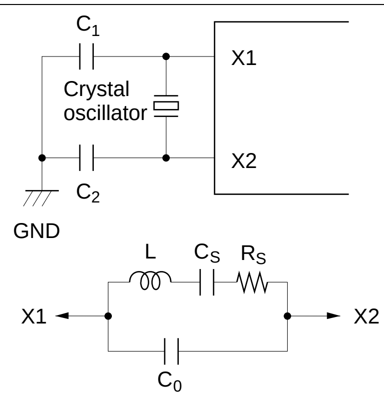


Figure 27 Typical Layouts of Crystal and Ceramic Oscillator

Table 22 Oscillator Circuit Examples

	Circuit Configuration	Circuit Constants
External clock operation		
Ceramic oscillator (OSC$_1$, OSC$_2$)		Ceramic oscillator: CSA4.00MG (Murata) R_f = 1 MΩ ± 20% $C_1 = C_2$ = 30 pF ± 20%
Crystal oscillator (OSC$_1$, OSC$_2$)		R_f = 1 MΩ ± 20% $C_1 = C_2$ = 10–22 pF ± 20% Crystal: Equivalent to circuit shown below C_0 = 7 pF max R_S = 100 Ω max
Crystal oscillator (X1, X2)		Ceramic: 32.768 kHz: MX38T (Nippon Denpa Kogyo) $C_1 = C_2$ = 20 pF ± 20% R_S: 14 kΩ C_0: 1.5 pF

Notes: 1. Since the circuit constants change depending on the crystal or ceramic resonator and stray capacitance of the board, the user should consult with the crystal or ceramic oscillator manufacturer to determine the circuit parameters.

2. Wiring among OSC$_1$, OSC$_2$, X1, X2, and elements should be as short as possible, and must not cross other wiring (see figure 27).

3. If the 32.768-kHz crystal oscillator is not used, the X1 pin must be fixed to GND and X2 must be open.

Input/Output

The MCU has 64 input/output pins (D_0–D_{11}, $R0_0$–RC_3) and 2 input pins (D_{12}, D_{13}). The features are described below.

- 10 pins (D_0–D_9) are high-current input/output pins.
- The D_{12}, D_{13}, $R0_0$–$R0_2$, and $R3_0$–$R5_3$ input/output pins are multiplexed with peripheral function pins such as for the timers or serial interface. For these pins, the peripheral function setting is done prior to the D or R port setting. Therefore, when a peripheral function is selected for a pin, the pin function and input/output selection are automatically switched according to the setting.
- Input or output selection for input/output pins and port or peripheral function selection for multiplexed pins are set by software.
- Peripheral function output pins are CMOS output pins. Only the $R4_3/SO_1$ and $R5_3/SO_2$ pins can be set to NMOS open-drain output by software.
- In stop mode, the MCU is reset, and therefore peripheral function selection is cancelled. Input/output pins are in high-impedance state.
- Each input/output pin has a built-in pull-up MOS, which can be individually turned on or off by software.

I/O buffer configuration is shown in figure 28, programmable I/O circuits are listed in table 23, and I/O pin circuit types are shown in table 24.

Table 23 Programmable I/O Circuits

MIS3 (Bit 3 of MIS)		**0**				**1**			
DCD, DCR		**0**		**1**		**0**		**1**	
PDR		**0**	**1**	**0**	**1**	**0**	**1**	**0**	**1**
CMOS buffer	PMOS	—	—	—	On	—	—	—	On
	NMOS	—	—	On	—	—	—	On	—
Pull-up MOS		—	—	—	—	—	On	—	On

Note: — indicates off status.

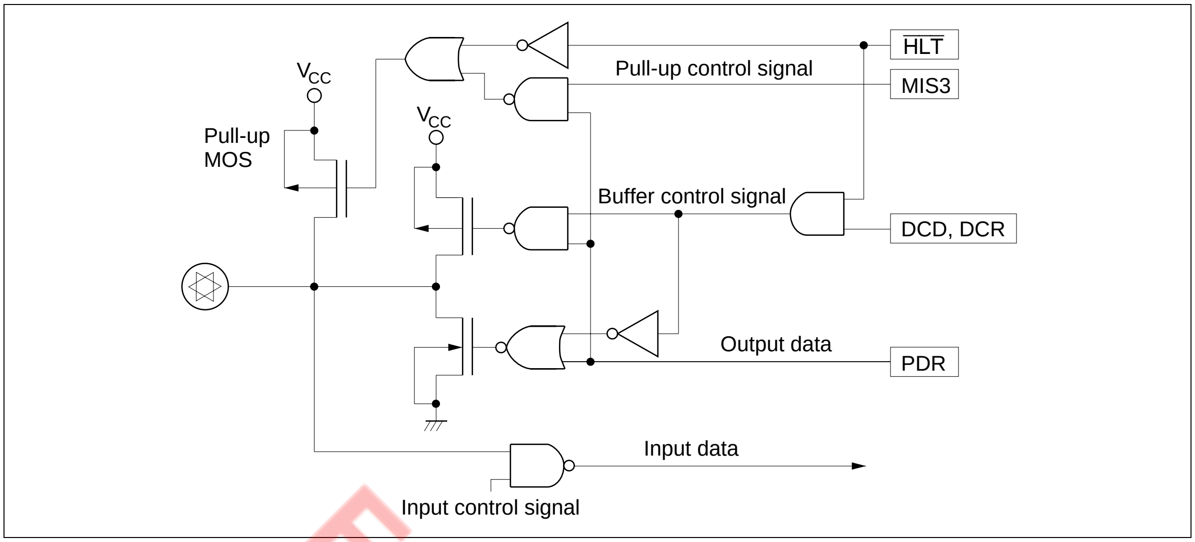
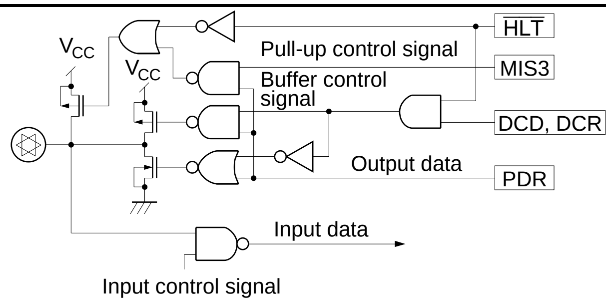
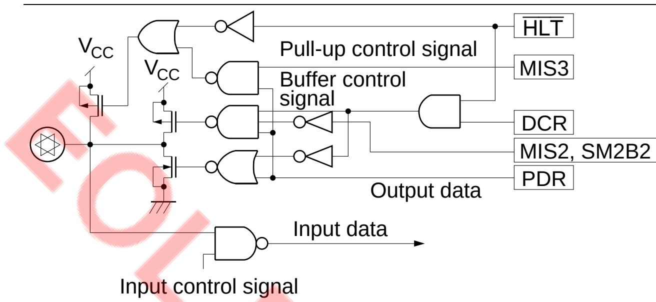
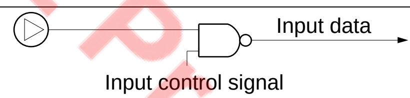
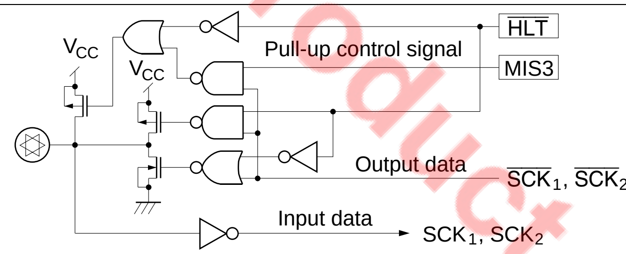
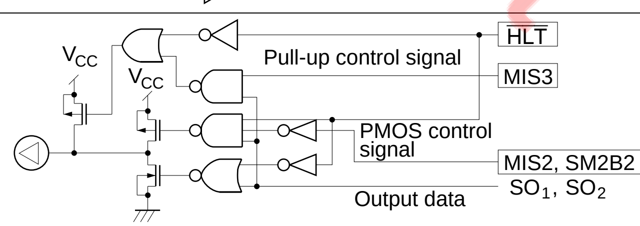
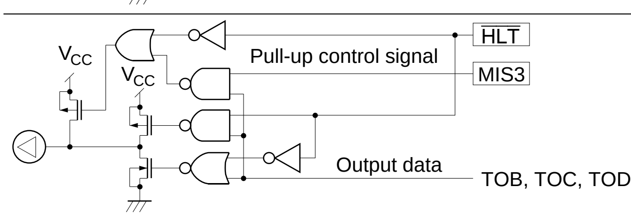
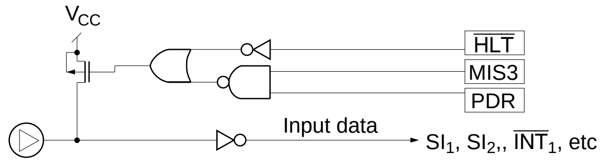


Figure 28 I/O Buffer Configuration

Table 24 Circuit Configurations of I/O Pins

I/O Pin Type		Circuit	Pins
Input/output pins			D_0–D_{11}, $R0_0$–$R0_3$, $R1_0$–$R1_3$, $R2_0$–$R2_3$, $R3_0$–$R3_3$, $R4_0$–$R4_2$, $R5_0$–$R5_2$, $R6_0$–$R6_3$, $R7_0$–$R7_3$, $R8_0$–$R8_3$, $R9_0$–$R9_3$, RA_0–RA_3, RB_0–RB_3, RC_0–RC_3
			$R4_3$, $R5_3$
Input pins			D_{12}, D_{13}
Peripheral function pins	Input/ output pins		$\overline{SCK_1}$, $\overline{SCK_2}$
	Output pins		SO_1, SO_2
			TOB, TOC, TOD

I/O Pin Type	Circuit	Pins
Input pins		SI_1, SI_2, $\overline{INT}_1$, INT_2, INT_3, $\overline{EVNB}$, EVND
		$\overline{INT}_0$, $\overline{STOPC}$

Notes: 1. The MCU is reset in stop mode, and peripheral function selection is cancelled. The $\overline{HLT}$ signal becomes low, and input/output pins enter high-impedance state.

2. The $\overline{HLT}$ signal is 1 in watch and subactive modes.

D Port (D_0–D_{13}): Consist of 12 input/output pins and 2 input pins addressed by one bit. D_0–D_{11} are high-current I/O pins, and D_{12} and D_{13} are input-only pins.

Pins D_0–D_{11} are set by the SED and SEDD instructions, and reset by the RED and REDD instructions. Output data is stored in the port data register (PDR) for each pin. All pins D_0–D_{13} are tested by the TD and TDD instructions.

The on/off statuses of the output buffers are controlled by D-port data control registers (DCD0–DCD2: $02C–$02E) that are mapped to memory addresses (figure 29).

Pins D_{12} and D_{13} are multiplexed with peripheral function pins $\overline{STOPC}$ and $\overline{INT}_0$, respectively. The peripheral function modes of these pins are selected by bits 2 and 3 (PMRC2, PMRC3) of port mode register C (PMRC: $025) (figure 30).

R Ports ($R0_0$–RC_3): 52 input/output pins addressed in 4-bit units. Data is input to these ports by the LAR and LBR instructions, and output from them by the LRA and LRB instructions. Output data is stored in the port data register (PDR) for each pin. The on/off statuses of the output buffers of the R ports are controlled by R-port data control registers (DCR0–DCRC: $030–$03C) that are mapped to memory addresses (figure 29).

Pins $R0_0$–$R0_2$ are multiplexed with peripheral pins $\overline{INT}_1$–INT_3, respectively. The peripheral function modes of these pins are selected by bits 0–2 (PMRB0–PMRB2) of port mode register B (PMRB: $024) (figure 31).

Pins $R3_0$–$R3_2$ are multiplexed with peripheral pins TOB, TOC, and TOD, respectively. The peripheral function modes of these pins are selected by bits 0 and 1 (TMB20, TMB21) of timer mode register B2 (TMB2: $013), bits 0–2 (TMC20–TMC22) of timer mode register C2 (TMC2: $014), and bits 0–3 (TMD20–TMD23) of timer mode register D2 (TMD2: $015) (figures 32, 33, and 34).

Pins $R3_3$ and $R4_0$ are multiplexed with peripheral pins $\overline{EVNB}$ and EVND, respectively. The peripheral function modes of these pins are selected by bits 0 and 1 (PMRC0, PMRC1) of port mode register C (PMRC: $025) (figure 30).

Pins $R4_1$–$R4_3$ are multiplexed with peripheral pins $\overline{SCK}_1$, SI_1, and SO_1, respectively. The peripheral function modes of these pins are selected by bit 3 (SM1A3) of serial mode register 1A (SM1A: $005), and bits 0 and 1 (PMRA0, PMRA1) of port mode register A (PMRA: $004), as shown in figures 35 and 36.

Ports $R5_1$–$R5_3$ are multiplexed with peripheral function pins $\overline{SCK}_2$, SI_2, SO_2, respectively. The function modes of these pins can be selected by individual pins, by 2A setting bit 3 (SM2A3) of serial mode register 2A (SM2A: \$01B), and bits 2 and 3 (PMRA2, PMRA3) of port mode register A (PMRA: \$004) (figures 36 and 37).

Pull-Up MOS Transistor Control: A program-controlled pull-up MOS transistor is provided for each input/output pin other than input-only pins D_{12} and D_{13}. The on/off status of all these transistors is controlled by bit 3 (MIS3) of the miscellaneous register (MIS: \$00C), and the on/off status of an individual transistor can also be controlled by the port data register (PDR) of the corresponding pin—enabling on/off control of that pin alone (table 23 and figure 38).

The on/off status of each transistor and the peripheral function mode of each pin can be set independently.

How to Deal with Unused I/O Pins: I/O pins that are not needed by the user system (floating) must be connected to V_{CC} to prevent LSI malfunctions due to noise. These pins must either be pulled up to V_{CC} by their pull-up MOS transistors or by resistors of about 100 kΩ.

Data control register **(DCD0 to 2: $02C to $02E)**
(DCR0 to C: $030 to $03C)

DCD0, DCD1

Bit	3	2	1	0
Initial value	0	0	0	0
Read/Write	W	W	W	W
Bit name	DCD03, DCD13	DCD02, DCD12	DCD01, DCD11	DCD00, DCD10

DCD2

Bit	3	2	1	0
Initial value	0	0	0	0
Read/Write	W	W	W	W
Bit name	DCD23	DCD22	DCD21	DCD20

DCR0 to DCRC

Bit	3	2	1	0
Initial value	0	0	0	0
Read/Write	W	W	W	W
Bit name	DCR03–DCRC3	DCR02–DCRC2	DCR01–DCRC1	DCR00–DCRC0

All Bits	CMOS Buffer On/Off Selection
0	Off (high-impedance)
1	On

Correspondence between ports and DCD/DCR bits

Register Name	Bit 3	Bit 2	Bit 1	Bit 0
DCD0	D_3	D_2	D_1	D_0
DCD1	D_7	D_6	D_5	D_4
DCD2	D_{11}	D_{10}	D_9	D_8
DCR0	$R0_3$	$R0_2$	$R0_1$	$R0_0$
DCR1	$R1_3$	$R1_2$	$R1_1$	$R1_0$
DCR2	$R2_3$	$R2_2$	$R2_1$	$R2_0$
DCR3	$R3_3$	$R3_2$	$R3_1$	$R3_0$
DCR4	$R4_3$	$R4_2$	$R4_1$	$R4_0$
DCR5	$R5_3$	$R5_2$	$R5_1$	$R5_0$
DCR6	$R6_3$	$R6_2$	$R6_1$	$R6_0$
DCR7	$R7_3$	$R7_2$	$R7_1$	$R7_0$
DCR8	$R8_3$	$R8_2$	$R8_1$	$R8_0$
DCR9	$R9_3$	$R9_2$	$R9_1$	$R9_0$
DCRA	RA_3	RA_2	RA_1	RA_0
DCRB	RB_3	RB_2	RB_1	RB_0
DCRC	RC_3	RC_2	RC_1	RC_0

Figure 29 Data Control Registers (DCD, DCR)

Port mode register C (PMRC: $025)

Bit	3	2	1	0
Initial value	0	0	0	0
Read/Write	W	W	W	W
Bit name	PMRC3	PMRC2*	PMRC1	PMRC0

PMRC0	$R3_3/\overline{EVNB}$ mode selection
0	$R3_3$
1	$\overline{EVNB}$

PMRC1	$R4_0$/EVND mode selection
0	$R4_0$
1	EVND

PMRC2	$D_{12}/\overline{STOPC}$ mode selection
0	D_{12}
1	$\overline{STOPC}$

PMRC3	$D_{13}/\overline{INT_0}$ mode selection
0	D_{13}
1	$\overline{INT_0}$

Note: *PMRC2 is reset to 0 only by RESET input. When $\overline{STOPC}$ is input in stop mode, PMRC2 is not reset but retains its value.

Figure 30 Port Mode Register C (PMRC)

Port mode register B (PMRB: $024)

Bit	3	2	1	0
Initial value	—	0	0	0
Read/Write	—	W	W	W
Bit name	Not used	PMRB2	PMRB1	PMRB0

PMRB0	$R0_0/\overline{INT_1}$ mode selection
0	$R0_0$
1	$\overline{INT_1}$

PMRB1	$R0_1/INT_2$ mode selection
0	$R0_1$
1	INT_2

PMRB2	$R0_2/INT_3$ mode selection
0	$R0_2$
1	INT_3

Figure 31 Port Mode Register B (PMRB)

Timer mode register B2 (TMB2: $013)

Bit	3	2	1	0
Initial value	—	—	0	0
Read/Write	—	—	R/W	R/W
Bit name	Not used	Not used	TMB21	TMB20

TMB21	TMB20	$R3_0$/TOB mode selection	
0	0	$R3_0$	$R3_0$ port
	1	TOB	Toggle output
1	0	TOB	0 output
	1	TOB	1 output

Figure 32 Timer Mode Register B2 (TMB2)

Timer mode register C2 (TMC2: $014)

Bit	3	2	1	0
Initial value	—	0	0	0
Read/Write	—	R/W	R/W	R/W
Bit name	Not used	TMC22	TMC21	TMC20

TMC22	TMC21	TMC20	$R3_1$/TOC mode selection	
0	0	0	$R3_1$	$R3_1$ port
		1	TOC	Toggle output
	1	0	TOC	0 output
		1	TOC	1 output
1	0	0	—	Inhibited
		1		
	1	0		
		1	TOC	PWM output

Figure 33 Timer Mode Register C2 (TMC2)

Timer mode register D2 (TMD2: $015)

Bit	3	2	1	0
Initial value	0	0	0	0
Read/Write	R/W	R/W	R/W	R/W
Bit name	TMD23	TMD22	TMD21	TMD20

TMD23	TMD22	TMD21	TMD20	$R3_2$/TOD mode selection	
0	0	0	0	$R3_2$	$R3_2$ port
			1	TOD	Toggle output
		1	0	TOD	0 output
			1	TOD	1 output
	1	0	0	—	Inhibited
			1		
		1	0		
			1	TOD	PWM output
1	Don't care	Don't care	Don't care	$R3_2$	Input capture ($R3_2$ port)

Figure 34 Timer Mode Register D2 (TMD2)

Serial mode register 1A (SM1A: $005)

Bit	3	2	1	0
Initial value	0	0	0	0
Read/Write	W	W	W	W
Bit name	SM1A3	SM1A2	SM1A1	SM1A0

SM1A3	$R4_1/\overline{SCK_1}$ mode selection
0	$R4_1$
1	$\overline{SCK_1}$

SM1A2	SM1A1	SM1A0	$\overline{SCK_1}$	Clock source	Prescaler division ratio
0	0	0	Output	Prescaler	÷2048
		1	Output	Prescaler	÷512
	1	0	Output	Prescaler	÷128
		1	Output	Prescaler	÷32
1	0	0	Output	Prescaler	÷8
		1	Output	Prescaler	÷2
	1	0	Output	System clock	—
		1	Input	External clock	—

Figure 35 Serial Mode Register 1A (SM1A)

Port mode register A (PMRA: $004)

Bit	3	2	1	0
Initial value	0	0	0	0
Read/Write	W	W	W	W
Bit name	PMRA3	PMRA2	PMRA1	PMRA0

PMRA0	$R4_3/SO_1$ mode selection
0	$R4_3$
1	SO_1

PMRA1	$R4_2/SI_1$ mode selection
0	$R4_2$
1	SI_1

PMRA2	$R5_3/SO_2$ mode selection
0	$R5_3$
1	SO_2

PMRA3	$R5_2/SI_2$ mode selection
0	$R5_2$
1	SI_2

Figure 36 Port Mode Register A (PMRA)

Serial mode register 2A (SM2A: $01B)

Bit	3	2	1	0
Initial value	0	0	0	0
Read/Write	W	W	W	W
Bit name	SM2A3	SM2A2	SM2A1	SM2A0

SM2A3	$R5_1/\overline{SCK_2}$ mode selection
0	$R5_1$
1	$\overline{SCK_2}$

SM2A2	SM2A1	SM2A0	$\overline{SCK_2}$	Clock source	Prescaler division ratio
0	0	0	Output	Prescaler	÷2048
		1	Output	Prescaler	÷512
	1	0	Output	Prescaler	÷128
		1	Output	Prescaler	÷32
1	0	0	Output	Prescaler	÷8
		1	Output	Prescaler	÷2
	1	0	Output	System clock	—
		1	Input	External clock	—

Figure 37 Serial Mode Register 2A (SM2A)

Miscellaneous register (MIS: $00C)

Bit	3	2	1	0
Initial value	0	0	0	0
Read/Write	W	W	W	W
Bit name	MIS3	MIS2	MIS1	MIS0

MIS3	Pull-up MOS on/off selection
0	Off
1	On

MIS2	CMOS buffer on/off selection for pin $R4_3/SO_1$
0	On
1	Off

MIS1	MIS0
t_{RC} selection. Refer to figure 18 in the operation modes section.	

Figure 38 Miscellaneous Register (MIS)

RENESAS

Prescalers

The MCU has the following two prescalers, S and W.

The prescaler operating conditions are listed in table 25, and the prescaler output supply is shown in figure 39. The timer A–D input clocks except external events and the serial transmit clock except the external clock are selected from the prescaler outputs, depending on corresponding mode registers.

Prescaler Operation

Prescaler S: 11-bit counter that inputs a system clock signal. After being reset to $000 by MCU reset, prescaler S divides the system clock. Prescaler S keeps counting, except in stop, watch, and subactive modes and at MCU reset.

Prescaler W: Five-bit counter that inputs the X1 input clock signal (32-kHz crystal oscillation) divided by eight. After being reset to $00 by MCU reset, prescaler W divides the input clock. Prescaler W can be reset by software.

