PRELIMINARY PRODUCT INFORMATION

NEC

MOS INTEGRATED CIRCUITS

μPD789144,789146,789154,789156

8-BIT SINGLE-CHIP MICROCONTROLLER

The μPD789146 and μPD789156 sub-series[Note] products are 8-bit single-chip microcontrollers of the 78K/0S series.

These microcontrollers feature an 8-bit CPU and hardware that includes an EEPROM™ that can be read or written by a program, and an A/D converter.

In addition, a flash memory product (μPD78F9156) that can operate within the same voltage range as the masked ROM models, and a range of related development tools are being developed.

The functions of these microcontrollers are described in the following user's manuals. Refer to these manuals when designing a system based on any of these microcontrollers.

μPD789146, 789156 Sub-Series User's Manual : To be created
78K/0S Series User's Manual, Instruction : U11047E

Note The μPD789144 and μPD789146 are referred to as the μPD789146 sub-series. The μPD789154 and μPD789156 are referred to as the μPD789156 sub-series.

FEATURES

- Built-in EEPROM
- ROM and RAM sizes

Item / Product name	Program memory (ROM)	Data memory	
		High-speed RAM	EEPROM
μPD789144, μPD789154	8 Kbytes	256 bytes	256 bytes
μPD789146, μPD789156	16 Kbytes		

- Variable minimum instruction execution time: From high speed (0.4 μs) to low speed (1.6 μs) at a system clock frequency of 5.0 MHz
- 20 I/O ports
- • Four A/D converters with an 8-bit resolution (μPD789146 sub-series)
  • Four A/D converters with a 10-bit resolution (μPD789156 sub-series)
- Three timers:
  - 16-bit timer counter
  - 8-bit timer/event counter
  - Watchdog timer
- Serial interface:
  Switchable between three-wire serial I/O and UART modes
- Multiplier: 8 × 8 bits = 16 bits
- Low-voltage indicator
- Power supply voltage: V_{DD} = 1.8 to 5.5 V

APPLICATIONS

Motor control equipment, electrical equipment for automobiles, and battery chargers

The information contained in this document is being issued in advance of the production cycle for the device. The parameters for the device may change before final production or NEC Corporation, at its own discretion, may withdraw the device prior to its production.

Document No. U13478EJ1V0PM00 (1st edition)

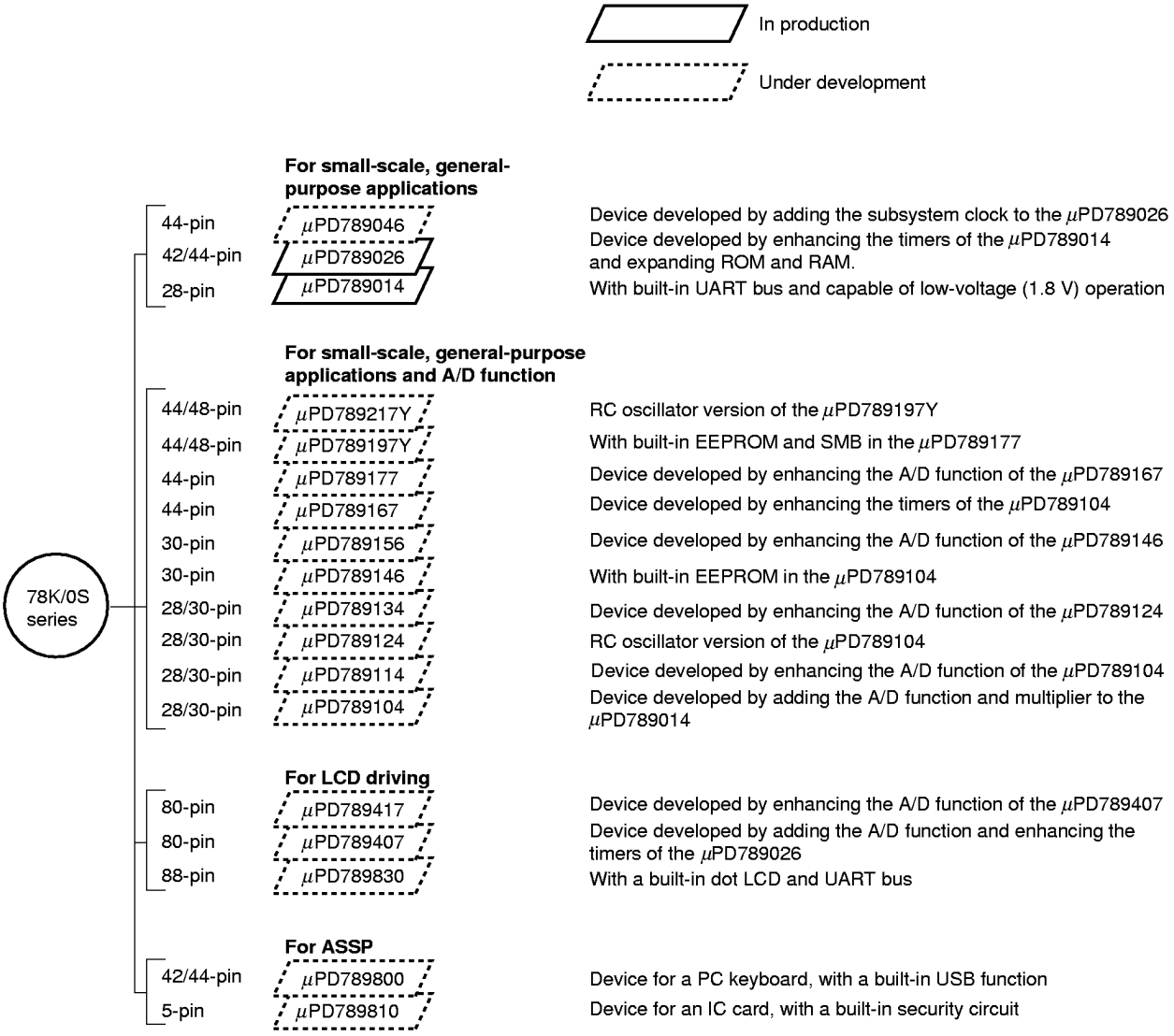
ORDERING INFORMATION

Part number	Package
μPD789144GS-×××	30-pin plastic shrink SOP (300 mil)
μPD789146GS-×××	30-pin plastic shrink SOP (300 mil)
μPD789154GS-×××	30-pin plastic shrink SOP (300 mil)
μPD789156GS-×××	30-pin plastic shrink SOP (300 mil)

Remark ××× indicates ROM code suffix.

78K/0S SERIES DEVELOPMENT

The 78K/0S series products are shown below. The sub-series names are indicated in frames.



The following table lists the major differences in functions between the sub-series.

Sub-series \ Function		ROM size	Timer				8-bit A/D	10-bit A/D	Serial interface	I/O	Minimum V_{DD} value	Remarks
			8-bit	16-bit	Clock	WDT						
Small-scale, general-purpose applications	μPD789046	16 K	1 ch	1 ch	1 ch	1 ch	-	-	1 ch (UART: 1 ch)	34 pins	1.8 V	-
	μPD789026	4 K-16 K			-							
	μPD789014	2 K-4 K	2 ch	-						22 pins		
Small-scale, general-purpose applications and A/D function	μPD789217Y	16 K-24 K	3 ch	1 ch	1 ch	1 ch	-	8 ch	2 ch [UART: 1 ch, SMB: 1 ch]	31 pins	1.8 V	RC-oscillator version, with built-in EEPROM
	μPD789197Y											With built-in EEPROIM
	μPD789177								1 ch (UART: 1 ch)			-
	μPD789167						8 ch	-				
	μPD789156	8 K-16 K	1 ch		-		-	4 ch		20 pins		With built-in EEPROM
	μPD789146						4 ch	-				
	μPD789134	2 K-8 K					-	4 ch				RC-oscillation version
	μPD789124						4 ch	-				
	μPD789114						-	4 ch				-
	μPD789104						4 ch	-				
LCD driving	μPD789417	12 K-24 K	3 ch	1 ch	1 ch	1 ch	-	7 ch	1 ch (UART: 1 ch)	43 pins	1.8 V	-
	μPD789407						7 ch	-				
	μPD789830	24 K	1 ch				-			30 pins	2.7 V	
ASSP	μPD789800	8 K	2 ch	-	-	1 ch	-	-	2 ch (USB: 1 ch)	31 pins	4.0 V	-
	μPD789810	6 K	-						-	1 pin	1.8 V	With built-in EEPROM

FUNCTIONS

Item		μPD789144 μPD789154	μPD789146 μPD789156
Internal memory	ROM	8 Kbytes	16 Kbytes
	High-speed RAM	256 bytes	
	EEPROM	256 bytes	
Minimum instruction execution time		0.4/1.6 μs (operation with system clock running at 5.0 MHz)	
General-purpose registers		8 bits × 8 registers	
Instruction set		• 16-bit operations • Bit manipulations (such as set, reset, and test)	
Multiplier		8 bits × 8 bits = 16 bits	
I/O ports		Total of 20 port pins • 4 CMOS input pins • 12 CMOS input/output pins • 4 N-ch open-drain pins (12 V withstand voltage)	
A/D converters		• Four channels with 8-bit resolution (μPD789144, μPD789146) • Four channels with 10-bit resolution (μPD789154, μPD789156)	
Serial interface		• Switchable between three-wire serial I/O and UART modes	
Timers		• 16-bit timer counter • 8-bit timer/event counter • Watchdog timer	
Timer output		One output	
Vectored interrupt sources	Maskable	8 internal and 3 external interrupts	
	Nonmaskable	Internal interrupt	
Power supply voltage		V_{DD} = 1.8 to 5.5 V	
Operating ambient temperature		T_A = -40°C to+85°C	
Package		30-pin plastic shrink SOP (300 mil)	

CONTENTS

1. PIN CONFIGURATION (TOP VIEW) 8

2. BLOCK DIAGRAM 9

3. PIN FUNCTIONS 10

3.1 Port Pins 10
3.2 Non-Port Pins 11
3.3 Pin Input/Output Circuits and Handling of Unused Pins 12

4. CPU ARCHITECTURE 14

4.1 Memory Space 14
4.2 Data Memory Addressing 15
4.3 Processor Registers 16

5. EEPROM 20

5.1 EEPROM Functions 20
5.2 EEPROM Configuration 20
5.3 EEPROM Control Register 20
5.4 Cautions Related to EEPROM Writing 22

6. PERIPHERAL HARDWARE FUNCTIONS 24

6.1 Ports 24
6.2 Clock Generator 29
6.3 16-Bit Timer Counter 31
6.4 8-Bit Timer/Event Counter 37
6.5 Watchdog Timer 42
6.6 A/D Converter 45
6.7 Serial Interface 50
6.8 Multiplier 61
6.9 Low-Voltage Indicator 64

7. INTERRUPT FUNCTIONS 66

7.1 Interrupt Function Types 66
7.2 Interrupt Sources and Configuration 66
7.3 Interrupt Function Control Registers 69

8. STANDBY FUNCTION 75

8.1 Standby Function 75
8.2 Standby Function Control Register 77

9. RESET FUNCTIONS 78

10. MASK OPTIONS 81

11. INSTRUCTION SET OVERVIEW 82

11.1 Legend 82
11.2 Operations 84

12. ELECTRICAL CHARACTERISTICS 89

13. PACKAGE DRAWINGS 100

APPENDIX A DEVELOPMENT TOOLS 101

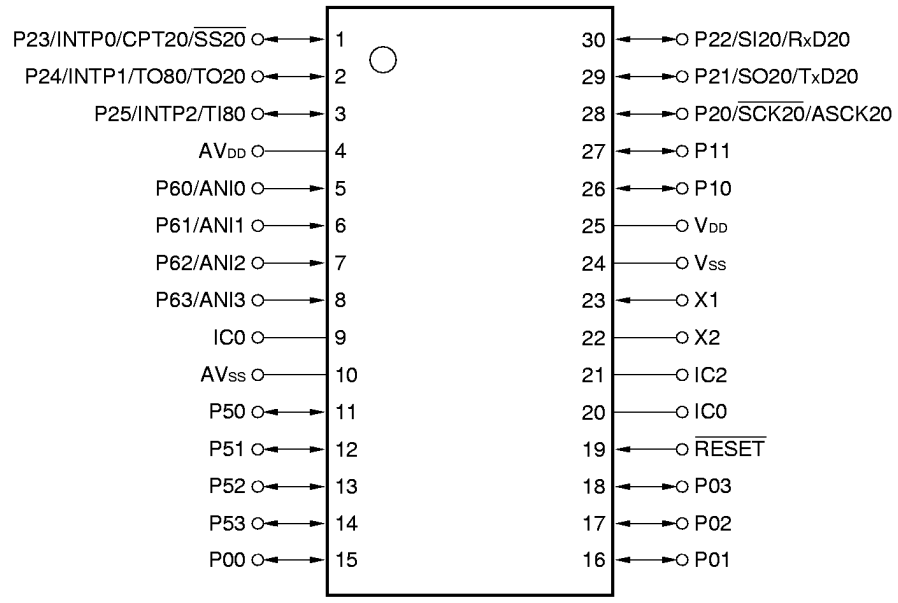
APPENDIX B RELATED DOCUMENTS 103

1. PIN CONFIGURATION (TOP VIEW)

- **30-pin plastic shrink SOP (300 mil)**

  μPD789144GS-××× μPD789154GS-×××

  μPD789146GS-××× μPD789156GS-×××



Cautions 1. Connect the IC0 (internally connected) pin directly to the V_{SS} pin.

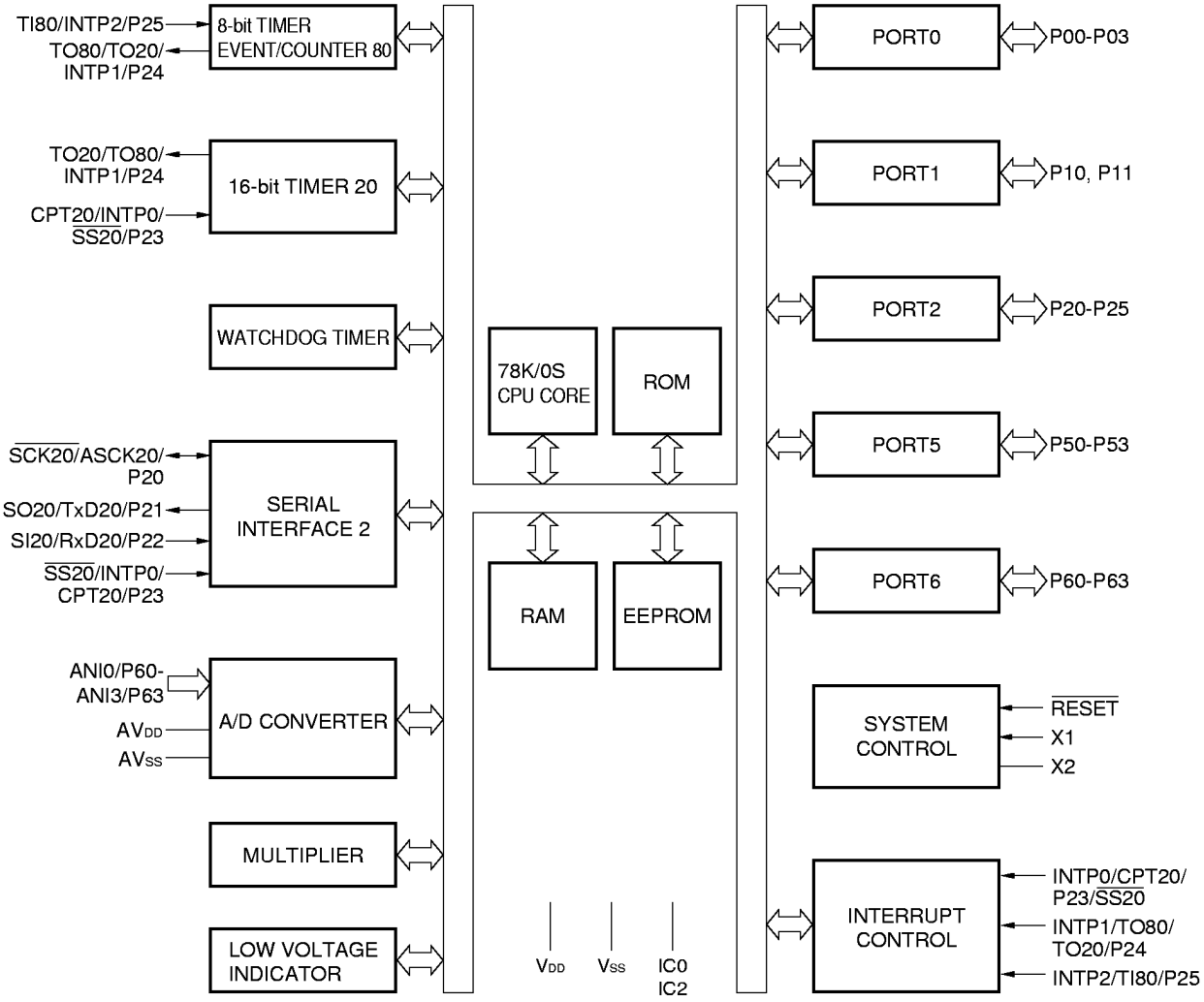
2. Leave the IC2 pin open.

3. Connect the AV_{DD} pin to the V_{DD} pin.

4. Connect the AV_{SS} pin to the V_{SS} pin.

ANI0-ANI3	: Analog Input	$\overline{\text{RESET}}$	: Reset
ASCK20	: Asynchronous Serial Input	RxD20	: Receive Data
AV_{DD}	: Analog Power Supply	$\overline{\text{SCK20}}$	: Serial Clock
AV_{SS}	: Analog Ground	SI20	: Serial Input
CPT20	: Capture Trigger Input	SO20	: Serial Output
IC0, IC2	: Internally Connected	$\overline{\text{SS20}}$	: Chip Select Input
INTP0-INTP2	: Interrupt from Peripherals	TI80	: Timer Input
P00-P03	: Port 0	TO20, TO80	: Timer Output
P10, P11	: Port 1	TxD20	: Transmit Data
P20-P25	: Port 2	V_{DD}	: Power Supply
P50-P53	: Port 5	V_{SS}	: Ground
P60-P63	: Port 6	X1, X2	: Crystal 1, 2

2. BLOCK DIAGRAM



Remark The size of the internal ROM varies depending on the model.

3. PIN FUNCTIONS

3.1 Port Pins

Pin name	I/O	Function	When reset	Also used as
P00-P03	I/O	Port 0 4-bit input/output port Can be set to either input or output in 1-bit units When used as an input port, whether the on-chip pull-up resistor is to be used can be specified by the setting of pull-up resistor option register 0 (PU0).	Input	-
P10, P11	I/O	Port 1 2-bit input/output port Can be set to either input or output in 1-bit units When used as an input port, whether the on-chip pull-up resistor is to be used can be specified by the setting of pull-up resistor option register 0 (PU0).	Input	-
P20	I/O	Port 2 6-bit input/output port Can be set to either input or output in 1-bit units Whether to use the on-chip pull-up resistor can be specified by the setting of pull-up resistor option register B2 (PUB2).	Input	$\overline{SCK20}$/ASCK20
P21				SO20/TxD20
P22				SI20/RxD20
P23				INTP0/CPT20/$\overline{SS20}$
P24				INTP1/TO80/TO20
P25				INTP2/TI80
P50-P53	I/O	Port 5 4-bit N-ch open-drain input/output port Can be set to either input or output in 1-bit units Whether to incorporate a pull-up resistor can be specified by a mask option.	Input	-
P60-P63	Input	Port 6 4-bit input-only port	Input	ANI0-ANI3

3.2 Non-Port Pins

Pin name	I/O	Function	When reset	Also used as
INTP0	Input	External interrupt input for which effective edges (rising and/or falling edges) can be specified	Input	P23/CPT20/$\overline{SS20}$
INTP1				P24/TO80/TO20
INTP2				P25/TI80
SI20	Input	Serial data input to serial interface	Input	P22/RxD20
SO20	Output	Serial data output from serial interface	Input	P21/TxD20
$\overline{SCK20}$	I/O	Serial clock input/output for serial interface	Input	P20/ASCK20
ASCK20	Input	Serial clock input to asynchronous serial interface	Input	P20/$\overline{SCK20}$
$\overline{SS20}$	Input	Chip select input to serial interface	Input	P23/CPT20/INTP0
RxD20	Input	Serial data input to asynchronous serial interface	Input	P22/SI20
TxD20	Output	Serial data output from asynchronous serial interface	Input	P21/SO20
TI80	Input	External count clock input to 8-bit timer (TM80)	Input	P25/INTP2
TO80	Output	8-bit timer (TM80) output	Input	P24/INTP1/TO20
TO20	Output	16-bit timer (TM20) output	Input	P24/INTP1/TO80
CPT20	Input	Capture edge input	Input	P23/INTP0/$\overline{SS20}$
ANI0-ANI3	Input	A/D converter analog input	Input	P60-P63
AV_{SS}	-	A/D converter ground potential	-	-
AV_{DD}	-	A/D converter analog power supply	-	-
X1	Input	Connected to crystal for system clock oscillation	-	-
X2	-		-	-
$\overline{RESET}$	Input	System reset input	Input	-
V_{DD}	-	Positive supply voltage	-	-
V_{SS}	-	Ground potential	-	-
IC0	-	Internally connected. Connect this pin directly to the V_{SS} pin.	-	-
IC2	-	Internally connected. Leave this pin open.	-	-

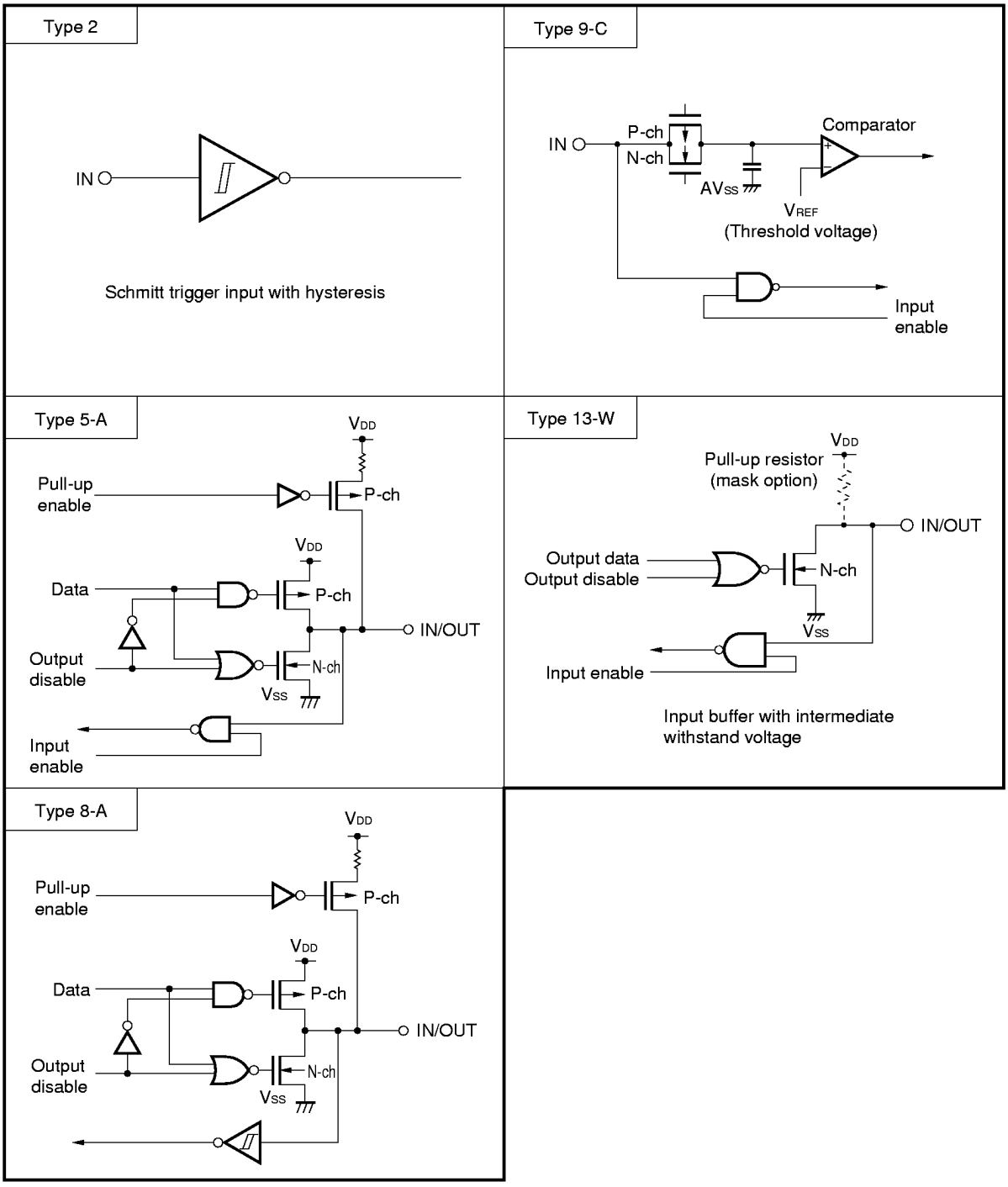
3.3 Pin Input/Output Circuits and Handling of Unused Pins

Table 3-1 lists the types of input/output circuits for each pin and explains how unused pins are handled.
Figure 3-1 shows the configuration of each type of input/output circuit.

Table 3-1. Type of Input/Output Circuit for Each Pin and Handling of Unused Pins

Pin name	I/O circuit type	I/O	Recommended connection of unused pins
P00-P03	5-A	I/O	Input: Connect these pins to the V_{DD} or V_{SS} pin through a separate resistor. Output: Leave these pins open.
P10, P11			
P20/$\overline{SCK20}$/ASCK20	8-A		
P21/SO20/TxD20			
P22/SI20/RxD20			
P23/INTP0/CPT20/$\overline{SS20}$			
P24/INTP1/TO80/TO20			
P25/INTP2/TI80			
P50-P53	13-W		Input: Connect these pins to the V_{DD} pin through a separate resistor. Output: Leave these pins open.
P60/ANI0-P63/ANI3	9-C	Input	Leave these pins open.
AV_{DD}	-	-	Connect this pin to the V_{DD} pin through a resistor.
AV_{SS}	-	-	Connect this pin to the V_{SS} pin through a resistor.
$\overline{RESET}$	2	Input	-
IC0	-	-	Connect this pin directly to the V_{SS} pin.
IC2	-	-	Leave this pin open.

