PRELIMINARY PRODUCT INFORMATION

NEC

MOS INTEGRATED CIRCUITS

μPD78F9076

8-BIT SINGLE-CHIP MICROCONTROLLERS

The μPD78F9076 is a μPD789074 Subseries product (Small, general-purpose) of the 78K/0S Series.
The μPD78F9076 replaces the internal masked ROM of the μPD789071, 789072 and 789074 with flash memory, which enables the writing/erasing of a program while the device is mounted on the board.

Because the device can be programmed by the user, it is ideally suited to the evaluation stages of system development, the manufacture of small batches of multiple products, and the rapid development of new products.

Detailed function descriptions are provided in the following user's manuals. Be sure to read them before designing.

μPD789074 Subseries User's Manual: To be prepared
78K/0S Series User's Manual Instruction: U11047E

FEATURES

- Pin-compatible with masked ROM version (excluding V_{PP} pin)
- Flash memory: 16K bytes
- Minimum instruction execution time can be changed from high-speed (0.4 μs) to low-speed (1.6 μs) (@ 5.0-MHz operation with system clock)
- I/O ports: 24
- Serial interface: 1 channel: Switchable between 3-wire serial I/O and UART modes
- Timers: 3 channels
  - 16-bit timer: 1 channel
  - 8-bit timer/event counter: 1 channel
  - Watchdog timer: 1 channel
- Power supply voltage: V_{DD} = 1.8 to 5.5 V

APPLICATIONS

General small home appliances and telephone

ORDERING INFORMATION

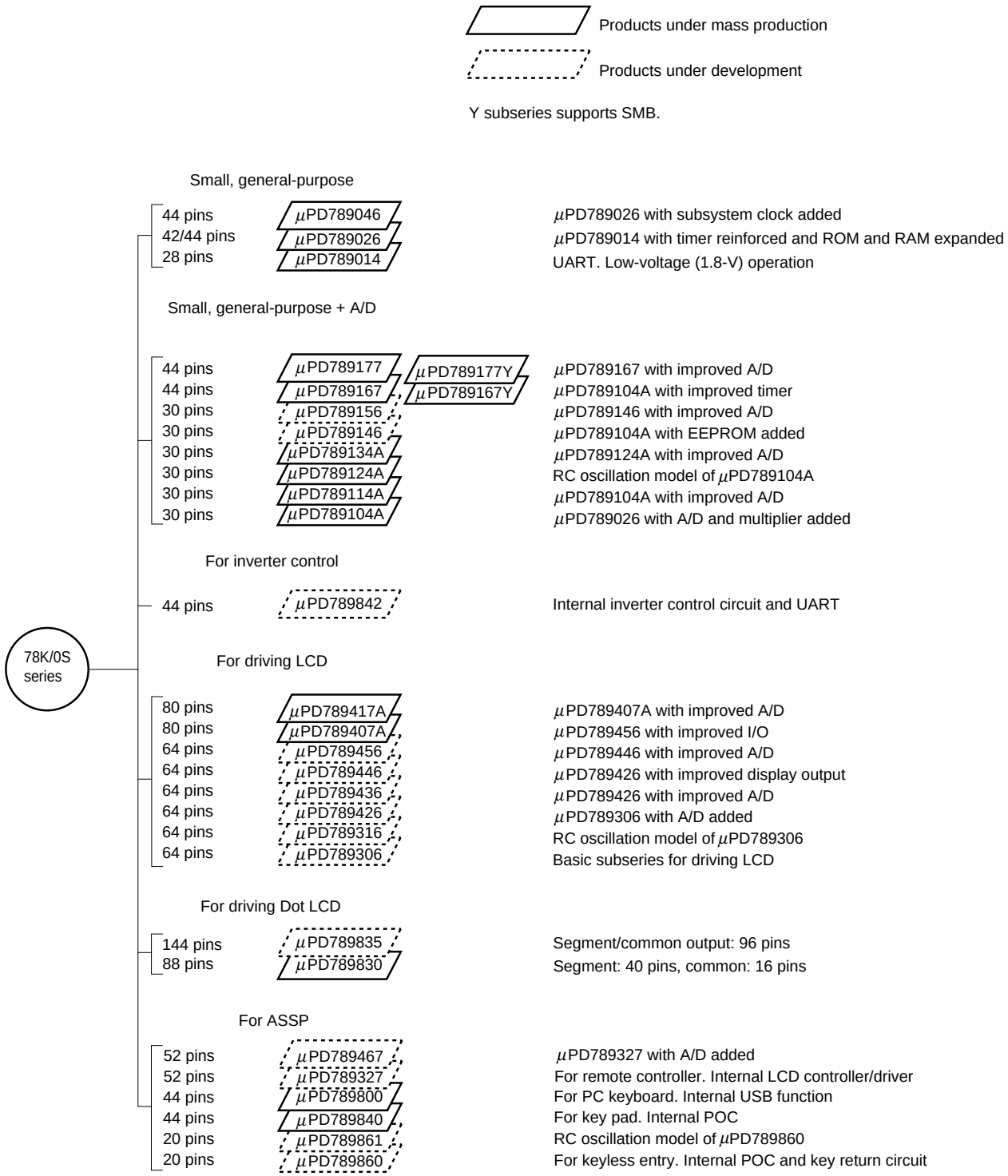
Part number	Package
μPD78F9076MC-5A4	30-pin plastic SSOP (7.62mm (300))