Table 25 Prescaler Operating Conditions

Prescaler	Input Clock	Reset Conditions	Stop Conditions
Prescaler S	System clock (in active and standby mode), Subsystem clock (in subactive mode)	MCU reset	MCU reset, stop mode, watch mode
Prescaler W	32-kHz crystal oscillation	Software	MCU reset, stop mode

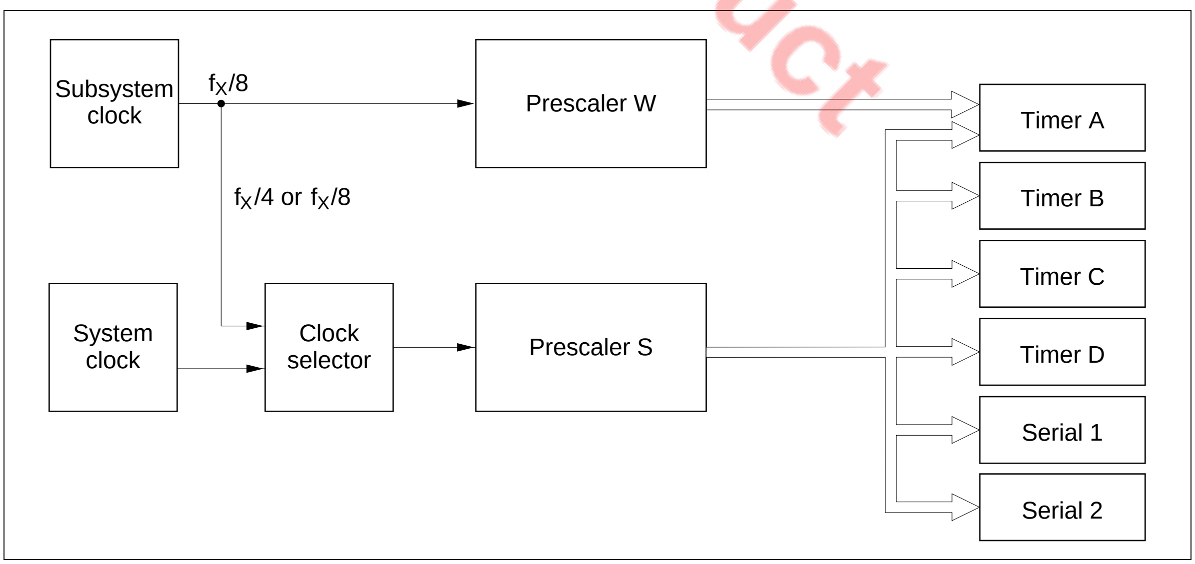


Figure 39 Prescaler Output Supply

Timers

The MCU has four timer/counters (A to D).

- Timer A: Free-running timer
- Timer B: Multifunction timer
- Timer C: Multifunction timer
- Timer D: Multifunction timer

Timer A is an 8-bit free-running timer. Timers B–D are 8-bit multifunction timers, whose functions are listed in table 26. The operating modes are selected by software.

Table 26 Timer Functions

Functions		Timer A	Timer B	Timer C	Timer D
Clock source	Prescaler S	Available	Available	Available	Available
	Prescaler W	Available	—	—	—
	External event	—	Available	—	Available
Timer functions	Free-running	Available	Available	Available	Available
	Time-base	Available	—	—	—
	Event counter	—	Available	—	Available
	Reload	—	Available	Available	Available
	Watchdog	—	—	Available	—
	Input capture	—	—	—	Available
Timer outputs	Toggle	—	Available	Available	Available
	0 output	—	Available	Available	Available
	1 output	—	Available	Available	Available
	PWM	—	—	Available	Available

Note: — means not available.

Timer A

Timer A Functions: Timer A has the following functions.

- Free-running timer
- Clock time-base

The block diagram of timer A is shown in figure 40.

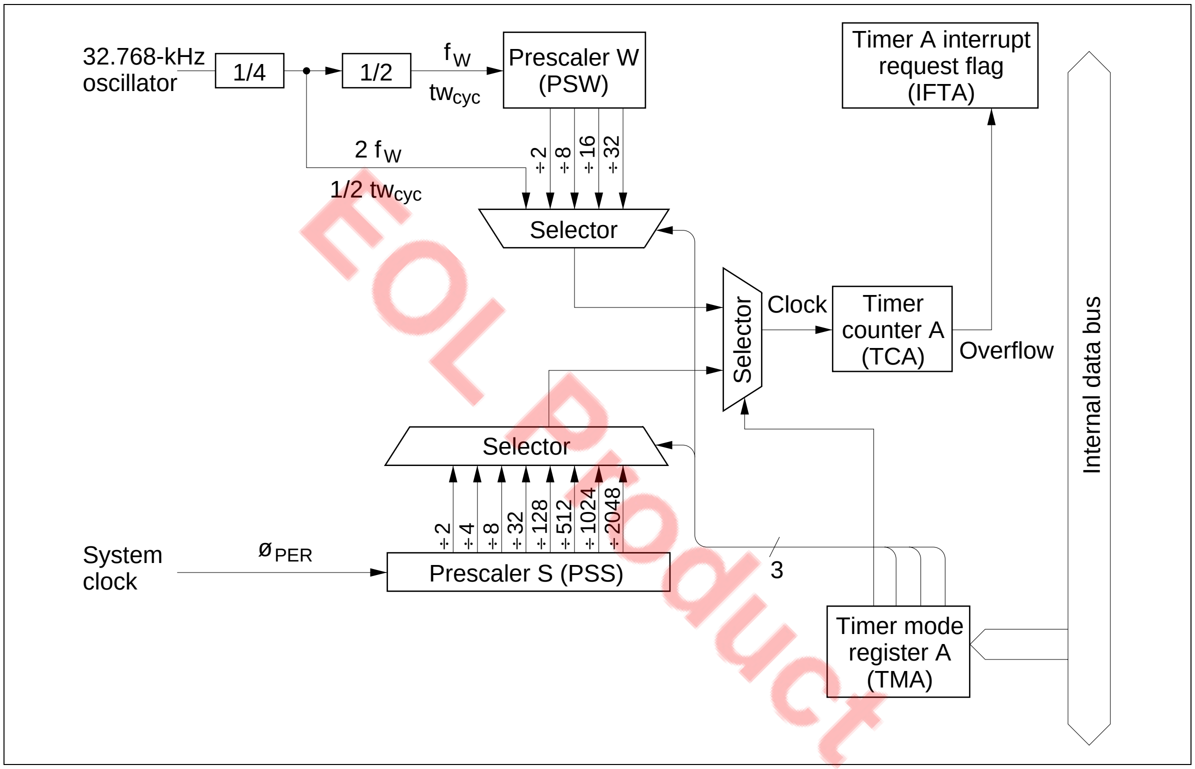


Figure 40 Timer A Block Diagram

Timer A Operations:

- Free-running timer operation: The input clock for timer A is selected by timer mode register A (TMA: $008).

  Timer A is reset to $00 by MCU reset and incremented at each input clock. If an input clock is applied to timer A after it has reached $FF, an overflow is generated, and timer A is reset to $00. The overflow sets the timer A interrupt request flag (IFTA: $001, bit 2). Timer A continues to be incremented after reset to $00, and therefore it generates regular interrupts every 256 clocks.

- Clock time-base operation: Timer A is used as a clock time-base by setting bit 3 (TMA3) of timer mode register A (TMA: $008) to 1. The prescaler W output is applied to timer A, and timer A generates interrupts at the correct timing based on the 32.768-kHz crystal oscillation. In this case, prescaler W and timer A can be reset to $00 by software.

Registers for Timer A Operation: Timer A operating modes are set by the following registers.

- Timer mode register A (TMA: $008): Four-bit write-only register that selects timer A's operating mode and input clock source as shown in figure 41.

Timer mode register A (TMA: $008)

Bit	3	2	1	0
Initial value	0	0	0	0
Read/Write	W	W	W	W
Bit name	TMA3	TMA2	TMA1	TMA0

TMA3	TMA2	TMA1	TMA0	Source prescaler	Input clock frequency	Operating mode
0	0	0	0	PSS	$2048t_{cyc}$	Timer A mode
			1	PSS	$1024t_{cyc}$	
		1	0	PSS	$512t_{cyc}$	
			1	PSS	$128t_{cyc}$	
	1	0	0	PSS	$32t_{cyc}$	
			1	PSS	$8t_{cyc}$	
		1	0	PSS	$4t_{cyc}$	
			1	PSS	$2t_{cyc}$	
1	0	0	0	PSW	$32t_{Wcyc}$	Time-base mode
			1	PSW	$16t_{Wcyc}$	
		1	0	PSW	$8t_{Wcyc}$	
			1	PSW	$2t_{Wcyc}$	
	1	0	0	PSW	$1/2t_{Wcyc}$	
			1	Inhibited		
		1	Don't care	PSW and TCA reset		

Notes: 1. t_{Wcyc} = 244.14 µs (when a 32.768-kHz crystal oscillator is used)
2. Timer counter overflow output period (seconds) = input clock period (seconds) × 256.
3. The division ratio must not be modified during time-base mode operation, otherwise an overflow cycle error will occur.

Figure 41 Timer Mode Register A (TMA)

Timer B

Timer B Functions: Timer B has the following functions.

- Free-running/reload timer
- External event counter
- Timer output operation (toggle, 0, and 1 outputs)

The block diagram of timer B is shown in figure 42.

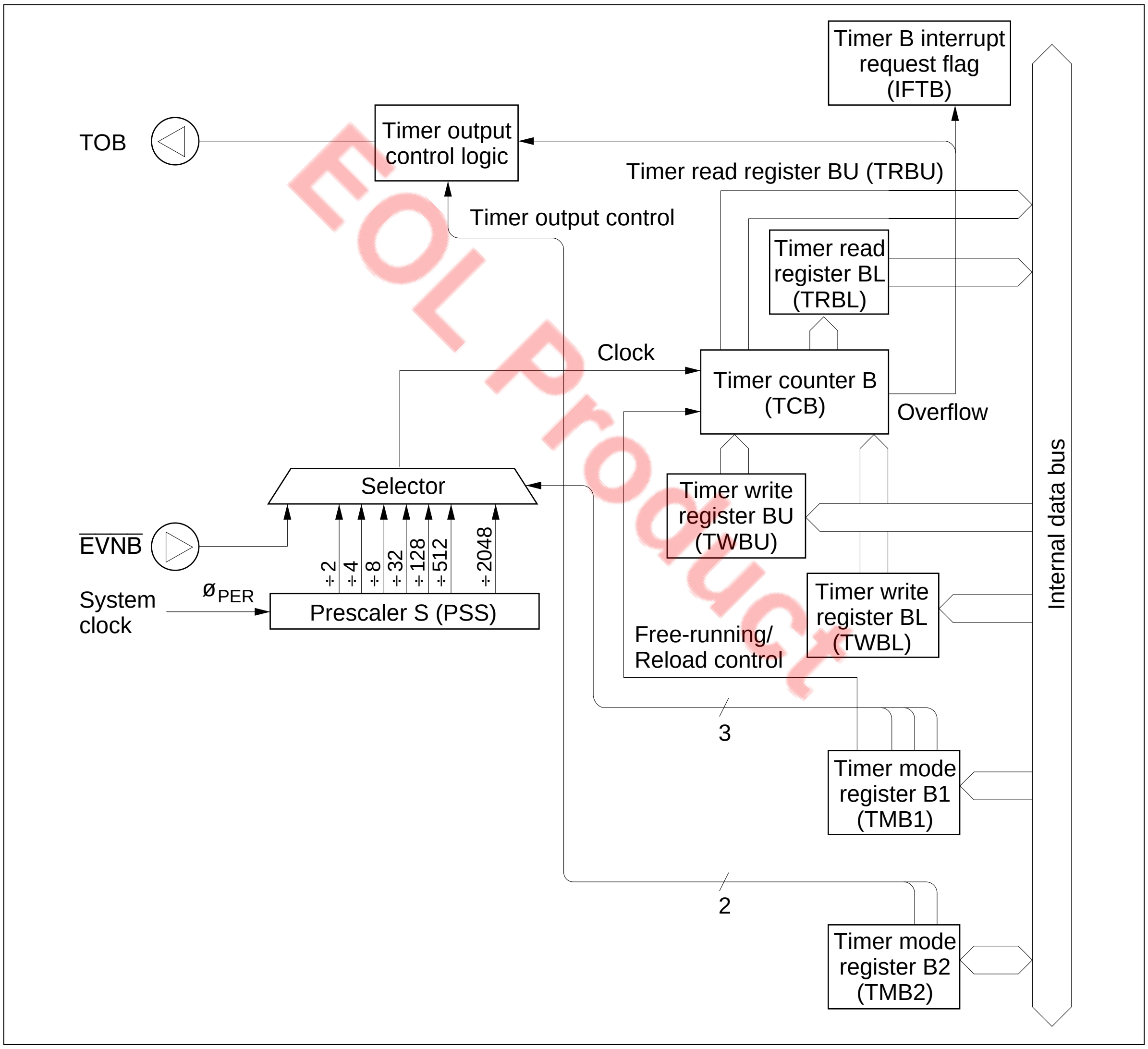


Figure 42 Timer B Block Diagram

Timer B Operations:

- Free-running/reload timer operation: The free-running/reload operation, input clock source, and prescaler division ratio are selected by timer mode register B1 (TMB1: $009).

  Timer B is initialized to the value set in timer write register B (TWBL: $00A, TWBU: $00B) by software and incremented by one at each clock input. If an input clock is applied to timer B after it has reached $FF, an overflow is generated. In this case, if the reload timer function is enabled, timer B is initialized to its initial value set in timer write register B; if the free-running timer function is enabled, the timer is initialized to $00 and then incremented again.

  The overflow sets the timer B interrupt request flag (IFTB: $002, bit 0). IFTB is reset by software or MCU reset. Refer to figure 3 and table 1 for details.

- External event counter operation: Timer B is used as an external event counter by selecting external event input as the input clock source. In this case, pin $R3_3/\overline{EVNB}$ must be set to $\overline{EVNB}$ by port mode register C (PMRC: $025).

  Timer B is incremented by one at each falling edge of signals input to pin $\overline{EVNB}$. Other operations are basically the same as the free-running/ reload timer operation.

- Timer output operation: The following three output modes can be selected for timer B by setting timer mode register B2 (TMB2: $013).

  — Toggle
  — 0 output
  — 1 output

  By selecting the timer output mode, pin $R3_0$/TOB is set to TOB. The output from TOB is reset low by MCU reset.

  — Toggle output: When toggle output mode is selected, the output level is inverted if a clock is input after timer B has reached $FF. By using this function and reload timer function, clock signals can be output at a required frequency for the buzzer. The output waveform is shown in figure 43.

  — 0 output: When 0 output mode is selected, the output level is pulled low if a clock is input after timer B has reached $FF. Note that this function must be used only when the output level is high.

  — 1 output: When 1 output mode is selected, the output level is set high if a clock is input after timer B has reached $FF. Note that this function must be used only when the output level is low.

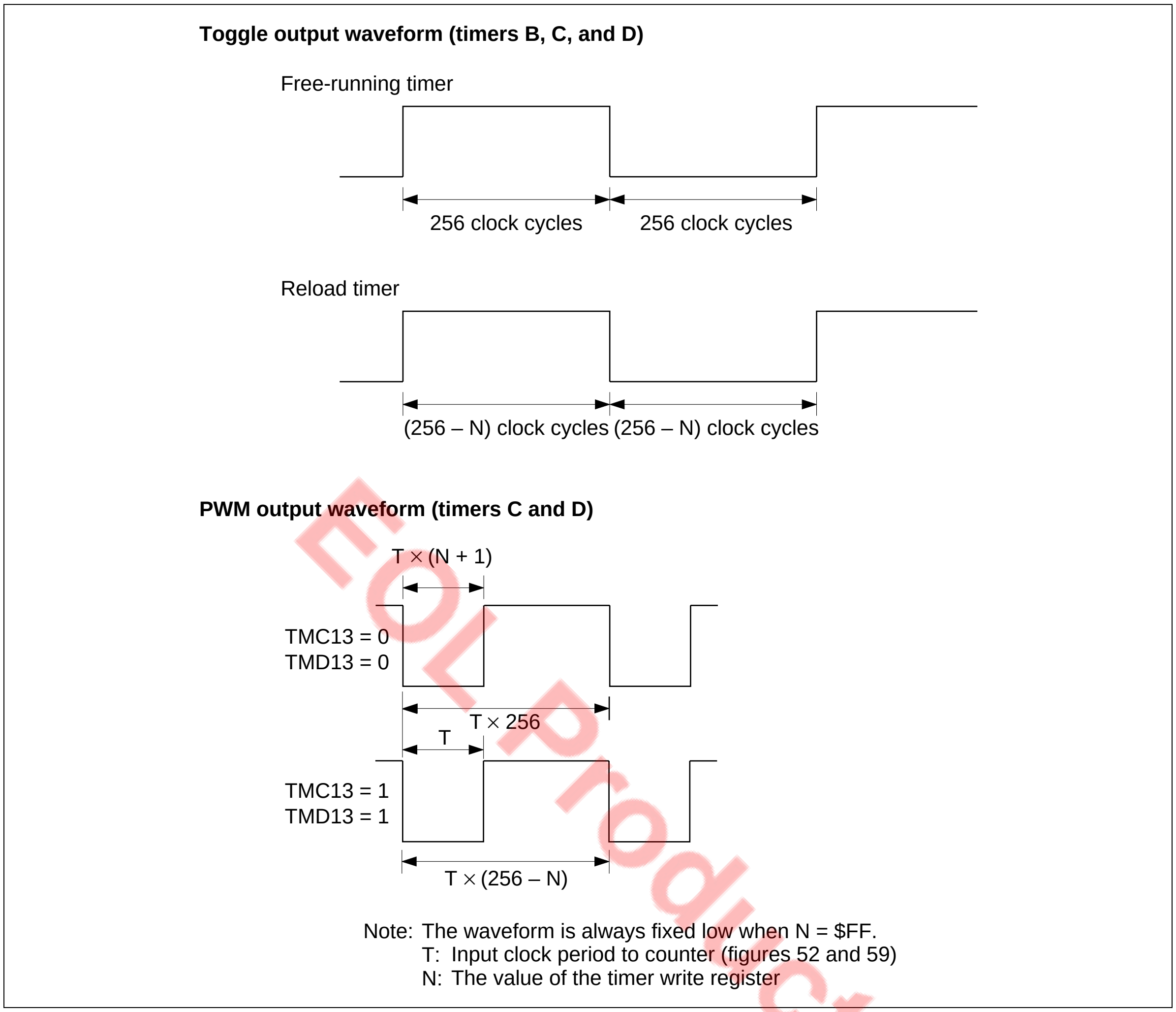


Figure 43 Timer Output Waveform

Registers for Timer B Operation: By using the following registers, timer B operation modes are selected and the timer B count is read and written.

- — Timer mode register B1 (TMB1: $009)
- — Timer mode register B2 (TMB2: $013)
- — Timer write register B (TWBL: $00A, TWBU: $00B)
- — Timer read register B (TRBL: $00A, TRBU: $00B)
- — Port mode register C (PMRC: $025)

- Timer mode register B1 (TMB1: $009): Four-bit write-only register that selects the free-running/reload timer function, input clock source, and the prescaler division ratio as shown in figure 44. It is reset to $0 by MCU reset.

Writing to this register is valid from the second instruction execution cycle after the execution of the previous timer mode register B1 write instruction. Setting timer B's initialization by writing to timer write register B (TWBL: $00A, TWBU: $00B) must be done after a mode change becomes valid.

Timer mode register B1 (TMB1: $009)

Bit	3	2	1	0
Initial value	0	0	0	0
Read/Write	W	W	W	W
Bit name	TMB13	TMB12	TMB11	TMB10

TMB13	Free-running/reload timer selection
0	Free-running timer
1	Reload timer

TMB12	TMB11	TMB10	Input clock period and input clock source
0	0	0	$2048t_{cyc}$
		1	$512t_{cyc}$
	1	0	$128t_{cyc}$
		1	$32t_{cyc}$
1	0	0	$8t_{cyc}$
		1	$4t_{cyc}$
	1	0	$2t_{cyc}$
		1	$R3_3/\overline{EVNB}$ (External event input)

Figure 44 Timer Mode Register B1 (TMB1)

- Timer mode register B2 (TMB2: $013): Two-bit read/write register that selects the timer B output mode as shown in figure 45. It is reset to $0 by MCU reset.

Timer mode register B2 (TMB2: $013)

Bit	3	2	1	0
Initial value	—	—	0	0
Read/Write	—	—	R/W	R/W
Bit name	Not used	Not used	TMB21	TMB20

TMB21	TMB20	$R3_0$/TOB mode selection	
0	0	$R3_0$	$R3_0$ port
	1	TOB	Toggle output
1	0	TOB	0 output
	1	TOB	1 output

Figure 45 Timer Mode Register B2 (TMB2)

- Timer write register B (TWBL: $00A, TWBU: $00B): Write-only register consisting of the lower digit (TWBL) and the upper digit (TWBU) as shown in figures 46 and 47. The lower digit is reset to $0 by MCU reset, but the upper digit value is invalid.

  Timer B is initialized by writing to timer write register B (TWBL: $00A, TWBU: $00B). In this case, the lower digit (TWBL) must be written to first, but writing only to the lower digit does not change the timer B value. Timer B is initialized to the value in timer write register B at the same time the upper digit (TWBU) is written to. When timer write register B is written to again and if the lower digit value needs no change, writing only to the upper digit initializes timer B.

Timer write register B (lower digit) (TWBL: $00A)

Bit	3	2	1	0
Initial value	0	0	0	0
Read/Write	W	W	W	W
Bit name	TWBL3	TWBL2	TWBL1	TWBL0

Figure 46 Timer Write Register B Lower Digit (TWBL)

Timer write register B (upper digit) (TWBU: $00B)

Bit	3	2	1	0
Initial value	Undefined	Undefined	Undefined	Undefined
Read/Write	W	W	W	W
Bit name	TWBU3	TWBU2	TWBU1	TWBU0

Figure 47 Timer Write Register B Upper Digit (TWBU)

- Timer read register B (TRBL: $00A, TRBU: $00B): Read-only register consisting of the lower digit (TRBL) and the upper digit (TRBU) that holds the count of the timer B upper digit (figures 48 and 49). The upper digit (TRBU) must be read first. At this time, the count of the timer B upper digit is obtained, and the count of the timer B lower digit is latched to the lower digit (TRBL). After this, by reading TRBL, the count of timer B when TRBU is read can be obtained.