Figure 3-1. Pin Input/Output Circuits

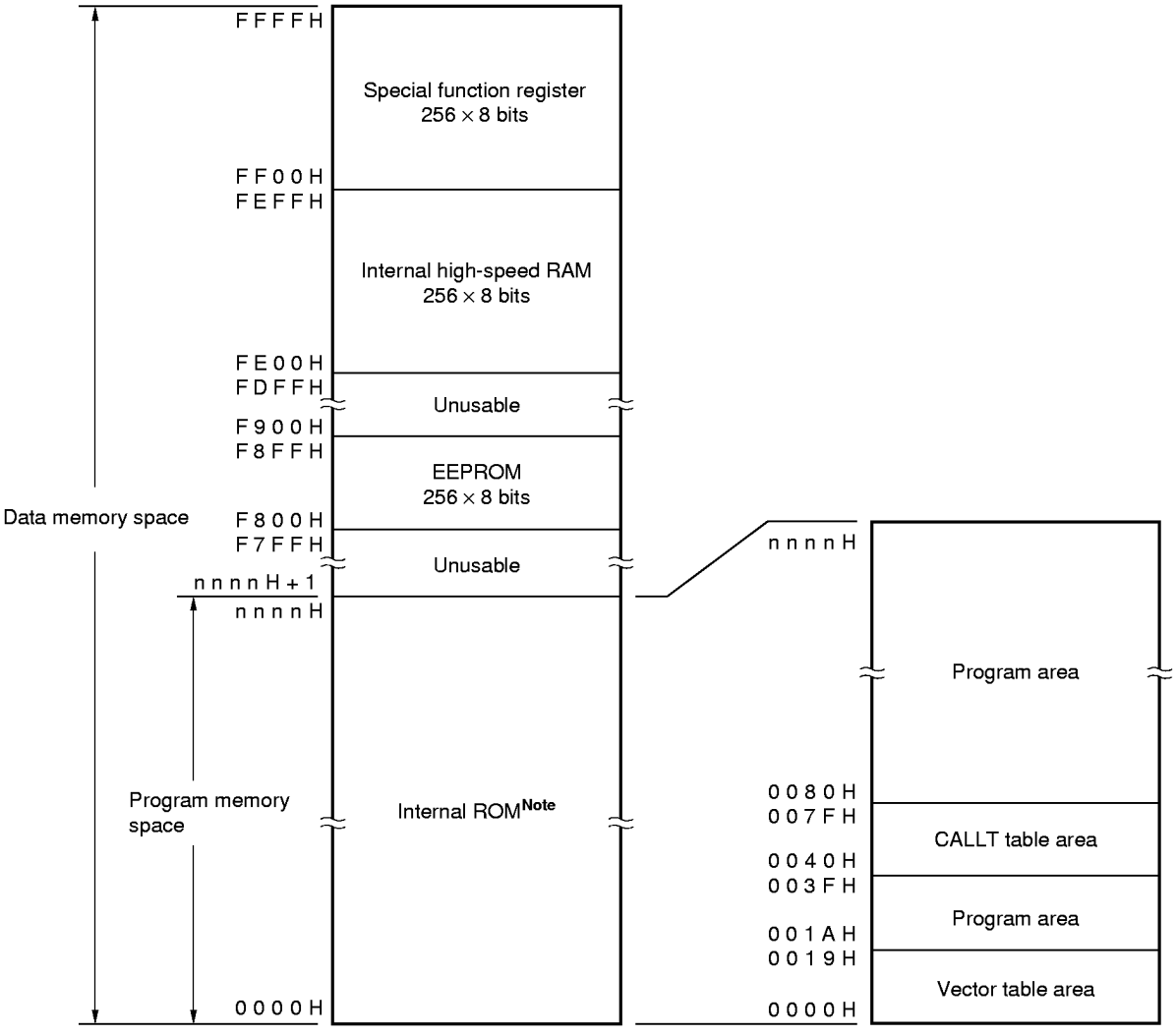


4. CPU ARCHITECTURE

4.1 Memory Space

The μPD789144, μPD789146, μPD789154, and μPD789156 can each access up to 64 Kbytes of memory space. Figure 4-1 shows the memory map.

Figure 4-1. Memory Map



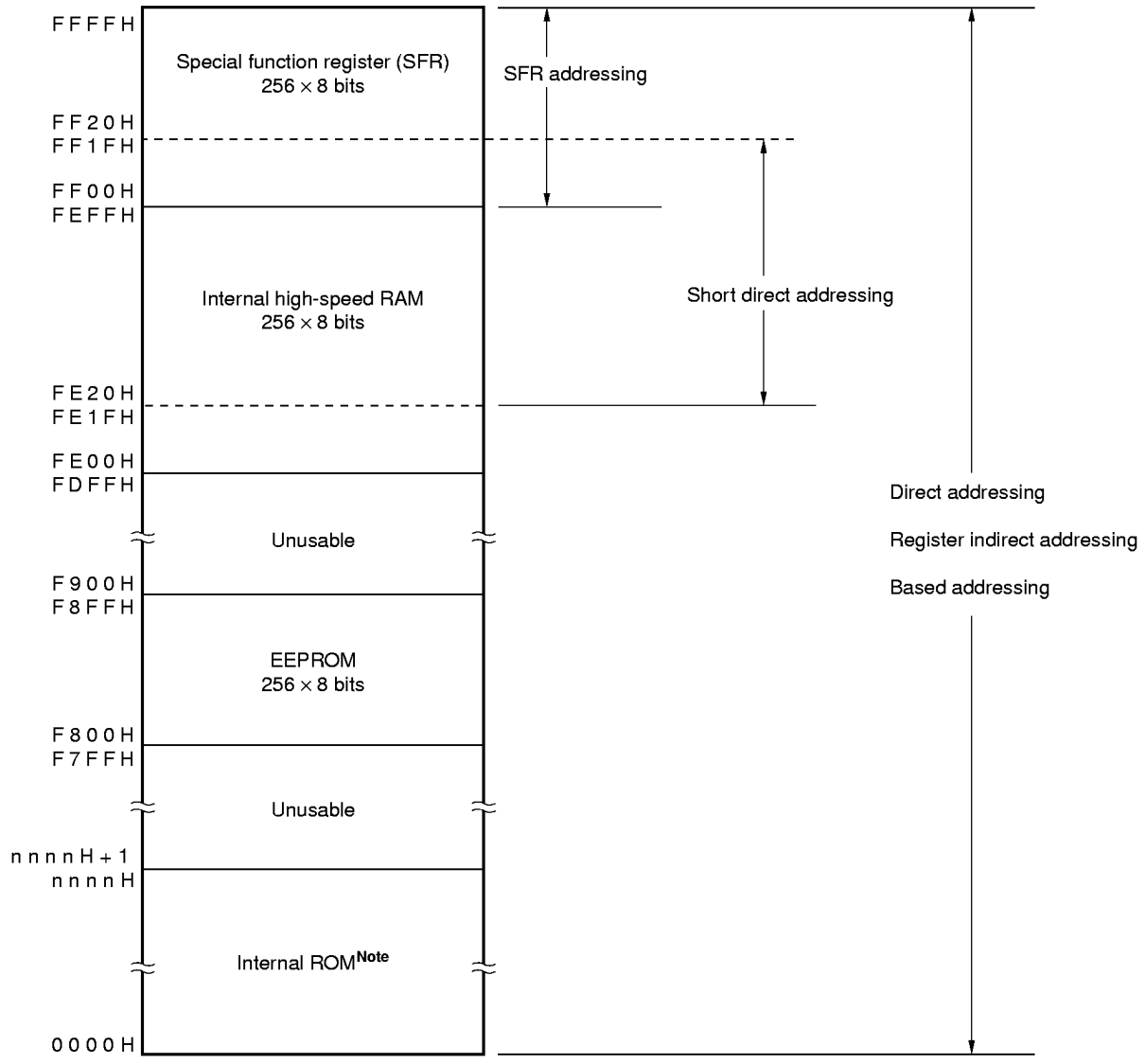
Note The size of the internal ROM varies depending on the model (see the following table).

Product name	Last address of internal ROM nnnnH
μPD789144, μPD789154	1FFFH
μPD789146, μPD789156	3FFFH

4.2 Data Memory Addressing

Each of the μPD789144, μPD789146, μPD789154, and μPD789156 is provided with a wide range of addressing modes to make memory manipulation as efficient as possible. A data memory area (FE00H to FFFFH) can be accessed using a unique addressing mode according to its use, such as a special function register (SFR). Figure 4-2 illustrates the data memory addressing modes.

Figure 4-2. Data Memory Addressing Modes



Note The size of internal ROM varies depending on the model (see the following table).

Product name	Last address of internal ROM nnnnH
μPD789144, μPD789154	1FFFH
μPD789146, μPD789156	3FFFH

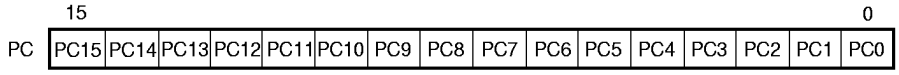
4.3 Processor Registers

4.3.1 Control registers

(1) Program counter (PC)

The PC is a 16-bit register for holding address information that indicates the next program to be executed.

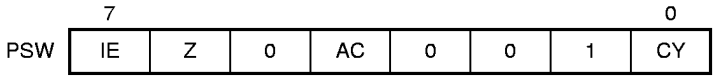
Figure 4-3. Program Counter Configuration



(2) Program status word (PSW)

The PSW is an 8-bit register for holding the status of the CPU according to the results of instruction execution.

Figure 4-4. Program Status Word Configuration



(a) Interrupt enable flag (IE)

IE is used to control whether interrupt requests are to be accepted by the CPU.

(b) Zero flag (Z)

Z is set (1) if the result of operation is zero. Otherwise, it is reset (0).

(c) Auxiliary carry flag (AC)

AC is set (1) if the result of the operation has a carry from bit 3 or a borrow to bit 3. Otherwise, it is reset (0).

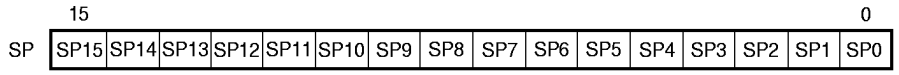
(d) Carry flag (CY)

CY is used to indicate whether an overflow or underflow has occurred during the execution of a subtract or add instruction.

(3) Stack pointer (SP)

SP is a 16-bit register for holding the start address of a stack area. The stack area can be specified only in an area (FE00H to FEFFH) of internal high-speed RAM.

Figure 4-5. Stack Pointer Configuration



Caution A $\overline{\text{RESET}}$ input makes the SP content undefined. Before executing an instruction, always initialize the SP.

4.3.2 General-purpose registers

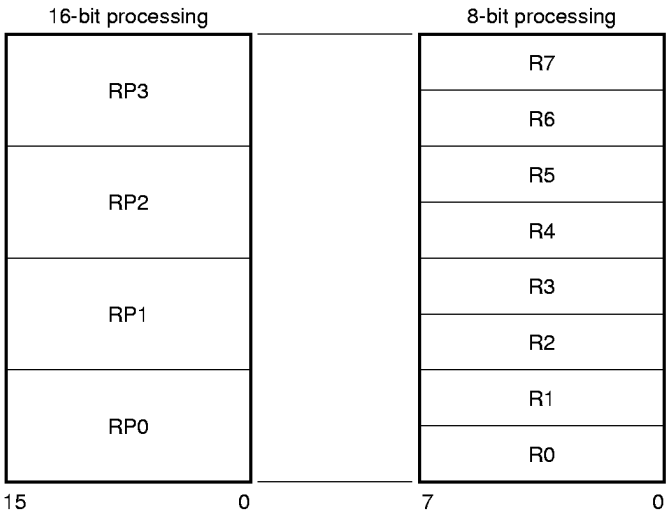
Each device has eight 8-bit general-purpose registers (X, A, C, B, E, D, L, and H).

These registers can be used as 16-bit registers (two 8-bit registers used in pairs like AX, BC, DE, and HL) as well as ordinary 8-bit registers.

These registers are identified using functional register names (X, A, C, B, E, D, L, H, AX, BC, DE, and HL) and absolute register names (R0 to R7 and RP0 to RP3).

Figure 4-6. General-Purpose Register Configuration

(a) Absolute register names



(b) Functional register names

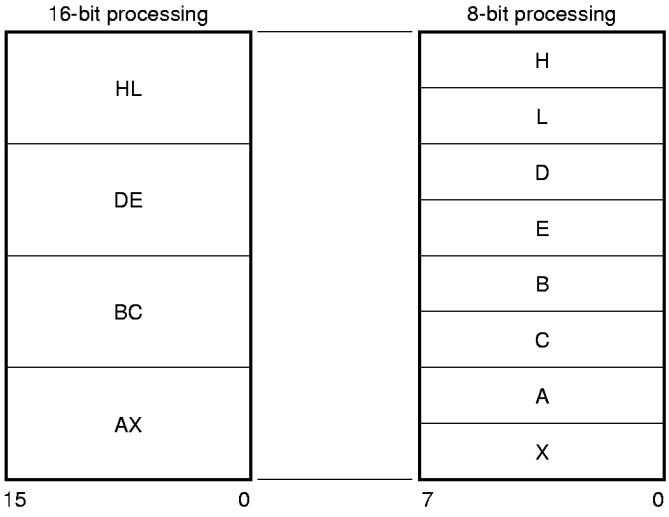


Table 4-1. Special Function Registers (1/2)

Address	Special function register (SFR) name	Symbol		R/W	Number of bits manipulated simultaneously			When reset
					1 bit	8 bits	16 bits	
FF00H	Port 0	P0		R/W	O	O	-	00H
FF01H	Port 1	P1			O	O	-	
FF02H	Port 2	P2			O	O	-	
FF05H	Port 5	P5			O	O	-	
FF06H	Port 6	P6		R	O	O	-	
FF10H	16-bit multiplication result storage register 0	MUL0L	MUL0		-	O[Note 1]	O[Note 2]	Undefined
FF11H		MUL0H						
FF14H	A/D conversion result register 0[Note 3]	ADCR0			-	O	O[Note 2]	
FF15H								
FF16H	16-bit compare register 20	CR20L	CR20	W	-	O[Note 1]	O[Note 2]	FFFFH
FF17H		CR20H						
FF18H	16-bit timer register 20	TM20L	TM20	R	-	O[Note 1]	O[Note 2]	0000H
FF19H		TM20H						
FF1AH	16-bit capture register 20	TCP20L	TCP20		-	O[Note 1]	O[Note 2]	Undefined
FF1BH		TCP20H						
FF20H	Port mode register 0	PM0		R/W	O	O	-	FFH
FF21H	Port mode register 1	PM1			O	O	-	
FF22H	Port mode register 2	PM2			O	O	-	
FF25H	Port mode register 5	PM5			O	O	-	
FF32H	Pull-up resistor option register B2	PUB2			O	O	-	00H
FF42H	Timer clock selection register 2	TCL2			-	O	-	
FF48H	16-bit timer mode control register 20	TMC20			O	O	-	
FF50H	8-bit compare register 80	CR80		W	-	O	-	Undefined
FF51H	8-bit timer register 80	TM80		R	-	O	-	00H
FF53H	8-bit timer mode control register 80	TMC80		R/W	O	O	-	

Notes 1. MUL0, CR20, TM20, and TCP20 are designed only for 16-bit access. With direct addressing, however, they can also be accessed in 8-bit mode.

2. 16-bit access is allowed only with short direct addressing.

3. When 8-bit A/D converters are used (for the μPD789146 sub-series), this register can be accessed only in 8-bit mode. In this case, the address is assumed to be FF15H. When 10-bit A/D converters are used (for the μPD789156 sub-series), this register can be accessed only in 16-bit mode. When the μPD78F9156 is used as flash memory for the μPD789144 or μPD789146, 8-bit access is allowed. However, only those object files generated by an assembler used with the μPD789144 or μPD789146 are supported for this access.

Table 4-1. Special Function Registers (2/2)

Address	Special function register (SFR) name	Symbol		R/W	Number of bits manipulated simultaneously			When reset
					1 bit	8 bits	16 bits	
FF70H	Asynchronous serial interface mode register 20	ASIM20		R/W	O	O	-	00H
FF71H	Asynchronous serial interface status register 20	ASIS20		R	O	O	-	
FF72H	Serial operation mode register 20	CSIM20		R/W	O	O	-	
FF73H	Baud rate generator control register 20	BRGC20			-	O	-	
FF74H	Transmission shift register 20	TXS20	SIO20	W	-	O	-	FFH
	Reception buffer register 20	RXB20		R	-	O	-	Undefined
FF80H	A/D converter mode register 0	ADM0		R/W	O	O	-	00H
FF84H	A/D input selection register 0	ADS0			O	O	-	
FFCCH	EEPROM write control register 1	EEWC1			O	O	-	08H
FFD0H	Multiplication data register A0	MRA0		W	O	O	-	Undefined
FFD1H	Multiplication data register B0	MRB0			O	O	-	
FFD2H	Multiplier control register 0	MULC0		R/W	O	O	-	00H
FFDFH	Low-voltage indicator register 0	LVI0			O	O	-	
FFE0H	Interrupt request flag register 0	IF0			O	O	-	
FFE1H	Interrupt request flag register 1	IF1			O	O	-	
FFE4H	Interrupt mask flag register 0	MK0			O	O	-	FFH
FFE5H	Interrupt mask flag register 1	MK1			O	O	-	
FFECH	External interrupt mode register 0	INTM0			-	O	-	00H
FFEDH	External interrupt mode register 1	INTM1			-	O	-	
FFF7H	Pull-up resistor option register 0	PU0			O	O	-	
FFF9H	Watchdog timer mode register	WDTM			O	O	-	
FFFAH	Oscillation settling time selection register	OSTS			-	O	-	04H
FFFBH	Processor clock control register	PCC			O	O	-	02H

5. EEPROM

5.1 EEPROM Functions

The μPD789144, μPD789146, μPD789154, and μPD789156 have a built-in EEPROM (electrically erasable PROM) (256 × 8 bits) as well as a built-in high-speed RAM as data memory.

Unlike ordinary RAM, the EEPROM can hold its data even when the power is turned off. Moreover, the EEPROM can erase its electrical data without using ultraviolet rays unlike EPROM.

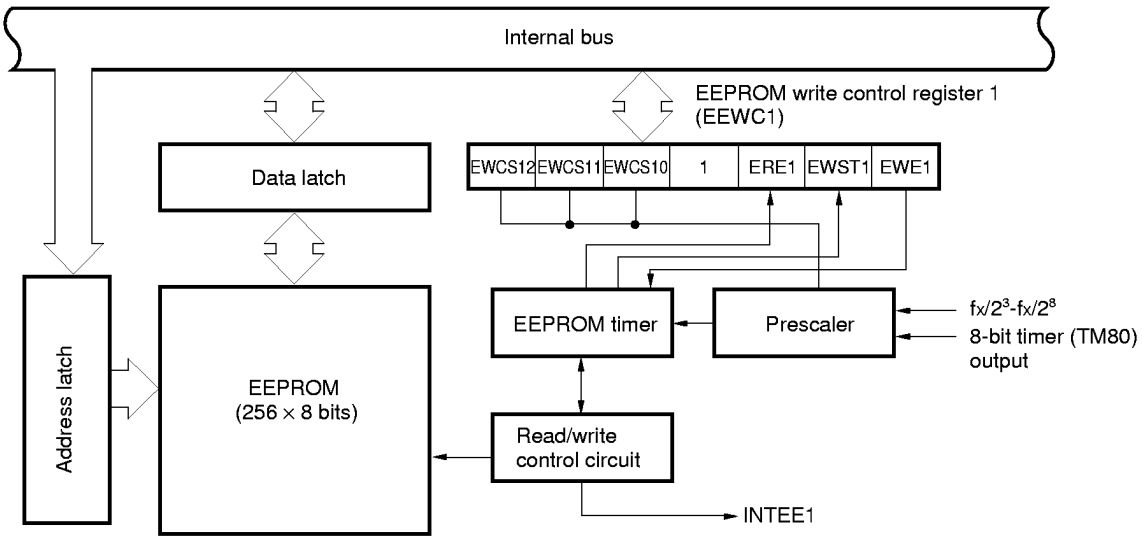
For EEPROM manipulation, an 8-bit memory manipulation instruction is used.

5.2 EEPROM Configuration

The EEPROM consists of an EEPROM body and control section.

The control section consists of EEPROM write control register 1 (EEWC1) for exercising EEPROM write control, and a section that generates an interrupt request signal (INTEE1) upon detection of termination of write operation.

Figure 5-1. Block Diagram of EEPROM



5.3 EEPROM Control Register

The EEPROM is controlled using EEPROM write control register 1 (EEWC1).

EEWC1 is the register set for selecting an EEPROM count clock and controlling writes to the EEPROM.

EEWC1 is set using a 1-bit or 8-bit memory manipulation instruction.

A $\overline{\text{RESET}}$ input loads 08H into EEWC1.

Figure 5-2. Format of EEPROM Write Control Register 1

Symbol	7	6	5	4	3	②	①	⓪	Address	When reset	R/W
EEWC1	0	EWCS12	EWCS11	EWCS10	1	ERE1	EWST1	EWE1	FFCCH	08H	R/W Note 1

EWCS12	EWCS11	EWCS10	EEPROM timer count clock selection	EEPROM write time Note 2
0	0	0	$f_X/2^3$ (625 kHz)	$2^3/f_X \times 145$ (Not to be set Note 3)
0	0	1	$f_X/2^4$ (313 kHz)	$2^4/f_X \times 145$ (Not to be set Note 3)
0	1	0	$f_X/2^5$ (156 kHz)	$2^5/f_X \times 145$ (Not to be set Note 3)
0	1	1	$f_X/2^6$ (78.1 kHz)	$2^6/f_X \times 145$ (Not to be set Note 3)
1	0	0	$f_X/2^7$ (39.1 kHz)	$2^7/f_X \times 145$ (3.71 ms)
1	0	1	$f_X/2^8$ (19.5 kHz)	$2^8/f_X \times 145$ (Not to be set Note 3)
1	1	0	8-bit timer (TM80) output	Output cycle of TM80 × 145
1	1	1	Not to be set	

ERE1	EWE1	Write	Read	Remarks
0	0	Disabled	Disabled	EEPROM is in the standby state (low power consumption mode).
0	1	Not to be set		
1	0	Disabled	Enabled	
1	1	Enabled	Enabled	

EWST1	EEPROM write status flag
0	Data is not being written to the EEPROM. (Data can be written to or read from the EEPROM. However, the EEPROM cannot be written to when EWE1 = 0.)
1	Data is being written to the EEPROM. (Data cannot be written to and read from the EEPROM.)

Notes 1. Bit 1 is read-only.

2. Set the EEPROM write time to a value between 3.3 and 6.6 ms. The value specified for the EEPROM write time is that value set as a target in the products development stage, and is subject to change after evaluation. When designing your system, refer to the data sheet that will be prepared after evaluation.

3. This setting is prohibited because the EEPROM write time falls outside the range of 3.3 to 6.6 ms.

Caution Bit 3 must be set to 1 and bit 7 must be set to 0.

Remarks 1. f_X: System clock oscillation frequency

2. The parenthesized values apply to operation at f_X = 5.0 MHz.

5.4 Cautions Related to EEPROM Writing

Note the following when writing data to the EEPROM.

(1) Set the count clock while the selected clock is active. If the selected clock is stopped, any attempt to start the clock and enable EEPROM write (EWE1 = 1) will fail to set write-enabled status.

(2) To switch the EEPROM timer count clock to the output of 8-bit timer/event counter 80 (TM80), first start TM80 in square wave output mode. If TM80 is operating in PWM mode, any attempt to enable EEPROM write will fail to set write-enabled status.

(3) The EEPROM write time must be set to a value that falls within the range of 3.3 to 6.6 ms.

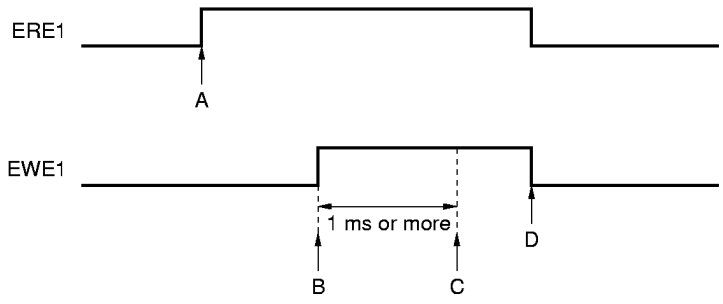
(4) To set ERE1 and EWE1, perform the following procedure; otherwise, the EEPROM write-enabled status is not set.

<1> Set ERE1 to 1 (with EWE1 set to 0).

<2> Set EWE1 to 1 (with ERE1 set to 1).

<3> Insert a wait of 1 ms or more (using software).

<4> The EEPROM enters write-enabled status.



A (ERE1 = 1) : Transition to read-enabled status

B (EWE1 = 1) : Set the count clock before reaching this point.

C : Transition to write-enabled status

D : When ERE1 is cleared (ERE1 = 0), EWE1 is also cleared (EWE1 = 0). The EEPROM cannot be read or written in this status.

(5) Before writing data to the EEPROM, check that EWST1 is set to 0. If executing a write instruction while EWST1 is set to 1, the instruction is ignored.

(6) Do not perform the following operations while writing data to the EEPROM. Otherwise, the values written to the addresses in the EEPROM cannot be guaranteed.