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.
Not all devices/types available in every country. Please check with local NEC representative for availability and additional information.

Document No. U14708EJ1V0PM00 (1st edition)
Date Published June 2000 NS CP(K)
Printed in Japan

78K/0S SERIES LINEUP

The products in the 78K/0S Series are listed below. The names enclosed in boxes are subseries names.



The major differences between subseries are shown below.

Subseries Name \ Function		ROM Capacity	Timer				8-bit A/D	10-bit A/D	Serial Interface	I/O	V_{DD} MIN Value	Remark
			8-bit	16-bit	Watch	WDT						
Small, general-purpose	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, general-purpose + A/D	μPD789177	16 K-24 K	3 ch	1 ch	1 ch	1 ch	–	8 ch	1 ch (UART: 1 ch)	31 pins	1.8 V	–
	μPD789167						8 ch	–				
	μPD789156	8 K-16 K	1 ch		–		–	4 ch		20 pins		Internal EEPROM
	μPD789146						4 ch	–				
	μPD789134A	2 K-8 K						4 ch				RC oscillation version
	μPD789124A						4 ch	–				
	μPD789114A						–	4 ch				–
	μPD789104A						4 ch	–				
For inverter control	μPD789842	8 K-16 K	3 ch	**Note**	1 ch	1 ch	8 ch	–	1 ch (UART: 1 ch)	30 pins	4.0 V	–
For LCD driving	μPD789417A	12 K-24 K	3 ch	1 ch	1 ch	1 ch		7 ch	1 ch (UART: 1 ch)	43 pins	1.8 V	–
	μPD789407A						7 ch	–				
	μPD789456	12 K-16 K	2 ch				–	6 ch		30 pins		
	μPD789446						6 ch	–				
	μPD789436						–	6 ch		40 pins		
	μPD789426						6 ch	–				
	μPD789316	8 K to 16K					–		2 ch (UART: 1 ch)	23 pins		RC oscillation version
	μPD789306											–
For Dot LCD driving	μPD789835	24 K-60 K	6 ch	–	1 ch	1 ch	3 ch	–	1 ch	28 pins	1.8 V	–
	μPD789830	24 K	1 ch	1 ch			–			30 pins	2.7 V	
ASSP	μPD789467	4 K-24 K	2 ch	–	1 ch	1 ch	1 ch	–	–	18 pins	1.8 V	Internal LCD
	μPD789327						–		1 ch	21 pins		
	μPD789800	8 K	2 ch	1 ch	–	1 ch	–		2 ch (USB: 1 ch)	31 pins	4.0 V	–
	μPD789840						4 ch		1 ch	29 pins	2.8 V	
	μPD789861	4 K		–			–		–	14 pins	1.8 V	RC oscillation version, Internal EEPROM
	μPD789860											Internal EEPROM