Timer read register B (lower digit) (TRBL: $00A)

Bit	3	2	1	0
Initial value	Undefined	Undefined	Undefined	Undefined
Read/Write	R	R	R	R
Bit name	TRBL3	TRBL2	TRBL1	TRBL0

Figure 48 Timer Read Register B Lower Digit (TRBL)

Timer read register B (upper digit) (TRBU: $00B)

Bit	3	2	1	0
Initial value	Undefined	Undefined	Undefined	Undefined
Read/Write	R	R	R	R
Bit name	TRBU3	TRBU2	TRBU1	TRBU0

Figure 49 Timer Read Register B Upper Digit (TRBU)

- Port mode register C (PMRC: $025): Write-only register that selects $R3_3/\overline{EVNB}$ pin function as shown in figure 50. It is reset to $0 by MCU reset.

Port mode register C (PMRC: $025)

Bit	3	2	1	0
Initial value	0	0	0	0
Read/Write	W	W	W	W
Bit name	PMRC3	PMRC2	PMRC1	PMRC0

PMRC0	R3$_3$/$\overline{\text{EVNB}}$ mode selection
0	R3$_3$
1	$\overline{\text{EVNB}}$

PMRC1	R4$_0$/EVND mode selection
0	R4$_0$
1	EVND

PMRC2	D$_{12}$/$\overline{\text{STOPC}}$ mode selection
0	D$_{12}$
1	$\overline{\text{STOPC}}$

PMRC3	D$_{13}$/$\overline{\text{INT}}_0$ mode selection
0	D$_{13}$
1	$\overline{\text{INT}}_0$

Figure 50 Port Mode Register C (PMRC)

Timer C

Timer C Functions: Timer C has the following functions.

- Free-running/reload timer
- Watchdog timer
- Timer output operation (toggle, 0, 1, and PWM outputs)

The block diagram of timer C is shown in figure 51.

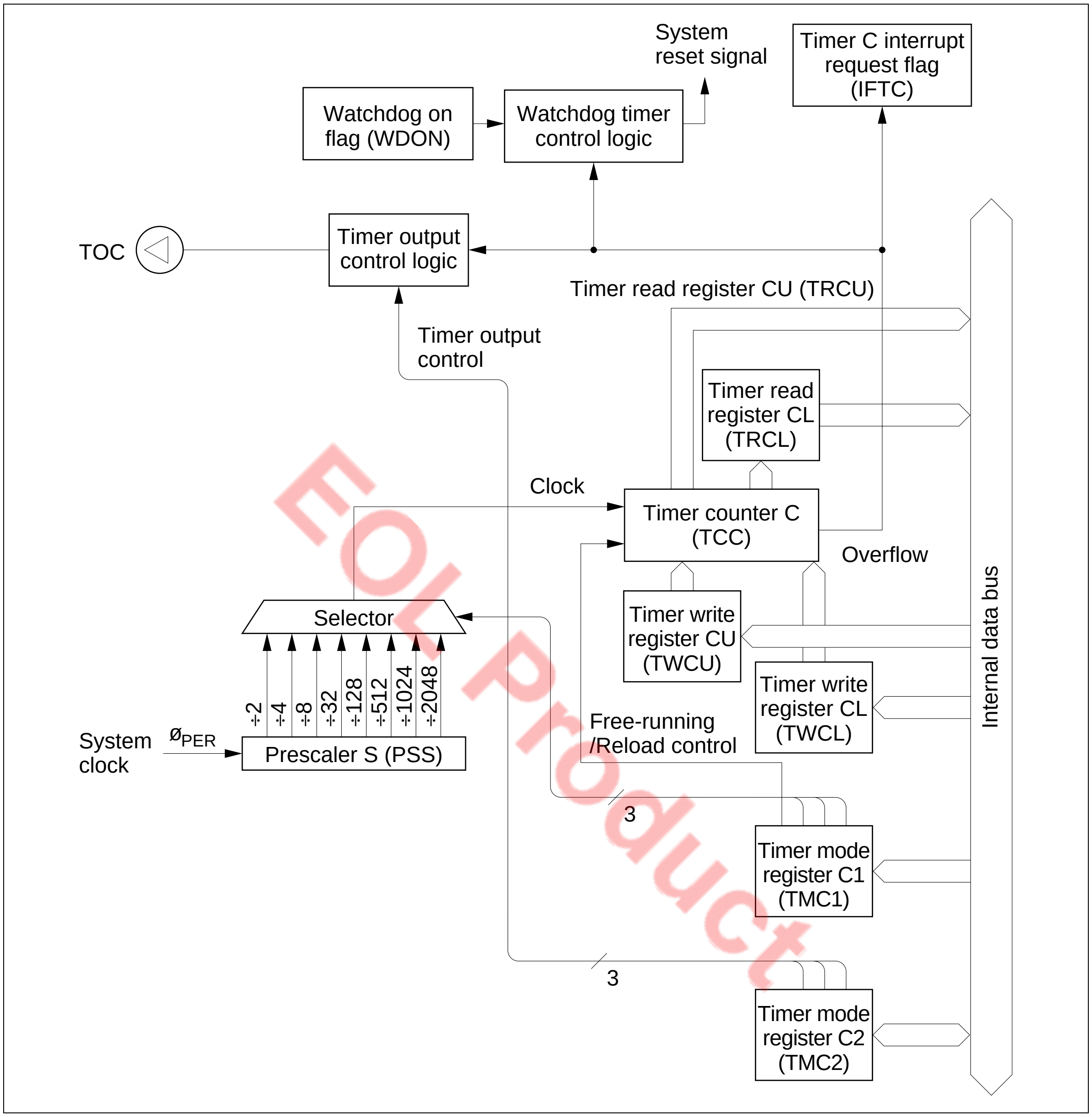


Figure 51 Timer C Block Diagram

Timer C Operations:

- Free-running/reload timer operation: The free-running/reload operation, input clock source, and prescaler division ratio are selected by timer mode register C1 (TMC1: $00D).

  Timer C is initialized to the value set in timer write register C (TWCL: $00E, TWCU: $00F) by software and incremented by one at each clock input. If an input clock is applied to timer C after it has reached $FF, an overflow is generated. In this case, if the reload timer function is enabled, timer C is initialized to its initial value set in timer write register C; if the free-running timer function is enabled, the timer is initialized to $00 and then incremented again.

The overflow sets the timer C interrupt request flag (IFTC: $002, bit 2). IFTC is reset by software or MCU reset. Refer to figure 3 and table 1 for details.

- Watchdog timer operation: Timer C is used as a watchdog timer for detecting out-of-control program routines by setting the watchdog on flag (WDON: $020, bit 1) to 1. If a program routine runs out of control and an overflow is generated, the MCU is reset. Program run can be controlled by initializing timer C by software before it reaches $FF.

- Timer output operation: The following four output modes can be selected for timer C by setting timer mode register C2 (TMC2: $014).

  — Toggle
  — 0 output
  — 1 output
  — PWM output

  By selecting the timer output mode, pin $R3_1$/TOC is set to TOC. The output from TOC is reset low by MCU reset.

  — Toggle output: The operation is basically the same as that of timer-B's toggle output.
  — 0 output: The operation is basically the same as that of timer-B's 0 output.
  — 1 output: The operation is basically the same as that of timer-B's 1 output.
  — PWM output: When PWM output mode is selected, timer C provides the variable-duty pulse output function. The output waveform differs depending on the contents of timer mode register C1 (TMC1: $00D) and timer write register C (TWCL: $00E, TWCU: $00F). The output waveform is shown in figure 43.

Registers for Timer C Operation: By using the following registers, timer C operation modes are selected and the timer C count is read and written.

— Timer mode register C1 (TMC1: $00D)
— Timer mode register C2 (TMC2: $014)
— Timer write register C (TWCL: $00E, TWCU: $00F)
— Timer read register C (TRCL: $00E, TRCU: $00F)

- Timer mode register C1 (TMC1: $00D): Four-bit write-only register that selects the free-running/reload timer function, input clock source, and prescaler division ratio as shown in figure 52. It is reset to $0 by MCU reset.

  Writing to this register is valid from the second instruction execution cycle after the execution of the previous timer mode register C1 write instruction. Setting timer C's initialization by writing to timer write register C (TWCL: $00E, TWCU: $00F) must be done after a mode change becomes valid.

- Timer mode register C2 (TMC2: $014): Three-bit read/write register that selects the timer C output mode as shown in figure 53. It is reset to $0 by MCU reset.

- Timer write register C (TWCL: $00E, TWCU: $00F): Write-only register consisting of a lower digit (TWCL) and an upper digit (TWCU) as shown in figures 54 and 55. The operation of timer write register C is basically the same as that of timer write register B (TWBL: $00A, TWBU: $00B).

- Timer read register C (TRCL: $00E, TRCU: $00F): Read-only register consisting of a lower digit (TRCL) and an upper digit (TRCU) that holds the count of the timer C upper digit as shown in figures 56 and 57. The operation of timer read register C is basically the same as that of timer read register B (TRBL: $00A, TRBU: $00B).

Timer mode register C1 (TMC1: $00D)

Bit	3	2	1	0
Initial value	0	0	0	0
Read/Write	W	W	W	W
Bit name	TMC13	TMC12	TMC11	TMC10

TMC13	Free-running/reload timer selection
0	Free-running timer
1	Reload timer

TMC12	TMC11	TMC10	Input clock period
0	0	0	2048t_{cyc}
		1	1024t_{cyc}
	1	0	512t_{cyc}
		1	128t_{cyc}
1	0	0	32t_{cyc}
		1	8t_{cyc}
	1	0	4t_{cyc}
		1	2t_{cyc}

Figure 52 Timer Mode Register C1 (TMC1)

Timer mode register C2 (TMC2: $014)

Bit	3	2	1	0
Initial value	—	0	0	0
Read/Write	—	R/W	R/W	R/W
Bit name	Not used	TMC22	TMC21	TMC20

TMC22	TMC21	TMC20	$R3_1$/TOC mode selection	
0	0	0	$R3_1$	$R3_1$ port
		1	TOC	Toggle output
	1	0	TOC	0 output
		1	TOC	1 output
1	0	0	—	Inhibited
		1		
	1	0		
		1	TOC	PWM output

Figure 53 Timer Mode Register C2 (TMC2)

Timer write register C (lower digit) (TWCL: $00E)

Bit	3	2	1	0
Initial value	0	0	0	0
Read/Write	W	W	W	W
Bit name	TWCL3	TWCL2	TWCL1	TWCL0

Figure 54 Timer Write Register C Lower Digit (TWCL)

Timer write register C (upper digit) (TWCU: $00F)

Bit	3	2	1	0
Initial value	Undefined	Undefined	Undefined	Undefined
Read/Write	W	W	W	W
Bit name	TWCU3	TWCU2	TWCU1	TWCU0

Figure 55 Timer Write Register C Upper Digit (TWCU)

Timer read register C (lower digit) (TRCL: $00E)

Bit	3	2	1	0
Initial value	Undefined	Undefined	Undefined	Undefined
Read/Write	R	R	R	R
Bit name	TRCL3	TRCL2	TRCL1	TRCL0

Figure 56 Timer Read Register C Lower Digit (TRCL)

Timer read register C (upper digit) (TRCU: $00F)

Bit	3	2	1	0
Initial value	Undefined	Undefined	Undefined	Undefined
Read/Write	R	R	R	R
Bit name	TRCU3	TRCU2	TRCU1	TRCU0

Figure 57 Timer Read Register C Upper Digit (TRCU)

Timer D

Timer D Functions: Timer D has the following functions.

- Free-running/reload timer
- External event counter
- Timer output operation (toggle, 0, 1, and PWM outputs)
- Input capture timer

The block diagram for each operation mode of timer D is shown in figures 58 (A) and (B).

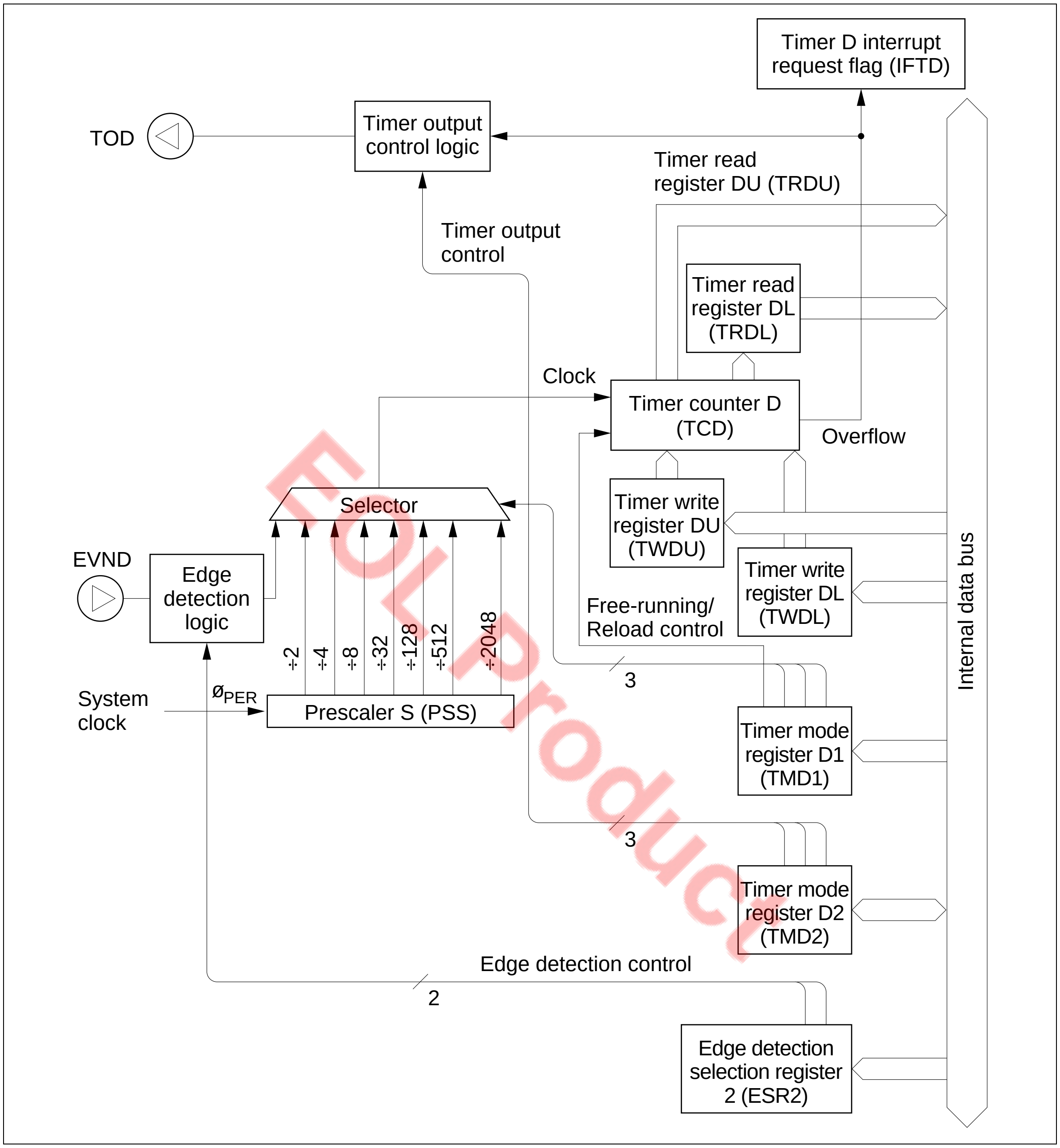


Figure 58 (A) Timer D Block Diagram (Free-Running/Reload Timer)

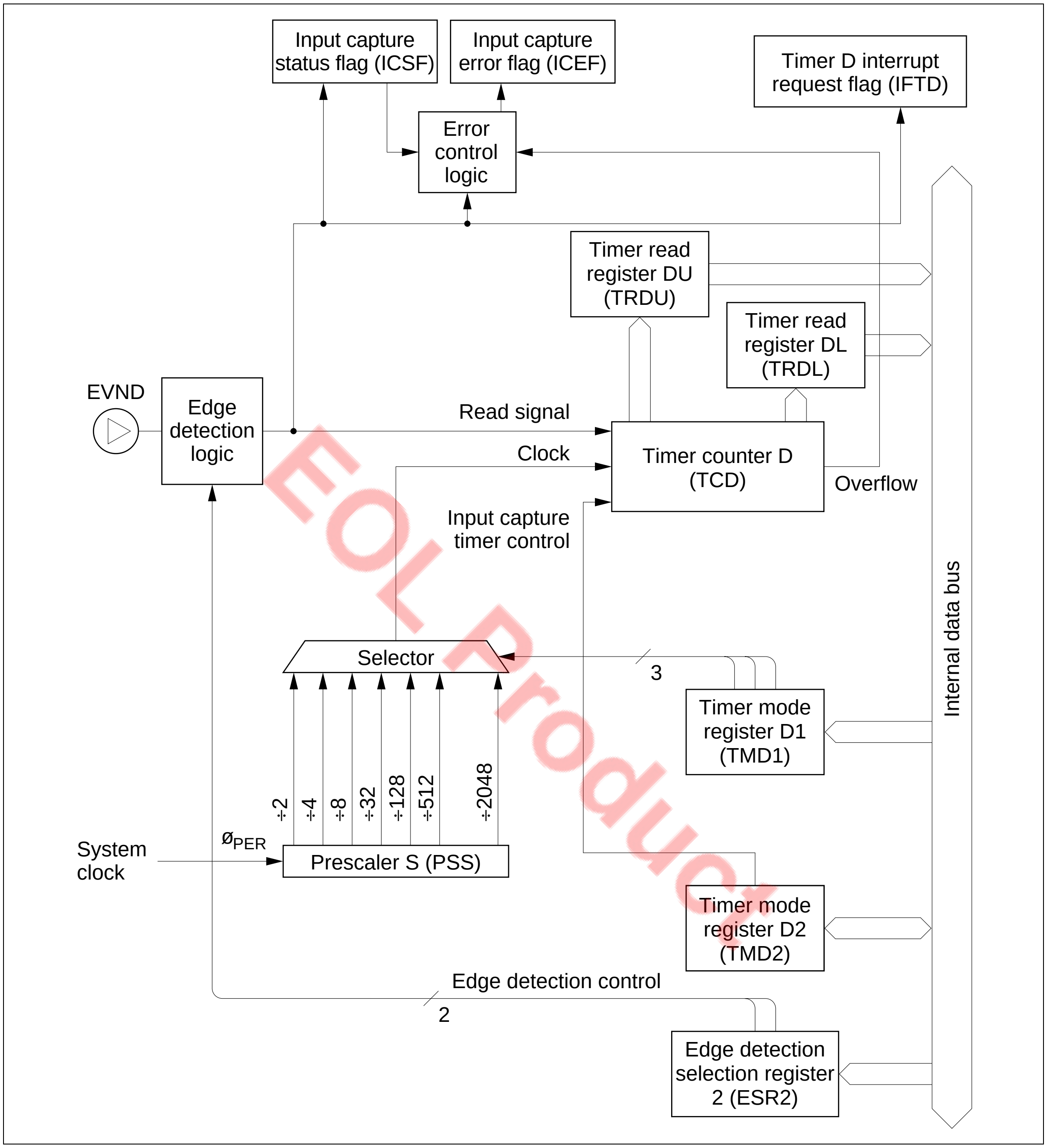


Figure 58 (B) Timer D Block Diagram (Input Capture Timer)

Timer D Operations:

- Free-running/reload timer operation: The free-running/reload operation, input clock source, and prescaler division ratio are selected by timer mode register D1 (TMD1: $010).

  Timer D is initialized to the value set in timer write register D (TWDL: $011, TWDU: $012) by software and incremented by one at each clock input. If an input clock is applied to timer D after it has reached $FF, an overflow is generated. In this case, if the reload timer function is enabled, timer D is initialized to its initial value set in timer write register D; if the free-running timer function is enabled, the timer is initialized to $00 and then incremented again.

  The overflow sets the timer D interrupt request flag (IFTD: $003, bit 0). IFTD is reset by software or MCU reset. Refer to figure 3 and table 1 for details.

- External event counter operation: Timer D is used as an external event counter by selecting the external event input as an input clock source. In this case, pin $R4_0$/EVND must be set to EVND by port mode register C (PMRC: $025).

  Either falling or rising edge, or both falling and rising edges of input signals can be selected as the external event detection edge by detection edge select register 2 (ESR2: $027). When both rising and falling edges detection is selected, the time between the falling edge and rising edge of input signals must be $2t_{cyc}$ or longer.

  Timer D is incremented by one at each detection edge selected by detection edge select register 2 (ESR2: $027). The other operation is basically the same as the free-running/reload timer operation.

- Timer output operation: The following four output modes can be selected for timer D by setting timer mode register D2 (TMD2: $015).

  — Toggle
  — 0 output
  — 1 output
  — PWM output

  By selecting the timer output mode, pin $R3_2$/TOD is set to TOD. The output from TOD is reset low by MCU reset.

  — Toggle output: The operation is basically the same as that of timer-B’s toggle output.
  — 0 output: The operation is basically the same as that of timer-B’s 0 output.
  — 1 output: The operation is basically the same as that of timer-B’s 1 output.
  — PWM output: The operation is basically the same as that of timer-C’s PWM output.

- Input capture timer operation: The input capture timer counts the clock cycles between trigger edges input to pin EVND.

  Either falling or rising edge, or both falling and rising edges of input signals can be selected as the trigger input edge by detection edge select register 2 (ESR2: $027).

When a trigger edge is input to EVND, the count of timer D is written to timer read register D (TRDL: $011, TRDU: $012), and the timer D interrupt request flag (IFTD: $003, bit 0) and the input capture status flag (ICSF: $021, bit 0) are set. Timer D is reset to $00, and then incremented again. While ICSF is set, if a trigger input edge is applied to timer D, or if timer D generates an overflow, the input capture error flag (ICEF: $021, bit 1) is set. ICSF and ICEF are reset to 0 by MCU reset or by writing 0.

By selecting the input capture operation, pin $R3_2$/TOD is set to $R3_2$ and timer D is reset to $00.

Registers for Timer D Operation: By using the following registers, timer D operation modes are selected and the timer D count is read and written.

— Timer mode register D1 (TMD1: $010)
— Timer mode register D2 (TMD2: $015)
— Timer write register D (TWDL: $011, TWDU: $012)
— Timer read register D (TRDL: $011, TRDU: $012)
— Port mode register C (PMRC: $025)
— Detection edge select register 2 (ESR2: $027)

- Timer mode register D1 (TMD1: $010): Four-bit write-only register that selects the free-running/reload timer function, input clock source, and the prescaler division ratio as shown in figure 59. It is reset to $0 by MCU reset.

  Writing to this register is valid from the second instruction execution cycle after the execution of the previous timer mode register D1 (TMD1: $010) write instruction. Setting timer D's initialization by writing to timer write register D (TWDL: $011, TWDU: $012) must be done after a mode change becomes valid.