- Turning the power off
- Executing a reset
- Setting ERE1 to 0
- Setting EWE1 to 0
- Switching the count clock of the EEPROM timer

(7) Do not perform the following operation while writing data to the EEPROM with the scaled system clock selected as the count clock of the EEPROM timer. Otherwise, the values written to the addresses in the EEPROM cell cannot be guaranteed.

- Executing a STOP instruction

(8) Do not perform the following operations while writing data to the EEPROM with the TM80 output selected as the count clock of the EEPROM timer. Otherwise, the values written to the addresses in the EEPROM cell cannot be guaranteed.

- Switching TM80 operation mode from square wave output mode to PWM output mode
- Executing a STOP instruction
- Stopping TM80 timer output
- Stopping TM80 operation

(9) Do not perform the following operations when executing an instruction from the EEPROM. Otherwise, the data to be read from the EEPROM after such operations cannot be guaranteed and the CPU may enter an infinite loop.

- Setting ERE1 to 0
- Performing EEPROM write

(10) When data is not written to or read from the EEPROM, setting ERE1 to 0 may cause the system to enter low power consumption mode. While ERE1 is set to 1, a current of approximately 500 μA (V_{DD} = 5.5 V) is always applied. A current of approximately 2.0 mA (V_{DD} = 5.5 V) is applied during EEPROM read instruction execution. While EWE1 is set to 1, an additional 50 μA (approx.) (V_{DD} = 5.5 V) is applied. A current of approximately 1 mA (V_{DD} = 1.8 to 5.5 V) is applied during EEPROM write.

(11) When a STOP instruction is executed, the system automatically enters low power consumption mode regardless of the settings of ERE1 and EWE1. In this case, the statuses of ERE1 and EWE1 are retained. During the wait time, while STOP mode is released, a current of approximately 550 μA (V_{DD} = 5.5 V) is applied. When a HALT instruction is executed, the system does not enter low power consumption mode.

6. PERIPHERAL HARDWARE FUNCTIONS

6.1 Ports

6.1.1 Port functions

The μPD789144, μPD789146, μPD789154, and μPD789156 are provided with the ports shown in Figure 6-1. These ports are used to enable several types of control.

These ports, while originally designed as digital input/output ports, can also be used for other functions, as summarized in **Chapter 3**.

Figure 6-1. Port Types

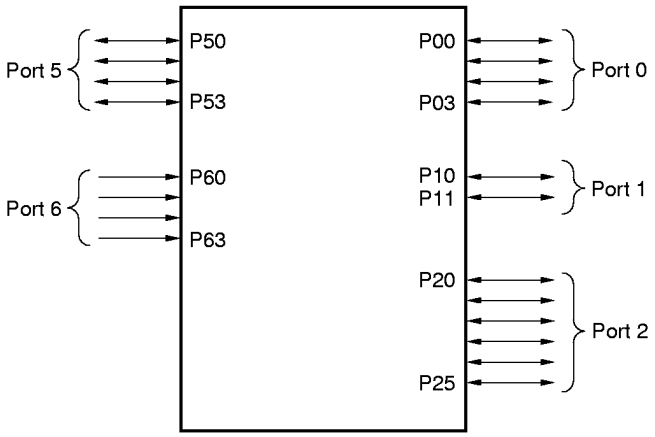


Table 6-1. Port Functions

Port name	Pin name	Description
Port 0	P00-P03	Input/output port. Each bit of the port can be separately specified as being for input or output. A port used for input can be connected to an on-chip pull-up resistor by means of pull-up resistor option register 0 (PU0).
Port 1	P10, P11	Input/output port. Each bit of the port can be separately specified as being for input or output. A port used for input can be connected to an on-chip pull-up resistor by means of pull-up resistor option register 0 (PU0).
Port 2	P20-P25	Input/output port. Each bit of the port can be separately specified as being for input or output. This port can be connected to an on-chip pull-up resistor by means of pull-up resistor option register B2 (PUB2).
Port 5	P50-P53	N-ch open-drain input/output port. Each bit of the port can be separately specified as being for input or output. Whether the port itself is to contain a pull-up resistor is specified with a mask option.
Port 6	P60-P63	Input-only port

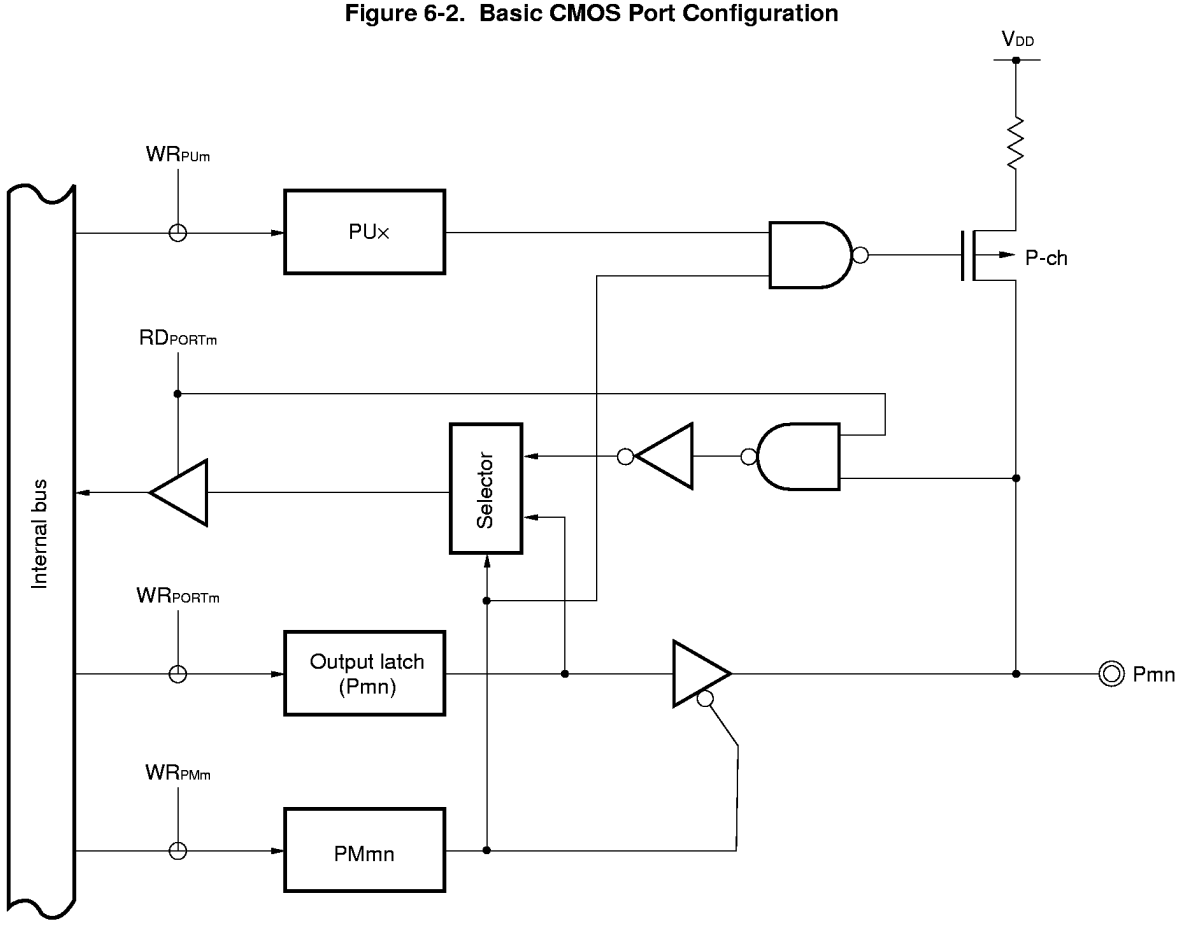
6.1.2 Port configuration

The hardware configuration of the ports is as follows.

Table 6-2. Port Configuration

Item	Configuration
Control register	Port mode register (PMm, where m = 0, 1, 2, or 5)
	Pull-up resistor option registers (PU0 and PUB2)
Port pins	Total: 20 (12 CMOS input/output, 4 CMOS input-only, and 4 N-ch open-drain input/output pins)
Pull-up resistors	Total: 16 (12 for software control and 4 for mask option control)

Figure 6-2. Basic CMOS Port Configuration



Caution **Figure 6-2 shows the basic configuration of the CMOS input/output ports. (Port 5 is not included in the basic configuration because they are used as N-ch open-drain input/output ports.) The configuration differs depending on the functions assigned to the pull-up resistor option registers and dual-function pins.**

Remark PU× : Pull-up resistor option register (× = 0 or B2)
PMmn : Bit n of port mode register m, where m = 0, 1, 2, or 5 and n = 0 to 7
Pmn : Bit n of port m
RD : Port read signal
WR : Port write signal
For details, see **(2)** in **Section 6.1.3**.

6.1.3 Port function control registers

The following three types of registers are used to control the ports.

- Port mode registers (PM0 to PM2, and PM5)
- Pull-up resistor option register 0 (PU0)
- Pull-up resistor option register B2 (PUB2)

(1) Port mode registers (PM0 to PM2, and PM5)

The port mode registers separately specify each port bit as being for input or output.

Each port mode register is manipulated using a 1-bit or 8-bit memory manipulation instruction.

A $\overline{\text{RESET}}$ input writes FFH into the port mode registers.

When port pins are used for secondary functions, the corresponding port mode register and output latch must be set or reset as described in Table 6-3.

Caution When port 2 is acting as an output port, and its output level is changed, an interrupt request flag is set, because this port is also used as the input for an external interrupt. To use port 2 in output mode, therefore, the interrupt mask flag must be set to 1 in advance.

Table 6-3. Port Mode Register and Output Latch Settings for Using Secondary Functions

Pin name	Secondary function		PM××	P××
	Name	Input/output		
P23	INTP0	Input	1	×
	CPT20	Input	1	×
P24	INTP1	Input	1	×
	TO80	Output	0	0
	TO20	Output	0	0
P25	INTP2	Input	1	×
	TI80	Input	1	×

Caution When port 2 is being used as a serial interface, it is necessary to specify whether the port is an input or output port, and to set the output latch accordingly. See Table 6-10 for an explanation of how to make this specification.

Remark × : Don't care
PM×× : Port mode register
P×× : Port output latch

Figure 6-3. Format of Port Mode Register

Symbol	7	6	5	4	3	2	1	0	Address	When reset	R/W
PM0	1	1	1	1	PM03	PM02	PM01	PM00	FF20H	FFH	R/W
PM1	1	1	1	1	1	1	PM11	PM10	FF21H	FFH	R/W
PM2	1	1	PM25	PM24	PM23	PM22	PM21	PM20	FF22H	FFH	R/W
PM5	1	1	1	1	PM53	PM52	PM51	PM50	FF25H	FFH	R/W

PMmn	Pmn pin input/output mode selection (m = 0, 1, 2, or 5; n = 0 to 5)
0	Output mode (output buffer ON)
1	Input mode (output buffer OFF)

(2) Pull-up resistor option register 0 (PU0)

This register is used to specify whether an on-chip pull-up resistor on ports 0 and 1 is used. An on-chip pull-up resistor can be used only for those bits set to the input mode of a port for which the use of the on-chip pull-up resistor is specified using PU0. For those bits set to the output mode, on-chip pull-up resistors cannot be used, regardless of the setting of PU0. This also applies to a dual-function pin used as an output pin.

PU0 is manipulated using a 1-bit or 8-bit memory manipulation instruction.

A $\overline{\text{RESET}}$ input clears PU0 to 00H.

Figure 6-4. Format of Pull-Up Resistor Option Register 0

Symbol	7	6	5	4	3	2	①	⓪	Address	When reset	R/W
PU0	0	0	0	0	0	0	PU01	PU00	FFF7H	00H	R/W

PU0m	Pm on-chip pull-up resistor selection (m = 0 or 1)
0	On-chip pull-up resistor not used
1	On-chip pull-up resistor used

(3) Pull-up resistor option register B2 (PUB2)

This register is used to specify whether to use an on-chip pull-up resistor on each port 2 pin. By setting PUB2, an on-chip pull-up resistor can be used, regardless of the setting of the port mode register.

PUB2 is manipulated using a 1-bit or 8-bit memory manipulation instruction.

A $\overline{\text{RESET}}$ input clears PUB2 to 00H.

Figure 6-5. Format of Pull-Up Resistor Option Register B2

Symbol	7	6	⑤	④	③	②	①	⓪	Address	When reset	R/W
PUB2	0	0	PUB25	PUB24	PUB23	PUB22	PUB21	PUB20	FF32H	00H	R/W

PUB2n	P2n on-chip pull-up resistor selection (n = 0 to 5)
0	On-chip pull-up resistor not used
1	On-chip pull-up resistor used

6.2 Clock Generator

6.2.1 Clock generator functions

The clock generator generates the clock pulse to be supplied to the CPU and peripheral hardware. There is the following system clock oscillator:

- System clock oscillator
  This circuit generates a frequency of 1.0 to 5.0 MHz. Oscillation can be stopped by executing the STOP instruction.

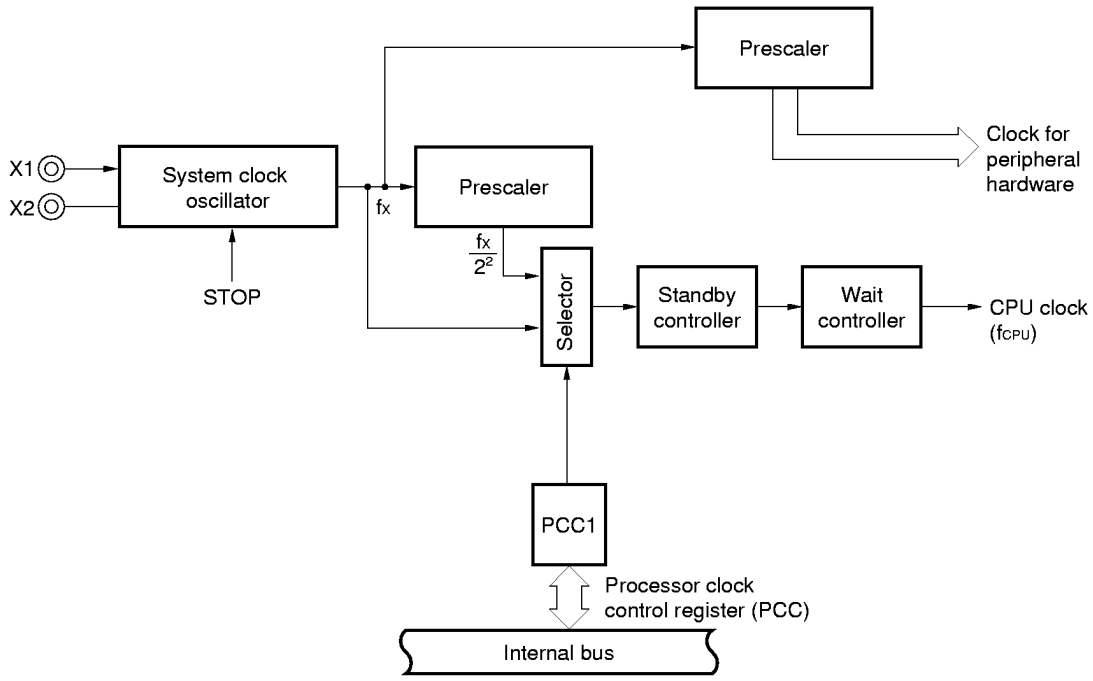
6.2.2 Clock generator configuration

The clock generator consists of the following hardware.

Table 6-4. Clock Generator Configuration

Item	Configuration
Control register	Processor clock control register (PCC)
Oscillator	System clock oscillator

Figure 6-6. Block Diagram of Clock Generator



6.2.3 Clock generator control register

The clock generator is controlled using the following register.

- Processor clock control register (PCC)

(1) Processor clock control register (PCC)

PCC selects a CPU clock and specifies a corresponding frequency division ratio.

It is manipulated using a 1-bit or 8-bit memory manipulation instruction.

A $\overline{\text{RESET}}$ input loads 02H into PCC.

Figure 6-7. Format of Processor Clock Control Register

Symbol	7	6	5	4	3	2	1	0	Address	When reset	R/W
PCC	0	0	0	0	0	0	PCC1	0	FFFBH	02H	R/W

PCC	CPU clock (f_{CPU}) selection
0	f_X (0.2 μs)
1	$f_X/2^2$ (0.8 μs)

Caution Bits 0 and 2 to 7 must be fixed to 0.

Remarks 1. f_X: System clock oscillation frequency

2. The parenthesized values (minimum instruction execution time) apply to operation at f_X = 5.0 MHz.

3. Minimum instruction execution time: 2 f_{CPU}

- f_{CPU} = 0.2 μs: 0.4 μs
- f_{CPU} = 0.8 μs: 1.6 μs

6.3 16-Bit Timer Counter

6.3.1 16-Bit timer counter functions

16-bit timer counter 20 (TM20) has the following functions.

- Timer interrupt
- Timer output
- Count capture

(1) Timer interrupt

An interrupt is generated if the TM20 count matches a comparison value.

(2) Timer output

The timer output can be controlled if the count matches a comparison value.

(3) Count capture

The count in TM20 is captured into the capture register in synchronization with a capture trigger.

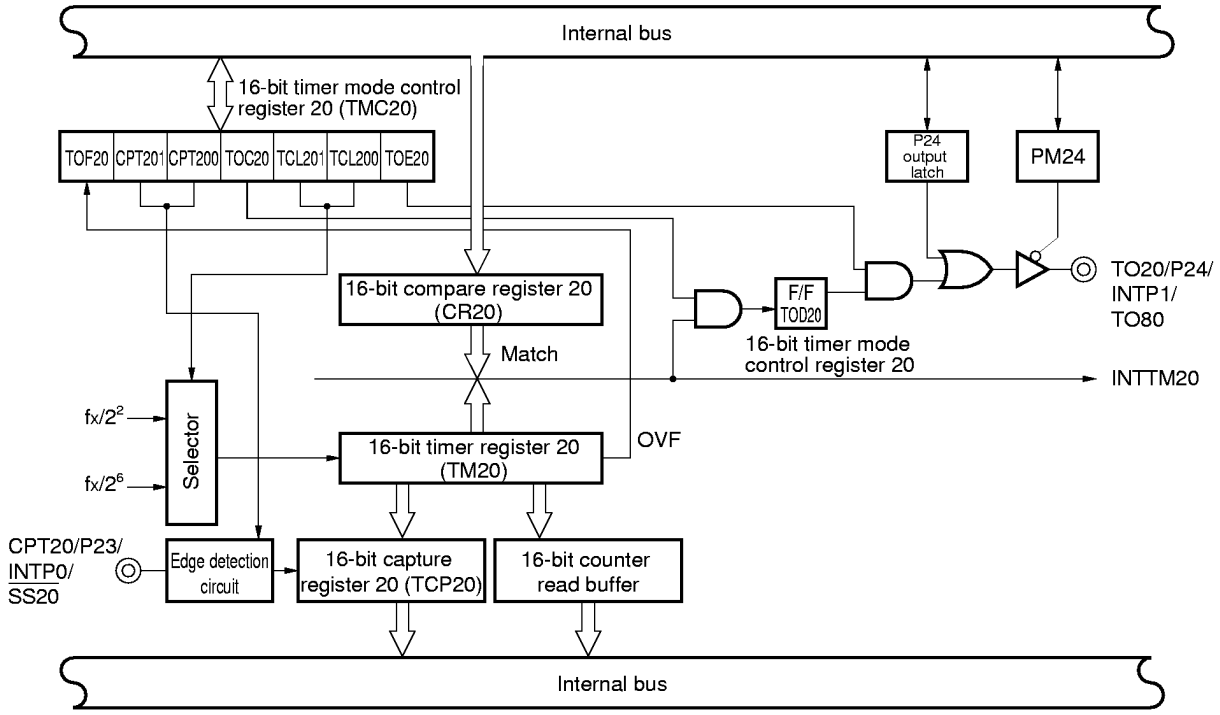
6.3.2 16-bit timer counter configuration

16-bit timer counter 20 (TM20) consists of the following hardware.

Table 6-5. 16-Bit Timer Counter Configuration

Item	Configuration
Timer register	16 bits × 1 (TM20)
Register	Compare register : 16 bits × 1 (CR20) Capture register : 16 bits × 1 (TCP20)
Timer output	1 (TO20)
Control register	16-bit timer mode control register 20 (TMC20) Port mode register 2 (PM2)

Figure 6-8. Block Diagram of 16-Bit Timer Counter



(1) 16-bit compare register 20 (CR20)

A value specified in CR20 is compared with the count in 16-bit timer register 20 (TM20). If they match, an interrupt request (INTTM20) is issued.

CR20 is manipulated using a 16-bit memory manipulation instruction. Any value from 0000H to FFFFH can be set.

A $\overline{\text{RESET}}$ input loads FFFFH into CR20.

Cautions 1. CR20 is designed to be manipulated using a 16-bit memory manipulation instruction. It can also be manipulated using 8-bit memory manipulation instructions, however. When an 8-bit memory manipulation instruction is used to set CR20, it must be in a direct addressing access mode.

2. To re-set CR20 during count operation, it is necessary to disable interrupts in advance, using an interrupt mask flag register 1 (MK1). It is also necessary to disable inversion of the timer output data, using 16-bit timer mode control register 20 (TMC20).

(2) 16-bit timer register 20 (TM20)

TM20 is used to count the number of pulses.

The contents of TM20 are read using a 16-bit memory manipulation instruction.

This register is in free running during count clock input.

A $\overline{\text{RESET}}$ input clears TM20 to 0000H and after that to be in free running.

Cautions 1. The count becomes undefined when STOP mode is deselected, because the count operation is performed before oscillation settles.

2. TM20 is designed to be manipulated using a 16-bit memory manipulation instruction. It can also be manipulated using 8-bit memory manipulation instructions, however. When an 8-bit memory instruction is used to manipulate TM20, it must be in a direct addressing access mode.

3. When an 8-bit memory manipulation instruction is used to manipulate TM20, the lower and upper bytes must be read as a pair, in this order.

(3) 16-bit capture register 20 (TCP20)

TCP20 captures the contents of 16-bit timer register 20 (TM20).

It is manipulated using a 16-bit memory manipulation instruction.

A $\overline{\text{RESET}}$ input makes TCP20 undefined.

Caution TCP20 is designed to be manipulated using a 16-bit memory manipulation instruction. It can also be manipulated using 8-bit memory manipulation instructions, however. When an 8-bit memory manipulation instruction is used to manipulate TCP20, it must be in a direct addressing access mode.

(4) 16-bit counter read buffer

This buffer is used to latch and hold the count for 16-bit timer register 20 (TM20).

6.3.3 16-bit timer/counter control registers

The following two types of registers are used to control 16-bit timer counter 20 (TM20).

- 16-bit timer mode control register 20 (TMC20)
- Port mode register 2 (PM2)

(1) 16-bit timer mode control register 20 (TMC20)

TMC20 controls the count clock and capture edge settings.

It is manipulated using a 1-bit or 8-bit memory manipulation instruction.

A $\overline{\text{RESET}}$ input clears TMC20 to 00H.

Figure 6-9. Format of 16-Bit Timer Mode Control Register 20

Symbol	7	⑥	5	4	3	2	1	⓪	Address	When reset	R/W
TMC20	TOD20	TOF20	CPT201	CPT200	TOC20	TCL201	TCL200	TOE20	FF48H	00H	R/W[Note]

TOD20	Stores output data of the 16-bit timer.

TOF20	Overflow flag setting
0	Reset or cleared by software
1	Set when the 16-bit timer overflows

CPT201	CPT200	Capture edge selection
0	0	Capture operation disabled
0	1	Captured at the rising edge at the CPT20 pin
1	0	Captured at the falling edge at the CPT20 pin
1	1	Captured at both the rising and falling edges at the CPT20 pin

TOC20	Timer output data inversion control
0	Inversion disabled
1	Inversion enabled

TCL201	TCL200	16-bit timer register 20 count clock selection
0	0	$f_X/2^2$ (1.25 MHz)
0	1	$f_X/2^6$ (78.1 kHz)
Other settings		Not to be set

TOE20	16-bit timer counter 20 output control
0	Output disabled (port mode)
1	Output enabled

Note Bit 7 is read-only.

Remarks 1. fx: System clock oscillation frequency

2. The parenthesized values apply to operation at fx = 5.0 MHz.

(2) Port mode register 2 (PM2)

PM2 is used to specify port 2 input/output on a bit-by-bit basis.

When the P24/TO20/INTP1/TO80 pin is used for timer output, set 0 in the output latch of PM24 and P24.

PM2 is manipulated using a 1-bit or 8-bit memory manipulation instruction.

A $\overline{\text{RESET}}$ input loads FFH into PM2.

Figure 6-10. Format of Port Mode Register 2

Symbol	7	6	5	4	3	2	1	0	Address	When reset	R/W
PM2	1	1	PM25	PM24	PM23	PM22	PM21	PM20	FF22H	FFH	R/W

PM24	P24 pin I/O mode selection
0	Output mode (output buffer ON)
1	Input mode (output buffer OFF)

6.4 8-Bit Timer/Event Counter

6.4.1 8-bit timer/event counter functions

8-bit timer/event counter 80 (TM80) has the following functions.

- Interval timer
- External event counter
- Square wave output
- PWM output

(1) Interval timer

This timer causes interrupts to be issued at specified intervals.

(2) External event counter

This counter is used to count the number of pulses input from an external source.

(3) Square wave output

A square wave of any frequency can be output.

(4) PWM output

PWM output with an 8-bit resolution is supported.