Note 10-bit timer: 1 channel

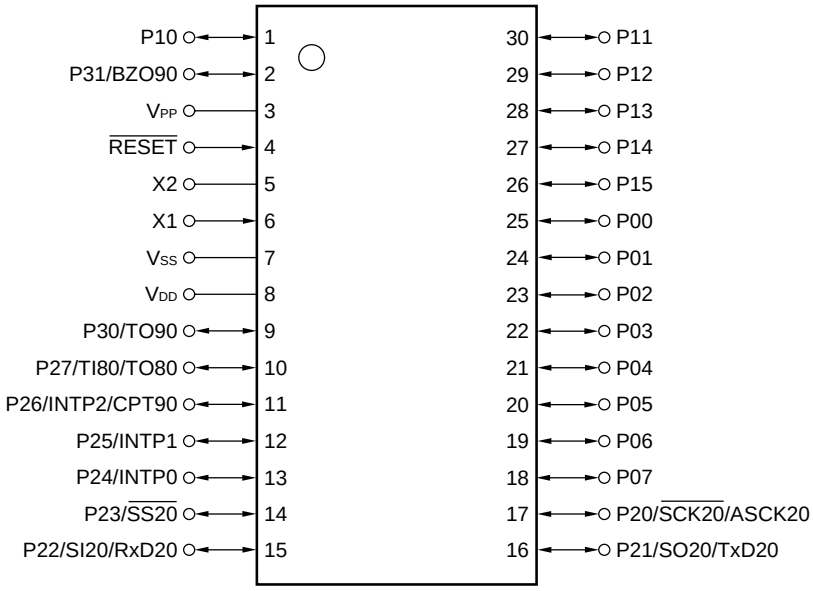
OVERVIEW OF FUNCTIONS

Item		function
Internal memory	Flash memory	16 Kbytes
	High-speed RAM	256 bytes
Minimum instruction execution time		0.4/1.6 μs (@ 5.0-MHz operation with system clock)
General-purpose registers		8 bits × 8 registers
Instruction set		• 16-bit operations • Bit manipulations (set, reset, and test)
Multiplier		8 bits × 8 bits = 16 bits
I/O ports		CMOS I/O:24
Timer		• 16-bit timer: 1 channel • 8-bit timer/event counter: 1 channel • Watchdog timer: 1 channel
Serial interface		Switchable between 3-wire serial I/O and UART modes
Vectored interrupt sources	Maskable	Internal: 4, External: 3
	Non-maskable	Internal: 1
Power supply voltage		V_{DD} = 1.8 to 5.5 V
Operating ambient temperature		T_A = –40 to +85°C
Package		30-pin plastic SSOP (7.62 mm (300))