  When selecting the input capture timer operation, select the internal clock as the input clock source.

- Timer mode register D2 (TMD2: $015): Four-bit read/write register that selects the timer D output mode and input capture operation as shown in figure 60. It is reset to $0 by MCU reset.

Timer mode register D1 (TMD1: $010)

Bit	3	2	1	0
Initial value	0	0	0	0
Read/Write	W	W	W	W
Bit name	TMD13	TMD12	TMD11	TMD10

TMD13	Free-running/reload timer selection
0	Free-running timer
1	Reload timer

TMD12	TMD11	TMD10	Input clock period and input clock source
0	0	0	$2048t_{cyc}$
		1	$512t_{cyc}$
	1	0	$128t_{cyc}$
		1	$32t_{cyc}$
1	0	0	$8t_{cyc}$
		1	$4t_{cyc}$
	1	0	$2t_{cyc}$
		1	$R4_0$/EVND (External event input)

Figure 59 Timer Mode Register D1 (TMD1)

- Timer write register D (TWDL: $011, TWDU: $012): Write-only register consisting of a lower digit (TWDL) and an upper digit (TWDU) as shown in figures 61 and 62. The operation of timer write register D is basically the same as that of timer write register B (TWBL: $00A, TWBU: $00B).

- Timer read register D (TRDL: $011, TRDU: $012): Read-only register consisting of a lower digit (TRDL) and an upper digit (TRDU) as shown in figures 63 and 64. The operation of timer read register D is basically the same as that of timer read register B (TRBL: $00A, TRBU: $00B).
  When the input capture timer operation is selected and if the count of timer D is read after a trigger is input, either the lower or upper digit can be read first.

- Port mode register C (PMRC: $025): Write-only register that selects $R4_0$/EVND pin function as shown in figure 50. It is reset to $0 by MCU reset.

- Detection edge select register 2 (ESR2: $027): Write-only register that selects the detection edge of signals input to pin EVND as shown in figure 65. It is reset to $0 by MCU reset.

Timer mode register D2 (TMD2: $015)

Bit	3	2	1	0
Initial value	0	0	0	0
Read/Write	R/W	R/W	R/W	R/W
Bit name	TMD23	TMD22	TMD21	TMD20

TMD23	TMD22	TMD21	TMD20	$R3_2$/TOD mode selection	
0	0	0	0	$R3_2$	$R3_2$ port
			1	TOD	Toggle output
		1	0	TOD	0 output
			1	TOD	1 output
	1	0	0	—	Inhibited
			1		
		1	0		
			1	TOD	PWM output
1	Don't care	Don't care	Don't care	$R3_2$	Input capture ($R3_2$ port)

Figure 60 Timer Mode Register D2 (TMD2)

Timer write register D (lower digit) (TWDL: $011)

Bit	3	2	1	0
Initial value	0	0	0	0
Read/Write	W	W	W	W
Bit name	TWDL3	TWDL2	TWDL1	TWDL0

Figure 61 Timer Write Register D Lower Digit (TWDL)

Timer write register D (upper digit) (TWDU: $012)

Bit	3	2	1	0
Initial value	Undefined	Undefined	Undefined	Undefined
Read/Write	W	W	W	W
Bit name	TWDU3	TWDU2	TWDU1	TWDU0

Figure 62 Timer Write Register D Upper Digit (TWDU)

Timer read register D (lower digit) (TRDL: $011)

Bit	3	2	1	0
Initial value	Undefined	Undefined	Undefined	Undefined
Read/Write	R	R	R	R
Bit name	TRDL3	TRDL2	TRDL1	TRDL0

Figure 63 Timer Read Register D Lower Digit (TRDL)

Timer read register D (upper digit) (TRDU: $012)

Bit	3	2	1	0
Initial value	Undefined	Undefined	Undefined	Undefined
Read/Write	R	R	R	R
Bit name	TRDU3	TRDU2	TRDU1	TRDU0

Figure 64 Timer Read Register D Upper Digit (TRDU)

Detection edge register 2 (ESR2: $027)

Bit	3	2	1	0
Initial value	0	0	—	—
Read/Write	W	W	—	—
Bit name	ESR23	ESR22	Not used	Not used

ESR23	ESR22	EVND detection edge
0	0	No detection
	1	Falling-edge detection
1	0	Rising-edge detection
	1	Double-edge detection*

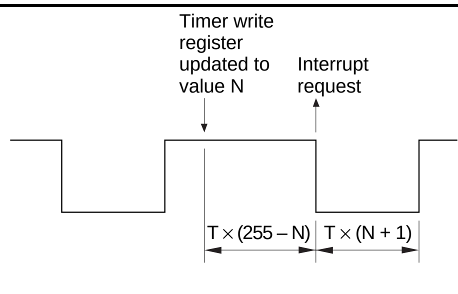
Note: * Both falling and rising edges are detected.

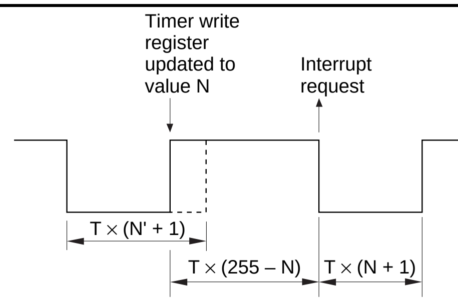
Figure 65 Detection Edge Select Register 2 (ESR2)

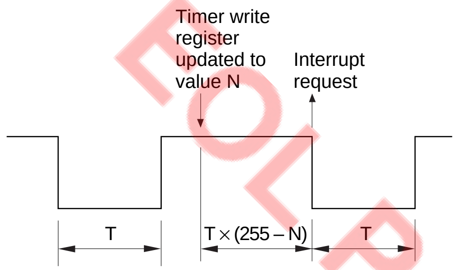
Notes on Use

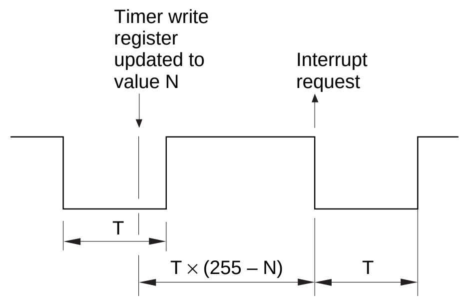
When using the timer output as PWM output, note the following point. From the update of the timer write register until the occurrence of the overflow interrupt, the PWM output differs from the period and duty settings, as shown in table 27. The PWM output should therefore not be used until after the overflow interrupt following the update of the timer write register. After the overflow, the PWM output will have the set period and duty cycle.

Table 27 PWM Output Following Update of Timer Write Register

<table>
<tr><th rowspan="2">Mode</th><th colspan="2">PWM Output</th></tr>
<tr><th>Timer Write Register is Updated during High PWM Output</th><th>Timer Write Register is Updated during Low PWM Output</th></tr>
<tr><td>Free running</td><td>


</td><td>


</td></tr>
<tr><td>Reload</td><td>


</td><td>


</td></tr>
</table>

Serial Interface

The MCU has two channels of serial interface. The transfer and receive start instructions differ according to the serial interface channel, but other functions are the same. The serial interface serially transfers or receives 8-bit data, and includes the following features.

- Multiple transmit clock sources
  - External clock
  - Internal prescaler output clock
  - System clock
- Output level control in idle states

Five registers, an octal counter, and a multiplexer are also configured for serial interfaces 1 and 2 as follows.

Serial interface 1

- Serial data register 1 (SR1L: $006, SR1U: $007)
- Serial mode register 1A (SM1A: $005)
- Serial mode register 1B (SM1B: $028)
- Port mode register A (PMRA: $004)
- Miscellaneous register (MIS: $00C)
- Octal counter (OC)
- Selector

Serial interface 2

- Serial data register 2 (SR2L: $01D, SR2U: $01E)
- Serial mode register 2A (SM2A: $01B)
- Serial mode register 2B (SM2B: $01C)
- Port mode register A (PMRA: $004)
- Octal counter (OC)
- Selector

The block diagram of serial interfaces 1 and 2 are shown in figure 66.

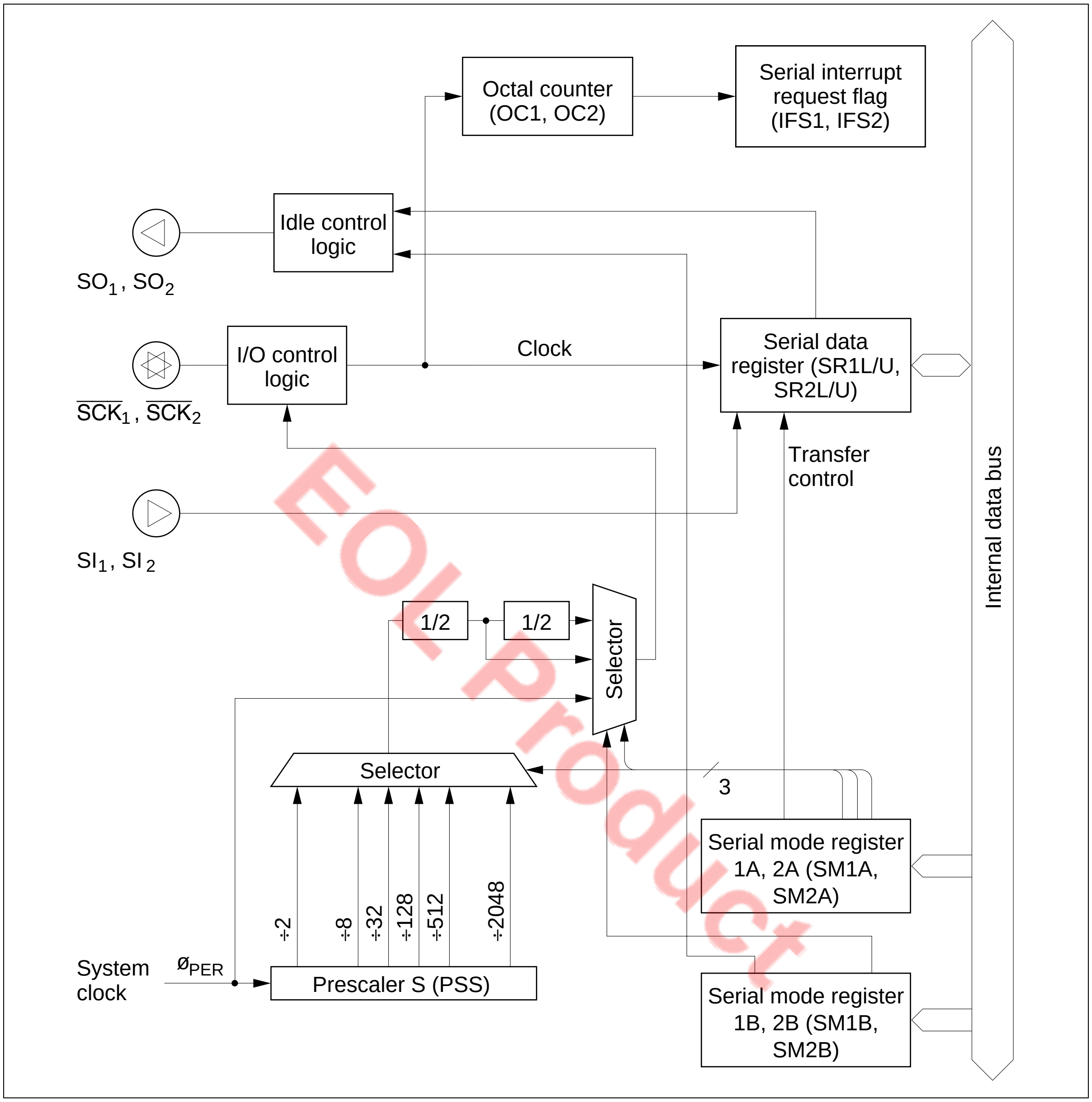


Figure 66 Serial Interfaces 1 and 2 Block Diagram

Serial Interface Operation

Selecting and Changing the Operating Mode: Tables 28 (A) and 28 (B) list the serial interfaces' operating modes. To select an operating mode, use one of these combinations of port mode register A (PMRA: $004), serial mode register 1A (SM1A: $005), and serial mode register 2A (SM2A: $01B) settings; to change the operating mode of serial interface 1, always initialize the serial interface internally by writing data to serial mode register 1A; and to change the operating mode of serial interface 2, always initialize the serial interface internally by writing data to serial mode register 2A. Note that serial interface

1 is initialized by writing data to serial mode register 1A, and serial interface 2 is initialized by writing data to serial mode register 2A. Refer to the following section Registers for Serial Interface for details.

Pin Setting: The $R4_1/\overline{SCK}_1$ pin is controlled by writing data to serial mode register 1A (SM1A: $005). The $R5_1/\overline{SCK}_2$ pin is controlled by writing data to serial mode register 2A (SM2A: $01B). Pins $R4_2/SI_1$, $R4_3/SO_1$, $R5_2/SI_2$, and $R5_3/SO_2$ are controlled by writing data to port mode register A (PMRA: $004). Refer to the following section Registers for Serial Interface for details.

Transmit Clock Source Setting: The transmit clock source of serial interface 1 is set by writing data to serial mode register 1A (SM1A: $005) and serial mode register 1B (SM1B: $028). The transmit clock source of serial interface 2 is set by writing data to serial mode register 2A (SM2A: $01B) and serial mode register 2B (SM2B: $01C). Refer to the following section Registers for Serial Interface for details.

Data Setting: Transmit data of serial interface 1 is set by writing data to serial data register 1 (SR1L: $006, SR1U: $007). Transmit data of serial interface 2 is set by writing data to serial data register 2 (SR2L: $01D, SR2U: $01E). Receive data of serial interface 1 is obtained by reading the contents of serial data register 1. Receive data of serial interface 2 is obtained by reading the contents of serial data register 2. The serial data is shifted by each serial interface transmit clock and is input from or output to an external system.

The output level of the SO_1 and SO_2 pins is invalid until the first data of each serial interface is output after MCU reset, or until the output level control in idle states is performed.

Transfer Control: Serial interface 1 is activated by the STS instruction. Serial interface 2 is activated by a dummy read of serial mode register 2A (SM2A: $01B), which will be referred to as SM2A read. The octal counter is reset to 000 by the STS instruction (serial interface 2 is SM2A read), and it increments at the rising edge of the transmit clock for each serial interface. When the eighth transmit clock signal is input or when serial transmission/reception is discontinued, the octal counter is reset to 000, the serial 1 interrupt request flag (IFS1: $003, bit 2) for serial interface 1 and serial 2 interrupt request flag (IFS2: $023, bit 2) for serial interface 2 are set, and the transfer stops.

When the prescaler output is selected as the transmit clock of serial interface 1, the transmit clock frequency is selected as $4t_{cyc}$ to $8192t_{cyc}$ by setting bits 0 to 2 (SM1A0–SM1A2) of serial mode register 1A (SM1A: $005) and bit 0 (SM1B0) of serial mode register 1B (SM1B: $028) as listed in table 29. When the prescaler output is selected as the transmit clock of serial interface 2, the transmit clock frequency is selected as $4t_{cyc}$ to $8192t_{cyc}$ by setting bits 0 to 2 (SM2A0– SM2A2) of serial mode register 2A (SM2A: $01B) and bit 0 (SM2B0) of serial mode register 2B (SM2B: $01C).

Note: To start serial interface 2, simply read serial mode register 2A by using the instruction that compares serial mode register 2A (SM2A: $01B) with the accumulator.
Serial mode register 2A (SM2A: $01B) is a read-only register, so $0 can be read.

Table 28 (A) Serial Interface 1 Operating Modes

SM1A	PMRA		
Bit 3	Bit 1	Bit 0	Operating Mode
1	0	0	Continuous clock output mode
		1	Transmit mode
	1	0	Receive mode
		1	Transmit/receive mode

Table 28 (B) Serial Interface 2 Operating Modes

SM2A	PMRA		
Bit 3	Bit 3	Bit 2	Operating Mode
1	0	0	Continuous clock output mode
		1	Transmit mode
	1	0	Receive mode
		1	Transmit/receive mode

Table 29 Serial Transmit Clock (Prescaler Output)

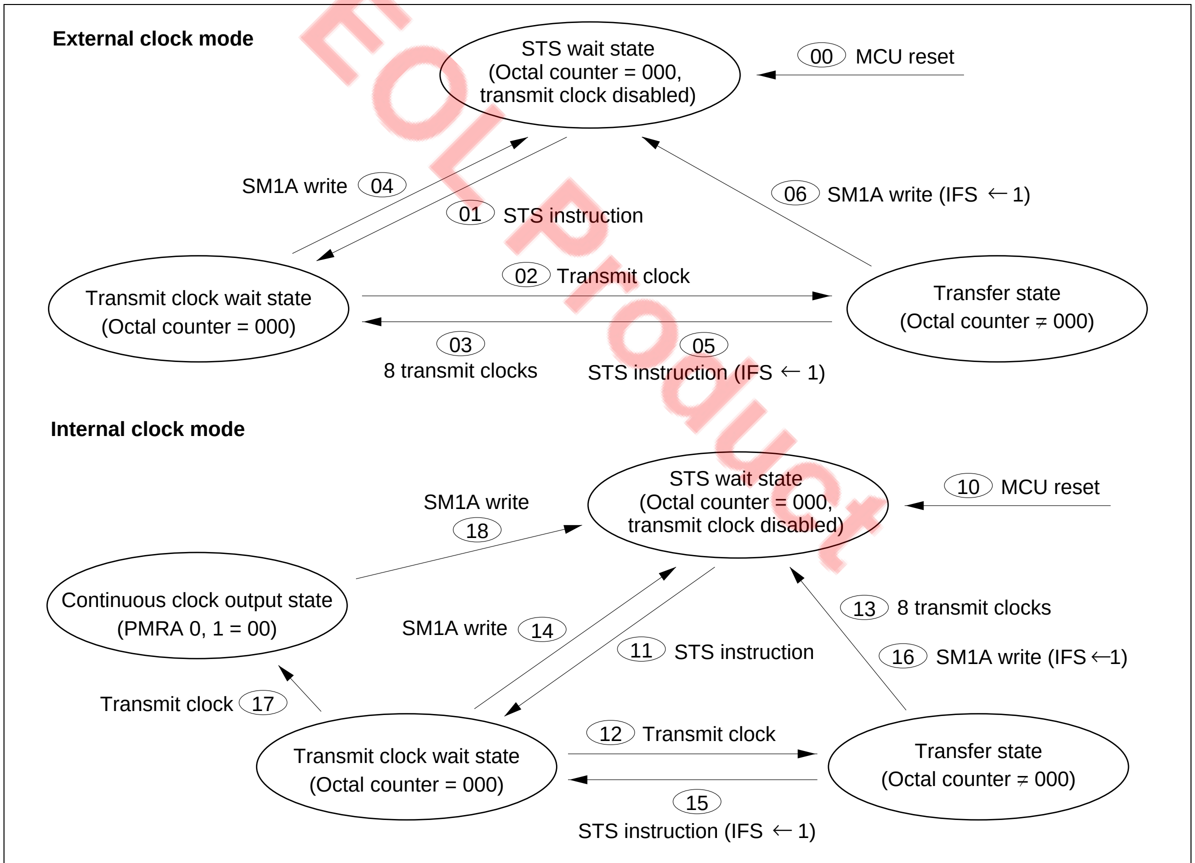
SM1B/ SM2B	SM1A/ SM2A				
Bit 0	Bit 2	Bit 1	Bit 0	Prescaler Division Ratio	Transmit Clock Frequency
0	0	0	0	÷ 2048	$4096t_{cyc}$
			1	÷ 512	$1024t_{cyc}$
		1	0	÷ 128	$256t_{cyc}$
			1	÷ 32	$64t_{cyc}$
	1	0	0	÷ 8	$16t_{cyc}$
			1	÷ 2	$4t_{cyc}$
1	0	0	0	÷ 4096	$8192t_{cyc}$
			1	÷ 1024	$2048t_{cyc}$
		1	0	÷ 256	$512t_{cyc}$
			1	÷ 64	$128t_{cyc}$
	1	0	0	÷ 16	$32t_{cyc}$
			1	÷ 4	$8t_{cyc}$

Operating States: The serial interface has the following operating states; transitions between them are shown in figure 67.

- — STS wait state (serial interface 2 is in SM2A read wait state)
- — Transmit clock wait state
- — Transfer state
- — Continuous clock output state (only in internal clock mode)

The operation state of serial interface 2 is the same as serial interface 1 except that the STS instruction of serial interface 1 changes to SM2A read. The following shows the operation state of serial interface 1.

- STS wait state: The serial interface enters STS wait state by MCU reset (00, 10 in figure 67). In STS wait state, serial interface 1 is initialized and the transmit clock is ignored. If the STS instruction is then executed (01, 11), serial interface 1 enters transmit clock wait state.



Note: Refer to the Operating States section for the corresponding encircled numbers.

Figure 67 Serial Interface State Transitions

- Transmit clock wait state: Transmit clock wait state is the period between the STS execution and the falling edge of the first transmit clock. In transmit clock wait state, input of the transmit clock (02, 12) increments the octal counter, shifts serial data register 1 (SR1L: $006, SR1U: $007), and enters the serial interface in transfer state. However, note that if continuous clock output mode is selected in internal clock mode, the serial interface does not enter transfer state but enters continuous clock output state (17).

The serial interface enters STS wait state by writing data to serial mode register 1A (SM1A: $005) (04, 14) in transmit clock wait state.

- Transfer state: Transfer state is the period between the falling edge of the first clock and the rising edge of the eighth clock. In transfer state, the input of eight clocks or the execution of the STS instruction sets the octal counter to 000, and the serial interface enters another state. When the STS instruction is executed (05, 15), transmit clock wait state is entered. When eight clocks are input, transmit clock wait state is entered (03) in external clock mode, and STS wait state is entered (13) in internal clock mode. In internal clock mode, the transmit clock stops after outputting eight clocks.

  In transfer state, writing data to serial mode register 1A (SM1A: $005) (06, 16) initializes serial interface 1, and STS wait state is entered.

  If the state changes from transfer to another state, the serial 1 interrupt request flag (IFS1: $003, bit 2) is set by the octal counter that is reset to 000.

- Continuous clock output state (only in internal clock mode): Continuous clock output state is entered only in internal clock mode. In this state, the serial interface does not transmit/receive data but only outputs the transmit clock from the $\overline{SCK}_1$ pin.

  When bits 0 and 1 (PMRA0, PMRA1) of port mode register A (PMRA: $004) are 00 in transmit clock wait state and if the transmit clock is input (17), the serial interface enters continuous clock output state. If serial mode register 1A (SM1A: $005) is written to in continuous clock output mode (18), STS wait state is entered.