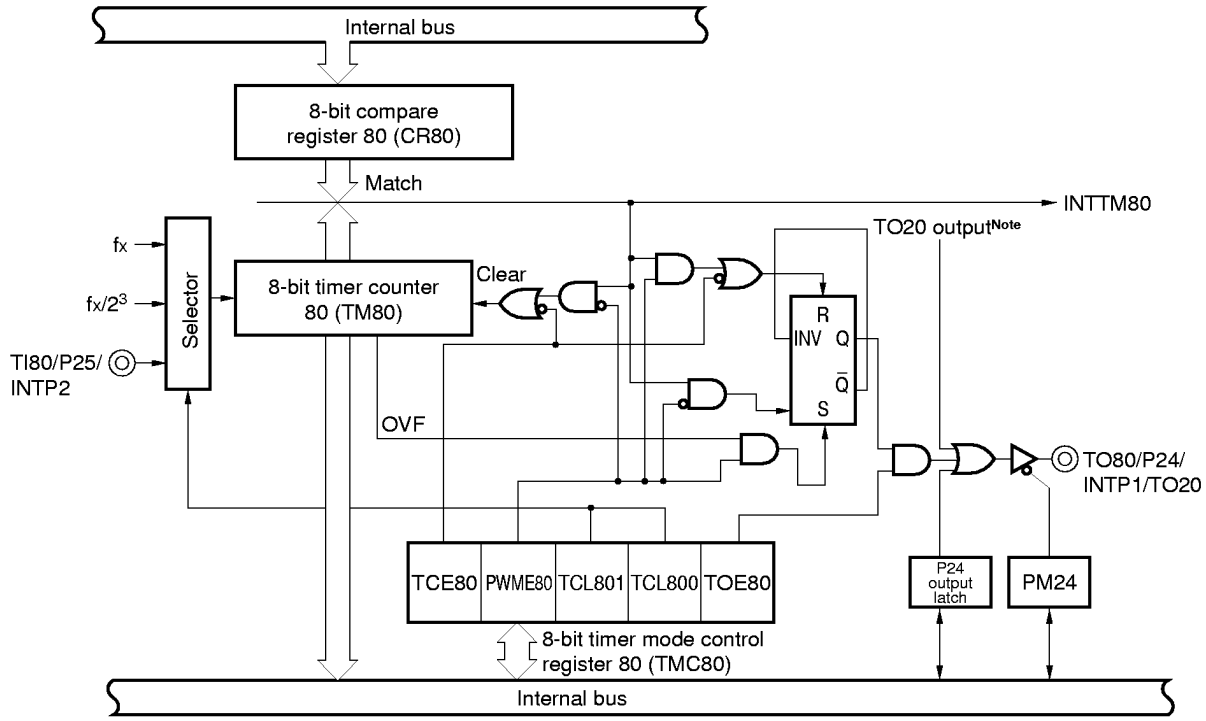
6.4.2 8-bit timer/event counter configuration

8-bit timer/event counter 80 (TM80) consists of the following hardware.

Table 6-6. 8-Bit Timer/Event Counter 80 Configuration

Item	Configuration
Timer register	8 bits × 1 (TM80)
Register	Compare registers: 8 bits × 1 (CR80)
Timer output	1 (TO80)
Control register	8-bit timer mode control register 80 (TMC80) Port mode register 2 (PM2)

Figure 6-11. Block Diagram of 8-Bit Timer/Event Counter



Note See Figure 6-8.

(1) 8-bit compare register 80 (CR80)

A value specified in CR80 is compared with the count in 8-bit timer register 80 (TM80). If they match, an interrupt request (INTTM80) is issued.

CR80 is manipulated using an 8-bit memory manipulation instruction. Any value from 00H to FFH can be set.

A $\overline{\text{RESET}}$ input makes CR80 undefined.

Caution Do not set CR80 to 00H in PWM output mode (when PWME80 = 1: bit 6 of 8-bit timer mode control register 80 (TMC80)); otherwise, PWM may not be output normally.

(2) 8-bit timer register 80 (TM80)

TM80 is used to count the number of pulses.

Its contents are read using an 8-bit memory manipulation instruction.

A $\overline{\text{RESET}}$ input clears TM80 to 00H.

6.4.3 8-bit timer/event counter control registers

The following two types of registers are used to control the 8-bit timer/event counter.

- 8-bit timer mode control register 80 (TMC80)
- Port mode register 2 (PM2)

(1) 8-bit timer mode control register 80 (TMC80)

TMC80 determines whether to enable or disable 8-bit timer register 80 (TM80) and specifies the count clock for TM80. It also controls the operation of the output control circuit of the 8-bit timer/event counter.

TMC80 is manipulated using a 1-bit or 8-bit memory manipulation instruction.

A $\overline{\text{RESET}}$ input clears TMC80 to 00H.

Figure 6-12. Format of 8-Bit Timer Mode Control Register 80

Symbol	⑦	⑥	5	4	3	2	1	⓪	Address	When reset	R/W
TMC80	TCE80	PWME80	0	0	0	TCL801	TCL800	TOE80	FF53H	00H	R/W

TCE80	8-bit timer register 80 operation control
0	Operation disabled (TM80 is cleared to 0.)
1	Operation enabled

PWME80	Operation mode selection
0	Timer counter operating mode
1	PWM output operating mode

TCL801	TCL800	8-bit timer/event counter count clock selection
0	0	fx (5.0 MHz)
0	1	$fx/2^3$ (625 kHz)
1	0	Rising edge of TI80
1	1	Falling edge of TI80

TOE80	8-bit timer/event counter output control
0	Output disabled (port mode)
1	Output enabled

Caution Always stop the timer before setting TMC80.

Remarks 1. fx: System clock oscillation frequency

2. The parenthesized values apply to operation at fx = 5.0 MHz.

(2) Port mode register 2 (PM2)

PM2 specifies whether each bit of port 2 is used for input or output.

To use the P24/TO80/INTP1/TO20 pin for timer output, the PM24 and P24 output latches must be reset to 0.

PM2 is manipulated using a 1-bit or 8-bit memory manipulation instruction.

A $\overline{\text{RESET}}$ input loads FFH into PM2.

Figure 6-13. Format of Port Mode Register 2

Symbol	7	6	5	4	3	2	1	0	Address	When reset	R/W
PM2	1	1	PM25	PM24	PM23	PM22	PM21	PM20	FF22H	FFH	R/W

PM24	P24 pin input/output mode setting
0	Output mode (output buffer ON)
1	Input mode (output buffer OFF)

6.5 Watchdog Timer

6.5.1 Watchdog timer functions

The watchdog timer has the following functions.

(1) Watchdog timer

The watchdog timer is used to detect unintended program loops. If an unintended program loop is detected, a nonmaskable interrupt or $\overline{\text{RESET}}$ signal is generated.

(2) Interval timer

The interval timer is used to generate interrupts at specified intervals.

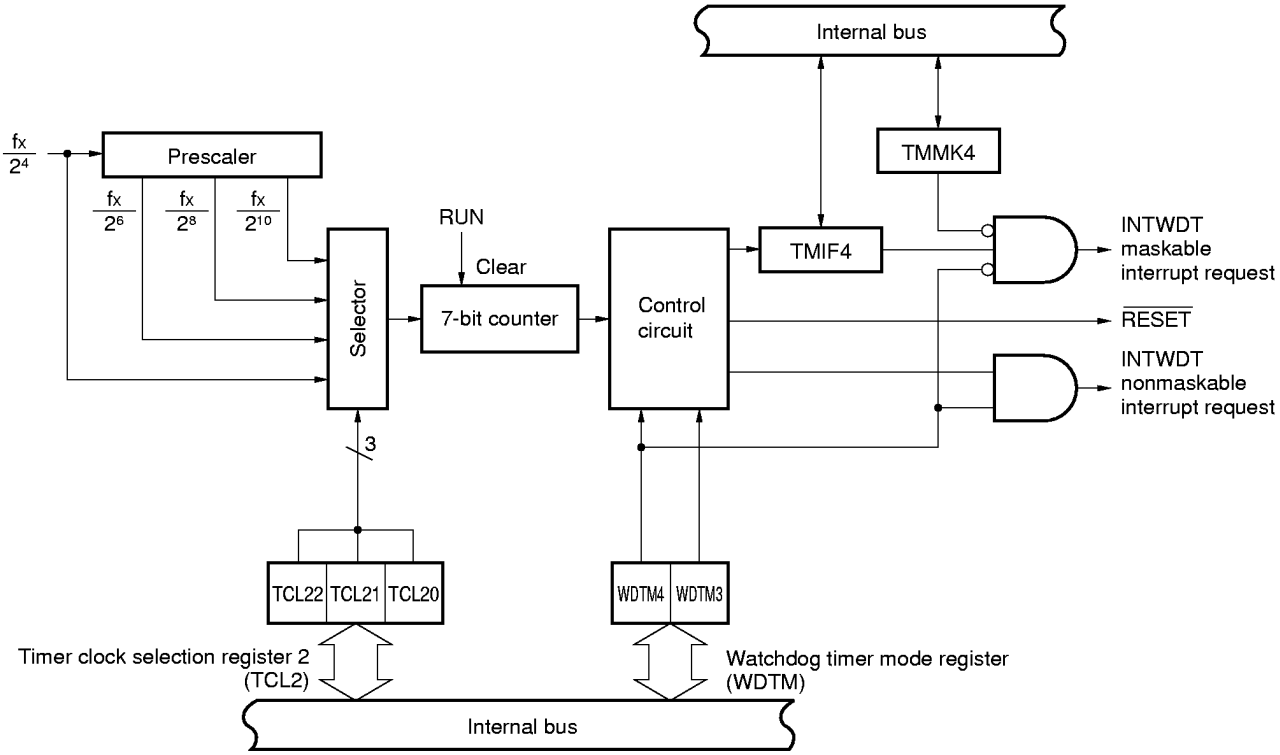
6.5.2 Watchdog timer configuration

The watchdog timer consists of the following hardware.

Table 6-7. Watchdog Timer Configuration

Item	Configuration
Control register	Timer clock selection register 2 (TCL2) Watchdog timer mode register (WDTM)

Figure 6-14. Block Diagram of Watchdog Timer



6.5.3 Watchdog timer control registers

The following two types of registers are used to control the watchdog timer.

- Timer clock selection register 2 (TCL2)
- Watchdog timer mode register (WDTM)

(1) Timer clock selection register 2 (TCL2)

TCL2 specifies the count clock for the watchdog timer.

TCL2 is manipulated using an 8-bit memory manipulation instruction.

A $\overline{\text{RESET}}$ input clears TCL2 to 00H.

Figure 6-15. Format of Timer Clock Selection Register 2

Symbol	7	6	5	4	3	2	1	0	Address	When reset	R/W
TCL2	0	0	0	0	0	TCL22	TCL21	TCL20	FF42H	00H	R/W

TCL22	TCL21	TCL20	Watchdog timer count clock selection	Interval time
0	0	0	$f_X/2^4$ (312.5 kHz)	$2^{11}/f_X$ (410 μs)
0	1	0	$f_X/2^6$ (78.1 kHz)	$2^{13}/f_X$ (1.64 ms)
1	0	0	$f_X/2^8$ (19.5 kHz)	$2^{15}/f_X$ (6.55 ms)
1	1	0	$f_X/2^{10}$ (4.88 kHz)	$2^{17}/f_X$ (26.2 ms)
Other settings			Not to be set	

Remarks 1. f_X: System clock oscillation frequency

2. The parenthesized values apply to operation at f_X = 5.0 MHz.

(2) Watchdog timer mode register (WDTM)

WDTM specifies the watchdog timer operation mode and whether to enable or disable counting.

WDTM is manipulated using a 1-bit or 8-bit memory manipulation instruction.

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

Figure 6-16. Format of Watchdog Timer Mode Register

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

RUN	Watchdog timer operation selection[Note 1]
0	Stops counting.
1	Clears the counter and causes it to start.

WDTM4	WDTM3	Watchdog timer operation mode selection[Note 2]
0	0	Operation disabled
0	1	Interval timer mode (When an overflow occurs, a maskable interrupt is issued.)[Note 3]
1	0	Watchdog timer mode 1 (When an overflow occurs, a nonmaskable interrupt is issued.)
1	1	Watchdog timer mode 2 (When an overflow occurs, a reset operation is started.)

Notes 1. Once the RUN bit has been set (1), it is impossible to zero-clear it by software. So, once counting begins, it cannot be stopped by any means other than a $\overline{\text{RESET}}$ input.

2. Once WDTM3 and WDTM4 have been set (1), it is impossible to zero-clear them by software.

3. The interval timer starts operating when the RUN bit is set to 1.

Cautions 1. If the RUN bit is set to 1, and the watchdog timer is cleared, the actual overflow time becomes 0.8% (maximum) less than the time specified in timer clock selection register 2 (TCL2).

2. To use watchdog timer mode 1 or 2, ensure that TMIF4 (bit 0 of interrupt request flag 0 (IF0)) is set to 0, before setting TMMK4 (bit 0 of interrupt mask flag register 0 (MK0)) to 1. If TMIF4 is set to 1, selecting watchdog timer mode 1 or 2 causes a nonmaskable interrupt to be issued at the instant rewriting ends.

6.6 A/D Converter

A/D converters support different conversion resolutions, depending on the microcontroller model, as shown below:

- A/D converters with an 8-bit resolution: for μPD789144 and μPD789146
- A/D converters with a 10-bit resolution: for μPD789154 and μPD789156

6.6.1 A/D converter functions

The A/D converter converts input analog voltages to digital signals with an 8-bit or 10-bit resolution. It can control up to four analog input channels (ANI0 to ANI3).

A/D conversion can be started only by software.

One of analog inputs ANI0 to ANI3 is selected for A/D conversion. A/D conversion is performed repeatedly, with an interrupt request (INTAD0) being issued each time an A/D conversion is completed.

Caution In standby mode, the A/D converter operation is disabled.

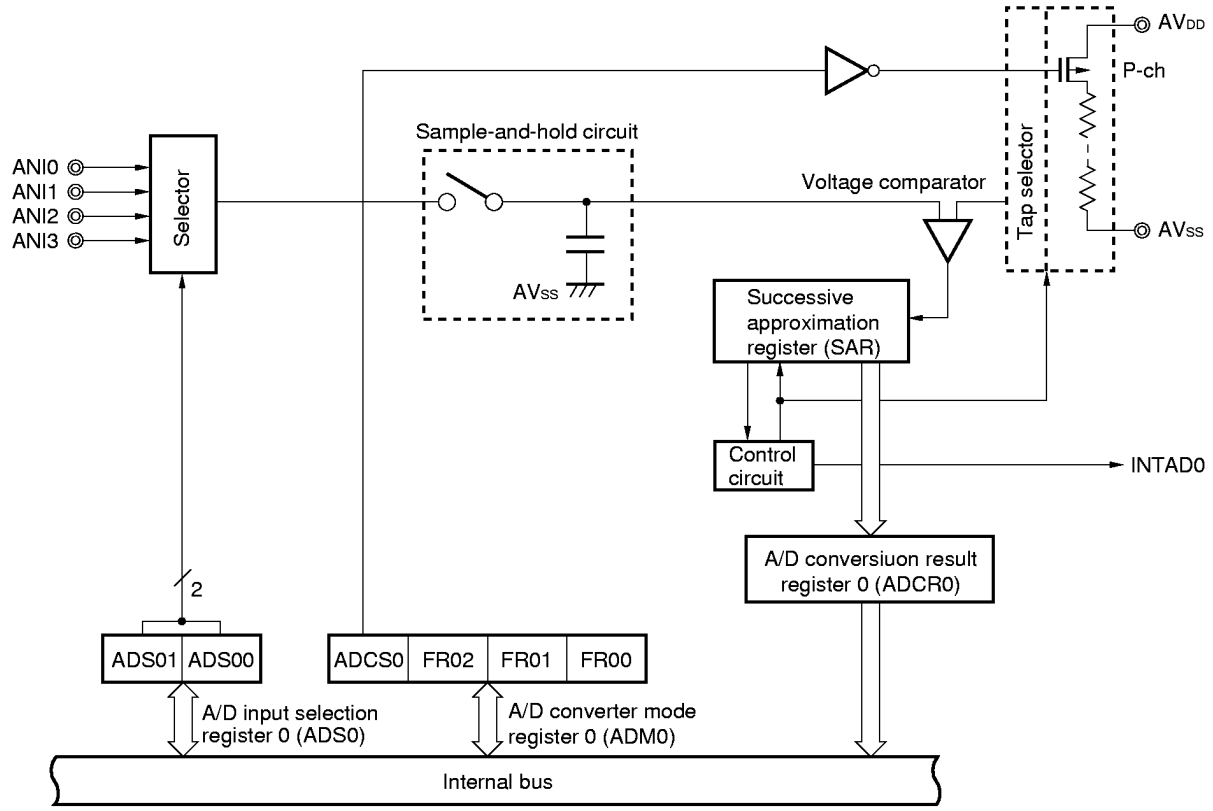
6.6.2 A/D converter configuration

The A/D converter consists of the following hardware.

Table 6-8. A/D Converter Configuration

Item	Configuration
Analog input	4 channels (ANI0 to ANI3)
Register	Successive approximation register (SAR) A/D conversion result register 0 (ADCR0)
Control register	A/D converter mode register 0 (ADM0) A/D input selection register 0 (ADS0)

Figure 6-17. Block Diagram of A/D Converter



(1) Successive approximation register (SAR)

SAR receives the result of comparing an analog input voltage and a voltage at a voltage tap (comparison voltage), received from the serial resistor string, starting from the most significant bit (MSB).

Upon receiving all the bits, down to the least significant bit (LSB), that is, upon the completion of A/D conversion, the SAR sends its contents to A/D conversion result register 0.

(2) A/D conversion result register 0 (ADCR0)

ADCR0 holds the result of A/D conversion. Each time A/D conversion ends, the conversion result received from the successive approximation register is loaded into ADCR0.

For the μPD789144 and μPD789146 (featuring 8-bit A/D converters), the value of ADCR0 is read using an 8-bit memory manipulation instruction.

For the μPD789154 and μPD789156 (featuring 10-bit A/D converters), the value of ADCR0 is read using a 16-bit memory manipulation instruction.

A $\overline{\text{RESET}}$ input clears ADCR0 to 00H.

Caution When 8-bit A/D converters are used (for the μPD789144 and μPD789146), this register can be accessed only in 8-bit mode. In this case, the address is assumed to be FF15H.

When 10-bit A/D converters are used (for the μPD789154 and μPD789156), this register can be accessed only in 16-bit mode.

When the μPD78F9156 is used as flash memory for the μPD789144 or μPD789146, 8-bit access is allowed. However, only those object files generated by an assembler used with the μPD789144 or μPD789146 are supported for this access.

(3) Sample-and-hold circuit

The sample-and-hold circuit samples consecutive analog inputs from the input circuit, one by one, and sends them to the voltage comparator. The sampled analog input voltage is held during A/D conversion.

(4) Voltage comparator

The voltage comparator compares an analog input with the voltage output by the serial resistor string.

(5) Serial resistor string

The serial resistor string is configured between AV_{DD} and AV_{SS}. It generates the reference voltages against which analog inputs are compared.

(6) ANI0 to ANI3 pins

Pins ANI0 to ANI3 are 4-channel analog input pins for the A/D converter. They are used to receive the analog signals to be subject to A/D conversion.

Caution Do not supply pins ANI0 to ANI3 with voltages that fall outside the rated range. If a voltage greater than AV_{DD} or less than AV_{SS} (even if within the absolute maximum rating) is supplied to any of these pins, the conversion value for the corresponding channel will be undefined. Furthermore, the conversion values for the other channels may also be affected.

(7) AV_{SS} pin

The AV_{SS} pin is a ground potential pin for the A/D converter. This pin must be held at the same potential as the V_{SS} pin, even while the A/D converter is not being used.

(8) AV_{DD} pin

The AV_{DD} pin is an analog power supply pin for the A/D converter. This pin must be held at the same potential as the V_{DD} pin, even while the A/D converter is not being used.

6.6.3 A/D converter control registers

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

- A/D converter mode register 0 (ADM0)
- A/D input selection register 0 (ADS0)

(1) A/D converter mode register 0 (ADM0)

ADM0 specifies the conversion time for analog inputs. It also specifies whether to enable conversion.

ADM0 is manipulated using a 1-bit or 8-bit memory manipulation instruction.

A $\overline{\text{RESET}}$ input clears ADM0 to 00H.

Figure 6-18. Format of A/D Converter Mode Register 0

Symbol	⑦	6	5	4	3	2	1	0	Address	When reset	R/W
ADM0	ADCS0	0	FR02	FR01	FR00	0	0	0	FF80H	00H	R/W

ADCS0	A/D conversion control
0	Conversion disabled
1	Conversion enabled

FR02	FR01	FR00	A/D conversion time selection[Note 1]
0	0	0	144/fx (28.8 μs)
0	0	1	120/fx (24 μs)
0	1	0	96/fx (19.2 μs)
1	0	0	72/fx (14.4 μs)
1	0	1	60/fx (Not to be set[Note 2])
1	1	0	48/fx (Not to be set[Note 2])
Other settings			Not to be set

Notes 1. The specifications of FR02, FR01, and FR00 must be such that the A/D conversion time is at least 14 μs.

2. These bit combinations must not be used, as the A/D conversion time will fall below 14 μs.

Caution The result of conversion performed immediately after bit 7 (ADCS0) is set is undefined.

Remark fx: System clock oscillation frequency

(2) A/D input selection register 0 (ADS0)

ADS0 specifies the port used to input the analog voltages to be converted to a digital signal. ADS0 is manipulated using a 1-bit or 8-bit memory manipulation instruction.

A $\overline{\text{RESET}}$ input clears ADS0 to 00H.

Figure 6-19. Format of A/D Input Selection Register 0

Symbol	7	6	5	4	3	2	1	0	Address	When reset	R/W
ADS0	0	0	0	0	0	0	ADS01	ADS00	FF84H	00H	R/W

ADS01	ADS00	Analog input channel specification
0	0	ANI0
0	1	ANI1
1	0	ANI2
1	1	ANI3

6.7 Serial Interface

6.7.1 Serial interface 20 functions

Serial interface 20 has the following three types of modes.

- Operation stopped mode
- Asynchronous serial interface (UART) mode
- Three-wire serial I/O mode

(1) Operation stopped mode

This mode is used when serial transfer is not performed. Power consumption is minimized in this mode.

(2) Asynchronous serial interface (UART) mode

This mode is used to send and receive the one byte of data that follows a start bit. It supports full-duplex communication.

Serial interface 20 contains a dedicated UART baud rate generator, enabling communication over a wide range of baud rates. It is also possible to define baud rates by dividing the frequency of the input clock pulse at the ASCK20 pin.

It is recommended that the ceramic/crystal oscillation be used for the system clock in UART mode, because the baud rate error is greater if an RC oscillation is used.

(3) Three-wire serial I/O mode (switchable between MSB-first and LSB-first transmission)

This mode is used to transmit 8-bit data, using three lines: a serial clock ($\overline{\text{SCK20}}$) line and two serial data lines (SI20 and SO20).

As it supports simultaneous transmission and reception, three-wire serial I/O mode requires less processing time for data transmission than asynchronous serial interface mode.

Because, in three-wire serial I/O mode, it is possible to select whether 8-bit data transmission begins with the MSB or LSB, serial interface 20 can be connected to any device regardless of whether that device is designed for MSB-first or LSB-first transmission.

Three-wire serial I/O mode is useful for connecting peripheral I/O circuits and display controllers having conventional clock synchronous serial interfaces, such as those of the 75X/XL, 78K, and 17K series devices.

6.7.2 Serial interface 20 configuration

Serial interface 20 consists of the following hardware.

Table 6-9. Serial Interface 20 Configuration

Item	Configuration
Register	Transmission shift register 20 (TXS20) Reception shift register 20 (RXS20) Reception buffer register 20 (RXB20)
Control register	Serial operation mode register 20 (CSIM20) Asynchronous serial interface mode register 20 (ASIM20) Asynchronous serial interface status register 20 (ASIS20) Baud rate generator control register 20 (BRGC20)

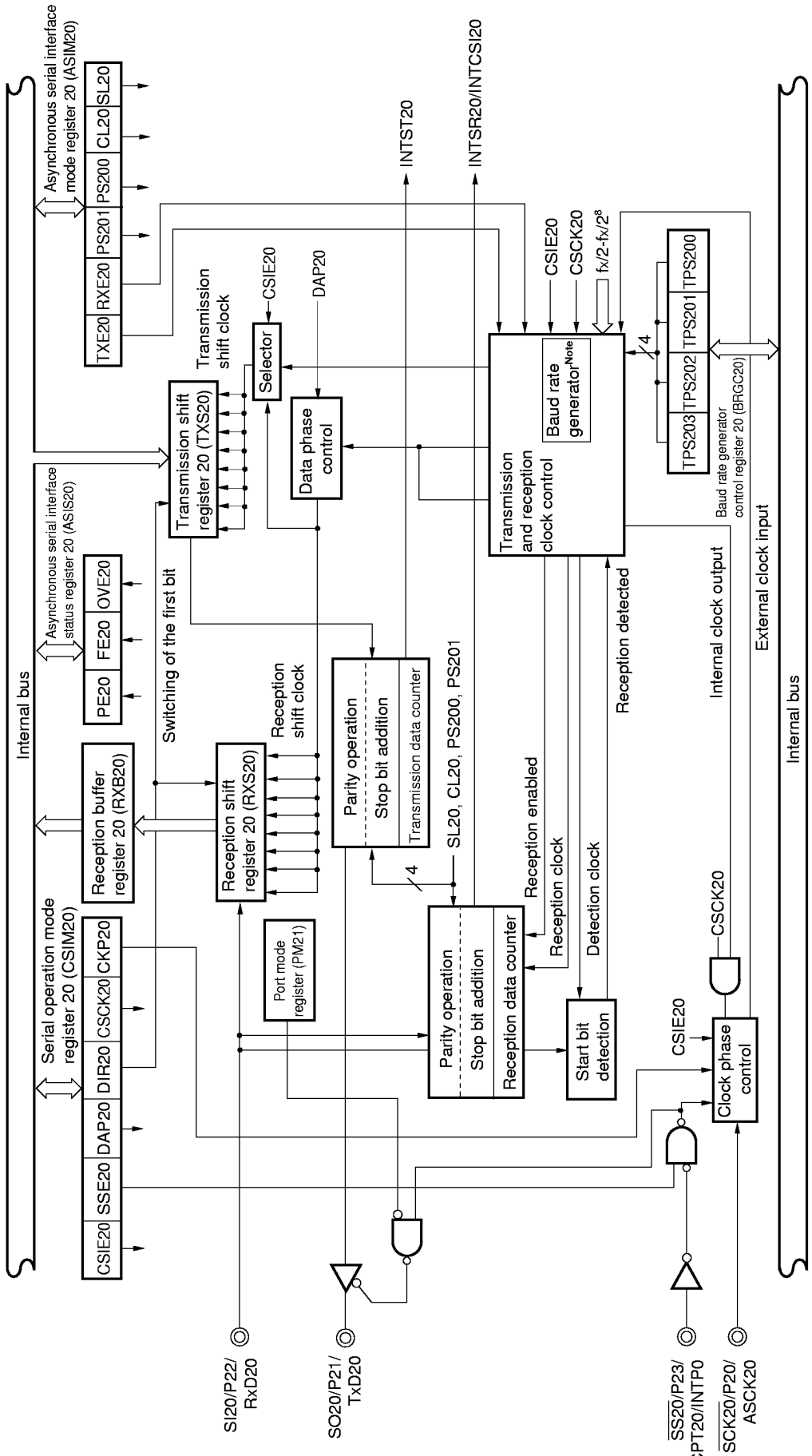


Figure 6-20. Block Diagram of Serial Interface 20

Note See **Figure 6-21** for the configuration of the baud rate generator.