CONTENTS

1. PIN CONFIGURATION (TOP VIEW)

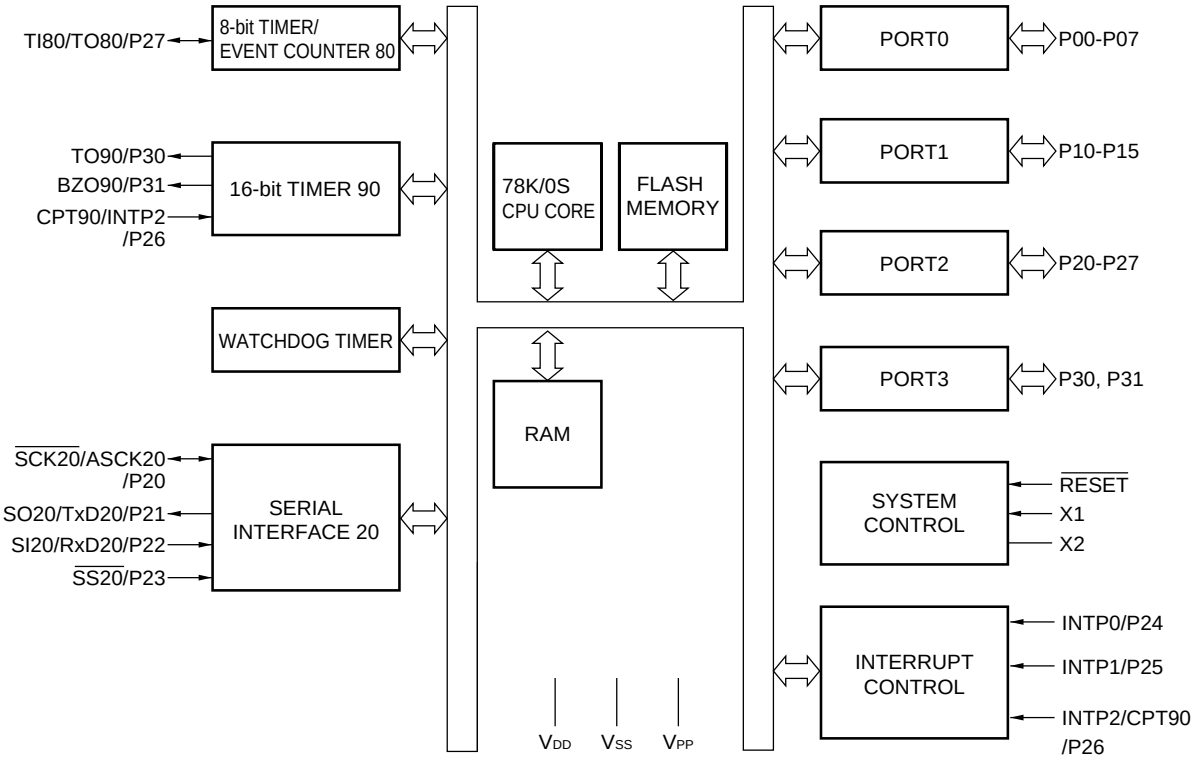
- 30-pin plastic SSOP (7.62 mm (300))
  μPD78F9076MC-5A4



Cautions Connect the V_{PP} pin directly to V_{SS} in normal operation mode.

ASCK20:	Asynchronous Serial Input	SCK20:	Serial Clock Input/Output
BZO90:	Buzzer Output	SI20:	Serial Data Input
CPT90:	Capture Trigger Input	SO20:	Serial Data Output
INTP0 to INTP2:	Interrupt from Peripherals	SS20:	Chip Select Input
P00 to P07:	Port0	TI80:	Timer Input
P10 to P15:	Port1	TO80, TO90:	Timer Output
P20 to P27:	Port2	TxD20:	Transmit Data
P30, P31:	Port3	V_{DD}:	Power Supply
RESET:	Reset	V_{PP}:	Programming Power Supply
RxD20:	Receive Data	V_{SS}:	Ground
		X1, X2:	Crystal 1, 2

2. BLOCK DIAGRAM



3.PIN FUNCTIONS

3.1 Port Pins

Pin Name	I/O	Function	After Reset	Alternate Function
P00 to P07	I/O	Port 0 8-bit input/output port Input/output can be specified in 1-bit units When used as an input port, an on-chip pull-up resistor can be specified by means of pull-up resistor option register 0 (PU0).	Input	–
P10 to P15	I/O	Port 1 6-bit input/output port Input/output can be specified in 1-bit units When used as an input port, an on-chip pull-up resistor can be specified by means of pull-up resistor option register 0 (PU0).	Input	–
P20	I/O	Port 2 8-bit input/output port Input/output can be specified in 1-bit units An input port, an on-chip pull-up resistor can be specified by means of pull-up resistor option register B2 (PUB2).	Input	$\overline{\text{SCK20}}$/ASCK20
P21				SO20/TxD20
P22				SI20/RxD20
P23				$\overline{\text{SS20}}$
P24				INTP0
P25				INTP1
P26				INTP2/CPT90
P27				TI80/TO80
P30	I/O	Port 3 2-bit input/output port Input/output can be specified in 1-bit units When used as an input port, an on-chip pull-up resistor can be specified by means of pull-up resistor option register 0 (PU0).	Input	TO90
P31				BZO90