Output Level Control in Idle States: When serial interface 1 is in STS instruction wait state and when serial interface 2 is in SM2A read wait state and transmit clock state, the output of each serial output pin, SO_1 and SO_2, can be controlled by setting bit 1 (SM1B1) of serial mode register 1B (SM1B: $028) to 0 or 1, or bit 1 (SM2B1) of serial mode register 2B (SM2B: $01C) to 0 or 1. The output level control example of serial interface 1 is shown in figure 68. Note that the output level cannot be controlled in transfer state.

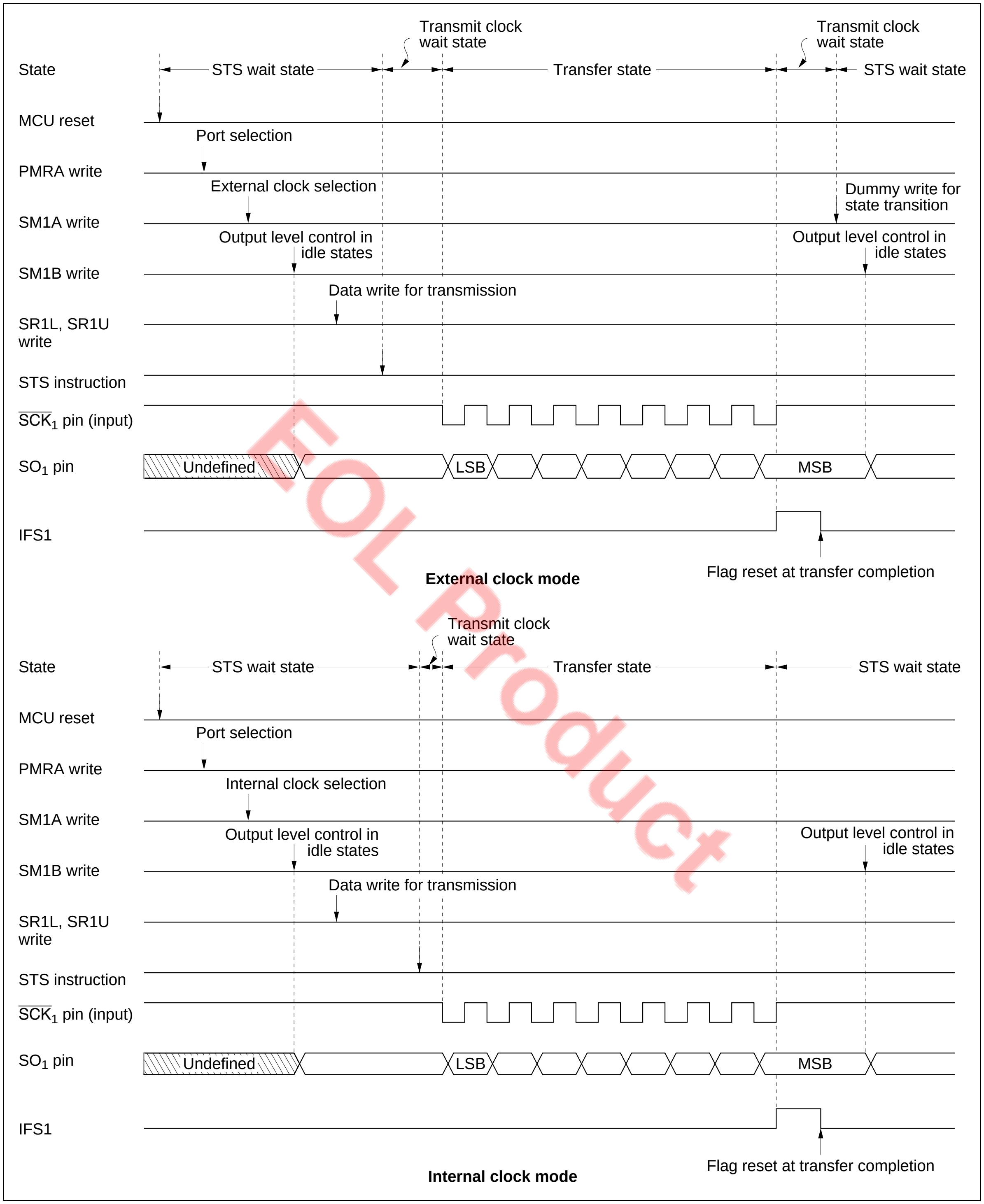


Figure 68 Example of Serial Interface 1 Operation Sequence

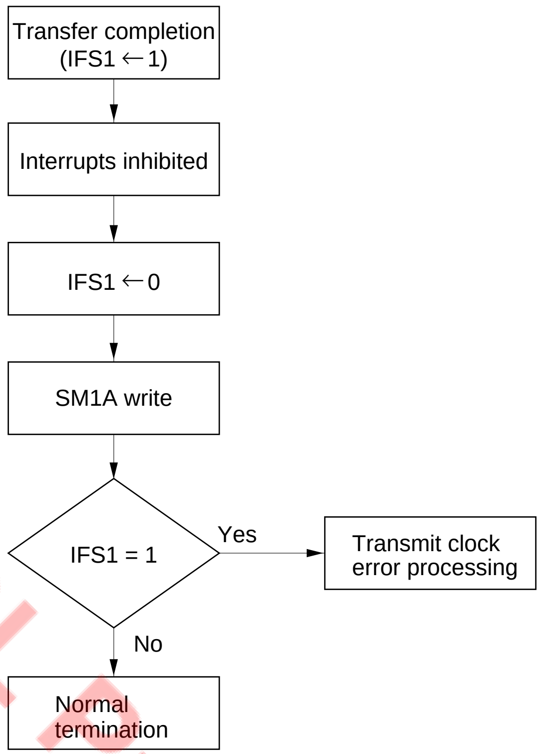
Transmit Clock Error Detection (In External Clock Mode): Each serial interface will malfunction if a spurious pulse caused by external noise conflicts with a normal transmit clock during transfer. A transmit clock error of this type can be detected as shown in figure 69.

If more than eight transmit clocks are input in transfer state, at the eighth clock including a spurious pulse by noise, the octal counter reaches 000, the serial 1 interrupt request flag (IFS1: $003, bit 2) is set, and transmit clock wait state is entered. At the falling edge of the next normal clock signal, the transfer state is entered. After the transfer is completed and IFS1 is reset, writing to serial mode register 1A (SM1A: $005) changes the state from transfer to STS wait. At this time serial interface 1 is in the transfer state, and the serial 1 interrupt request flag (IFS1: $003, bit 2) is set again, and therefore the error can be detected. The same applies to serial interface 2.

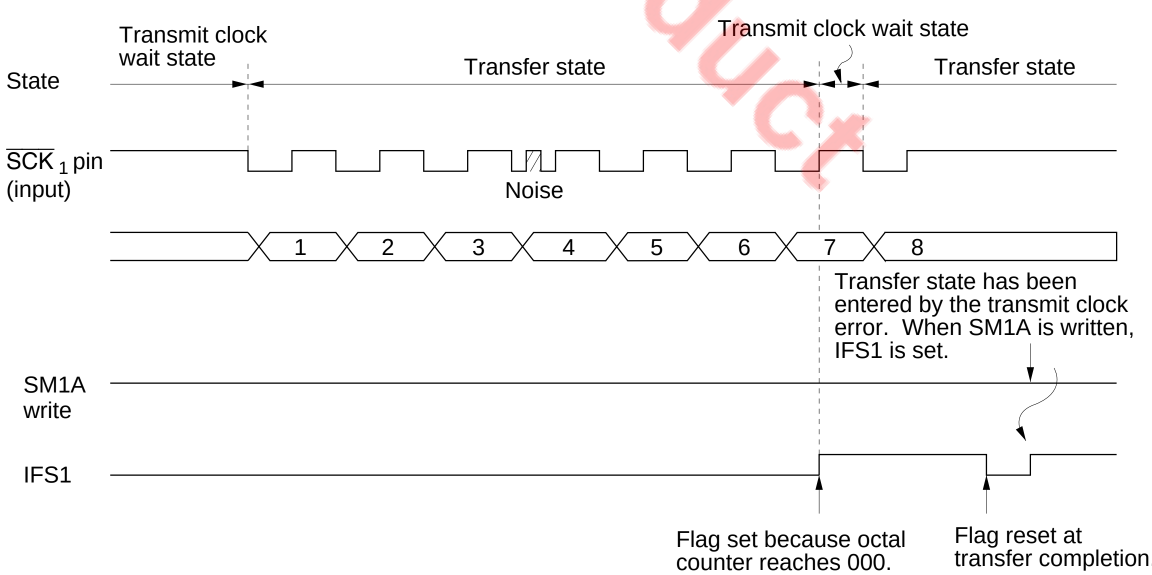
Notes on Use:

- Initialization after writing to registers: If port mode register A (PMRA: $004) is written to in transmit clock wait state or in transfer state, the serial interface must be initialized by writing to serial mode register 1A (SM1A: $005) and serial mode register 2A (SM2A: $01B) again.

- Serial 1 interrupt request flag (IFS1: $003, bit 2) and serial 2 interrupt request flag (IFS2: $023, bit 2) set: For serial interface 1, if the state is changed from transfer state to another by writing to serial mode register 1A (SM1A: $005) or executing the STS instruction during the first low pulse of the transmit clock, the serial 1 interrupt request flag (IFS1: $003, bit 2) is not set. In the same way for serial interface 2, if the state is changed from transfer state to another by writing to serial mode register 2A (SM2A: $01B) or by executing the STS instruction during the first low pulse of the transmit clock, the serial 2 interrupt request flag (IFS2: $023, bit 2) is not set. To set the serial 1 interrupt request flag (IFS1: $003, bit 2), a serial mode register 1A (SM1A: $005) write or STS instruction execution must be programmed to be executed after confirming that the $\overline{SCK_1}$ pin is at 1, that is, after executing the input instruction to port R4. To set the serial 2 interrupt request flag (IFS2: $023, bit 2), a serial mode register 2A (SM2A: $01B) write or SM2A instruction execution must be programmed to be executed after confirming that the $\overline{SCK_2}$ pin is at 1, that is, after executing the input instruction to port R5.



Transmit clock error detection flowchart



Transmit clock error detection procedures

Figure 69 Transmit Clock Error Detection

Registers for Serial Interface

The serial interface operation is selected, and serial data is read and written by the following registers.

For serial interface 1

- Serial mode register 1A (SM1A: $005)
- Serial mode register 1B (SM1B: $028)
- Serial data register 1
  (SR1L: $006, SR1U: $007)
- Port mode register A (PMRA: $004)
- Miscellaneous register (MIS: $00C)

For serial interface 2

- Serial mode register 2A (SM2A: $01B)
- Serial mode register 2B (SM2B: $01C)
- Serial data register 2
  (SR2L: $01D, SR2U: $01E)
- Port mode register A (PMRA: $004)

Serial Mode Register 1A (SM1A: $005): This register has the following functions (figure 70).

- $R4_1/\overline{SCK}_1$ pin function selection
- Serial interface 1 transmit clock selection
- Serial interface 1 prescaler division ratio selection
- Serial interface 1 initialization

Serial mode register 1A (SM1A: $005) is a 4-bit write-only register. It is reset to $0 by MCU reset.

A write signal input to serial mode register 1A (SM1A: $005) discontinues the input of the transmit clock to serial data register 1 (SR1L: $006, SR1U: $007) and the octal counter, and the octal counter is reset to 000. Therefore, if a write is performed during data transfer, the serial 1 interrupt request flag (IFS1: $003, bit 2) is set.

Written data is valid from the second instruction execution cycle after the write operation, so the STS instruction must be executed at least two cycles after that.

Serial mode register 1A (SM1A: $005)

Bit	3	2	1	0
Initial value	0	0	0	0
Read/Write	W	W	W	W
Bit name	SM1A3	SM1A2	SM1A1	SM1A0

SM1A3	$R4_1/\overline{SCK_1}$ mode selection
0	$R4_1$
1	$\overline{SCK_1}$

SM1A2	SM1A1	SM1A0	$\overline{SCK_1}$	Clock source	Prescaler division ratio
0	0	0	Output	Prescaler	Refer to table 29
		1			
	1	0			
		1			
1	0	0			
		1			
	1	0	Output	System clock	—
		1	Input	External clock	—

Figure 70 Serial Mode Register 1A (SM1A)

Serial Mode Register 1B (SM1B: $028): This register has the following functions (figure 71).

- Serial interface 1 prescaler division ratio selection
- Serial interface 1 output level control in idle states

Serial mode register 1B (SM1B: $028) is a 2-bit write-only register. It cannot be written during data transfer.

By setting bit 0 (SM1B0) of this register, the serial interface 1 prescaler division ratio is selected. Only bit 0 (SM1B0) can be reset to 0 by MCU reset. By setting bit 1 (SM1B1), the output level of the SO_1 pin is controlled in idle states of serial interface 1. The output level changes at the same time that SM1B1 is written to.

Serial mode register 1B (SM1B: $028)

Bit	3	2	1	0
Initial value	—	—	Undefined	0
Read/Write	—	—	W	W
Bit name	Not used	Not used	SM1B1	SM1B0

SM1B1	Output level control in idle states
0	Low level
1	High level

SM1B0	Transmit clock division ratio
0	Prescaler output divided by 2
1	Prescaler output divided by 4

Figure 71 Serial Mode Register 1B (SM1B)

Serial Data Register 1 (SR1L: $006, SR1U: $007): This register has the following functions (figures 72 and 73)

- Serial interface 1 transmission data write and shift
- Serial interface 1 receive data shift and read

Writing data in this register is output from the SO_1 pin, LSB first, synchronously with the falling edge of the transmit clock; data is input, LSB first, through the SI_1 pin at the rising edge of the transmit clock. Input/output timing is shown in figure 74.

Data cannot be read or written during serial data transfer. If a read/write occurs during transfer, the accuracy of the resultant data cannot be guaranteed.

Serial data register 1 (lower digit) (SR1L: $006)

Bit	3	2	1	0
Initial value	Undefined	Undefined	Undefined	Undefined
Read/Write	R/W	R/W	R/W	R/W
Bit name	SR13	SR12	SR11	SR10

Figure 72 Serial Data Register 1 (SR1L)

Serial data register 1 (upper digit) (SR1U: $007)

Bit	3	2	1	0
Initial value	Undefined	Undefined	Undefined	Undefined
Read/Write	R/W	R/W	R/W	R/W
Bit name	SR17	SR16	SR15	SR14

Figure 73 Serial Data Register 1 (SR1U)

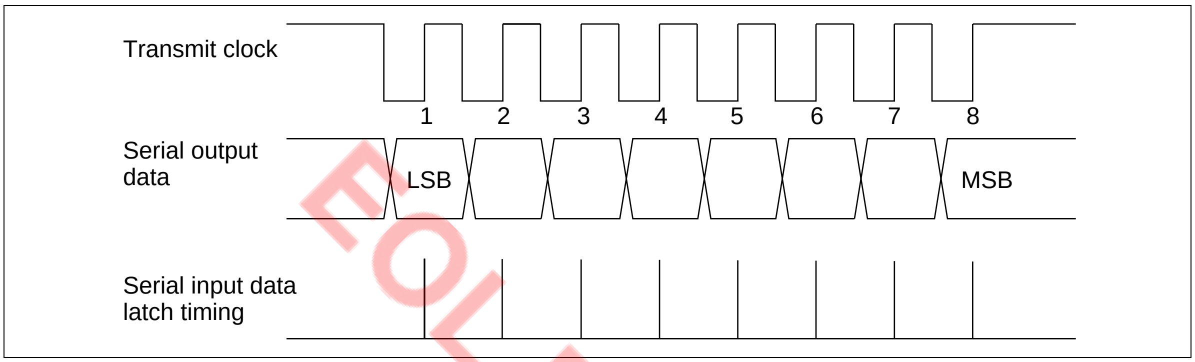


Figure 74 Serial Interface Output Timing

Port Mode Register A (PMRA: $004): This register has the following functions (figure 75).

- $R4_2/SI_1$ pin function selection
- $R4_3/SO_1$ pin function selection
- $R5_2/SI_2$ pin function selection
- $R5_3/SO_2$ pin function selection

Port mode register A (PMRA: $004) is a 4-bit write-only register, and is reset to $0 by MCU reset.

Port mode register A (PMRA: $004)

Bit	3	2	1	0
Initial value	0	0	0	0
Read/Write	W	W	W	W
Bit name	PMRA3	PMRA2	PMRA1	PMRA0

PMRA0	$R4_3/SO_1$ mode selection
0	$R4_3$
1	SO_1

PMRA1	$R4_2/SI_1$ mode selection
0	$R4_2$
1	SI_1

PMRA2	$R5_3/SO_2$ mode selection
0	$R5_3$
1	SO_2

PMRA3	$R5_2/SI_2$ mode selection
0	$R5_2$
1	SI_2

Figure 75 Port Mode Register A (PMRA)

Miscellaneous Register (MIS: $00C): This register has the following functions (figure 76).

- $R4_3/SO_1$ pin PMOS control

Miscellaneous register (MIS: $00C) is a 4-bit write-only register and is reset to $0 by MCU reset.

Miscellaneous register (MIS: $00C)

Bit	3	2	1	0
Initial value	0	0	0	0
Read/Write	W	W	W	W
Bit name	MIS3	MIS2	MIS1	MIS0

MIS1	MIS0	t_{RC}
0	0	0.12207 ms
		0.24414 ms*
	1	7.8125 ms
1	0	62.5 ms
	1	Not used

MIS2	$R4_3/SO_1$ PMOS on/off selection
0	On
1	Off

MIS3	Pull-up MOS on/off selection
0	Off
1	On

Note: *This value is valid only for direct transfer operation.

Figure 76 Miscellaneous Register (MIS)

Serial Mode Register 2A (SM2A: $01B): This register has the following functions (figure 77).

- $R5_1/\overline{SCK}_2$ pin function selection
- Serial interface 2 transmit clock selection
- Serial interface 2 prescaler division ratio selection
- Serial interface 2 initialization

Serial mode register 2A (SM2A: $01B) is a 4-bit write-only register. It is reset to $0 by MCU reset.

A write signal input to serial mode register 2A (SM2A: $01B) discontinues the input of the transmit clock to serial data register 2 (SR2L: $01D, SR1U: $01E) and the octal counter, and the octal counter is reset to

000. Therefore, if a write is performed during data transfer, the serial 2 interrupt request flag (IFS2: \$023, bit 2) is set.

Written data is valid from the second instruction execution cycle after the write operation, so the SM2A read instruction must be executed at least two cycles after that.

Serial mode register 2A (SM2A: \$01B)

Bit	3	2	1	0
Initial value	0	0	0	0
Read/Write	W	W	W	W
Bit name	SM2A3	SM2A2	SM2A1	SM2A0

SM2A3	$R5_1/\overline{SCK_2}$ mode selection
0	$R5_1$
1	$\overline{SCK_2}$

SM2A2	SM2A1	SM2A0	$\overline{SCK_2}$	Clock source	Prescaler division ratio
0	0	0	Output	Prescaler	Refer to table 29
		1			
	1	0			
		1			
1	0	0			
		1			
	1	0	Output	System clock	—
		1	Input	External clock	—

Figure 77 Serial Mode Register 2A (SM2A)

Serial Mode Register 2B (SM2B: \$01C): This register has the following functions (figure 78).

- Serial interface 2 prescaler division ratio selection
- Serial interface 2 output level control in idle states
- $R5_3/SO_2$ pin PMOS control

Serial mode register 2B (SM2B: \$01C) is a 3-bit write-only register. It cannot be written during serial interface 2 data transfer. Bit 0 (SM2B0) and bit 2 (SM2B2) is reset to \$0 by MCU reset.

By setting bit 0 (SM2B0) of this register, the serial interface 2 prescaler division ratio of serial interface 2 is selected. By resetting bit 1 (SM2B1), the output level of the SO_2 pin is controlled in idle states of serial interface 2. The output level changes at the same time that SM2B1 is written to.

Serial mode register 2B (SM2B: $01C)

Bit	3	2	1	0
Initial value	—	0	Undefined	0
Read/Write	—	W	W	W
Bit name	Not used	SM2B2	SM2B1	SM2B0

SM2B2	$R5_3/SO_2$ PMOS
0	On
1	Off

SM2B1	Output level control in idle states
0	Low level
1	High level

SM2B0	Transmit clock division ratio
0	Prescaler output divided by 2
1	Prescaler output divided by 4

Figure 78 Serial Mode Register 2B (SM2B)

Serial Data Register 2 (SR2L: $01D, SR2U: $01E): This register has the following functions (figures 79 and 80).

- Serial interface 2 transmission data write and shift
- Serial interface 2 receive data shift and read

Writing data in this register is output from the SO_2 pin, LSB first, synchronously with the falling edge of the transmit clock; data is input, LSB first, through the SI_2 pin at the rising edge of the transmit clock.

Data cannot be read or written during serial data transfer. If a read/write occurs during transfer, the accuracy of the resultant data cannot be guaranteed.

Serial data register 2 (lower digit) (SR2L: $01D)

Bit	3	2	1	0
Initial value	Undefined	Undefined	Undefined	Undefined
Read/Write	R/W	R/W	R/W	R/W
Bit name	SR23	SR22	SR21	SR20

Figure 79 Serial Data Register 2 (SR2L)

Serial data register 2 (upper digit) (SR2U: $01E)

Bit	3	2	1	0
Initial value	Undefined	Undefined	Undefined	Undefined
Read/Write	R/W	R/W	R/W	R/W
Bit name	SR27	SR26	SR25	SR24

Figure 80   Serial Data Register 2 (SR2U)

A/D Converter

The MCU has a built-in A/D converter that uses a sequential comparison method with a resistor ladder. It can measure four analog inputs with 8-bit resolution. As shown in the block diagram of figure 81, the A/D converter has a 4-bit A/D mode register, a 1-bit A/D start flag, and a 4-bit plus 4-bit A/D data register.