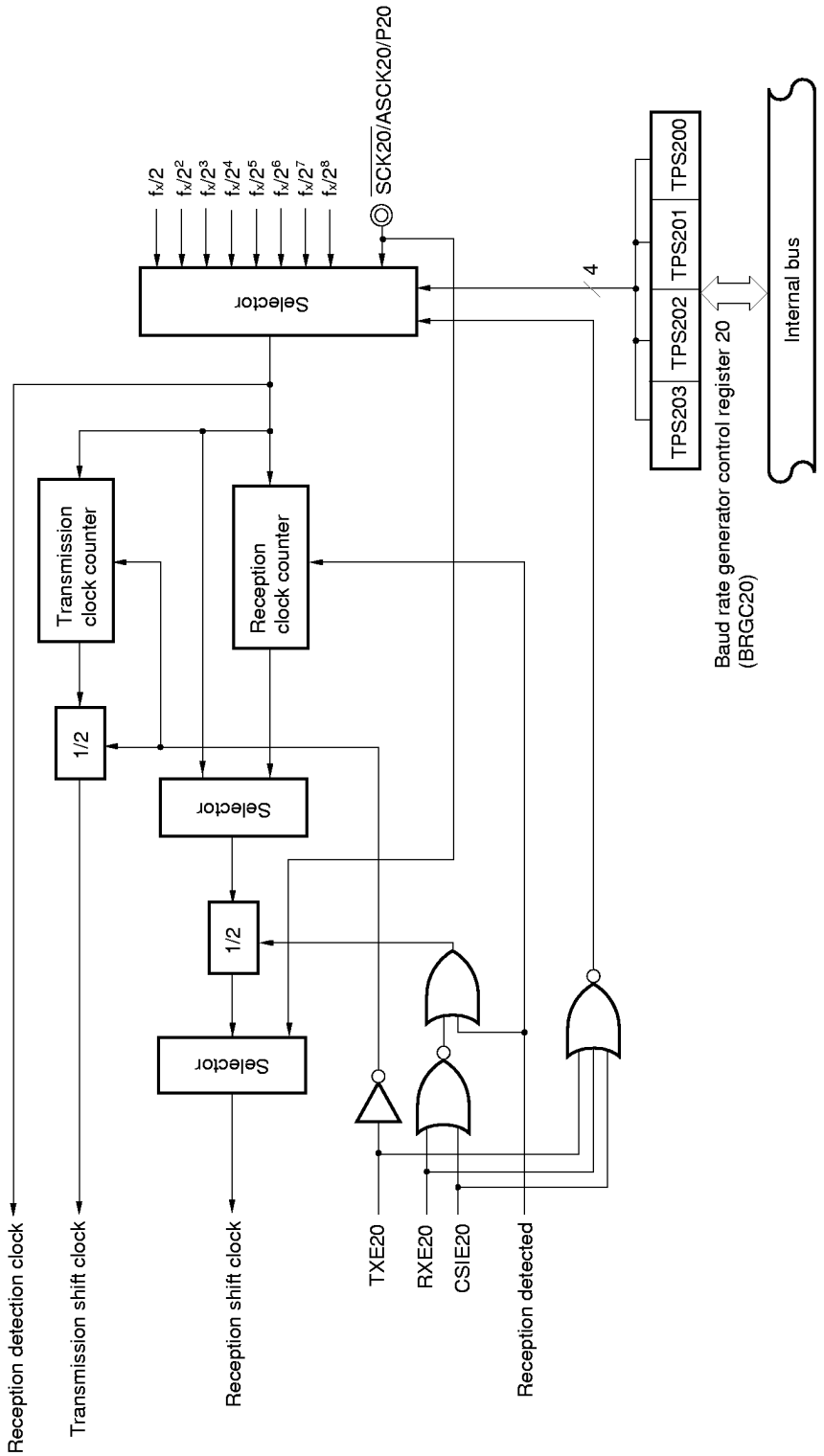


Figure 6-21. Block Diagram of Baud Rate Generator

(1) Transmission shift register 20 (TXS20)

TXS20 is a register in which transmission data is prepared. The transmission data is output from TXS20 bit-serially.

When the data length is seven bits, bits 0 to 6 of the data in TXS20 will be transmission data. Writing data to TXS20 triggers transmission.

TXS20 can be write-accessed, using an 8-bit memory manipulation instruction, but cannot be read-accessed.

A $\overline{\text{RESET}}$ input loads FFH into TXS20.

Caution Do not write to TXS20 during transmission.
TXS20 and the reception buffer register 20 (RXB20) are mapped at the same address, such that any attempt to read from TXS20 results in a value being read from RXB20.

(2) Reception shift register 20 (RXS20)

RXS20 is a register in which serial data, received at the RxD20 pin, is converted to parallel data. Once one entire byte has been received, RXS20 feeds the reception data to the reception buffer register 20 (RXB20).

RXS20 cannot be manipulated directly by a program.

(3) Reception buffer register 20 (RXB20)

RXB20 is used to hold reception data. Once RXS20 has received one entire byte of data, it feeds that data into RXB20.

When the data length is seven bits, the reception data is sent to bits 0 to 6 of RXB20, in which the MSB is fixed to 0.

RXB20 can be read-accessed, using an 8-bit memory manipulation instruction, but cannot be write-accessed.

A $\overline{\text{RESET}}$ input makes RXB20 undefined.

Caution RXB20 and the transmission shift register 20 (TXS20) are mapped at the same address, such that any attempt to write to RXB20 results in a value being written to TXS20.

(4) Transmission control circuit

The transmission control circuit controls transmission. For example, it adds start, parity, and stop bits to the data in transmission shift register 20 (TXS20), according to the setting of the asynchronous serial interface mode register 20 (ASIM20).

(5) Reception control circuit

The reception control circuit controls reception according to the setting of the asynchronous serial interface mode register 20 (ASIM20). It also checks for errors, such as parity errors, during reception. If an error is detected, the asynchronous serial interface status register 20 (ASIS20) is set according to the status of the error.

6.7.3 Serial interface 20 control registers

The following four types of registers are used to control serial interface 20.

- Serial operation mode register 20 (CSIM20)
- Asynchronous serial interface mode register 20 (ASIM20)
- Asynchronous serial interface status register 20 (ASIS20)
- Baud rate generator control register 20 (BRGC20)

(1) Serial operation mode register 20 (CSIM20)

CSIM20 is used to make the settings related to three-wire serial I/O mode.

CSIM20 is manipulated using a 1-bit or 8-bit memory manipulation instruction.

A $\overline{\text{RESET}}$ input clears CSIM20 to 00H.

Figure 6-22. Format of Serial Operation Mode Register 20

Symbol	⑦	6	5	4	3	2	1	0	Address	When reset	R/W
CSIM20	CSIE20	SSE20	0	0	DAP20	DIR20	CSCK20	CKP20	FF72H	00H	R/W

CSIE20	Three-wire serial I/O mode operation control
0	Operation disabled
1	Operation enabled

SSE20	$\overline{\text{SS20}}$-pin selection	Functions of the $\overline{\text{SS20}}$/P23 pin	Communication status
0	Not used	Port function	Communication enabled
1	Used	0	Communication enabled
		1	Communication disabled

DAP20	Three-wire serial I/O mode data phase selection
0	Outputs at the falling edge of $\overline{\text{SCK20}}$.
1	Outputs at the rising edge of $\overline{\text{SCK20}}$.

DIR20	First-bit specification
0	MSB
1	LSB

CSCK20	Three-wire serial I/O mode clock selection
0	External clock pulse input to the $\overline{\text{SCK20}}$ pin
1	Output of the dedicated baud rate generator

CKP20	Three-wire serial I/O mode clock phase selection
0	Clock is low active; $\overline{\text{SCK20}}$ is high in the idle state
1	Clock is high active; $\overline{\text{SCK20}}$ is low in the idle state

Cautions 1. Bits 4 and 5 must be fixed to 0.

2. CSIM20 must be cleared to 00H, if UART mode is selected.

(2) Asynchronous serial interface mode register 20 (ASIM20)

ASIM20 is used to make the settings related to serial interface 20 used in asynchronous serial interface mode.

ASIM20 is manipulated using a 1-bit or 8-bit memory manipulation instruction.

A $\overline{\text{RESET}}$ input clears ASIM20 to 00H.

Figure 6-23. Format of Asynchronous Serial Interface Mode Register 20

Symbol	⑦	⑥	5	4	3	2	1	0	Address	When reset	R/W
ASIM20	TXE20	RXE20	PS201	PS200	CL20	SL20	0	0	FF70H	00H	R/W

TXE20	Transmission control
0	Transmission disabled
1	Transmission enabled

RXE20	Reception control
0	Reception disabled
1	Reception enabled

PS201	PS200	Parity bit specification
0	0	No parity
0	1	At transmission, the parity bit is fixed to 0. At reception, a parity check is not made; no parity error is reported.
1	0	Odd parity
1	1	Even parity

CL20	Transmission data character length specification
0	7 bits
1	8 bits

SL20	Transmission data stop bit length
0	1 bit
1	2 bits

Cautions 1. Bits 0 and 1 must be fixed to 0.

2. If three-wire serial I/O mode is selected, ASIM20 must be cleared to 00H.

3. Switch operation mode from one mode to another after stopping both serial transmission and reception.

Table 6-10. Serial Interface 20 Operation Mode Settings

(1) Operation stopped mode

ASIM20		CSIM20			PM22	P22	PM21	P21	PM20	P20	First bit	Shift clock	P22/SI20/ RxD20 pin function	P21/SO20/ TxD20 pin function	P20/$\overline{\text{SCK20}}$/ ASCK20 pin function
TXE20	RXE20	CSIE20	DIR20	CSCK20											
0	0	0	×	×	×Note 1	×Note 1	×Note 1	×Note 1	×Note 1	×Note 1	-	-	P22	P21	P20
Other settings											Not to be set				

(2) Three-wire serial I/O mode

ASIM20		CSIM20			PM22	P22	PM21	P21	PM20	P20	First bit	Shift clock	P22/SI20/ RxD20 pin function	P21/SO20/ TxD20 pin function	P20/$\overline{\text{SCK20}}$/ ASCK20 pin function
TXE20	RXE20	CSIE20	DIR20	CSCK20											
0	0	1	0	0	1Note 2	×Note 2	0	1	1	×	MSB	External clock	SI20Note 2	SO20 (CMOS output)	$\overline{\text{SCK20}}$ input
				1					0	1		Internal clock			$\overline{\text{SCK20}}$ output
		1	1	0					1	×	LSB	External clock			$\overline{\text{SCK20}}$ input
				1					0	1		Internal clock			$\overline{\text{SCK20}}$ output
Other settings											Not to be set				

(3) Asynchronous serial interface mode

ASIM20		CSIM20			PM22	P22	PM21	P21	PM20	P20	First bit	Shift clock	P22/SI20/ RxD20 pin function	P21/SO20/ TxD20 pin function	P20/$\overline{\text{SCK20}}$/ ASCK20 pin function
TXE20	RXE20	CSIE20	DIR20	CSCK20											
1	0	0	0	0	×Note 1	×Note 1	0	1	1	×	LSB	External clock	P22	TxD20 (CMOS output)	ASCK20 input
									×Note 1	×Note 1		Internal clock			P20
0	1	0	0	0	1	×	×Note 1	×Note 1	1	×		External clock	RxD20	P21	ASCK20 input
									×Note 1	×Note 1		Internal clock			P20
1	1	0	0	0	1	×	0	1	1	×		External clock		TxD20 (CMOS output)	ASCK20 input
									×Note 1	×Note 1		Internal clock			P20
Other settings											Not to be set				

Notes 1. These pins can be used for port functions.

2. When only transmission is used, these pins can be used as P22 (CMOS input/output).

Remark ×: Don't care.

(3) Asynchronous serial interface status register 20 (ASIS20)

ASIS20 is used to display the type of a reception error, if it occurs while asynchronous serial interface mode is set.

ASIS20 is manipulated using a 1-bit or 8-bit memory manipulation instruction.

The contents of ASIS20 are undefined in three-wire serial I/O mode.

A $\overline{\text{RESET}}$ input clears ASIS20 to 00H.

Figure 6-24. Format of Asynchronous Serial Interface Status Register 20

Symbol	7	6	5	4	3	2	1	0	Address	When reset	R/W
ASIS20	0	0	0	0	0	PE20	FE20	OVE20	FF71H	00H	R

PE20	Parity error flag
0	No parity error has occurred.
1	A parity error has occurred (parity mismatch in transmission data).

FE20	Framing error flag
0	No framing error has occurred.
1	A framing error has occurred (no stop bit detected).Note 1

OVE20	Overrun error flag
0	No overrun error has occurred.
1	An overrun error has occurredNote 2 (Before data was read from the reception buffer register, the subsequent recepiton sequence was completed.)

Notes 1. Even if 2 is specified for the number of stop bits (using bit 2 (SL20) of ASIM20), only one stop bit is detected at reception.

2. After an overrun error occurs, read-access the reception buffer register 20 (RXB20). Otherwise, the overrun error will recur each time data is received.

(4) Baud rate generator control register 20 (BRGC20)

BRGC20 is used to specify the serial clock for serial interface 20.

BRGC20 is manipulated using an 8-bit memory manipulation instruction.

A $\overline{\text{RESET}}$ input clears BRGC20 to 00H.

Figure 6-25. Format of Baud Rate Generator Control Register 20

Symbol	7	6	5	4	3	2	1	0	Address	When reset	R/W
BRGC20	TPS203	TPS202	TPS201	TPS200	0	0	0	0	FF73H	00H	R/W

TPS203	TPS202	TPS201	TPS200	3-bit counter source clock selection	n
0	0	0	0	fx/2 (2.5 MHz)	1
0	0	0	1	$fx/2^2$ (1.25 MHz)	2
0	0	1	0	$fx/2^3$ (625 kHz)	3
0	0	1	1	$fx/2^4$ (313 kHz)	4
0	1	0	0	$fx/2^5$ (156 kHz)	5
0	1	0	1	$fx/2^6$ (78.1 kHz)	6
0	1	1	0	$fx/2^7$ (39.1 kHz)	7
0	1	1	1	$fx/2^8$ (19.5 kHz)	8
1	0	0	0	External clock pulse input at the ASCK20 pin[Note]	-
Other settings				Not to be set	

Note An external clock can be used only in UART mode.

Cautions 1. Any attempt to write to BRGC20 during communication adversely affects the output of the baud rate generator, thus hampering normal operation. Therefore, do not write to BRGC20 during communication.

2. Do not select n = 1 during operation at fx = 5.0 MHz, as n = 1 causes the rated baud rate to be exceeded.

3. When the external input clock is selected, set port mode register 2 (PM2) in input mode.

Remarks 1. fx: System clock oscillation frequency

2. The parenthesized values apply to operation at fx = 5.0 MHz.

The transmission and reception clock pulses used to generate the baud rate are obtained by dividing the frequency of the system clock pulse or a signal input to the ASCK20 pin.

(a) Generating transmission and reception clock pulses for baud rates based on the system clock

The frequency of the system clock is divided to generate the transmission and reception clock pulses. The baud rate generated based on the system clock is determined using the following expression.

$$[\text{Baud rate}] = \frac{fx}{2^{n+1} \times 8} \text{ [Hz]}$$

fx : System clock oscillation frequency

Table 6-11. Relationships between System Clock Frequencies and Baud Rates (Example)

Baud rate (bps)	n	BRGC20 setting	Error (%)	
			fx = 5.0 MHz	fx = 4.9152 MHz
1,200	8	70H	1.73	0
2,400	7	60H		
4,800	6	50H		
9,600	5	40H		
19,200	4	30H		
38,400	3	20H		
76,800	2	10H		

(b) Generating transmission and reception clock pulses for baud rates based on an external clock pulse received at the ASCK20 pin

The frequency of an external clock pulse received at the ASCK20 pin is used to generate the transmission and reception clock pulses. The baud rate generated based on the external clock pulse received at the ASCK20 pin is determined using the following expression.

$$[\text{Baud rate}] = \frac{f_{ASCK}}{16} \text{ [Hz]}$$

f_{ASCK}: Frequency of clock pulse received at the ASCK20 pin

Table 6-12. Relationships between ASCK20 Pin Input Frequencies and Baud Rates (When BRGC20 = 80H)

Baud rate (bps)	ASCK20 pin input frequency (kHz)
75	1.2
150	2.4
300	4.8
600	9.6
1,200	19.2
2,400	38.4
4,800	76.8
9,600	153.6
19,200	307.2
31,250	500.0
38,400	614.4

6.8 Multiplier

6.8.1 Multiplier function

The multiplier has the following function:

- Calculation of 8 bits × 8 bits = 16 bits

6.8.2 Multiplier configuration

(1) 16-bit multiplication result storage register 0 (MUL0)

This register stores 16-bit multiplication results.

This register holds the result of a multiplication after 16 CPU clock periods.

MUL0 is manipulated using a 16-bit memory manipulation instruction.

A $\overline{\text{RESET}}$ input makes MUL0 undefined.

Caution MUL0 is designed to be manipulated using a 16-bit memory manipulation instruction. It can also be manipulated using 8-bit memory manipulation instructions, however. When an 8-bit memory manipulation instruction is used to set MUL0, it must be in a direct addressing access mode.

(2) Multiplication data registers A and B (MRA0, MRB0)

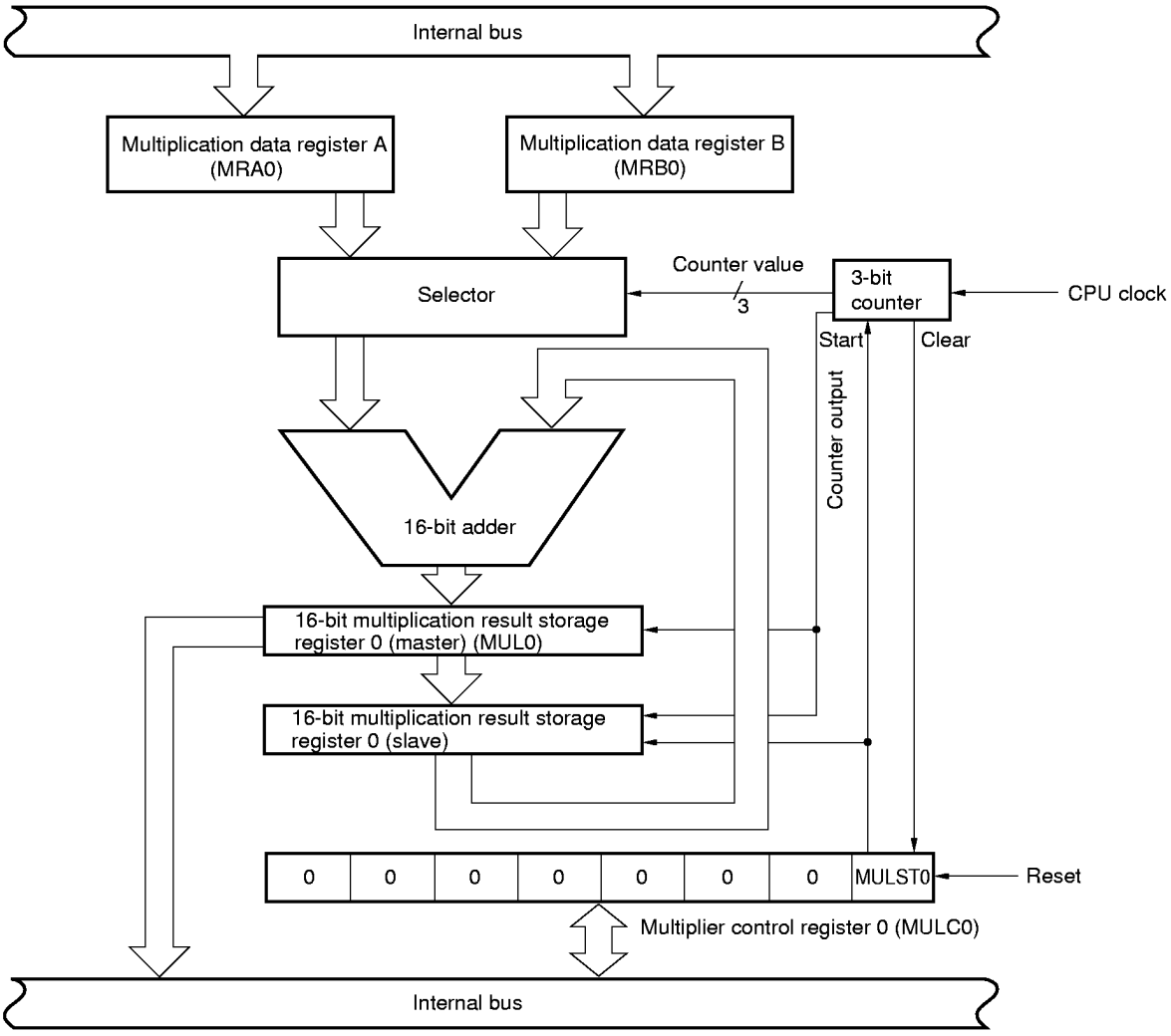
These registers store 8-bit multiplication data. The multiplier multiplies the value of MRA0 by the value of MRB0.

MRA0 and MRB0 are set using a 1-bit or 8-bit memory manipulation instruction.

A $\overline{\text{RESET}}$ input makes these registers undefined.

Figure 6-26 shows the block diagram of the multiplier.

Figure 6-26. Block Diagram of Multiplier



6.8.3 Multiplier control register

The following register is used to control the multiplier:

- Multiplier control register (MULC0)

MULC0 not only controls operations, but also indicates the operation status of the multiplier.
MULC0 is manipulated using a 1-bit or 8-bit memory manipulation instruction.
A $\overline{\text{RESET}}$ input clears MULC0 to 00H.

Figure 6-27. Format of Multiplier Control Register 0

Symbol	7	6	5	4	3	2	1	0	Address	When reset	R/W
MULC0	0	0	0	0	0	0	0	MULST0	FFD2H	00H	R/W

MULST0	Multiplier operation start control bit	Multiplier operation status
0	Operation is stopped after the counter is cleared to 0.	Operation is stopped.
1	Operation is enabled.	Operation is being executed.

Caution Bits 1 to 7 must be fixed to 0.

6.9 Low-Voltage Indicator

6.9.1 Low-voltage indicator function

The low-voltage indicator detects power supply voltage increases and decreases.

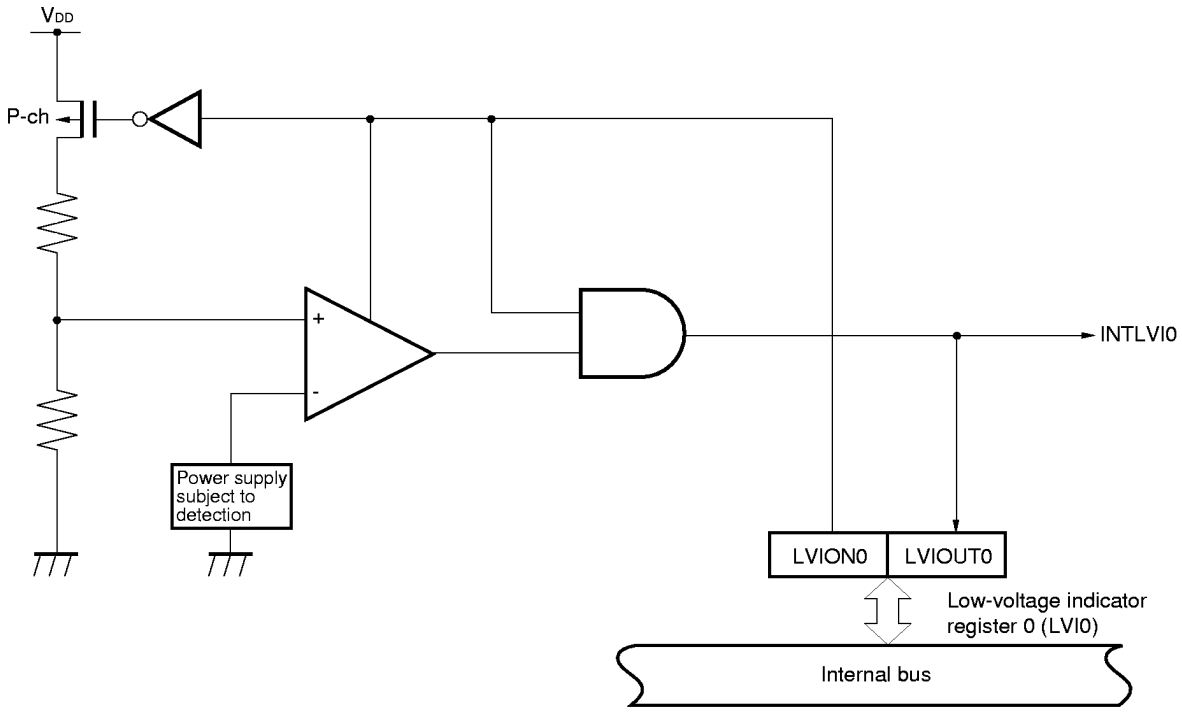
A power supply voltage decrease is detected when T_A = -40°C to +85°C and V_{DD} = 3.55 to 4.45 V. However, an abrupt power supply voltage change cannot be detected.