3.2 Non-Port Pins

Pin Name	I/O	Function	After Reset	Alternate Function
INTP0	Input	External interrupt request input for which the valid edge (rising edge, falling edge, or both rising and falling edges) can be specified	Input	P24
INTP1				P25
INTP2				P26/CPT90
$\overline{SCK20}$	I/O	Serial interface serial clock input/output	Input	P20/ASCK20
SI20	Input	Serial interface serial data input	Input	P22/RxD20
SO20	Output	Serial interface serial data output	Input	P21/TxD20
$\overline{SS20}$	Input	Chip select input for serial interface	Input	P23
ASCK20	Input	Serial clock input for asynchronous serial interface	Input	P20/$\overline{SCK20}$
RxD20	Input	Serial data input for asynchronous serial interface	Input	P22/SI20
TxD20	Output	Serial data output for asynchronous serial interface	Input	P21/SO20
TO90	Output	16-bit timer 90 output	Input	P30
BZO90	Output	Buzzer output	Input	P31
CPT90	Input	Capture edge input	Input	P26/INTP2
TI80	Input	External count clock input to 8-bit timer/event counter 80	Input	P27/TI80
TO80	Output	8-bit timer/event counter 80 output	Input	P27/TO80
X1	Input	Connecting crystal resonator for main system clock oscillation	–	–
X2	-		–	–
$\overline{RESET}$	Input	System reset input	Input	–
V_{DD}	-	Positive power supply	–	–
V_{SS}	-	Ground potential	–	–
V_{PP}	-	Sets flash memory programming mode. Applies high voltage when a program is written or verified. Connect directly to V_{SS} in normal operation mode.	–	–

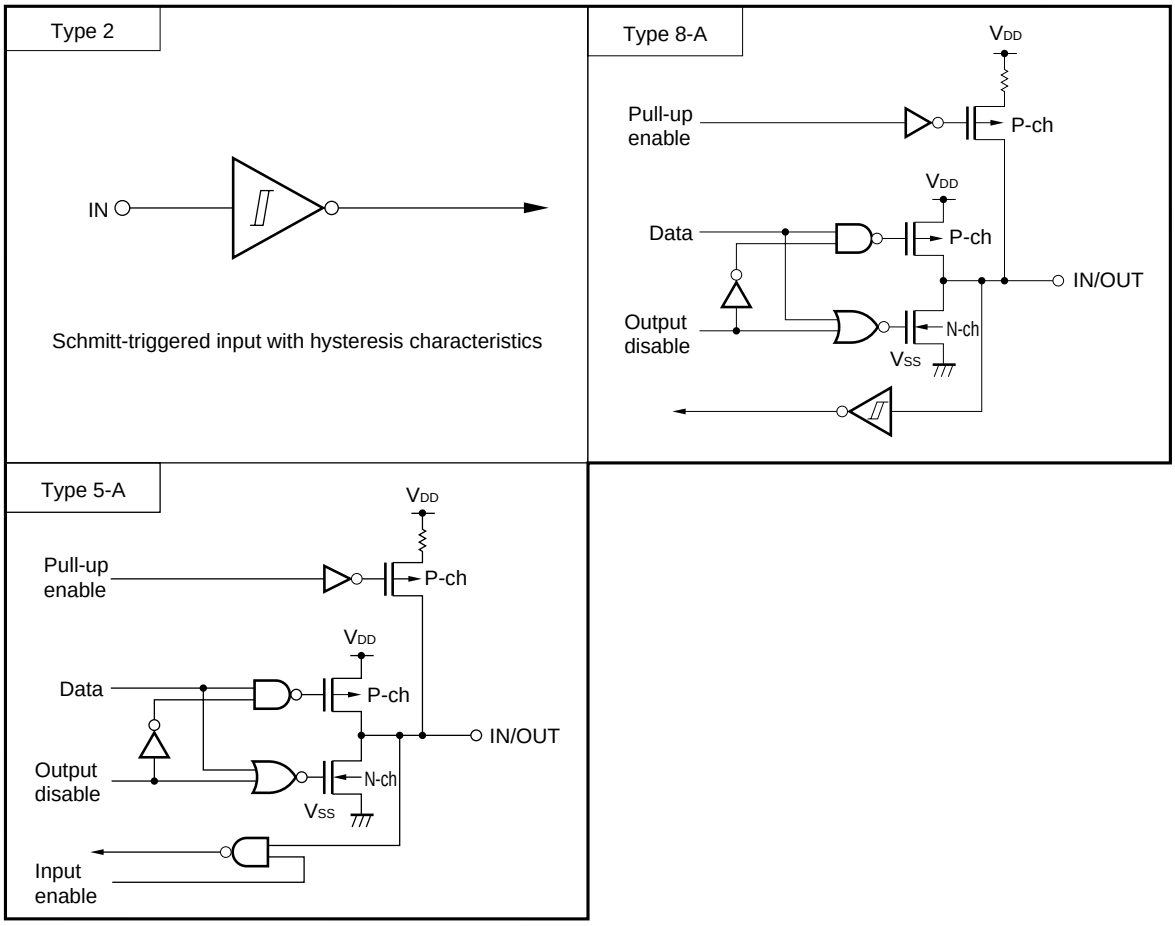
3.3 Pin I/O Circuits and Recommended Connection of Unused Pins