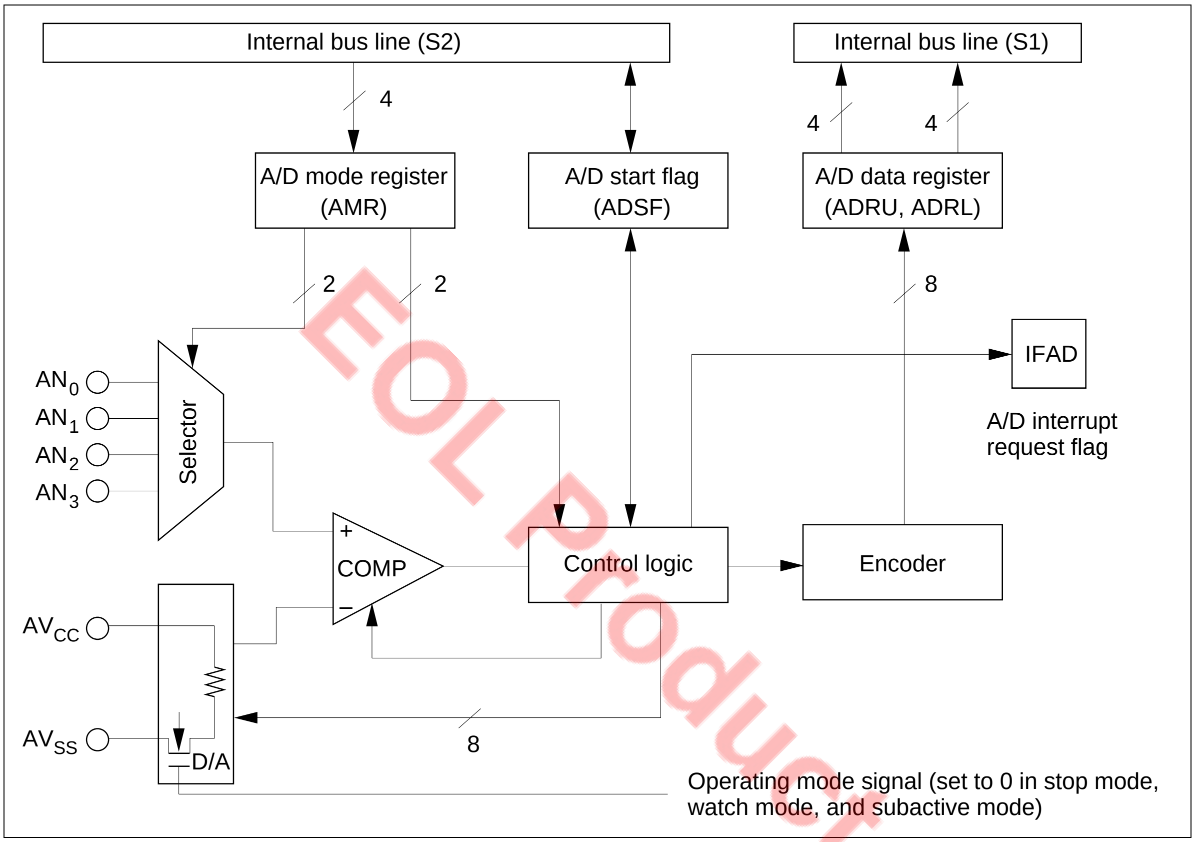


Figure 81 A/D Converter Block Diagram

A/D Mode Register (AMR: $016): Four-bit write-only register which selects the A/D conversion period and indicates analog input pin information. Bit 0 of the A/D mode register selects the A/D conversion period, and bits 2 and 3 select a channel, as shown in figure 82.

A/D Start Flag (ADSF: $020, Bit 2): One-bit flag that initiates A/D conversion when set to 1. At the completion of A/D conversion, the converted data is stored in the A/D data register and the A/D start flag is cleared. Refer to figure 86.

A/D Data Register (ADRL: $017, ADRU: $018): Eight-bit read-only register consisting of a 4-bit lower digit and 4-bit upper digit. This register is not cleared by reset. After the completion of A/D conversion, the resultant eight-bit data is held in this register until the start of the next conversion (figures 83, 84, and 85).

Note on Use: Use the SEM and SEMD instructions to write data to the A/D start flag (ADSF: $020, bit 2), but make sure that the A/D start flag is not written to during A/D conversion. Data read from the A/D data register (ADRL: $017, ADRU: $018) during A/D conversion cannot be guaranteed.

The A/D converter does not operate in the stop, watch, and subactive modes because of the OSC clock. During these low-power dissipation modes, current through the resistor ladder is cut off to decrease the power input.

A/D mode register (AMR: $016)

Bit	3	2	1	0
Initial value	0	0	—	0
Read/Write	W	W	—	W
Bit name	AMR3	AMR2	Not used	AMR0

AMR3	AMR2	Analog input selection
0	0	AN_0
0	1	AN_1
1	0	AN_2
1	1	AN_3

AMR0	Conversion time
0	$34t_{cyc}$
1	$67t_{cyc}$

Figure 82 A/D Mode Register (AMR)

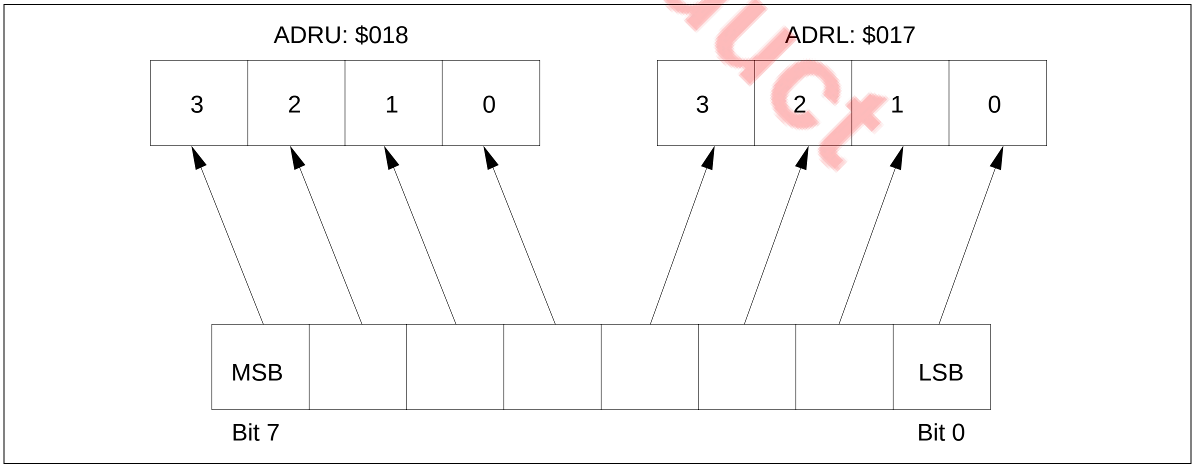


Figure 83 A/D Data Registers

A/D data register (lower digit) (ADRL: $017)

Bit	3	2	1	0
Initial value	0	0	0	0
Read/Write	R	R	R	R
Bit name	ADRL3	ADRL2	ADRL1	ADRL0

Figure 84 A/D Data Register Lower Digit (ADRL)

A/D data register (upper digit) (ADRU: $018)

Bit	3	2	1	0
Initial value	1	0	0	0
Read/Write	R	R	R	R
Bit name	ADRU3	ADRU2	ADRU1	ADRU0

Figure 85 A/D Data Register Upper Digit (ADRU)

A/D start flag (ADSF: $020, bit 2)

Bit	3	2	1	0
Initial value	0	0	0	0
Read/Write	R/W	R/W	R/W	R/W
Bit name	DTON	ADSF	WDON	LSON

LSON
Refer to the description of operating modes

WDON
Refer to the description of timers

ADSF (A/D start flag)	
1	A/D conversion started
0	A/D conversion completed

DTON
Refer to the description of operating modes

Figure 86 A/D Start Flag (ADSF)

Notes on Mounting

Assemble all parts including the HD404449 Series on a board, noting the points described below.

1. Connect layered ceramic type capacitors (about 0.1 µF) between AV_{CC} and AV_{SS}, between V_{CC} and GND, and between used analog pins and AV_{SS}.
2. Connect unused analog pins to AV_{SS}.

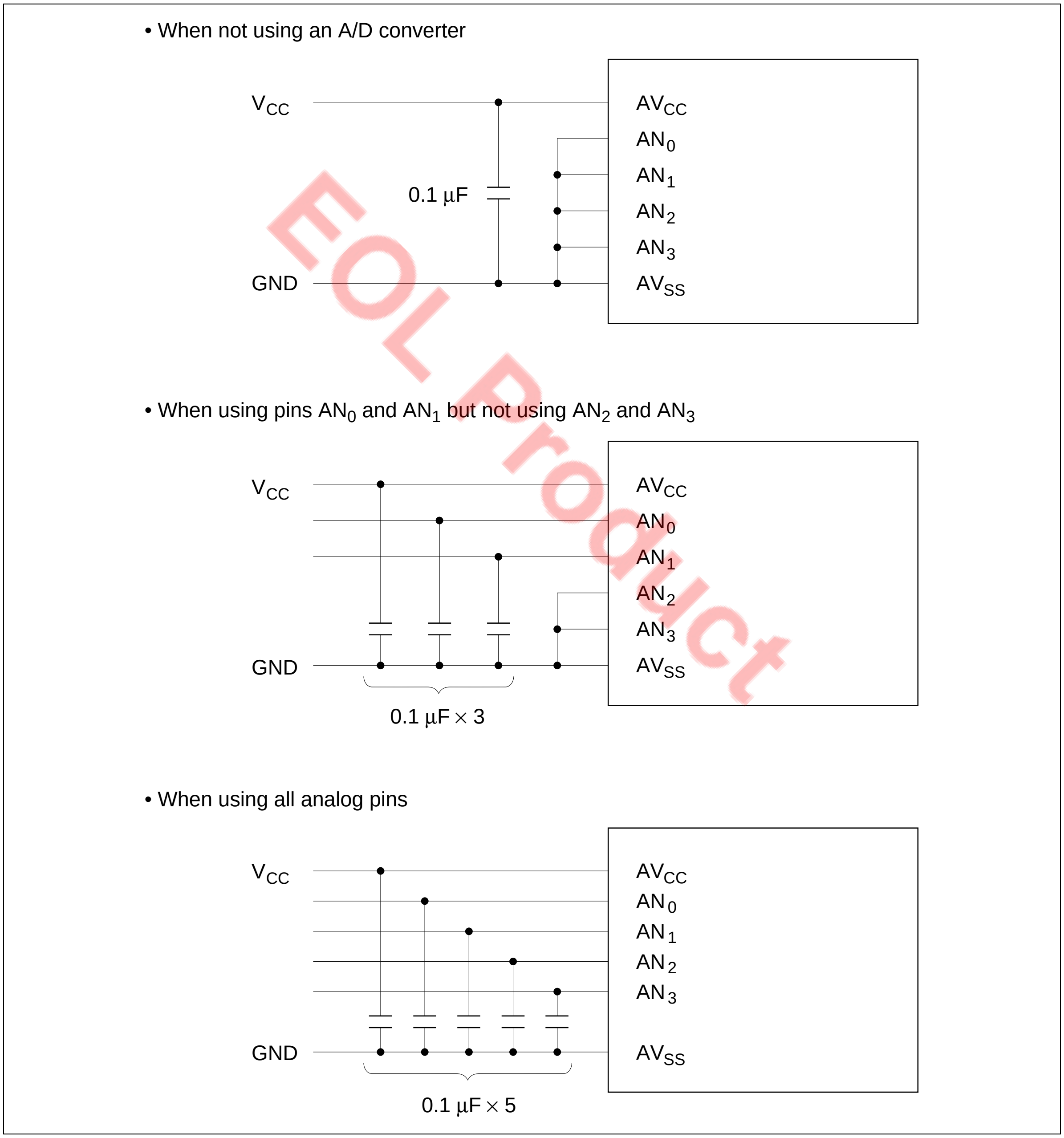


Figure 87 Example of Connections (1)

Between the V_{CC} and GND lines, connect capacitors designed for use in ordinary power supply circuits. An example connection is described in figure 88.

No resistors can be inserted in series in the power supply circuit, so the capacitors should be connected in parallel. The capacitors are a large capacitance C_1 and a small capacitance C_2.

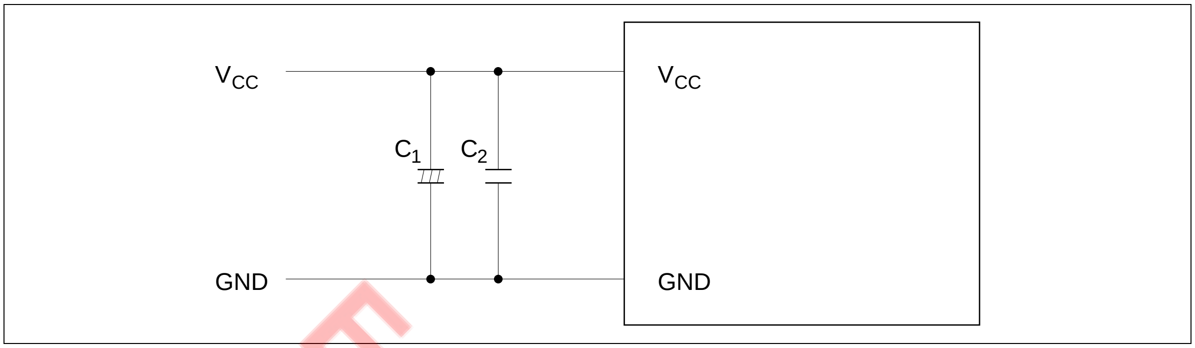


Figure 88 Example of Connections (2)

Programmable ROM (HD4074449)

The HD4074449 is a ZTAT™ microcomputer with built-in PROM that can be programmed in PROM mode.

PROM Mode Pin Description

	MCU Mode		PROM Mode			MCU Mode		PROM Mode	
Pin No.	**Pin Name**	**I/O**	**Pin Name**	**I/O**	**Pin No.**	**Pin Name**	**I/O**	**Pin Name**	**I/O**
1	AN_2	I			31	$R1_2$	I/O	A_7	I
2	AN_3	I			32	$R1_3$	I/O	A_8	I
3	AV_{SS}		GND		33	$R2_0$	I/O	A_0	I
4	$\overline{TEST}$	I	$\overline{TEST}$	I	34	$R2_1$	I/O	A_{10}	I
5	OSC_1	I	V_{CC}		35	$R2_2$	I/O	A_{11}	I
6	OSC_2	O			36	$R2_3$	I/O	A_{12}	I
7	RESET	I	RESET	I	37	$R3_0$/TOB	I/O		
8	X1	I	GND		38	$R3_1$/TOC	I/O		
9	X2	O			39	$R3_2$/TOD	I/O		
10	GND		GND		40	$R3_3/\overline{EVNB}$	I/O		
11	D_0	I/O	$\overline{CE}$	I	41	$R4_0$/EVND	I/O		
12	D_1	I/O	$\overline{OE}$	I	42	$R4_1/\overline{SCK_1}$	I/O		
13	D_2	I/O	V_{CC}		43	$R4_2/SI_1$	I/O		
14	D_3	I/O	V_{CC}		44	$R4_3/SO_1$	I/O		
15	D_4	I/O			45	$R5_0$	I/O		
16	D_5	I/O			46	$R5_1/\overline{SCK_2}$	I/O		
17	D_6	I/O			47	$R5_2/SI_2$	I/O		
18	D_7	I/O			48	$R5_3/SO_2$	I/O		
19	D_8	I/O			49	$R6_0$	I/O	A_1	I
20	D_9	I/O			50	$R6_1$	I/O	A_2	I
21	D_{10}	I/O	A_{13}	I	51	$R6_2$	I/O	A_3	I
22	D_{11}	I/O	A_{14}	I	52	$R6_3$	I/O	A_4	I
23	$D_{12}/\overline{STOPC}$	I	A_9	I	53	$R7_0$	I/O	O_0	I/O
24	$D_{13}/\overline{INT_0}$	I	V_{PP}		54	$R7_1$	I/O	O_1	I/O
25	$R0_0/\overline{INT_1}$	I/O	$\overline{M_0}$	I	55	$R7_2$	I/O	O_2	I/O
26	$R0_1/INT_2$	I/O	$\overline{M_1}$	I	56	$R7_3$	I/O	O_3	I/O
27	$R0_2/INT_3$	I/O			57	$R8_0$	I/O	O_4	I/O
28	$R0_3$	I/O			58	$R8_1$	I/O	O_5	I/O
29	$R1_0$	I/O	A_5	I	59	$R8_2$	I/O	O_6	I/O
30	$R1_1$	I/O	A_6	I	60	$R8_3$	I/O	O_7	I/O

	MCU Mode		PROM Mode			MCU Mode		PROM Mode	
Pin No.	Pin Name	I/O	Pin Name	I/O	Pin No.	Pin Name	I/O	Pin Name	I/O
61	$R9_0$	I/O	O_4	I/O	71	RB_2	I/O		
62	$R9_1$	I/O	O_3	I/O	72	RB_3	I/O		
63	$R9_2$	I/O	O_2	I/O	73	RC_0	I/O		
64	$R9_3$	I/O	O_1	I/O	74	RC_1	I/O		
65	RA_0	I/O	O_0	I/O	75	RC_2	I/O		
66	RA_1	I/O	V_{CC}		76	RC_3	I/O		
67	RA_2	I/O			77	V_{CC}		V_{CC}	
68	RA_3	I/O			78	AV_{CC}		V_{CC}	
69	RB_0	I/O			79	AN_0	I		
70	RB_1	I/O			80	AN_1	I		

Notes: 1. I/O: Input/output pin, I: Input pin, O: Output pin
2. Each of O_0–O_4 has two pins; before using, each pair must be connected together.

Programming the Built-In PROM

The MCU's built-in PROM is programmed in PROM mode. PROM mode is set by pulling $\overline{\text{TEST}}$, $\overline{M_0}$, and $\overline{M_1}$ low, and RESET high as shown in figure 89. In PROM mode, the MCU does not operate, but it can be programmed in the same way as any other commercial 27256-type EPROM using a standard PROM programmer and an 80-to-28-pin socket adapter. Recommended PROM programmers and socket adapters of the HD4074449 are listed in table 31.

Since an HMCS400-series instruction is ten bits long, the HMCS400-series MCU has a built-in conversion circuit to enable the use of a general-purpose PROM programmer. This circuit splits each instruction into five lower bits and five upper bits that are read from or written to consecutive addresses. This means that if, for example, 16 kwords of built-in PROM are to be programmed by a general-purpose PROM programmer, a 32-kbyte address space ($0000–$7FFF) must be specified.

Warnings

1. Always specify addresses $0000 to $7FFF when programming with a PROM programmer. If address $8000 or higher is accessed, the PROM may not be programmed or verified correctly. Set all data in unused addresses to $FF.

   Note that the plastic-package version cannot be erased or reprogrammed.
2. Make sure that the PROM programmer, socket adapter, and LSI are aligned correctly (their pin 1 positions match), otherwise overcurrents may damage the LSI. Before starting programming, make sure that the LSI is firmly fixed in the socket adapter and the socket adapter is firmly fixed onto the programmer.
3. PROM programmers have two voltages (V_{PP}): 12.5 V and 21 V. Remember that ZTAT™ devices require a V_{PP} of 12.5 V—the 21-V setting will damage them. 12.5 V is the Intel 27256 setting.

Programming and Verification

The built-in PROM of the MCU can be programmed at high speed without risk of voltage stress or damage to data reliability.

Programming and verification modes are selected as listed in table 30.

For details of PROM programming, refer to the following section, Notes on PROM Programming.

Table 30 PROM Mode Selection

	Pin			
Mode	$\overline{CE}$	$\overline{OE}$	V_{PP}	O_0–O_7
Programming	Low	High	V_{PP}	Data input
Verification	High	Low	V_{PP}	Data output
Programming inhibited	High	High	V_{PP}	High impedance

Table 31 Recommended PROM Programmers and Socket Adapters

PROM Programmer		Socket Adapter		
Manufacturer	**Model Name**	**Manufacturer**	**Package**	**Model Name**
DATA I/O Corp.	121B	Hitachi	FP-80A	HS444ESH01H
	29B			
			TFP-80F	HS4449ESN01H
AVAL Corp.	PKW-1000	Hitachi	FP-80A	HS444ESH01H
			TFP-80F	HS4449ESN01H

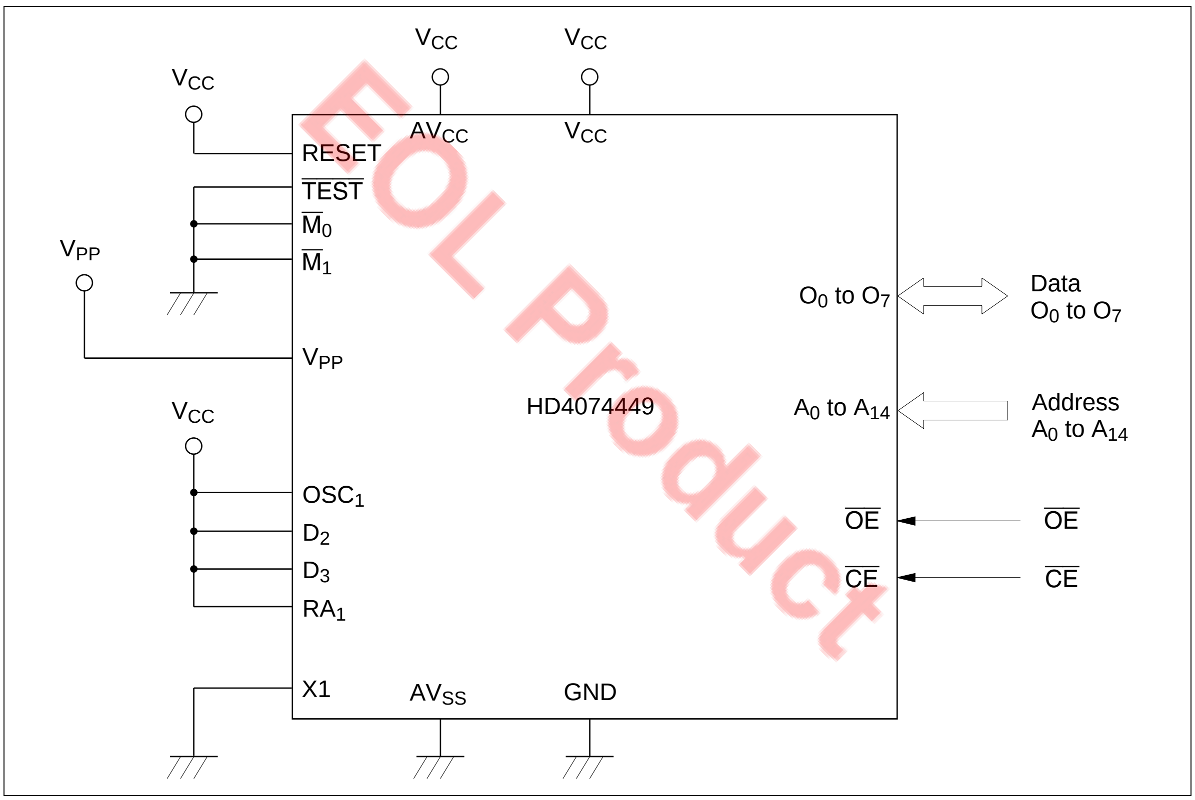


Figure 89 PROM Mode Connections

Addressing Modes

RAM Addressing Modes

The MCU has three RAM addressing modes, as shown in figure 90 and described below.