The result of a comparison made between a power supply voltage and detected voltage can be read using an instruction.

Based on the result of a comparison made between a power supply voltage and detected voltage, an interrupt request signal can be generated.

6.9.2 Low-voltage indicator configuration

Figure 6-28. Block Diagram of Low-Voltage Indicator



6.9.3 Low-voltage indicator control register

The following register is used to control the low-voltage indicator.

- Low-voltage indicator register 0 (LVI0)
  LVI0 is used to enable and disable low-voltage indicator operation.
  LVI0 is manipulated using a 1-bit or 8-bit memory manipulation instruction.
  A $\overline{\text{RESET}}$ input clears LVI0 to 00H.

Figure 6-29. Format of Low-Voltage Indicator Register 0

Symbol	7	6	5	4	3	2	1	0	Address	When reset	R/W
LVI0	LVION0	0	0	0	0	0	0	LVIOUT0	FFDFH	00H	R/W Note

LVION0	Low-voltage indiator operation start control bit
0	Low-voltage indicator operation is disabled.
1	Low-voltage indicator operation is enabled.

LVIOUT0	Low-voltage indicator result output bit
0	When power supply voltage ≤ detected voltage, or the low-voltage indicator is stopped
1	When power supply voltage > detected voltage

Note Bit 0 is read-only.

Cautions 1. Bits 1 to 6 must be fixed to 0.

2. The operation of the low-voltage indicator does not become stable for the operation start time (500 μs) after the indicator is enabled.

3. Before the generation of an interrupt can be specified using the low-voltage indicator, the current power supply voltage status must be checked by reading the value of the detection result output bit.

4. If operation is enabled at standby time (HALT or STOP), the standby state can be released by an interrupt request signal.

7. INTERRUPT FUNCTIONS

7.1 Interrupt Function Types

Two types of interrupt function are supported.

(1) Nonmaskable interrupt

A nonmaskable interrupt request is accepted unconditionally, that is, even when interrupts are disabled. A nonmaskable interrupt takes precedence over all other interrupts; it is not subjected to interrupt priority control.

A nonmaskable interrupt causes the standby release signal to be generated.

The μPD789144, μPD789146, μPD789154, and μPD789156 support one nonmaskable interrupt source namely, the watchdog timer interrupt.

(2) Maskable interrupt

Maskable interrupts are those which are subjected to mask control. If two or more maskable interrupts occur simultaneously, the priority listed in Table 7-1 applies.

The maskable interrupts cause the standby release signal to be generated.

The maskable interrupts supported by the μPD789144, μPD789146, μPD789154, and μPD789156 include 3 external interrupt sources and 8 internal interrupt sources.

7.2 Interrupt Sources and Configuration

The μPD789144, μPD789146, μPD789154, and μPD789156 each support a total of 12 maskable and nonmaskable interrupt sources. (See **Table 7-1**.)

Table 7-1. Interrupt Sources

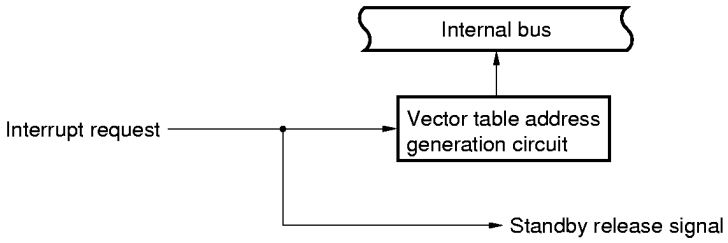
Interrupt type	Priority[Note 1]	Interrupt source		Internal/external	Vector table address	Basic configuration type[Note 2]
		Name	Trigger			
Nonmaskable interrupt	-	INTWDT	Watchdog timer overflow (when watchdog timer mode 1 is selected)	Internal	0004H	(A)
Maskable interrupt	0	INTWDT	Watchdog timer overflow (when the interval timer mode is selected)			(B)
	1	INTP0	Pin input edge detection	External	0006H	(C)
	2	INTP1			0008H	
	3	INTP2			000AH	
	4	INTSR20	End of UART reception on serial interface 20	Internal	000EH	(B)
		INTCSI20	End of three-wire SIO transfer reception on serial interface 20			
	5	INTST20	End of UART transmission on serial interface 20		000EH	
	6	INTTM80	Generation of match signal for 8-bit timer/event counter 80		0010H	
	7	INTTM20	Generation of match signal for 16-bit timer counter 20		0012H	
	8	INTAD0	A/D conversion completion signal		0014H	
	9	INTLVI0	LVI interrupt signal		0016H	
	10	INTEE1	EEPROM interrupt end signal		0018H	

Notes 1. The priority regulates which maskable interrupt is higher, when two or more maskable interrupts are requested simultaneously. Zero signifies the highest priority, while 10 is the lowest.

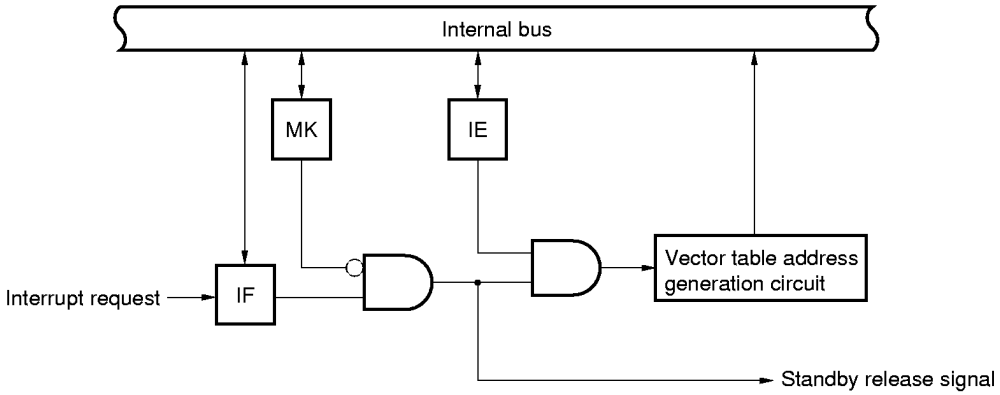
2. Basic configuration types (A), (B), and (C) correspond to (A), (B), and (C) in Figure 7-1, respectively.

Figure 7-1. Basic Configuration of Interrupt Functions

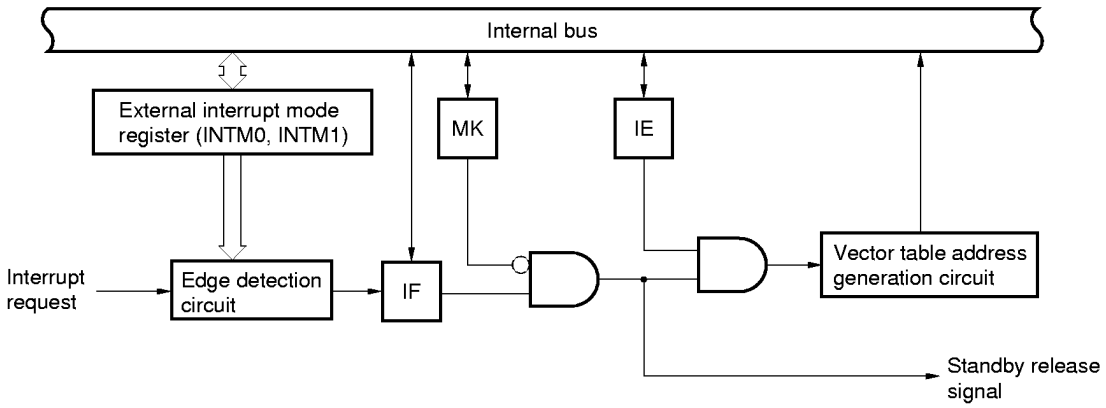
(A) Internal nonmaskable interrupt



(B) Internal maskable interrupt



(C) External maskable interrupt



IF : Interrupt request flag
IE : Interrupt enable flag
MK : Interrupt mask flag

7.3 Interrupt Function Control Registers

The interrupt functions are controlled by the following registers.

- Interrupt request flag registers 0 and 1 (IF0 and IF1)
- Interrupt mask flag registers 0 and 1 (MK0 and MK1)
- External interrupt mode registers 0 and 1 (INTM0 and INTM1)
- Program status word (PSW)

Table 7-2 lists interrupt requests, the corresponding interrupt request flags, and interrupt mask flags.

Table 7-2. Interrupt Request Signals and Corresponding Flags

Interrupt request signal	Interrupt request flag	Interrupt mask flag
INTWDT	TMIF4	TMMK4
INTP0	PIF0	PMK0
INTP1	PIF1	PMK1
INTP2	PIF2	PMK2
INTSR20/INTCSI20	SRIF20	SRMK20
INTST20	STIF20	STMK20
INTTM80	TMIF80	TMMK80
INTTM20	TMIF20	TMMK20
INTAD0	ADIF0	ADMK0
INTLVI0	LVIIF0	LVIMK0
INTEE1	EEIF1	EEMK1

(1) Interrupt request flag registers 0 and 1 (IF0 and IF1)

An interrupt request flag is set (1), when the corresponding interrupt request is issued, or when the related instruction is executed. It is cleared (0), when the interrupt request is accepted, when a $\overline{\text{RESET}}$ signal is input, or when a related instruction is executed.

IF0 and IF1 are manipulated using a 1-bit or 8-bit memory manipulation instruction.

A $\overline{\text{RESET}}$ input clears IF0 and IF1 to 00H.

Figure 7-2. Format of Interrupt Request Flag Register

Symbol	⑦	⑥	⑤	④	③	②	①	⓪	Address	When reset	R/W
IF0	TMIF20	TMIF80	STIF20	SRIF20	PIF2	PIF1	PIF0	TMIF4	FFE0H	00H	R/W

	7	6	5	4	3	②	①	⓪	Address	When reset	R/W
IF1	0	0	0	0	0	EEIF1	LVIIF0	ADIF0	FFE1H	00H	R/W

××IF×	Interrupt request flag
0	No interrupt request signal has been issued.
1	An interrupt request signal has been issued; an interrupt request has been made.

Cautions 1. The TMIF4 flag can be read- and write-accessed only when the watchdog timer is being used as an interval timer. It must be cleared to 0 if the watchdog timer is used in watchdog timer mode 1 or 2.

2. When port 2 is being used as an output port, and its output level is changed, an interrupt request flag is set, because this port is also used as an external interrupt input. To use port 2 in output mode, therefore, the interrupt mask flag must be set to 1 in advance.

(2) Interrupt mask flag registers 0 and 1 (MK0 and MK1)

The interrupt mask flags are used to enable and disable the corresponding maskable interrupts.

MK0 and MK1 are manipulated using a 1-bit or 8-bit memory manipulation instruction.

A $\overline{\text{RESET}}$ input loads FFH into MK0 and MK1.

Figure 7-3. Format of Interrupt Mask Flag Register

Symbol	⑦	⑥	⑤	④	③	②	①	⓪	Address	When reset	R/W
MK0	TMMK20	TMMK80	STMK20	SRMK20	PMK2	PMK1	PMK0	TMMK4	FFE4H	FFH	R/W

	7	6	5	4	3	②	①	⓪	Address	When reset	R/W
MK1	1	1	1	1	1	EEMK1	LVIMK0	ADMK0	FFE5H	FFH	R/W

××MK×	Interrupt handling control
0	Enable interrupt handling.
1	Disable interrupt handling.

Cautions 1. When the watchdog timer is being used in watchdog timer mode 1 or 2, any attempt to read TMMK4 flag results in an undefined value being detected.

2. When port 2 is being used as an output port, and its output level is changed, an interrupt request flag is set, because this port is also used as an external interrupt input. To use port 2 in output mode, therefore, the interrupt mask flag must be set to 1 in advance.

(3) External interrupt mode register 0 (INTM0)

INTM0 is used to specify an effective edge for INTP0 to INTP2.

INTM0 is manipulated using an 8-bit memory manipulation instruction.

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

Figure 7-4. Format of External Interrupt Mode Register 0

Symbol	7	6	5	4	3	2	1	0	Address	When reset	R/W
INTM0	ES21	ES20	ES11	ES10	ES01	ES00	0	0	FFECH	00H	R/W

ES21	ES20	INTP2 effective edge selection
0	0	Falling edge
0	1	Rising edge
1	0	Not to be set
1	1	Both rising and falling edges

ES11	ES10	INTP1 effective edge selection
0	0	Falling edge
0	1	Rising edge
1	0	Not to be set
1	1	Both rising and falling edges

ES01	ES00	INTP0 effective edge selection
0	0	Falling edge
0	1	Rising edge
1	0	Not to be set
1	1	Both rising and falling edges

Cautions 1. Bits 0 and 1 must be fixed to 0.

2. Before setting INTM0, set the corresponding interrupt mask flag to 1 (××MK× = 1) to disable interrupts.

To enable interrupts, clear (0) the corresponding interrupt request flag (××IF× = 0), then the corresponding interrupt mask flag (××MK× = 0).

(4) External interrupt mode register 1 (INTM1)

INTM1 is used to specify an effective edge for INTLVI0.

INTM1 is manipulated using an 8-bit memory manipulation instruction.

A $\overline{\text{RESET}}$ input clears INTM1 to 00H.

Figure 7-5. Format of External Interrupt Mode Register 1

Symbol	7	6	5	4	3	2	1	0	Address	When reset	R/W
INTM1	0	0	0	0	0	0	ES31	ES30	FFEDH	00H	R/W

ES31	ES30	INTLVI0 effective edge selection
0	0	Falling edge
0	1	Rising edge
1	0	Not to be set
1	1	Both rising and falling edges

Cautions 1. Bits 2 to 7 must be fixed to 0.

2. Before setting INTM1, set LVIMK0 (bit 1 of interrupt mask flag register 1 (MK)) to 1 (LVIMK0 = 1) to disable interrupts.

To enable interrupts, clear LVIIF0 (bit 1 of interrupt request flag 1 (IF1)) to 0 (LVIIF0 = 0), then LVIMK0.

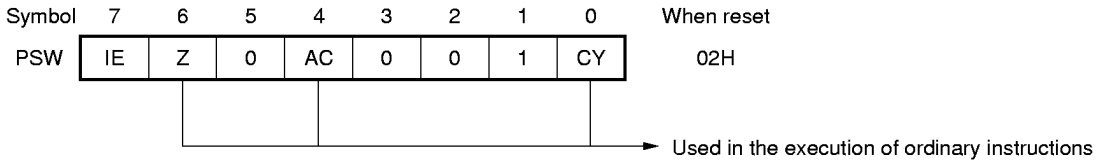
(5) Program status word (PSW)

The program status word is used to hold the instruction execution result and the current status of the interrupt requests. The IE flag, used to enable and disable maskable interrupts, is mapped to the PSW.

The PSW can be read- and write-accessed in 8-bit units, as well as in 1-bit units when using bit manipulation instructions and dedicated instructions (EI and DI). When a vector interrupt is accepted, the PSW is automatically saved to a stack, and the IE flag is reset (0). It is reset from the stack with the RETI and POP PSW instructions.

A $\overline{\text{RESET}}$ input loads 02H into the PSW.

Figure 7-6. Program Status Word Configuration



IE	Whether to enable/disable interrupt acceptance
0	Disable
1	Enable

8. STANDBY FUNCTION

8.1 Standby Function

The standby function is supported to minimize the system's power consumption. There are the following two standby modes.

(1) HALT mode

This mode is set when the HALT instruction is executed. HALT mode stops the operation clock of the CPU. The system clock oscillation circuit continues oscillating. This mode does not reduce the power consumption as much as STOP mode, but is useful for resuming processing immediately when an interrupt request is generated, or for intermittent operations.

(2) STOP mode

This mode is set when the STOP instruction is executed. STOP mode stops the main system clock oscillation circuit and stops the entire system. The power consumption of the CPU can be substantially reduced in this mode.

The low voltage (V_{DD} = 1.8 V) of the data memory can be retained. Therefore, this mode is useful for retaining the contents of the data memory at an extremely low current.

STOP mode can be released by an interrupt request, so that this mode can be used for intermittent operation. However, some time is required until the system clock oscillation circuit settles after STOP mode has been released. If processing must be resumed immediately by using an interrupt request, therefore, use HALT mode.

In both modes, the previous contents of the registers, flags, and data memory before setting the standby mode are all retained. In addition, the statuses of the output latch of the I/O ports and output buffer are also retained.

Caution Before shifting to STOP mode, first stop the operation of the peripheral hardware, then execute the STOP instruction.

Table 8-1. Operation Statuses in HALT Mode

Item	HALT mode operation status
Clock generator	Enables system clock oscillation. Stops clock supply to the CPU.
CPU	Operation disabled
EEPROM	Operation enabled
Port (output latch)	Remains in the state existing before the selection of HALT mode.
16-bit timer counter	Operation enabled
8-bit timer/event counter	Operation enabled
Watchdog timer	Operation enabled
Serial interface	Operation enabled
A/D converter	Operation disabled
Multiplier	Operation disabled
Low-voltage indicator	Operation enabled[Note]
External interrupt	Operation enabled[Note]

Note Maskable interrupt that is not masked

Table 8-2. Operation Statuses in STOP Mode

Item	STOP mode operation status
Clock generator	Stops system clock oscillation.
CPU	Operation disabled
EEPROM	Operation disabled
Port (output latch)	Remains in the state existing before the selection of STOP mode.
16-bit timer counter	Operation disabled
8-bit timer/event counter	Operation enabled[Note 1]
Watchdog timer	Operation disabled
Serial interface	Operation enabled[Note 2]
A/D converter	Operation disabled
Multiplier	Operation disabled
Low-voltage indicator	Operation enabled[Note 3]
External interrupt	Operation enabled[Note 3]

Notes 1. Operation is enabled only when TI80 is selected as the count clock.

2. Operation is enabled in both three-wire serial I/O and UART modes while an external clock is being used.

3. Maskable interrupt that is not masked

8.2 Standby Function Control Register

The oscillation settling time selection register (OSTS) is used to control the wait time, from the time STOP mode is deselected by an interrupt request, until oscillation settles.

The OSTS is manipulated using an 8-bit memory manipulation instruction.

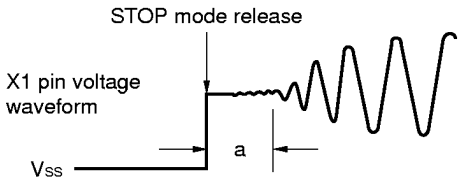
A $\overline{\text{RESET}}$ input loads 04H into the OSTS. If a $\overline{\text{RESET}}$ input is used to deselect STOP mode, the time required for oscillation to settle will be 2^{15}/fx, rather than 2^{17}/fx.

Figure 8-1. Format of Oscillation Settling Time Selection Register

Symbol	7	6	5	4	3	2	1	0	Address	When reset	R/W
OSTS	0	0	0	0	0	OSTS2	OSTS1	OSTS0	FFFAH	04H	R/W

OSTS2	OSTS1	OSTS0	Oscillation settling time selection
0	0	0	2^{12}/fx (819 μs)
0	1	0	2^{15}/fx (6.55 ms)
1	0	0	2^{17}/fx (26.2 ms)
Other settings			Not to be set

Caution The wait time required to deselect STOP mode does not include the time ("a" in the following figure) required for the clock oscillation to settle after STOP mode is deselected, regardless of whether STOP mode is deselected by a $\overline{\text{RESET}}$ input or interrupt.



Remarks 1. fx: System clock oscillation frequency

2. The parenthesized values apply to operation at fx = 5.0 MHz.

9. RESET FUNCTIONS

The μPD789144, μPD789146, μPD789154, and μPD789156 can be reset using the following signals.

(1) External reset signal input to the $\overline{\text{RESET}}$ pin
(2) Internal reset signal generated upon the elapse of the period set in the watchdog timer, used for detecting an unintended program loop

The external and internal reset signals are functionally equivalent. When $\overline{\text{RESET}}$ is input, they cause program execution to begin at the addresses indicated at addresses 0000H and 0001H, respectively.

If a low level signal is applied to the $\overline{\text{RESET}}$ pin, or if the watchdog timer overflows, a reset occurs, causing each piece of the hardware to enter the states listed in Table 9-1. While a reset signal is being input, or while the oscillation frequency is settling immediately after the end of a reset sequence, each pin remains in the high-impedance state.

If a high level signal is applied to the $\overline{\text{RESET}}$ pin, a reset sequence is terminated, and program execution begins once the oscillation settling time elapses. A watchdog timer overflow-based reset sequence is terminated automatically. Similarly, program execution begins upon the elapse of the oscillation settling time.

Cautions 1. To use an external reset sequence, supply a low level signal to the $\overline{\text{RESET}}$ pin and maintain the signal for at least 10 μs.

2. When a reset is used to deselect STOP mode, the information related to STOP mode is held during the reset sequence, that is, while the reset signal is applied. The port pins remain in the high-impedance state, however.

Figure 9-1. Block Diagram of Reset Function

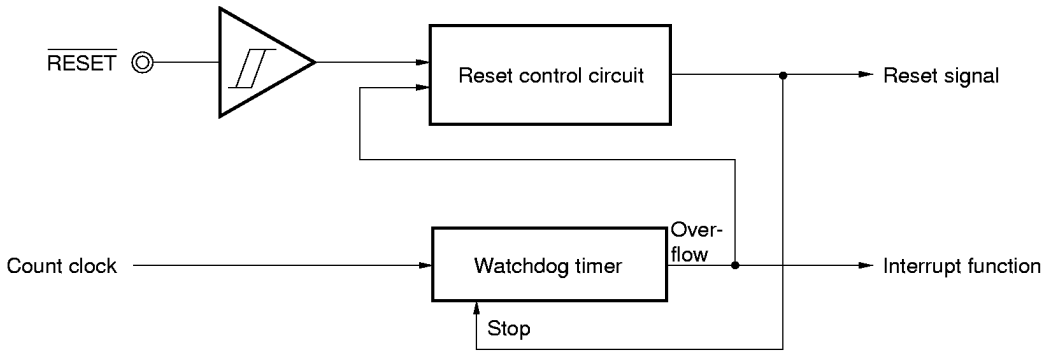


Table 9-1. State of the Hardware after a Reset (1/2)

Hardware		State after reset
Program counter (PC)[Note 1]		Loaded with the contents of the reset vector table (0000H, 0001H)
Stack pointer (SP)		Undefined
Program status word (PSW)		02H
RAM	Data memory	Undefined[Note 2]
	General-purpose register	Undefined[Note 2]
EEPROM	Write control register (EEWC1)	08H
Ports (P0 to P2, P5) (output latch)		00H
Port mode registers (PM0 to PM2, PM5)		FFH
Pull-up resistor option register 0 (PU0)		00H
Pull-up resistor option register B2 (PUB2)		00H
Processor clock control register (PCC)		02H
Oscillation settling time selection register (OSTS)		04H
16-bit timer/counter	Timer register (TM20)	0000H
	Compare register (CR20)	FFFFH
	Mode control register (TMC20)	00H
	Capture register (TCP20)	Undefined
8-bit timer/event counter	Timer register (TM80)	00H
	Compare register (CR80)	Undefined
	Mode control register (TMC80)	00H
Watchdog timer	Timer clock selection register (TCL2)	00H
	Mode register (WDTM)	00H
A/D converter	Mode register (ADM0)	00H
	Input selection register (ADS0)	00H
	Conversion result register (ADCR0)	Undefined

Notes 1. While a reset signal is being input, and during the oscillation settling period, the contents of the PC will be undefined, while the remainder of the hardware will be the same as after the reset.

2. In standby mode, the RAM enters the hold state after a reset.

Table 9-1. State of the Hardware after a Reset (2/2)

Hardware		State after reset
Serial interface	Mode register (CSIM20)	00H
	Asynchronous serial interface mode register (ASIM20)	00H
	Asynchronous serial interface status register (ASIS20)	00H
	Baud rate generator control register (BRGC20)	00H
	Transmission shift register (TXS20)	FFH
	Reception buffer register (RXB20)	Undefined
Multiplier	16-bit multiplication result storage register (MUL0)	Undefined
	Data register A (MRA0)	Undefined
	Data register B (MRB0)	Undefined
	Control register (MULC0)	00H
Low-voltage indicator	Low-voltage indicator register (LVI0)	00H
Interrupts	Request flag registers (IF0 and IF1)	00H
	Mask flag registers (MK0 and MK1)	FFH
	External interrupt mode registers (INTM0 and INTM1)	00H

10. MASK OPTIONS

Table 10-1. Selection of Mask Option for Pins

Pin	Mask option
P50 to P53	Whether to use an on-chip pull-up resistor can be specified in 1-bit units.

For P50 to P53 (port 5), the on-chip pull-up resistor can be specified by a mask option. The mask option can be specified in 1-bit units.

11. INSTRUCTION SET OVERVIEW

The instruction set for the μPD789144, μPD789146, μPD789154, and μPD789156 is listed later.

11.1 Legend

11.1.1 Operand formats and descriptions

The description made in the operand field of each instruction conforms to the operand format for the instructions listed below (the details conform with the assembly specification). If more than one operand format is listed for an instruction, one is selected. Uppercase letters, #, !, $, and a pair of [and] are used to specify keywords, which must be written exactly as they appear. The meanings of these special characters are as follows:

- #: Immediate data specification
- $: Relative address specification
- !: Absolute address specification
- [and]: Indirect address specification

Immediate data should be described using appropriate values or labels. The specification of values and labels must be accompanied by #, !, $, or a pair of [and].

Operand registers, expressed as r or rp in the formats, can be described using both functional names (X, A, C, etc.) and absolute names (R0, R1, R2, and other names listed in Table 11-1).