The input/output circuit type of each pin and recommended connection of unused pins are shown in Table 3-1. For the input/output circuit configuration of each type, refer to Figure 3-1.

Table 3-1. Types of Pin Input/Output Circuits and Recommended Connection of Unused Pins

Pin Name	Input/Output Circuit Type	I/O	Recommended Connection of Unused Pins
P00 to P07	5-A	I/O	Input: Independently connects to V_{DD} or V_{SS} via a resistor. Output: Leave open
P10 to P15			
P20/$\overline{SCK20}$/ASCK20	8-A		
P21/SO20/TxD20			
P22/SI20/RxD20			
P23/$\overline{SS20}$			
P24/INTP0			Input: Independently connects to V_{SS} via a resistor. Output: Leave open
P25/INTP1			
P26/INTP2/CPT90			
P27/TI80/TO80			Input: Independently connects to V_{DD} or V_{SS} via a resistor. Output: Leave open
P30/TO90	5-A		
P31/BZO90			
$\overline{RESET}$	2	Input	–
V_{PP}	–	–	Connect directly to V_{SS}.

Figure 3-1. Pin Input/Output Circuits

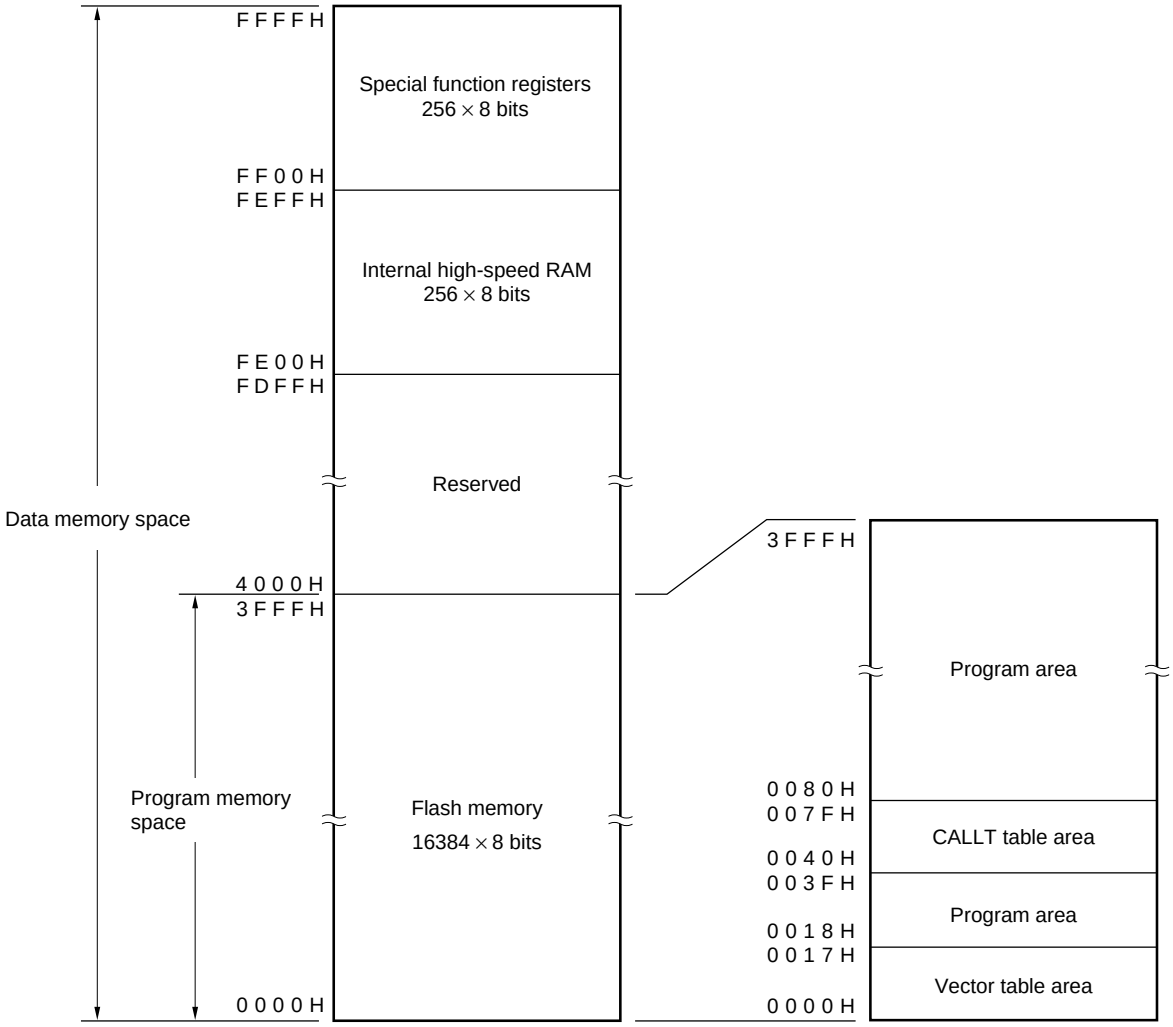


4. CPU ARCHITECTURE

4.1 Memory Space

The μPD78F9076 can access 64 Kbytes of memory space. Figure 4-1 shows the memory map of the μPD78F9076.