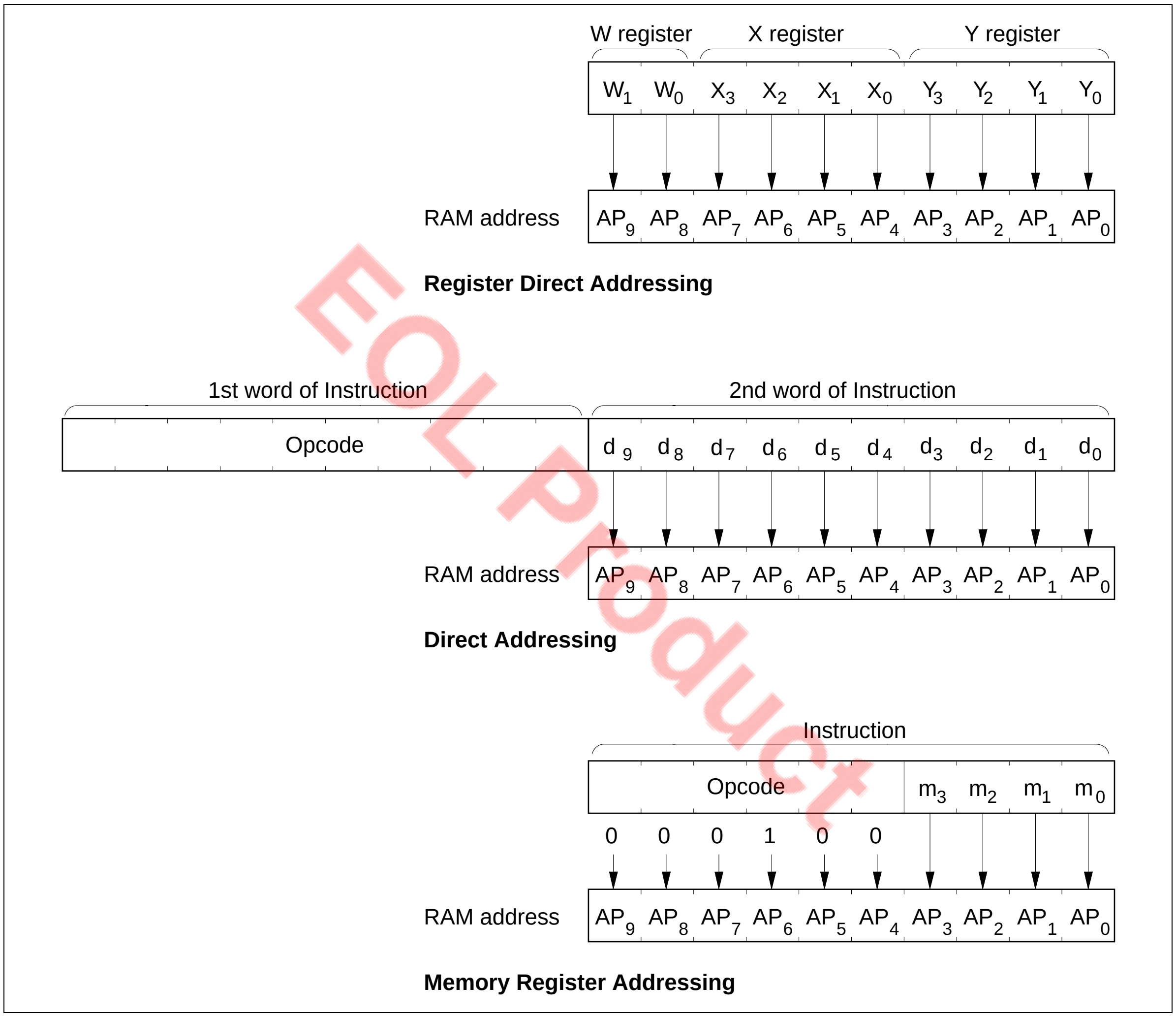


Figure 90 RAM Addressing Modes

Register Indirect Addressing Mode: The contents of the W, X, and Y registers (10 bits in total) are used as a RAM address. When the area from \$090 to \$25F is used, a bank must be selected by the bank register (V: \$03F).

Direct Addressing Mode: A direct addressing instruction consists of two words. The first word contains the opcode, and the contents of the second word (10 bits) are used as a RAM address.

Memory Register Addressing Mode: The memory registers (MR), which are located in 16 addresses from \$040 to \$04F, are accessed with the LAMR and XMRA instructions.

ROM Addressing Modes and the P Instruction

The MCU has four ROM addressing modes, as shown in figure 91 and described below.

Direct Addressing Mode: A program can branch to any address in the ROM memory space by executing the JMPL, BRL, or CALL instruction. Each of these instructions replaces the 14 program counter bits (PC_{13}–PC_0) with 14-bit immediate data.

Current Page Addressing Mode: The MCU has 64 pages of ROM with 256 words per page. A program can branch to any address in the current page by executing the BR instruction. This instruction replaces the eight low-order bits of the program counter (PC_7–PC_0) with eight-bit immediate data. If the BR instruction is on a page boundary (address 256n + 255), executing that instruction transfers the PC contents to the next physical page, as shown in figure 93. This means that the execution of the BR instruction on a page boundary will make the program branch to the next page.

Note that the HMCS400-series cross macroassembler has an automatic paging feature for ROM pages.

Zero-Page Addressing Mode: A program can branch to the zero-page subroutine area located at \$0000–\$003F by executing the CAL instruction. When the CAL instruction is executed, 6 bits of immediate data are placed in the six low-order bits of the program counter (PC_5–PC_0), and 0s are placed in the eight high-order bits (PC_{13}–PC_6).

Table Data Addressing Mode: A program can branch to an address determined by the contents of four-bit immediate data, the accumulator, and the B register by executing the TBR instruction.

P Instruction: ROM data addressed in table data addressing mode can be referenced with the P instruction as shown in figure 92. If bit 8 of the ROM data is 1, eight bits of ROM data are written to the accumulator and the B register. If bit 9 is 1, eight bits of ROM data are written to the R1 and R2 port output registers. If both bits 8 and 9 are 1, ROM data is written to the accumulator and the B register, and also to the R1 and R2 port output registers at the same time.

The P instruction has no effect on the program counter.

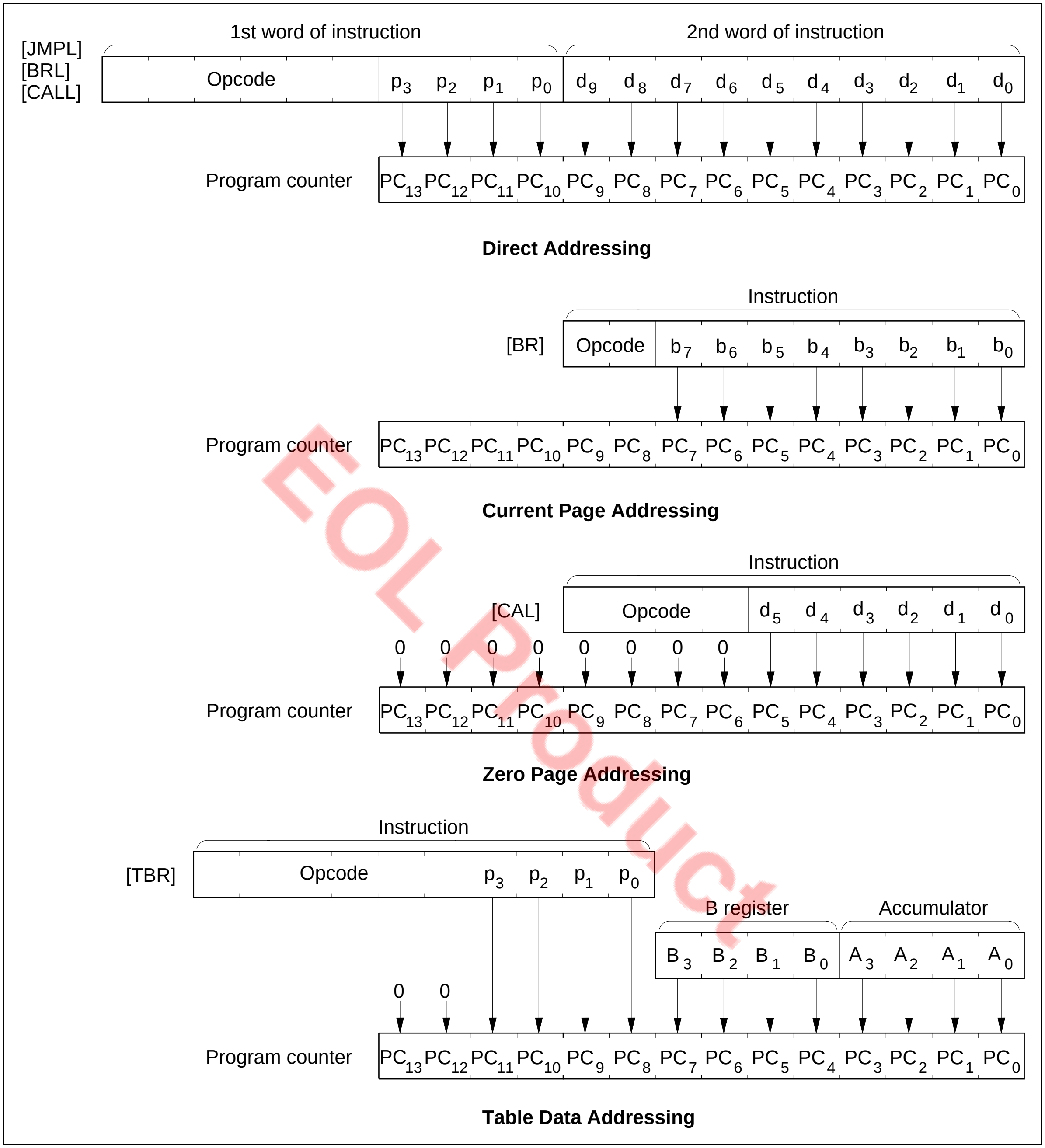


Figure 91 ROM Addressing Modes

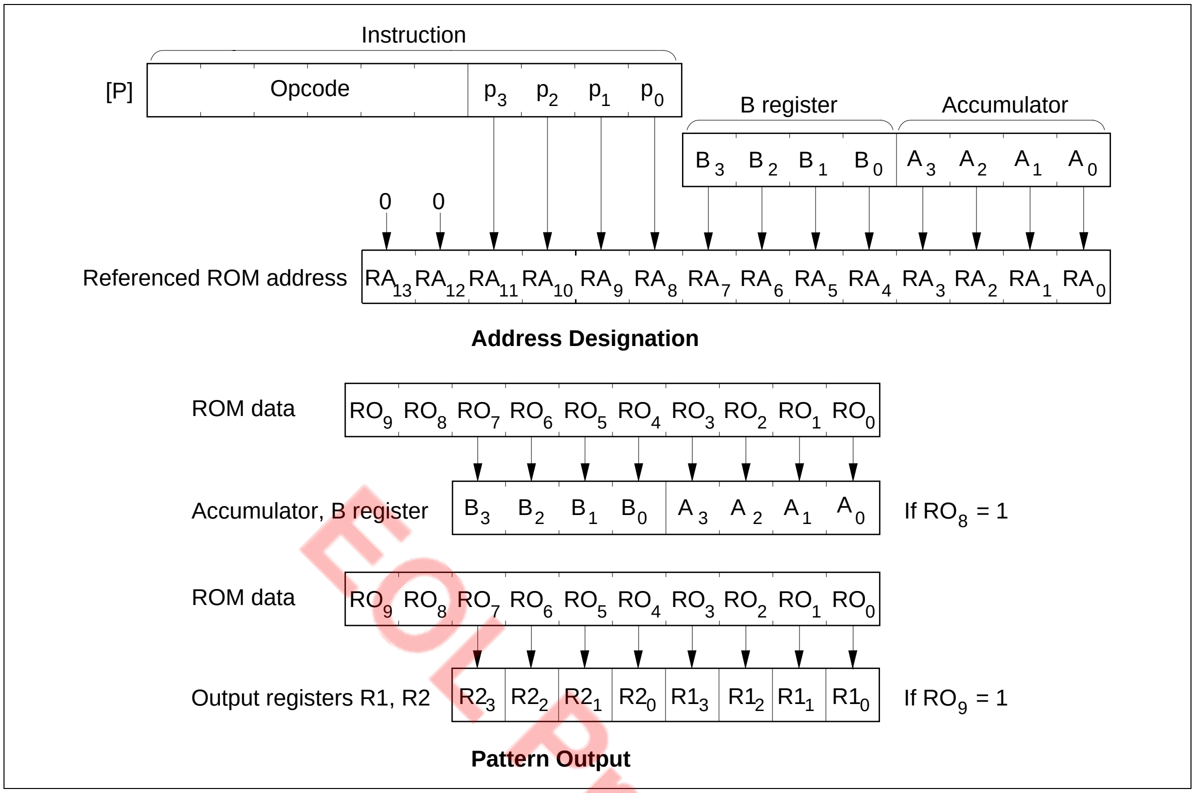


Figure 92 P Instruction

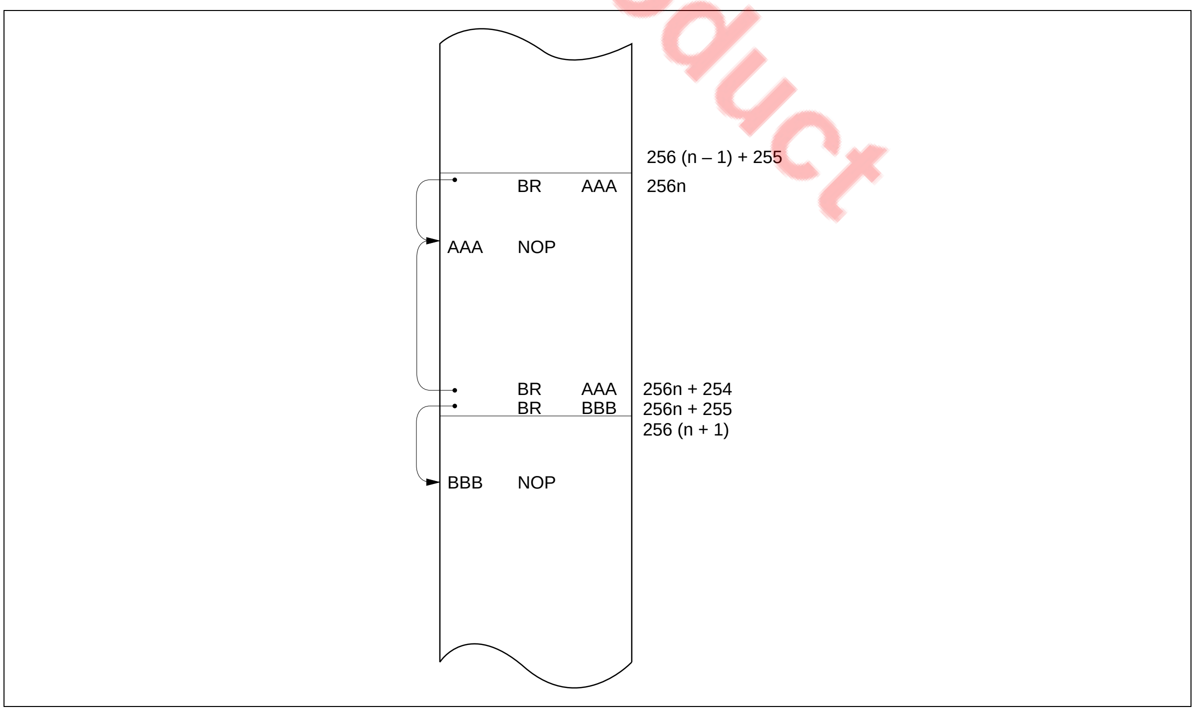


Figure 93 Branching when the Branch Destination is on a Page Boundary

Absolute Maximum Ratings

Item	Symbol	Value	Unit	Notes
Supply voltage	V_{CC}	–0.3 to +7.0	V	
Programming voltage	V_{PP}	–0.3 to +14.0	V	1
Pin voltage	V_T	–0.3 to (V_{CC} + 0.3)	V	
Total permissible input current	ΣI_o	100	mA	2
Total permissible output current	$-\Sigma I_o$	50	mA	3
Maximum input current	I_o	4	mA	4, 5
		30	mA	4, 6
Maximum output current	$-I_o$	4	mA	7, 8
Operating temperature	T_{opr}	–20 to +75	°C	
Storage temperature	T_{stg}	–55 to +125	°C	

Notes: Permanent damage may occur if these absolute maximum ratings are exceeded. Normal opera-tion must be under the conditions stated in the electrical characteristics tables. If these conditions are exceeded, the LSI may malfunction or its reliability may be affected.

1. Applies to D_{13} (V_{PP}) of the HD4074449.
2. The total permissible input current is the total of input currents simultaneously flowing in from all the I/O pins to ground.
3. The total permissible output current is the total of output currents simultaneously flowing out from V_{CC} to all I/O pins.
4. The maximum input current is the maximum current flowing from each I/O pin to ground.
5. Applies to D_{10}, D_{11}, and R0–RC.
6. Applies to D_0–D_9.
7. The maximum output current is the maximum current flowing out from V_{CC} to each I/O pin.
8. Applies to D_0–D_{11} and R0–RC.

Electrical Characteristics

DC Characteristics (HD404448, HD404449: V_{CC} = 2.7 to 6.0 V, GND = 0 V, T_a = –20°C to +75°C; HD4074449: V_{CC} = 2.7 to 5.5 V, GND = 0 V, T_a = –20°C to +75°C, unless otherwise specified)

Item	Symbol	Pin(s)	Min	Typ	Max	Unit	Test Condition	Notes
Input high voltage	V_{IH}	RESET, $\overline{\text{STOPC}}$, $\overline{\text{INT}_0}$, $\overline{\text{INT}_1}$, INT_2, INT_3, $\overline{\text{SCK}_1}$, SI_1, $\overline{\text{SCK}_2}$, SI_2, $\overline{\text{EVNB}}$, EVND	$0.9V_{CC}$	—	V_{CC} + 0.3	V	—	
		OSC_1	V_{CC} – 0.3	—	V_{CC} + 0.3	V	External clock operation	
Input low voltage	V_{IL}	RESET, $\overline{\text{STOPC}}$, $\overline{\text{INT}_0}$, $\overline{\text{INT}_1}$, INT_2, INT_3, $\overline{\text{SCK}_1}$, SI_1, $\overline{\text{SCK}_2}$, SI_2, $\overline{\text{EVNB}}$, EVND	–0.3	—	$0.1V_{CC}$	V	—	
		OSC_1	–0.3	—	0.3	V	External clock operation	
Output high voltage	V_{OH}	$\overline{\text{SCK}_1}$, SO_1, $\overline{\text{SCK}_2}$, SO_2, TOB, TOC, TOD	V_{CC} – 1.0	—	—	V	$–I_{OH}$ = 0.5 mA	
Output low voltage	V_{OL}	$\overline{\text{SCK}_1}$, SO_1, $\overline{\text{SCK}_2}$, SO_2, TOB, TOC, TOD	—	—	0.4	V	I_{OL} = 0.4 mA	
I/O leakage current	$\lvert I_{IL} \rvert$	RESET, $\overline{\text{STOPC}}$, $\overline{\text{INT}_0}$, $\overline{\text{INT}_1}$, INT_2, INT_3, $\overline{\text{SCK}_1}$, SI_1, $\overline{\text{SCK}_2}$, SI_2, SO_1, SO_2, $\overline{\text{EVNB}}$, EVND, OSC_1, TOB, TOC, TOD	—	—	1.0	µA	V_{in} = 0 V to V_{CC}	1
Current dissipation in active mode	I_{CC1}	V_{CC}	—	5	9	mA	V_{CC} = 5.0 V, f_{OSC} = 4 MHz	2
	I_{CC2}	V_{CC}	—	0.6	1.8	mA	V_{CC} = 3.0 V, f_{OSC} = 800 kHz	2

Item	Symbol	Pin(s)	Min	Typ	Max	Unit	Test Condition	Notes
Current dissipation in standby mode	I_{SBY1}	V_{CC}	—	1.2	3	mA	V_{CC} = 5.0 V, f_{OSC} = 4 MHz	3
	I_{SBY2}	V_{CC}	—	0.2	0.7	mA	V_{CC} = 3.0 V, f_{OSC} = 800 kHz	3
Current dissipation in subactive mode	I_{SUB}	V_{CC}	—	35	70	µA	V_{CC} = 3.0 V, 32-kHz oscillator	4
			—	70	150	µA	V_{CC} = 3.0 V, 32-kHz oscillator	5
Current dissipation in watch mode	I_{WTC}	V_{CC}	—	8	15	µA	V_{CC} = 3.0 V, 32-kHz oscillator	6
Current dissipation in stop mode	I_{STOP}	V_{CC}	—	1	10	µA	V_{CC} = 3.0 V, no 32-kHz oscillator	6
Stop mode retaining voltage	V_{STOP}	V_{CC}	2	—	—	V	No 32-kHz oscillator	7

Notes: 1. Output buffer current is excluded.

2. I_{CC1} and I_{CC2} are the source currents when no I/O current is flowing while the MCU is in reset state.

   Test conditions: MCU: Reset

   Pins: RESET at V_{CC} (V_{CC} – 0.3 V to V_{CC})

   $\overline{TEST}$ at V_{CC} (V_{CC} – 0.3 V to V_{CC})

3. I_{SBY1} and I_{SBY2} are the source currents when no I/O current is flowing while the MCU timer is operating.

   Test conditions: MCU: I/O reset

   Serial interface stopped

   Standby mode

   Pins: RESET at GND (0 V to 0.3 V)

   $\overline{TEST}$ at V_{CC} (V_{CC} – 0.3 V to V_{CC})

4. Applies to HD404448 and HD404449.
5. Applies to HD4074449.
6. These are the source currents when no I/O current is flowing.