Table 11-1. Operand Formats and Descriptions

Format	Description
r	X (R0), A (R1), C (R2), B (R3), E (R4), D (R5), L (R6), H (R7)
rp	AX (RP0), BC (RP1), DE (RP2), HL (RP3)
sfr	Special function register symbol
saddr	FE20H to FF1FH: Immediate data or label
saddrp	FE20H to FF1FH: Immediate data or label (even addresses only)
addr16	0000H to FFFFH: Immediate data or label (only even address for 16-bit data transfer instructions)
addr5	0040H to 007FH: Immediate data or label (even addresses only)
word	16-bit immediate data or label
byte	8-bit immediate data or label
bit	3-bit immediate data or label

Remark For the special function register symbols, see **Table 4-1**.

11.1.2 Descriptions of the operation field

A : A register (8-bit accumulator)
X : X register
B : B register
C : C register
D : D register
E : E register
H : H register
L : L register
AX : AX register pair (16-bit accumulator)
BC : BC register pair
DE : DE register pair
HL : HL register pair
PC : Program counter
SP : Stack pointer
PSW : Program status word
CY : Carry flag
AC : Auxiliary carry flag
Z : Zero flag
IE : Interrupt request enable flag
NMIS : Flag to indicate that a nonmaskable interrupt is being handled
() : Contents of a memory location indicated by a parenthesized address or register name
X_H, X_L : Upper and lower 8 bits of a 16-bit register
∧ : Logical product (AND)
∨ : Logical sum (OR)
∀ : Exclusive OR
— : Inverted data
addr16 : 16-bit immediate data or label
jdisp8 : Signed 8-bit data (displacement value)

11.1.3 Description of the flag operation field

(blank) : No change
0 : To be cleared to 0
1 : To be set to 1
× : To be set or cleared according to the result
R : To be restored to the previous value

11.2 Operations

Mnemonic	Operand	Byte	Clock	Operation	Flag Z	Flag AC	Flag CY
MOV	r, #byte	3	6	r ← byte			
	saddr, #byte	3	6	(saddr) ← byte			
	sfr, #byte	3	6	sfr ← byte			
	A, r Note 1	2	4	A ← r			
	r, A Note 1	2	4	r ← A			
	A, saddr	2	4	A ← (saddr)			
	saddr, A	2	4	(saddr) ← A			
	A, sfr	2	4	A ← sfr			
	sfr, A	2	4	sfr ← A			
	A, !addr16	3	8	A ← (addr16)			
	!addr16, A	3	8	(addr16) ← A			
	PSW, #byte	3	6	PSW ← byte	×	×	×
	A, PSW	2	4	A ← PSW			
	PSW, A	2	4	PSW ← A	×	×	×
	A, [DE]	1	6	A ← (DE)			
	[DE], A	1	6	(DE) ← A			
	A, [HL]	1	6	A ← (HL)			
	[HL], A	1	6	(HL) ← A			
	A, [HL + byte]	2	6	A ← (HL + byte)			
	[HL + byte], A	2	6	(HL + byte) ← A			
XCH	A, X	1	4	A ↔ X			
	A, r Note 2	2	6	A ↔ r			
	A, saddr	2	6	A ↔ (saddr)			
	A, sfr	2	6	A ↔ (sfr)			
	A, [DE]	1	8	A ↔ (DE)			
	A, [HL]	1	8	A ↔ (HL)			
	A, [HL + byte]	2	8	A ↔ (HL + byte)			
MOVW	rp, #word	3	6	rp ← word			
	AX, saddrp	2	6	AX ← (saddrp)			
	saddrp, AX	2	8	(saddrp) ← AX			
	AX, rp Note 3	1	4	AX ← rp			
	rp, AX Note 3	1	4	rp ← AX			
XCHW	AX, rp Note 3	1	8	AX ↔ rp			

Notes 1. Except when r = A.

2. Except when r = A or X.

3. Only when rp = BC, DE, or HL.

Remark The instruction clock cycle is based on the CPU clock (f_{CPU}), specified in the processor clock control register (PCC).

Mnemonic	Operand	Byte	Clock	Operation	Flag		
					Z	AC	CY
ADD	A, #byte	2	4	A, CY ← A + byte	×	×	×
	saddr, #byte	3	6	(saddr), CY ← (saddr) + byte	×	×	×
	A, r	2	4	A, CY ← A + r	×	×	×
	A, saddr	2	4	A, CY ← A + (saddr)	×	×	×
	A, !addr16	3	8	A, CY ← A + (addr16)	×	×	×
	A, [HL]	1	6	A, CY ← A + (HL)	×	×	×
	A, [HL + byte]	2	6	A, CY ← A + (HL + byte)	×	×	×
ADDC	A, #byte	2	4	A, CY ← A + byte + CY	×	×	×
	saddr, #byte	3	6	(saddr), CY ← (saddr) + byte + CY	×	×	×
	A, r	2	4	A, CY ← A + r + CY	×	×	×
	A, saddr	2	4	A, CY ← A + (saddr) + CY	×	×	×
	A, !addr16	3	8	A, CY ← A + (addr16) + CY	×	×	×
	A, [HL]	1	6	A, CY ← A + (HL) + CY	×	×	×
	A, [HL + byte]	2	6	A, CY ← A + (HL + byte) + CY	×	×	×
SUB	A, #byte	2	4	A, CY ← A – byte	×	×	×
	saddr, #byte	3	6	(saddr), CY ← (saddr) – byte	×	×	×
	A, r	2	4	A, CY ← A – r	×	×	×
	A, saddr	2	4	A, CY ← A – (saddr)	×	×	×
	A, !addr16	3	8	A, CY ← A – (addr16)	×	×	×
	A, [HL]	1	6	A, CY ← A – (HL)	×	×	×
	A, [HL + byte]	2	6	A, CY ← A – (HL + byte)	×	×	×
SUBC	A, #byte	2	4	A, CY ← A – byte – CY	×	×	×
	saddr, #byte	3	6	(saddr), CY ← (saddr) – byte – CY	×	×	×
	A, r	2	4	A, CY ← A – r – CY	×	×	×
	A, saddr	2	4	A, CY ← A – (saddr) – CY	×	×	×
	A, !addr16	3	8	A, CY ← A – (addr16) – CY	×	×	×
	A, [HL]	1	6	A, CY ← A – (HL) – CY	×	×	×
	A, [HL + byte]	2	6	A, CY ← A – (HL + byte) – CY	×	×	×
AND	A, #byte	2	4	A ← A ∧ byte	×		
	saddr, #byte	3	6	(saddr) ← (saddr) ∧ byte	×		
	A, r	2	4	A ← A ∧ r	×		
	A, saddr	2	4	A ← A ∧ (saddr)	×		
	A, !addr16	3	8	A ← A ∧ (addr16)	×		
	A, [HL]	1	6	A ← A ∧ (HL)	×		
	A, [HL + byte]	2	6	A ← A ∧ (HL + byte)	×		

Remark The instruction clock cycle is based on the CPU clock (f_{CPU}), specified in the processor clock control register (PCC).

Mnemonic	Operand	Byte	Clock	Operation	Flag		
					Z	AC	CY
OR	A, #byte	2	4	A ← A ∨ byte	×		
	saddr, #byte	3	6	(saddr) ← (saddr) ∨ byte	×		
	A, r	2	4	A ← A ∨ r	×		
	A, saddr	2	4	A ← A ∨ (saddr)	×		
	A, !addr16	3	8	A ← A ∨ (addr16)	×		
	A, [HL]	1	6	A ← A ∨ (HL)	×		
	A, [HL + byte]	2	6	A ← A ∨ (HL + byte)	×		
XOR	A, #byte	2	4	A ← A ⊻ byte	×		
	saddr, #byte	3	6	(saddr) ← (saddr) ⊻ byte	×		
	A, r	2	4	A ← A ⊻ r	×		
	A, saddr	2	4	A ← A ⊻ (saddr)	×		
	A, !addr16	3	8	A ← A ⊻ (addr16)	×		
	A, [HL]	1	6	A ← A ⊻ (HL)	×		
	A, [HL + byte]	2	6	A ← A ⊻ (HL + byte)	×		
CMP	A, #byte	2	4	A – byte	×	×	×
	saddr, #byte	3	6	(saddr) – byte	×	×	×
	A, r	2	4	A – r	×	×	×
	A, saddr	2	4	A – (saddr)	×	×	×
	A, !addr16	3	8	A – (addr16)	×	×	×
	A, [HL]	1	6	A – (HL)	×	×	×
	A, [HL + byte]	2	6	A – (HL + byte)	×	×	×
ADDW	AX, #word	3	6	AX, CY ← AX + word	×	×	×
SUBW	AX, #word	3	6	AX, CY ← AX – word	×	×	×
CMPW	AX, #word	3	6	AX – word	×	×	×
INC	r	2	4	r ← r + 1	×	×	
	saddr	2	4	(saddr) ← (saddr) + 1	×	×	
DEC	r	2	4	r ← r – 1	×	×	
	saddr	2	4	(saddr) ← (saddr) – 1	×	×	
INCW	rp	1	4	rp ← rp + 1			
DECW	rp	1	4	rp ← rp – 1			
ROR	A, 1	1	2	(CY, A_7 ← A_0, A_{m-1} ← A_m) × 1			×
ROL	A, 1	1	2	(CY, A_0 ← A_7, A_{m+1} ← A_m) × 1			×
RORC	A, 1	1	2	(CY ← A_0, A_7 ← CY, A_{m-1} ← A_m) × 1			×
ROLC	A, 1	1	2	(CY ← A_7, A_0 ← CY, A_{m+1} ← A_m) × 1			×

Remark The instruction clock cycle is based on the CPU clock (f_{CPU}), specified in the processor clock control register (PCC).

Mnemonic	Operand	Byte	Clock	Operation	Flag		
					Z	AC	CY
SET1	saddr. bit	3	6	(saddr. bit) ← 1			
	sfr. bit	3	6	sfr. bit ← 1			
	A. bit	2	4	A. bit ← 1			
	PSW. bit	3	6	PSW. bit ← 1	×	×	×
	[HL]. bit	2	10	(HL). bit ← 1			
CLR1	saddr. bit	3	6	(saddr. bit) ← 0			
	sfr. bit	3	6	sfr. bit ← 0			
	A. bit	2	4	A. bit ← 0			
	PSW. bit	3	6	PSW. bit ← 0	×	×	×
	[HL]. bit	2	10	(HL). bit ← 0			
SET1	CY	1	2	CY ← 1			1
CLR1	CY	1	2	CY ← 0			0
NOT1	CY	1	2	CY ← $\overline{CY}$			×
CALL	!addr16	3	6	$(SP-1) \leftarrow (PC+3)_H$, $(SP-2) \leftarrow (PC+3)_L$, PC ← addr16, SP ← SP – 2			
CALLT	[addr5]	1	8	$(SP-1) \leftarrow (PC+1)_H$, $(SP-2) \leftarrow (PC+1)_L$, $PC_H \leftarrow (00000000, addr5+1)$, $PC_L \leftarrow (00000000, addr5)$, SP ← SP – 2			
RET		1	6	$PC_H \leftarrow (SP+1)$, $PC_L \leftarrow (SP)$, SP ← SP + 2			
RETI		1	8	$PC_H \leftarrow (SP+1)$, $PC_L \leftarrow (SP)$, PSW ← (SP + 2), SP ← SP + 3, NMIS ← 0	R	R	R
PUSH	PSW	1	2	(SP – 1) ← PSW, SP ← SP – 1			
	rp	1	4	$(SP-1) \leftarrow rp_H$, $(SP-2) \leftarrow rp_L$, SP ← SP – 2			
POP	PSW	1	4	PSW ← (SP), SP ← SP + 1	R	R	R
	rp	1	6	$rp_H \leftarrow (SP+1)$, $rp_L \leftarrow (SP)$, SP ← SP + 2			
MOVW	SP, AX	2	8	SP ← AX			
	AX, SP	2	6	AX ← SP			
BR	!addr16	3	6	PC ← addr16			
	$addr16	2	6	PC ← PC + 2 + jdisp8			
	AX	1	6	$PC_H \leftarrow A$, $PC_L \leftarrow X$			

Remark The instruction clock cycle is based on the CPU clock (f_{CPU}), specified in the processor clock control register (PCC).

Mnemonic	Operand	Byte	Clock	Operation	Flag		
					Z	AC	CY
BC	$addr16	2	6	PC ← PC + 2 + jdisp8 if CY = 1			
BNC	$addr16	2	6	PC ← PC + 2 + jdisp8 if CY = 0			
BZ	$addr16	2	6	PC ← PC + 2 + jdisp8 if Z = 1			
BNZ	$addr16	2	6	PC ← PC + 2 + jdisp8 if Z = 0			
BT	saddr. bit, $addr16	4	10	PC ← PC + 4 + jdisp8 if (saddr. bit) = 1			
	sfr. bit, $addr16	4	10	PC ← PC + 4 + jdisp8 if sfr. bit = 1			
	A. bit, $addr16	3	8	PC ← PC + 3 + jdisp8 if A. bit = 1			
	PSW. bit, $addr16	4	10	PC ← PC + 4 + jdisp8 if PSW. bit = 1			
BF	saddr. bit, $addr16	4	10	PC ← PC + 4 + jdisp8 if (saddr. bit) = 0			
	sfr. bit, $addr16	4	10	PC ← PC + 4 + jdisp8 if sfr. bit = 0			
	A. bit, $addr16	3	8	PC ← PC + 3 + jdisp8 if A. bit = 0			
	PSW. bit, $addr16	4	10	PC ← PC + 4 + jdisp8 if PSW. bit = 0			
DBNZ	B, $addr16	2	6	B ← B – 1, then PC ← PC + 2 + jdisp8 if B ≠ 0			
	C, $addr16	2	6	C ← C – 1, then PC ← PC + 2 + jdisp8 if C ≠ 0			
	saddr, $addr16	3	8	(saddr) ← (saddr) – 1, then PC ← PC + 3 + jdisp8 if (saddr) ≠ 0			
NOP		1	2	No Operation			
EI		3	6	IE ← 1 (Enable Interrupt)			
DI		3	6	IE ← 0 (Disable Interrupt)			
HALT		1	2	Set HALT Mode			
STOP		1	2	Set STOP Mode			

Remark The instruction clock cycle is based on the CPU clock (f_{CPU}), specified in the processor clock control register (PCC).

12. ELECTRICAL CHARACTERISTICS

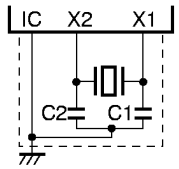
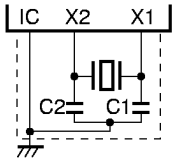
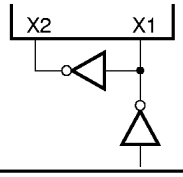
ABSOLUTE MAXIMUM RATINGS (T_A = 25°C)

Parameter	Symbol	Conditions		Rated value	Unit
Supply voltage	V_{DD}, AV_{DD}	V_{DD} = AV_{DD}		-0.3 to +6.5	V
Input voltage	V_{I1}	Pins other than P50 to P53		-0.3 to V_{DD} + 0.3	V
	V_{I2}	P50 to P53	N-ch open drain	-0.3 to +13	V
			With an on-chip pull-up resistor	-0.3 to V_{DD} + 0.3	V
Output voltage	V_O			-0.3 to V_{DD} + 0.3	V
High-level output current	I_{OH}	Each pin		-10	mA
		Total for all pins		-30	mA
Low-level output current	I_{OL}	Each pin		30	mA
		Total for all pins		160	mA
Operating ambient temperature	T_A			-40 to +85	°C
Storage temperature	T_{stg}			-65 to +150	°C

Caution Absolute maximum ratings are rated values beyond which physical damage will be caused to the product; if the rated value of any of the parameters in the above table is exceeded, even momentarily, the quality of the product may deteriorate. Always use the product within its rated values.

Remark The characteristic of a dual-function pin does not differ between the port function and the secondary function, unless otherwise stated.

CHARACTERISTICS OF THE SYSTEM CLOCK OSCILLATION CIRCUIT (T_A = -40°C to +85°C, V_{DD} = 1.8 to 5.5 V)

Resonator	Recommended circuit	Parameter	Conditions	MIN.	TYP.	MAX.	Unit
Ceramic resonator		Oscillator frequency (f_X)[Note 1]	V_{DD} = oscillation voltage range	1.0		5.0	MHz
		Oscillation settling time[Note 2]	After V_{DD} reaches MIN. of the oscillation voltage range			4	ms
Crystal		Oscillator frequency (f_X)[Note 1]		1.0		5.0	MHz
		Oscillation settling time[Note 2]	4.5 V ≤ V_{DD} ≤ 5.5 V			10	ms
						30	
External clock		X1 input frequency (f_X)[Note 1]		1.0		5.0	MHz
		X1 input high/low level width (t_{XH}, t_{XL})		85		500	ns

Notes 1. Only the characteristic of the oscillation circuit is indicated. See the description of the AC characteristic for the instruction execution time.

2. Time required for oscillation to settle once a reset sequence ends or STOP mode is deselected.

Caution When using the system clock oscillation circuit, observe the following conditions for the wiring of that section enclosed in dotted lines in the above diagrams, so as to avoid the influence of the wiring capacitance.

- **Keep the wiring as short as possible.**
- **Do not allow signal wires to cross one another.**
- **Keep the wiring away from wires that carry a high, non-stable current.**
- **Keep the grounding point of the capacitors at the same level as V_{SS}.**
- **Do not connect the grounding point to a grounding wire that carries a high current.**
- **Do not extract a signal from the oscillation circuit.**

DC CHARACTERISTICS (T_A = -40°C to +85°C, V_{DD} = 1.8 to 5.5 V)

Parameter	Symbol	Conditions			MIN.	TYP.	MAX.	Unit
Low-level output current	I_{OL}	Each pin					Undefined	mA
		Total for all pins					80	mA
High-level output current	I_{OH}	Each pin					Undefined	mA
		Total for all pins					-15	mA
High-level input voltage	V_{IH1}	P00-P03, P10, P11, P60-P63		V_{DD} = 2.7 to 5.5 V	0.7V_{DD}		V_{DD}	V
					0.9V_{DD}		V_{DD}	V
	V_{IH2}	P50-P53	With N-ch open drain	V_{DD} = 2.7 to 5.5 V	0.7V_{DD}		12	V
					0.9V_{DD}		12	V
			With an on-chip pull-up resistor	V_{DD} = 2.7 to 5.5 V	0.7V_{DD}		V_{DD}	V
					0.9V_{DD}		V_{DD}	V
	V_{IH3}	$\overline{\text{RESET}}$, P20-P25, P40-P45		V_{DD} = 2.7 to 5.5 V	0.8V_{DD}		V_{DD}	V
					0.9V_{DD}		V_{DD}	V
	V_{IH4}	X1, X2			V_{DD} - 0.1		V_{DD}	V
Low-level input voltage	V_{IL1}	P00-P03, P10, P11, P60-P63		V_{DD} = 2.7 to 5.5 V	0		0.3V_{DD}	V
					0		0.1V_{DD}	V
	V_{IL2}	P50-P53		V_{DD} = 2.7 to 5.5 V	0		0.3V_{DD}	V
					0		0.1V_{DD}	V
	V_{IL3}	$\overline{\text{RESET}}$, P20-P25, P40-P45		V_{DD} = 2.7 to 5.5 V	0		0.2V_{DD}	V
					0		0.1V_{DD}	V
	V_{IL4}	X1, X2			0		0.1	V
High-level output voltage	V_{OH}	V_{DD} = 4.5 to 5.5 V, I_{OH} = -1 mA			V_{DD} - 1.0			V
		V_{DD} = 1.8 to 5.5 V, I_{OH} = -100 μA			V_{DD} - 0.5			V
Low-level output voltage	V_{OL1}	Pins other than P50-P53		V_{DD} = 4.5 to 5.5 V, I_{OL} = 10 mA			1.0	V
				V_{DD} = 1.8 to 5.5 V, I_{OL} = 400 μA			0.5	V
	V_{OL2}	P50-P53		V_{DD} = 4.5 to 5.5 V, I_{OL} = 10 mA			1.0	V
				V_{DD} = 1.8 to 5.5 V, I_{OL} = 1.6 mA			0.4	V
High-level input leakage current	I_{LIH1}	V_{IN} = V_{DD}		Pins other than X1, X2, or P50-P53			3	μA
	I_{LIH2}			X1, X2			20	μA
	I_{LIH3}	V_{IN} = 12 V		P50-P53 (N-ch open drain)			20	μA
Low-level input leakage current	I_{LIL1}	V_{IN} = 0 V		Pins other than X1, X2, or P50-P53			-3	μA
	I_{LIL2}			X1, X2			-20	μA

Remark The characteristic of a dual-function pin does not differ between the port function and the secondary function, unless otherwise stated.

DC CHARACTERISTICS (T_A = -40°C to +85°C, V_{DD} = 1.8 to 5.5 V)

Parameter	Symbol	Conditions		MIN.	TYP.	MAX.	Unit
Low-level input leakage current	I_{LIL3}	V_{IN} = 0 V	P50-P53 (N-ch open drain) Except during input instruction execution			-3	μA
			P50-P53 (N-ch open drain) During input instruction execution			-30	μA
High-level output leakage current	I_{LOH}	V_{OUT} = V_{DD}				3	μA
Low-level output leakage current	I_{LOL}	V_{OUT} = 0 V				-3	μA
Software-specified pull-up resistor	R_1	V_{IN} = 0 V, for pins other than P50-P53		50	100	200	kΩ
Mask option-specified pull-up resistor	R_2	V_{IN} = 0 V, P50-P53		15	30	60	kΩ
Power supply current[Note 1]	I_{DD1}	5.0-MHz crystal oscillation operating mode	V_{DD} = 5.0 V ± 10 %[Note 2]		1.1	2.0	mA
			V_{DD} = 3.0 V ± 10 %[Note 3]		0.3	0.45	mA
			V_{DD} = 2.0 V ± 10 %[Note 3]		0.2	0.35	mA
	I_{DD2}	5.0-MHz crystal oscillation HALT mode	V_{DD} = 5.0 V ± 10 %[Note 2]		0.6	0.85	mA
			V_{DD} = 3.0 V ± 10 %[Note 3]		0.2	0.35	mA
			V_{DD} = 2.0 V ± 10 %[Note 3]		0.1	0.18	mA
	I_{DD3}	STOP mode	V_{DD} = 5.0 V ± 10 %		0.1	10.0	μA
			V_{DD} = 3.0 V ± 10 %		0.05	5.0	μA
			V_{DD} = 2.0 V ± 10 %		0.05	3.0	μA
	I_{DD4}	5.0-MHz crystal oscillation A/D operating mode	V_{DD} = 5.0 V ± 10 %		1.7	3.1	mA
			V_{DD} = 3.0 V ± 10 %		0.9	1.5	mA
			V_{DD} = 2.0 V ± 10 %		0.8	1.4	mA
	I_{DD5}	During operation of 5.0-MHz crystal oscillation EEPROM (ERE1 = 1, EWE1 = 1)	V_{DD} = 5.0 V ± 10 %		1.7[Notes 4, 7]	Undefined	mA
			V_{DD} = 3.0 V ± 10 %		0.6[Notes 5, 8]	Undefined	mA
			V_{DD} = 2.0 V ± 10 %		0.3[Notes 6, 9]	Undefined	mA
	I_{DD6}	During operation of 5.0-MHz crystal oscillation EEPROM (ERE1 = 1, EWE1 = 0)	V_{DD} = 5.0 V ± 10 %		1.6[Note 7]	Undefined	mA
			V_{DD} = 3.0 V ± 10 %		0.6[Note 8]	Undefined	mA
			V_{DD} = 2.0 V ± 10 %		0.3[Note 9]	Undefined	mA

Notes 1. The power supply current does not include AV_{DD} or the port current (including the current flowing through the on-chip pull-up resistor).

2. During high-speed mode operation (when the processor clock control register (PCC) is cleared to 00H.)

3. During low-speed mode operation (when the PCC is set to 02H.)

4. An additional current of 0.5 mA (TYP.) is also applied during writing.

5. An additional current of 0.7 mA (TYP.) is also applied during writing.

6. An additional current of 0.9 mA (TYP.) is also applied during writing.

7. An additional current of 1.5 mA (TYP.) is also applied during reading.

8. An additional current of 0.9 mA (TYP.) is also applied during reading.

9. An additional current of 0.3 mA (TYP.) is also applied during reading.

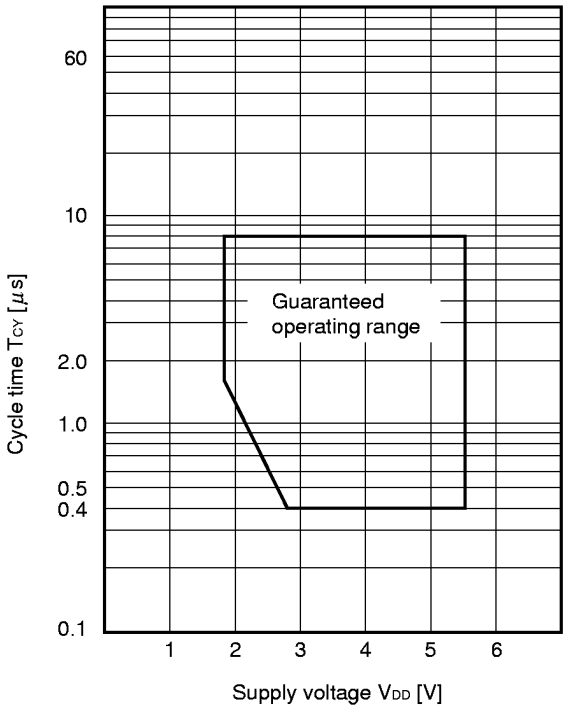
Remark The characteristic of a dual-function pin does not differ between the port function and the secondary function, unless otherwise stated.