   Test conditions: Pins: RESET at GND (0 V to 0.3 V)

   $\overline{TEST}$ at V_{CC} (V_{CC} – 0.3 V to V_{CC})

   D_{13} (V_{PP}) at V_{CC} (V_{CC} – 0.3 V to V_{CC}) for the HD4074449

7. RAM data retention.

I/O Characteristics for Standard Pins (HD404448, HD404449: V_{CC} = 2.7 to 6.0 V, GND = 0 V, T_a = –20°C to +75°C; HD4074449: V_{CC} = 2.7 to 5.5 V, GND = 0 V, T_a = –20°C to +75°C, unless otherwise specified)

Item	Symbol	Pin(s)	Min	Typ	Max	Unit	Test Condition	Notes
Input high voltage	V_{IH}	D_{10}–D_{13}, R0–RC	$0.7V_{CC}$	—	V_{CC} + 0.3	V	—	
Input low voltage	V_{IL}	D_{10}–D_{13}, R0–RC	–0.3	—	$0.3V_{CC}$	V	—	
Output high voltage	V_{OH}	D_{10}, D_{11}, R0–RC	V_{CC} – 1.0	—	—	V	$-I_{OH}$ = 0.5 mA	
Output low voltage	V_{OL}	D_{10}, D_{11}, R0–RC	—	—	0.4	V	I_{OL} = 0.4 mA	
I/O leakage current	$\lvert I_{IL} \rvert$	D_{10}–D_{13}, R0–RC	—	—	1	µA	V_{in} = 0 V to V_{CC}	1, 2
		D_{10}–D_{12}, R0–RC	—	—	1	µA	V_{in} = 0 V to V_{CC}	1, 3
		D_{13}	—	—	1	µA	V_{in} = V_{CC} – 0.3 V to V_{CC}	1, 3
		D_{13}	—	—	20	µA	V_{in} = 0 V to 0.3 V	1, 3
Pull-up MOS current	$-I_{PU}$	D_{10}, D_{11}, R0–RC	5	30	90	µA	V_{CC} = 3.0 V, V_{in} = 0 V	

Notes: 1. Output buffer current is excluded.
2. Applies to HD404448 and HD404449.
3. Applies to HD4074449.

I/O Characteristics for High-Current Pins (HD404448, HD404449: V_{CC} = 2.7 to 6.0 V, GND = 0 V, T_a = –20°C to +75°C; HD4074449: V_{CC} = 2.7 to 5.5 V, GND = 0 V, T_a = –20°C to +75°C, unless otherwise specified)

Item	Symbol	Pin(s)	Min	Typ	Max	Unit	Test Condition	Notes
Input high voltage	V_{IH}	D_0–D_9	$0.7V_{CC}$	—	V_{CC} + 0.3	V	—	
Input low voltage	V_{IL}	D_0–D_9	–0.3	—	$0.3V_{CC}$	V	—	
Output high voltage	V_{OH}	D_0–D_9	V_{CC} – 1.0	—	—	V	$-I_{OH}$ = 0.5 mA	
Output low voltage	V_{OL}	D_0–D_9	—	—	0.4	V	I_{OL} = 0.4 mA	
			—	—	2.0	V	I_{OL} = 15 mA, $V_{CC} \geq$ 4.5 V	
I/O leakage current	$\lvert I_{IL} \rvert$	D_0–D_9	—	—	1	µA	V_{in} = 0 V to V_{CC}	1
Pull-up MOS current	$-I_{PU}$	D_0–D_9	5	30	90	µA	V_{CC} = 3.0 V, V_{in} = 0 V	

Notes: 1. Output buffer current is excluded.

A/D Converter Characteristics (HD404448, HD404449: V_{CC} = 2.7 to 6.0 V, GND = 0 V, T_a = –20°C to +75°C; HD4074449: V_{CC} = 2.7 to 5.5 V, GND = 0 V, T_a = –20°C to +75°C, unless otherwise specified)

Item	Symbol	Pin(s)	Min	Typ	Max	Unit	Test Condition	Notes
Analog power voltage	AV_{CC}	AV_{CC}	V_{CC} – 0.3	V_{CC}	V_{CC} + 0.3	V	—	1
Analog input voltage	AV_{in}	AN_0–AN_3	AV_{SS}	—	AV_{CC}	V	—	
Current between AV_{CC} and AV_{SS}	I_{AD}	—	—	50	150	µA	$V_{CC} = AV_{CC}$ = 5.0 V	
Analog input capacitance	CA_{in}	AN_0–AN_3	—	15	—	pF	—	
Resolution	—	—	8	8	8	Bit		
Number of inputs	—	—	0	—	4	Channel	—	
Absolute accuracy	—	—	—	—	± 2.0	LSB	T_a = 25°C, V_{CC} = 4.5 V to 5.5 V	
Conversion time	—	—	34	—	67	t_{cyc}	—	
Input impedance	—	AN_0–AN_3	1	—	—	MΩ	f_{OSC} = 1 MHz, V_{in} = 0 V	

Note: 1. $AV_{CC} \geq$ 2.7 V

AC Characteristics (HD404448, HD404449: V_{CC} = 2.7 to 6.0 V, GND = 0 V, T_a = –20°C to +75°C; HD4074449: V_{CC} = 2.7 to 5.5 V, GND = 0 V, T_a = –20°C to +75°C, unless otherwise specified)

Item	Symbol	Pin(s)	Min	Typ	Max	Unit	Test Condition	Notes
Clock oscillation frequency	f_{OSC}	OSC_1, OSC_2	0.4		4.0	MHz	1/4 division	1
		X1, X2	—	32.768	—	kHz	—	
Instruction cycle time	t_{cyc}	—	1.0	—	10	µs	—	1
	t_{subcyc}	—	—	244.14	—	µs	32-kHz oscillator, 1/8 division	
			—	122.07	—	µs	32-kHz oscillator, 1/4 division	
Oscillation stabilization time (ceramic)	t_{RC}	OSC_1, OSC_2	—	—	7.5	ms	—	2
Oscillation stabilization time (crystal)	t_{RC}	OSC_1, OSC_2	—	—	40	ms	HD404448, HD404449 V_{CC}=3.0 to 6.0V HD4074449 V_{CC}=3.5 to 5.5V	2
			—	—	60	ms	—	2
		X1, X2	—	—	3	s	T_a = –10°C to +60°C	3
External clock high width	t_{CPH}	OSC_1	105	—	—	ns	—	4
External clock low width	t_{CPL}	OSC_1	105	—	—	ns	—	4
External clock rise time	t_{CPr}	OSC_1	—	—	20	ns	—	4
External clock fall time	t_{CPf}	OSC_1	—	—	20	ns	—	4
$\overline{INT_0}$–INT_3, $\overline{EVNB}$, EVND high widths	t_{IH}	$\overline{INT_0}$–INT_3, $\overline{EVNB}$, EVND	2	—	—	t_{cyc}/ t_{subcyc}	—	5
$\overline{INT_0}$–INT_3, $\overline{EVNB}$, EVND low widths	t_{IL}	$\overline{INT_0}$–INT_3, $\overline{EVNB}$, EVND	2	—	—	t_{cyc}/ t_{subcyc}	—	5
RESET high width	t_{RSTH}	RESET	2	—	—	t_{cyc}	—	6
$\overline{STOPC}$ low width	t_{STPL}	$\overline{STOPC}$	1	—	—	t_{RC}	—	7
RESET fall time	t_{RSTf}	RESET	—	—	20	ms	—	6
$\overline{STOPC}$ rise time	t_{STPr}	$\overline{STOPC}$	—	—	20	ms	—	7

Item	Symbol	Pin(s)	Min	Typ	Max	Unit	Test Condition	Notes
Input capacitance	C_{in}	All pins except D_{13}	—	—	15	pF	f = 1 MHz, V_{in} = 0 V	
		D_{13}	—	—	15	pF	HD404448, HD404449: f = 1 MHz, V_{in} = 0 V	
			—	—	180	pF	HD4074449: f = 1 MHz, V_{in} = 0 V	

Notes: 1. If the 32.768-kHz oscillator is used for the subsystem oscillator, f_{OSC} must be set as 0.4 MHz ≤ f_{OSC} ≤ 1.0 MHz or 1.6 MHz ≤ f_{OSC} ≤ 4.0 MHz, and bit 1 of the system clock selector register (SSR: $029) must be set to 0 or 1, respectively.

2. The oscillation stabilization time is the period required for the oscillator to stabilize after V_{CC} reaches 2.7 V at power-on, or after RESET input goes high or $\overline{STOPC}$ input goes low when stop mode is cancelled. At power-on or when stop mode is cancelled, RESET or $\overline{STOPC}$ must be input for at least t_{RC} to ensure the oscillation stabilization time. If using a ceramic oscillator, contact its manufacturer to determine what stabilization time is required, since it will depend on the circuit constants and stray capacitances. Set bits 0 and 1 (MIS0, MIS1) of the miscellaneous register (MIS: $00C) according to the system oscillation of the oscillation stabilization time.

3. The oscillation stabilization time is the period required for the oscillator to stabilize after V_{CC} reaches 2.7 V at power-on, or after RESET input goes high or $\overline{STOPC}$ input goes low when the 32-kHz oscillator stops in stop mode and stop mode is cancelled. If using a crystal oscillator, contact its manufacturer to determine what stabilization time is required, since it will depend on the circuit constants and stray capacitances.

4. Refer to figure 94.

5. Refer to figure 95. The t_{cyc} unit applies when the MCU is in standby or active mode. The t_{subcyc} unit applies when the MCU is in watch or subactive mode.

6. Refer to figure 96.

7. Refer to figure 97.

Serial Interface Timing Characteristics (HD404448, HD404449: V_{CC} = 2.7 to 6.0 V, GND = 0 V, T_a = –20°C to +75°C; HD4074449: V_{CC} = 2.7 to 5.5 V, GND = 0 V, T_a = –20°C to +75°C, unless otherwise specified)

During Transmit Clock Output

Item	Symbol	Pin	Min	Typ	Max	Unit	Test Condition	Note
Transmit clock cycle time	t_{Scyc}	$\overline{SCK_1}$, $\overline{SCK_2}$	1.0	—	—	t_{cyc}	Load shown in figure 99	1
Transmit clock high width	t_{SCKH}	$\overline{SCK_1}$, $\overline{SCK_2}$	0.5	—	—	t_{Scyc}	Load shown in figure 99	1
Transmit clock low width	t_{SCKL}	$\overline{SCK_1}$, $\overline{SCK_2}$	0.5	—	—	t_{Scyc}	Load shown in figure 99	1
Transmit clock rise time	t_{SCKr}	$\overline{SCK_1}$, $\overline{SCK_2}$	—	—	200	ns	Load shown in figure 99	1
Transmit clock fall time	t_{SCKf}	$\overline{SCK_1}$, $\overline{SCK_2}$	—	—	200	ns	Load shown in figure 99	1
Serial output data delay time	t_{DSO}	SO_1, SO_2	—	—	500	ns	Load shown in figure 99	1
Serial input data setup time	t_{SSI}	SI_1, SI_2	300	—	—	ns	—	1
Serial input data hold time	t_{HSI}	SI_1, SI_2	300	—	—	ns	—	1

Note: 1. Refer to figure 98.

During Transmit Clock Input

Item	Symbol	Pin	Min	Typ	Max	Unit	Test Condition	Note
Transmit clock cycle time	t_{Scyc}	$\overline{SCK_1}$, $\overline{SCK_2}$	1.0	—	—	t_{cyc}	—	1
Transmit clock high width	t_{SCKH}	$\overline{SCK_1}$, $\overline{SCK_2}$	0.5	—	—	t_{Scyc}	—	1
Transmit clock low width	t_{SCKL}	$\overline{SCK_1}$, $\overline{SCK_2}$	0.5	—	—	t_{Scyc}	—	1
Transmit clock rise time	t_{SCKr}	$\overline{SCK_1}$, $\overline{SCK_2}$	—	—	200	ns	—	1
Transmit clock fall time	t_{SCKf}	$\overline{SCK_1}$, $\overline{SCK_2}$	—	—	200	ns	—	1
Serial output data delay time	t_{DSO}	SO_1, SO_2	—	—	500	ns	Load shown in figure 99	1
Serial input data setup time	t_{SSI}	SI_1, SI_2	300	—	—	ns	—	1
Serial input data hold time	t_{HSI}	SI_1, SI_2	300	—	—	ns	—	1

Note: 1. Refer to figure 98.

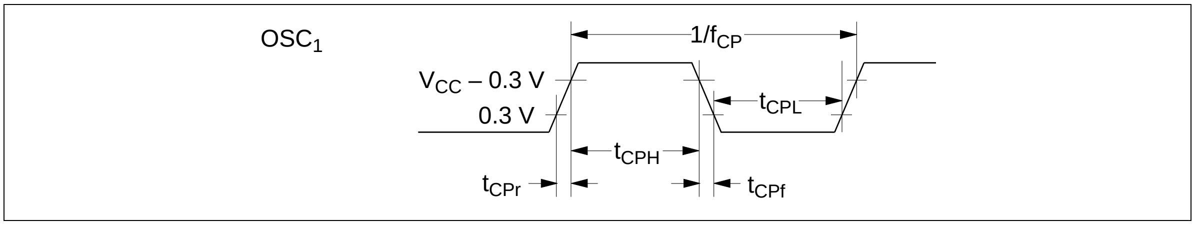


Figure 94 External Clock Timing

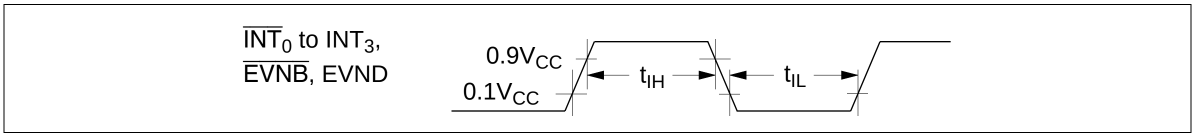


Figure 95 Interrupt Timing

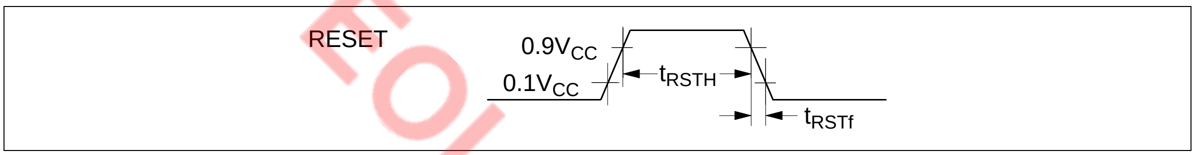


Figure 96 Reset Timing

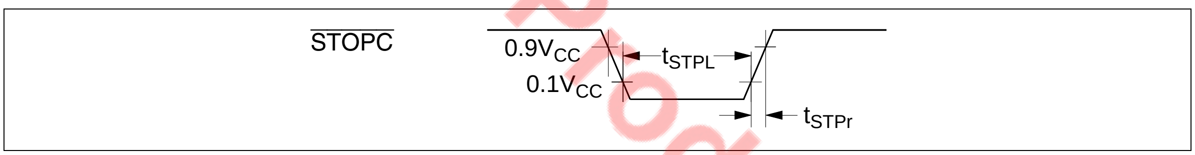
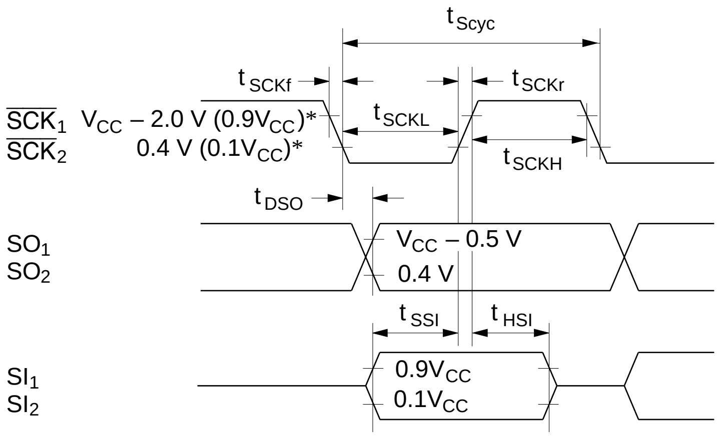


Figure 97 $\overline{\text{STOPC}}$ Timing



Note: * V_{CC} – 2.0 V and 0.4 V are the threshold voltages for transmit clock output, and 0.9V_{CC} and 0.1V_{CC} are the threshold voltages for transmit clock input.

Figure 98 Serial Interface Timing

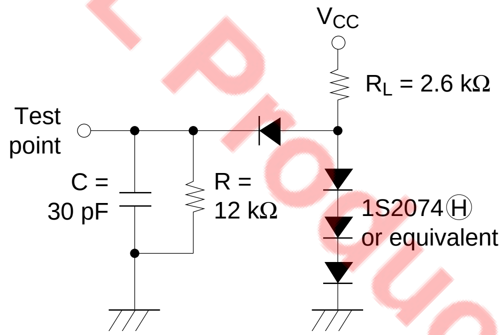


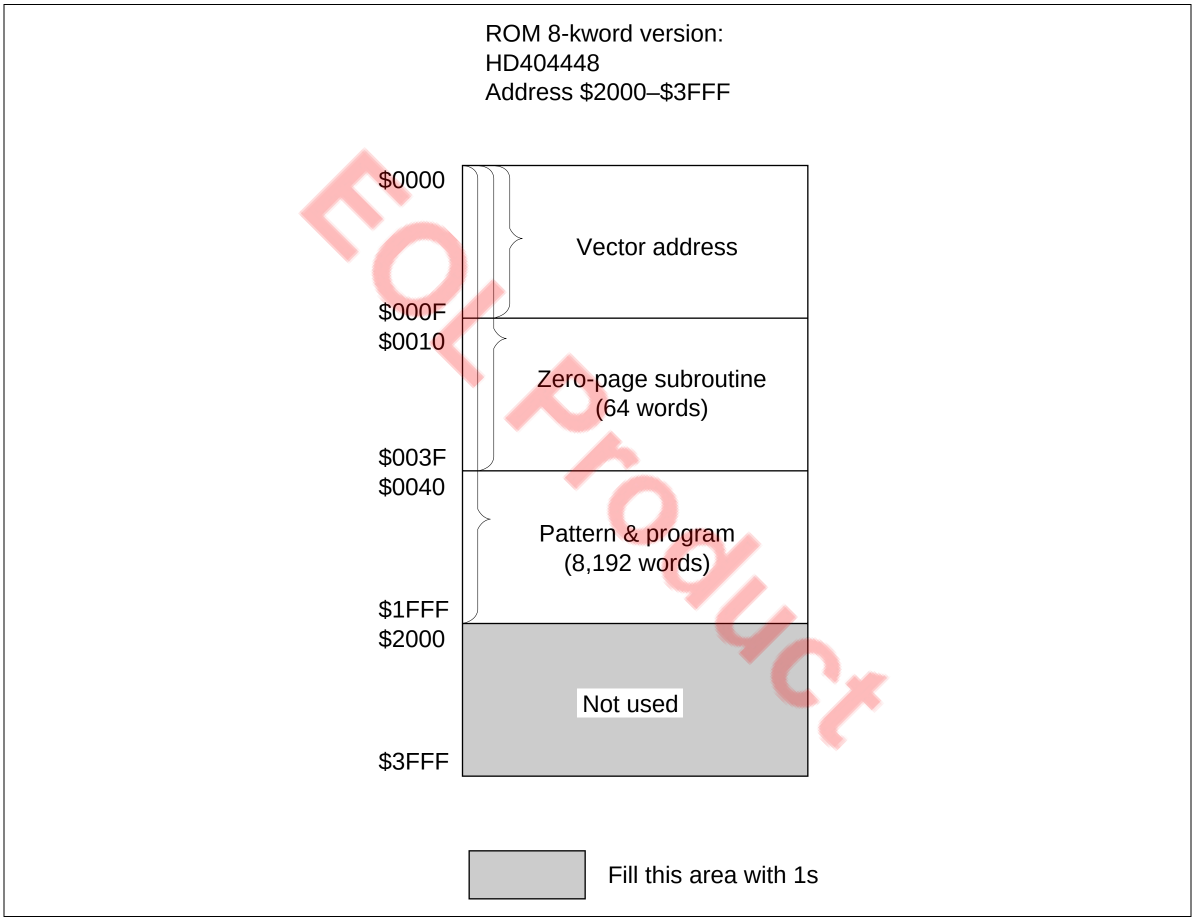
Figure 99 Timing Load Circuit

Notes on ROM Out

Please pay attention to the following items regarding ROM out.

On ROM out, fill the ROM area indicated below with 1s to create the same data size as a 16-kword version (HD404449). A 16-kword data size is required to change ROM data to mask manufacturing data since the program used is for a16-kword version.

This limitation applies when using an EPROM or a data base.



HD404448, HD404449 Option List

Please check off the appropriate applications and enter the necessary information.

Date of order	
Customer	
Department	
Name	
ROM code name	
LSI number	HD40444

1. ROM size

☐ HD404448	8-kword
☐ HD404449	16-kword

2. Optional Functions

* ☐ With 32-kHz CPU operation, with time-base for clock
* ☐ Without 32-kHz CPU operation, with time-base for clock
☐ Without 32-kHz CPU operation, without time-base

Note: * Options marked with an asterisk require a subsystem crystal oscillator (X1, X2).

3. ROM code media

Please specify the first type below (the upper bits and lower bits are mixed together), when using the EPROM on-package microcomputer type (including ZTAT™ version).

☐ EPROM: The upper bits and lower bits are mixed together. The upper five bits and lower five bits are programmed to the same EPROM in alternating order (i.e., LULULU...).
☐ EPROM: The upper bits and lower bits are separated. The upper five bits and lower five bits are programmed to different EPROMS.

4. Oscillator for OSC1 and OSC2

☐ Ceramic oscillator	f = MHz
☐ Crystal oscillator	f = MHz
☐ External clock	f = MHz

5. Stop mode

☐ Used
☐ Not used

6. Package

☐ FP-80A
☐ TFP-80F

Cautions

1. Hitachi neither warrants nor grants licenses of any rights of Hitachi's or any third party's patent, copyright, trademark, or other intellectual property rights for information contained in this document. Hitachi bears no responsibility for problems that may arise with third party's rights, including intellectual property rights, in connection with use of the information contained in this document.
2. Products and product specifications may be subject to change without notice. Confirm that you have received the latest product standards or specifications before final design, purchase or use.
3. Hitachi makes every attempt to ensure that its products are of high quality and reliability. However, contact Hitachi's sales office before using the product in an application that demands especially high quality and reliability or where its failure or malfunction may directly threaten human life or cause risk of bodily injury, such as aerospace, aeronautics, nuclear power, combustion control, transportation, traffic, safety equipment or medical equipment for life support.
4. Design your application so that the product is used within the ranges guaranteed by Hitachi particularly for maximum rating, operating supply voltage range, heat radiation characteristics, installation conditions and other characteristics. Hitachi bears no responsibility for failure or damage when used beyond the guaranteed ranges. Even within the guaranteed ranges, consider normally foreseeable failure rates or failure modes in semiconductor devices and employ systemic measures such as fail-safes, so that the equipment incorporating Hitachi product does not cause bodily injury, fire or other consequential damage due to operation of the Hitachi product.
5. This product is not designed to be radiation resistant.
6. No one is permitted to reproduce or duplicate, in any form, the whole or part of this document without written approval from Hitachi.
7. Contact Hitachi's sales office for any questions regarding this document or Hitachi semiconductor products.