AC CHARACTERISTICS

(1) Basic operations (T_A = -40°C to +85°C, V_{DD} = 1.8 to 5.5 V)

Parameter	Symbol	Conditions	MIN.	TYP.	MAX.	Unit
Cycle time (minimum instruction execution time)	T_{CY}	V_{DD} = 2.7 to 5.5 V	0.4		8	μs
			1.6		8	μs
TI80 input high/low level width	t_{TIH}, t_{TIL}	V_{DD} = 2.7 to 5.5 V	0.1			μs
			1.8			μs
TI80 input frequency	f_{TI}	V_{DD} = 2.7 to 5.5 V	0		4	MHz
			0		275	kHz
Interrupt input high/low level width	t_{INTH}, t_{INTL}	INTP0-INTP2	10			μs
$\overline{\text{RESET}}$ low level width	t_{RSL}		10			μs

T_{CY} vs V_{DD} (system clock)



(2) Serial interface (T_A = -40°C to +85°C, V_{DD} = 1.8 to 5.5 V)

(i) Three-wire serial I/O mode ($\overline{SCK20}$...Internal clock output)

Parameter	Symbol	Conditions		MIN.	TYP.	MAX.	Unit
$\overline{SCK20}$ cycle time	t_{KCY1}	V_{DD} = 2.7 to 5.5 V		800			ns
				3,200			ns
$\overline{SCK20}$ high/low level width	t_{KH1}, t_{KL1}	V_{DD} = 2.7 to 5.5 V		t_{KCY1}/2-50			ns
				t_{KCY1}/2-150			ns
SI20 setup time (for $\overline{SCK20}$ latch edge)	t_{SIK1}	V_{DD} = 2.7 to 5.5 V		150			ns
				500			ns
SI20 hold time (for $\overline{SCK20}$ latch edge)	t_{KSI1}	V_{DD} = 2.7 to 5.5 V		400			ns
				600			ns
Delay from $\overline{SCK20}$ shift edge to SO20 output	t_{KSO1}	R = 1 kΩ, C = 100 pF[Note]	V_{DD} = 2.7 to 5.5 V	0		250	ns
				0		1,000	ns

Note R and C are the resistance and capacitance of the SO20 output line, respectively.

(ii) Three-wire serial I/O mode ($\overline{SCK20}$...External clock input)

Parameter	Symbol	Conditions		MIN.	TYP.	MAX.	Unit
$\overline{SCK20}$ cycle time	t_{KCY2}	V_{DD} = 2.7 to 5.5 V		800			ns
				3,200			ns
$\overline{SCK20}$ high/low level width	t_{KH2}, t_{KL2}	V_{DD} = 2.7 to 5.5 V		400			ns
				1,600			ns
SI20 setup time (for $\overline{SCK20}$ latch edge)	t_{SIK2}	V_{DD} = 2.7 to 5.5 V		100			ns
				150			ns
SI20 hold time (for $\overline{SCK20}$ latch edge)	t_{KSI2}	V_{DD} = 2.7 to 5.5 V		400			ns
				600			ns
Delay from $\overline{SCK20}$ shift edge to SO20 output	t_{KSO2}	R = 1 kΩ, C = 100 pF[Note]	V_{DD} = 2.7 to 5.5 V	0		300	ns
				0		1,000	ns
SO20 setup time (for $\overline{SS20}$↓ when $\overline{SS20}$ is used)	t_{KAS2}	V_{DD} = 2.7 to 5.5 V				120	ns
						400	ns
SO20 disable time (for $\overline{SS20}$↑ when $\overline{SS20}$ is used)	t_{KDS2}	V_{DD} = 2.7 to 5.5 V				240	ns
						800	ns

Note R and C are the resistance and capacitance of the SO20 output line, respectively.

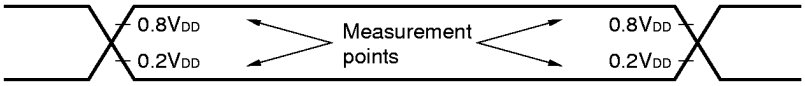
(iii) UART mode (internal clock output)

Parameter	Symbol	Conditions	MIN.	TYP.	MAX.	Unit
Transfer rate		V_{DD} = 2.7 to 5.5 V			78,125	bps
					19,531	bps

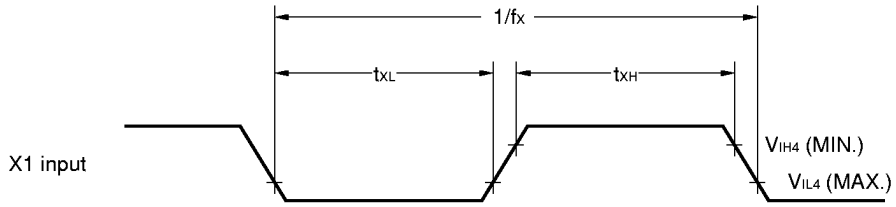
(iv) UART mode (external clock input)

Parameter	Symbol	Conditions	MIN.	TYP.	MAX.	Unit
ASCK20 cycle time	t_{KCY3}	V_{DD} = 2.7 to 5.5 V	800			ns
			3,200			ns
ASCK20 high/low level width	t_{KH3}, t_{KL3}	V_{DD} = 2.7 to 5.5 V	400			ns
			1,600			ns
Transfer rate		V_{DD} = 2.7 to 5.5 V			39,063	bps
					9,766	bps
ASCK20 rising time, falling time	t_R, t_F				1	μs

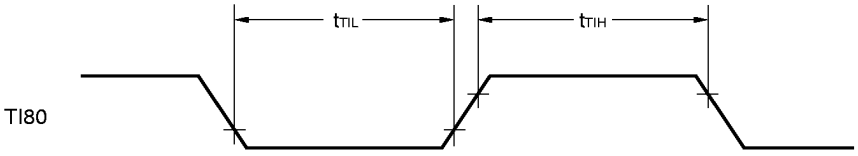
AC TIMING MEASUREMENT POINTS (except the X1 inputs)



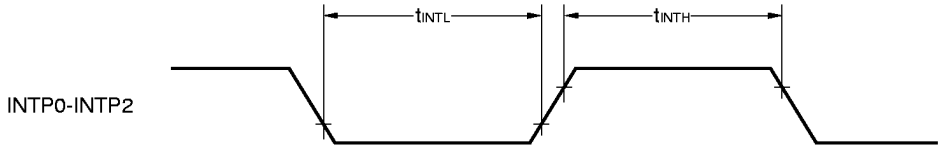
CLOCK TIMING



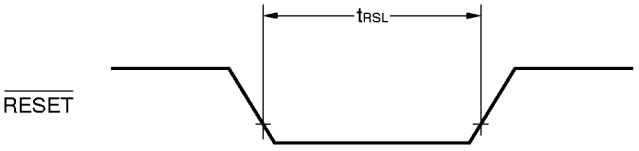
TI TIMING



INTERRUPT INPUT TIMING

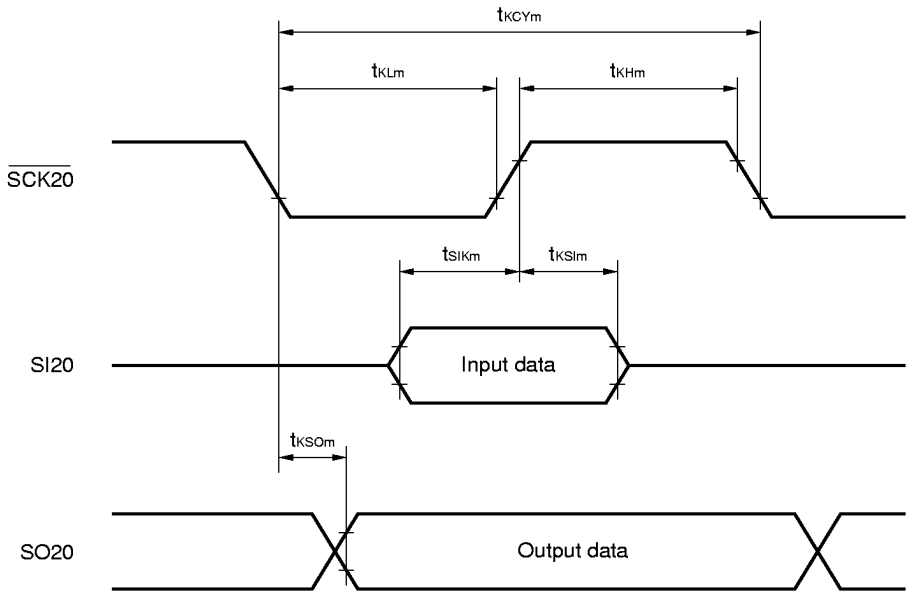


$\overline{\text{RESET}}$ INPUT TIMING



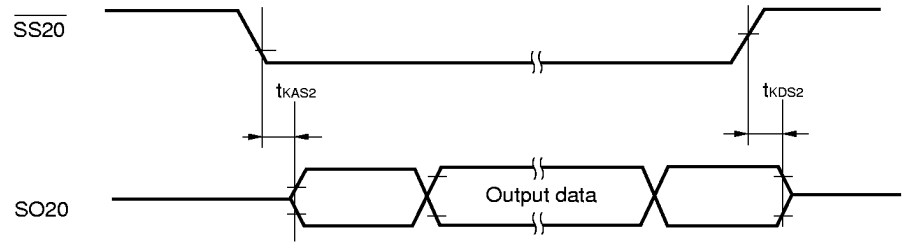
SERIAL TRANSFER TIMING

Three-Wire Serial I/O Mode:

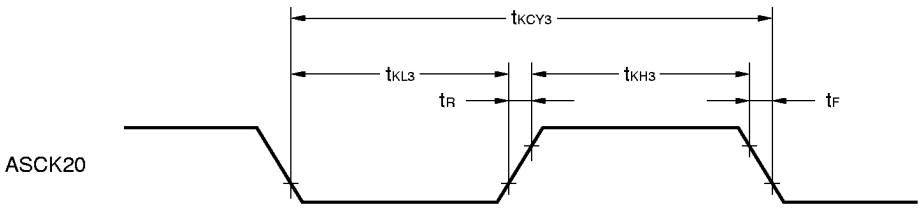


m = 1, 2

Three-Wire Serial I/O Mode (When $\overline{SS20}$ Is Used):



UART Mode (External Clock Input):



8-BIT A/D CONVERTER CHARACTERISTICS (T_A = -40°C to +85°C, V_{DD} = 1.8 to 5.5 V)

Parameter	Symbol	Conditions	MIN.	TYP.	MAX.	Unit
Resolution			8	8	8	bit
Total error[Note]					0.6	%
Conversion time	t_{CONV}		Undefined		Undefined	μs
Analog input voltage	V_{IAN}		0		AV_{DD}	V

Note No quantization error (±1/2 LSB) is included.

10-BIT A/D CONVERTER CHARACTERISTICS (T_A = -40°C to +85°C, V_{DD} = 1.8 to 5.5 V)

Parameter	Symbol	Conditions	MIN.	TYP.	MAX.	Unit
Resolution			10	10	10	bit
Total error[Note]		4.5 V ≤ V_{DD} ≤ 5.5 V		0.2	0.4	%
		2.7 V ≤ V_{DD} < 4.5 V		0.4	0.7	
		1.8 V ≤ V_{DD} < 2.7 V		Undefined	Undefined	
Conversion time	t_{CONV}	4.5 V ≤ V_{DD} ≤ 5.5 V	Undefined		Undefined	μs
		2.7 V ≤ V_{DD} < 4.5 V	Undefined		Undefined	
		1.8 V ≤ V_{DD} < 2.7 V	Undefined		Undefined	
Analog input voltage	V_{IAN}		AV_{SS}		AV_{DD}	V

Note No quantization error (±1/2 LSB) is included.

LOW-VOLTAGE INDICATION CHARACTERISTICS (T_A = -40°C to +85°C, V_{DD} = 1.8 to 5.5 V)

(1) DC characteristics

Parameter	Symbol	Conditions	MIN.	TYP.	MAX.	Unit
Detected voltage	V_{LVI}		3.55	4.00	4.45	V
Hysteresis	V_{HYS}			0.1		V
Operating current	I_{LVI0}	The low-voltage indication is operating.		30	100	μA
	I_{LVI1}	The low-voltage indication is stopped (STOP mode).		0		μA

(2) AC characteristics

Parameter	Symbol	Conditions	MIN.	TYP.	MAX.	Unit
Response delay time	t_D	| Detected voltage – V_{DD} | ≥ 0.45 V			200	μs
Minimum pulse width	tw		500			μs
Operation start time	t_{STA}	After operation is enabled			500	μs

EEPROM CHARACTERISTICS (T_A = -40°C to +85°C, V_{DD} = 1.8 to 5.5 V)

Parameter	Symbol	Conditions	MIN.	TYP.	MAX.	Unit
Write time[Note]	Undefined		3.3		6.6	ms
Rewrite count	Undefined				10	10,000 times

Note Write time = T × 145 (T = one cycle time of the clock selected in EWCS10 to EWCS12)

OSCILLATION SETTLING TIME (T_A = -40°C to +85°C, V_{DD} = 1.8 to 5.5 V)

Parameter	Symbol	Conditions	MIN.	TYP.	MAX.	Unit
Oscillation settling time		Reset by $\overline{\text{RESET}}$		$2^{15}/f_X$		s
		Reset by an interrupt		**Note**		s

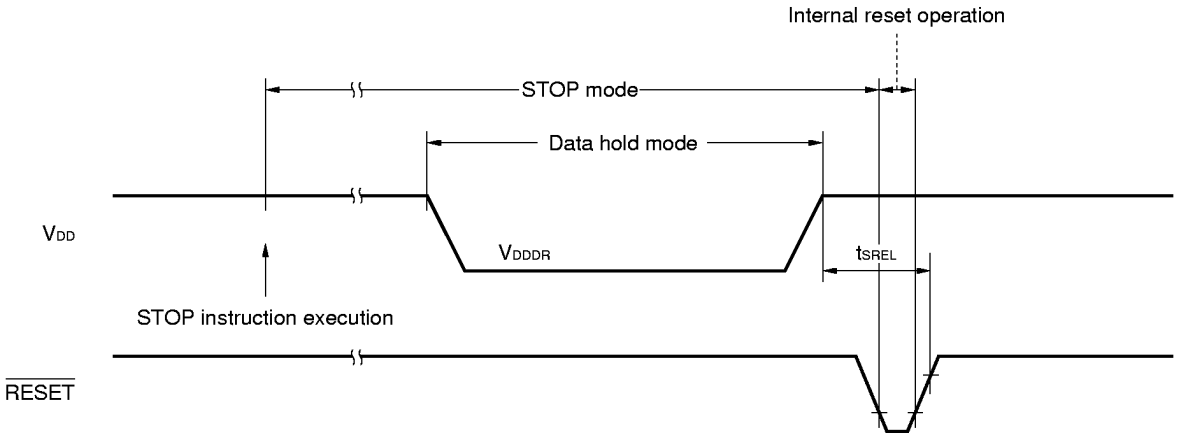
Note Bits 0 to 2 (OSTS0 to OSTS2) of the oscillation settling time selection register (OSTS) can be used to select $2^{12}/f_X$, $2^{15}/f_X$, or $2^{17}/f_X$.

DATA MEMORY STOP MODE LOW POWER SUPPLY VOLTAGE DATA HOLD CHARACTERISTICS (T_A = -40°C to +85°C)

Parameter	Symbol	Conditions	MIN.	TYP.	MAX.	Unit
Data hold supply voltage	V_{DDDR}		1.8		5.5	V
Release signal set time	t_{SREL}		0			μs

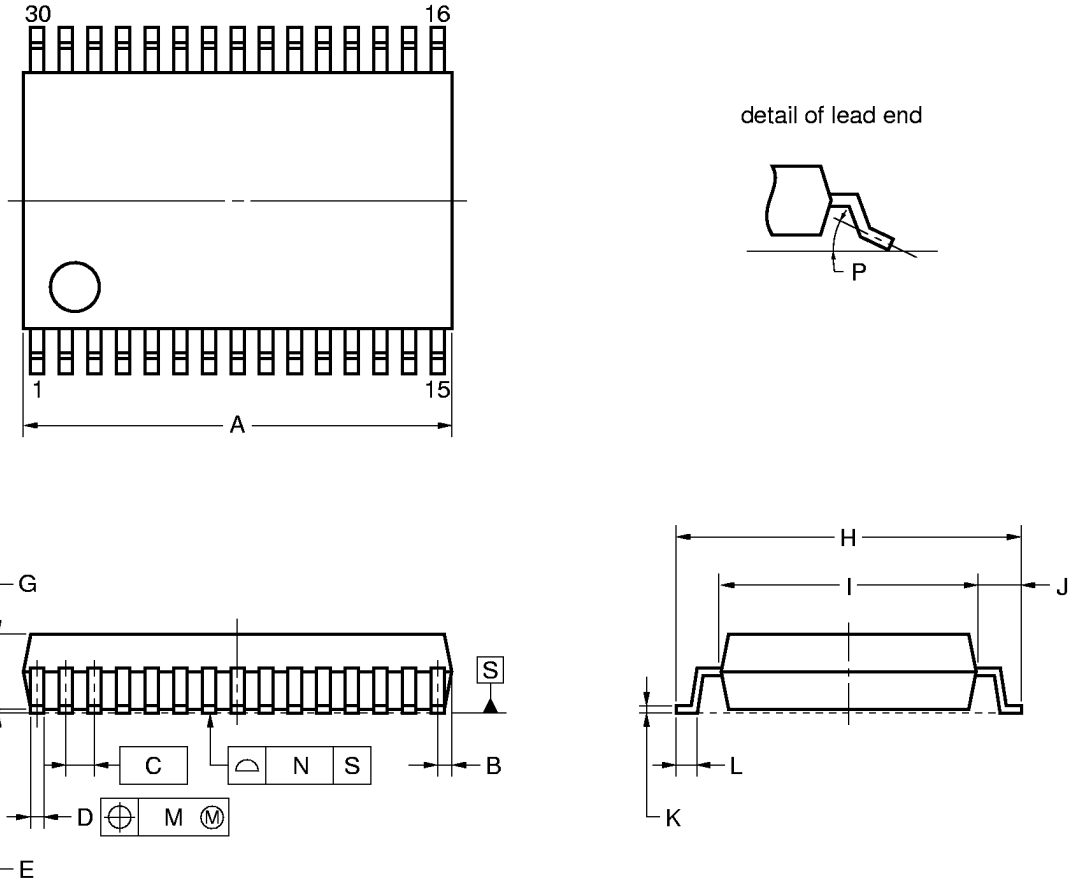
Remark f_X: System clock oscillation frequency

Data Hold Timing (STOP mode release by $\overline{\text{RESET}}$)



13. PACKAGE DRAWINGS

30 PIN PLASTIC SHRINK SOP (300 mil)



NOTES

1. Controlling dimension — millimeter.
2. Each lead centerline is located within 0.10 mm (0.004 inch) of its true position (T.P.) at maximum material condition.

ITEM	MILLIMETERS	INCHES
A	9.85±0.26	0.388±0.011
B	0.51 MAX.	0.020 MAX.
C	0.65 (T.P.)	0.026 (T.P.)
D	$0.32^{+0.08}_{-0.07}$	$0.013^{+0.003}_{-0.004}$
E	0.125±0.075	0.005±0.003
F	2.0 MAX.	0.079 MAX.
G	1.7±0.1	0.067±0.004
H	8.1±0.2	0.319±0.008
I	6.1±0.2	0.240±0.008
J	1.0±0.2	$0.039^{+0.009}_{-0.008}$
K	$0.17^{+0.08}_{-0.07}$	$0.007^{+0.003}_{-0.004}$
L	0.5±0.2	$0.020^{+0.008}_{-0.009}$
M	0.10	0.004
N	0.10	0.004
P	$3°^{+7°}_{-3°}$	$3°^{+7°}_{-3°}$

P30GS-65-300B-2

APPENDIX A DEVELOPMENT TOOLS

The following development tools are available for developing systems using the μPD789144, μPD789146, μPD789154, and μPD789156.

LANGUAGE PROCESSING SOFTWARE

RA78K0S[Notes 1, 2, 3]	Assembler package common to the 78K/0S series
CC78K0S[Notes 1, 2, 3]	C compiler package common to the 78K/0S series
DF789156[Notes 1, 2, 3, 5]	Device file for the μPD789146 and μPD789156 sub-series

FLASH MEMORY WRITE TOOLS

Flashpro II[Note 4]	Dedicated flash writer (formerly, Flashpro)
FA-30GS[Note 4]	Flash memory write adapter

DEBUGGING TOOLS

IE-78K0S-NS In-circuit emulator	This in-circuit emulator is used to debug hardware or software when application systems which use the 78K/0S series are developed. The IE-78K0S-NS supports the integrated debugger (ID78K0S-NS). The IE-78K0S-NS is used in combination with an interface adapter for connection to an AC adapter, emulation probe, or host machine.
IE-70000-MC-PS-B AC adapter	This adapter is used to supply power from a 100-VAC outlet.
IE-70000-98-IF-C Interface adapter	This adapter is required when a PC-9800 series computer (other than a notebook type) is used as the host machine for the IE-78K0S-NS.
IE-70000-CD-IF PC card/interface	These PC card and interface cable are required when a PC-9800 series computer is used as the host machine for the IE-78K0S-NS.
IE-70000-PC-IF-C Interface adapter	This adapter is required when an IBM PC/AT™ compatible is used as the host machine for the IE-78K0S-NS.
IE-789156-NS-EM1[Note 5] Emulation board	This board is used to emulate the peripheral hardware specific to the device. The IE-789156-NS-EM1 is used in combination with the in-circuit emulator.
NP-30GS[Note 4] Emulation probe	This probe is used to connect an in-circuit emulator to the target system. The probe is dedicated to the 30-pin plastic shrink SOP.
SM78K0S[Notes 1, 2]	System simulator common to all 78K/0S series units
DF789156[Notes 1, 2, 5]	Device file for the μPD789146 and μPD789156 sub-series

REAL-TIME OS

MX78K0S[Notes 1, 2, 5]	OS for the 78K/0S series

Notes 1. Based on the PC-9800 series (MS-DOS™ + Windows™)

2. Based on the IBM PC/AT™ compatibles (Japanese/English Windows)

3. Based on the HP9000 series 700™ (HP-UX™), SPARCstation™ (SunOS™), and NEWS™ (NEWS-OS™)

4. Product manufactured by Naito Densei Machida Mfg. Co., Ltd. (044-822-3813). Consult an NEC sales representative for purchase.

5. Under development

Remark The RA78K0S, CC78K0S, and SM78K0S can be used in combination with the DF789156.

APPENDIX B RELATED DOCUMENTS

DOCUMENTS RELATED TO DEVICES

Document name	Document No.	
	Japanese	English
μPD789144, 789146, 789154, 789156 Preliminary Product Information	U13478J	This manual
μPD78F9156 Preliminary Product Information	To be created	To be created
μPD789146, 789156 Sub-Series User's Manual	To be created	To be created
78K/0S Series User's Manual, Instruction	U11047J	U11047E

DOCUMENTS RELATED TO DEVELOPMENT TOOLS (USER'S MANUAL)

Document name		Document No.	
		Japanese	English
RA78K0S Assembler Package	Operation	U11622J	U11622E
	Language	U11599J	U11599E
	Structured Assembly Language	U11623J	U11623E
CC78K0S C Compiler	Operation	U11816J	U11816E
	Language	U11817J	U11817E
SM78K0S System Simulator Windows Base	Reference	U11489J	U11489E
SM78K Series System Simulator	External Parts User Open Interface Specifications	U10092J	U10092E
ID78K0S-NS Integrated Debugger Windows Base	Reference	U12901J	U12901E

DOCUMENTS RELATED TO SOFTWARE TO BE INCORPORATED INTO THE PRODUCT (USER'S MANUAL)

Document name		Document No.	
		Japanese	English
OS for 78K/0S Series MX78K0S	Fundamental	U12938J	U12938E

Caution The above documents may be revised without notice. Use the latest versions when you design application systems.

OTHER DOCUMENTS

Document name	Document No.	
	Japanese	English
IC PACKAGE MANUAL	C10943X	
Semiconductor Device Mounting Technology Manual	C10535J	C10535E
Quality Grades on NEC Semiconductor Device	C11531J	C11531E
NEC Semiconductor Device Reliability/Quality Control System	C10983J	C10983E
Guide to Prevent Damage for Semiconductor Devices by Electrostatic Discharge (ESD)	C11892J	C11892E
Semiconductor Device Quality Control/Reliability Handbook	C12769J	-
Guide for Products Related to Microcomputer: Other Companies	U11416J	-

Caution The above documents may be revised without notice. Use the latest versions when you design application systems.

Note that "Preliminary" is not indicated in this document, even though the related documents may be preliminary versions.

NOTES FOR CMOS DEVICES

① PRECAUTION AGAINST ESD FOR SEMICONDUCTORS

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

② HANDLING OF UNUSED INPUT PINS FOR CMOS

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

③ STATUS BEFORE INITIALIZATION OF MOS DEVICES

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

EEPROM is a trademark of NEC Corporation.

MS-DOS and Windows are either registered trademarks or trademarks of Microsoft Corporation in the United States and/or other countries.

PC/AT is a trademark of IBM Corporation.

HP9000 series 700 and HP-UX are trademarks of Hewlett-Packard Company.

SPARCstation is a trademark of SPARC International, Inc.

SunOS is a trademark of Sun Microsystems, Inc.

NEWS and NEWS-OS are trademarks of SONY Corporation.

Regional Information

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

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

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

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

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

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

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

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

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

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

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

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

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

NEC Electronics Singapore Pte. Ltd.
United Square, Singapore 1130
Tel: 65-253-8311
Fax: 65-250-3583

NEC Electronics Taiwan Ltd.
Taipei, Taiwan
Tel: 02-2719-2377
Fax: 02-2719-5951

NEC do Brasil S.A.
Electron Devices Division
Rodovia Presidente Dutra, Km 214
07210-902-Guarulhos-SP Brasil
Tel: 55-11-6465-6810
Fax: 55-11-6465-6829

J98. 8

The export of this product from Japan is regulated by the Japanese government. To export this product may be prohibited without governmental license, the need for which must be judged by the customer. The export or re-export of this product from a country other than Japan may also be prohibited without a license from that country. Please call an NEC sales representative.

These products are manufactured and sold based on a license contract with CP8 Transac regarding the EEPROM microcontroller patent.

These products cannot be used for an IC card (SMART CARD).