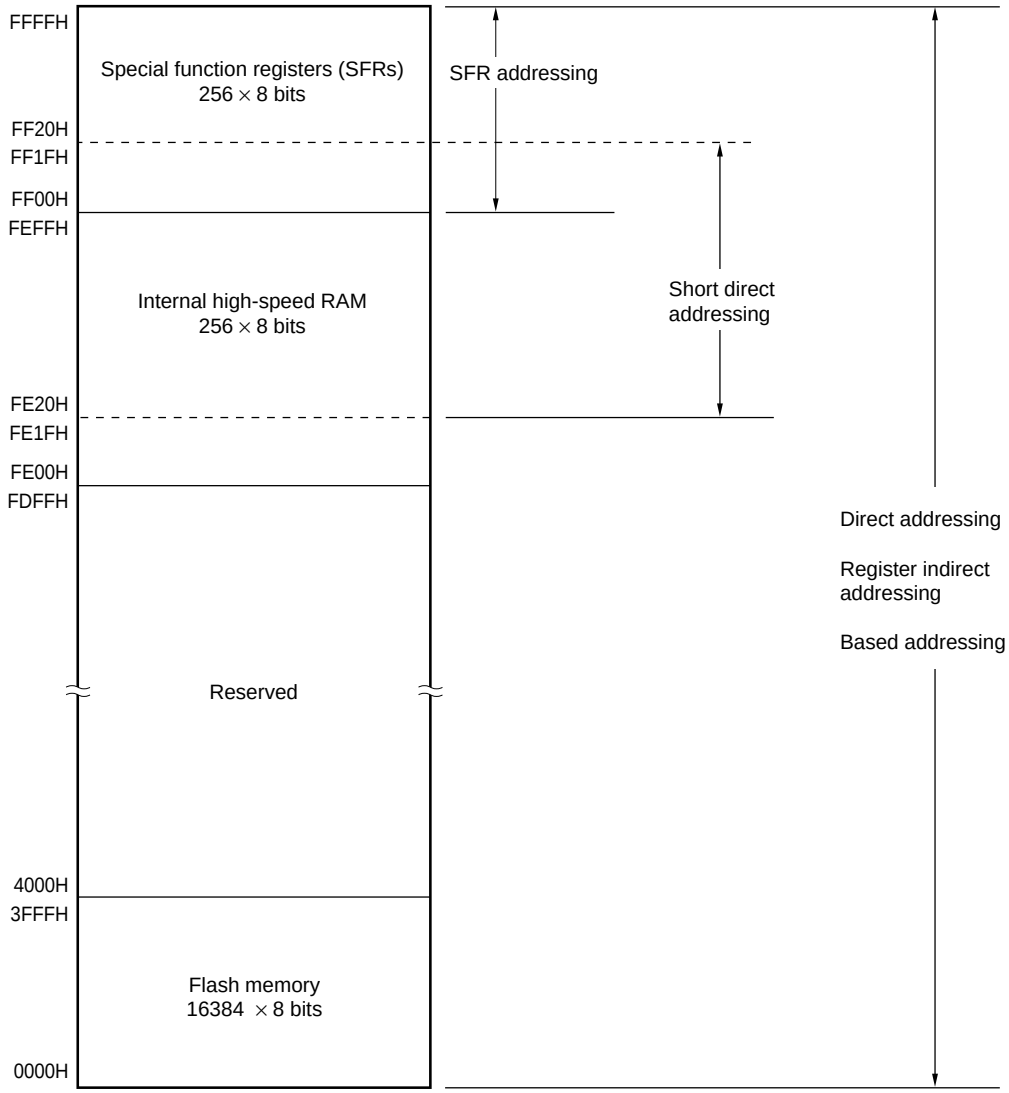
Figure 4-1. Memory Map



4.2 Data Memory Addressing

The μPD78F9076 provides a variety of addressing modes which take account of memory manipulability, etc. Especially at addresses corresponding to data memory area (FE00H to FEFFH), particular addressing modes are possible to meet the functions of the special function registers (SFRs) and general registers. Figure 4-2 shows the data memory addressing modes.

Figure 4-2. Data Memory Addressing



4.3 Processor Register

4.3.1 Control registers

(1) Program counter (PC)

The program counter is a 16-bit register that contains address information for the next program to be executed.

Figure 4-3. Configuration of Program Counter

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

(2) Program status word (PSW)

The program status word is an 8-bit register that indicates the CPU status after an instruction has been executed.

Figure 4-4. Configuration of Program Status Word

7							0
IE	Z	0	AC	0	0	1	CY

(a) Interrupt enable flag (IE)

This flag is used to control servicing of the CPU interrupt requests.

(b) Zero flag (Z)

This flag is set (to 1) when the calculation (arithmetic operation) result is 0 and is reset (to 0) for all other calculation results.

(c) Auxiliary carry flag (AC)

This flag is set (to 1) when the calculation result includes a carry from bit 3 or a borrow to bit 3 and is reset (to 0) for all other calculation results.

(d) Carry flag (CY)

This flag is used to record overflow or underflow conditions that may occur when executing an add/subtract instruction.

(3) Stack pointer (SP)

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

Figure 4-5. Configuration of Stack Pointer

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

Caution The contents of SP become undefined when $\overline{\text{RESET}}$ is input, so be sure to initialize before executing an instruction.

4.3.2 General-purpose registers

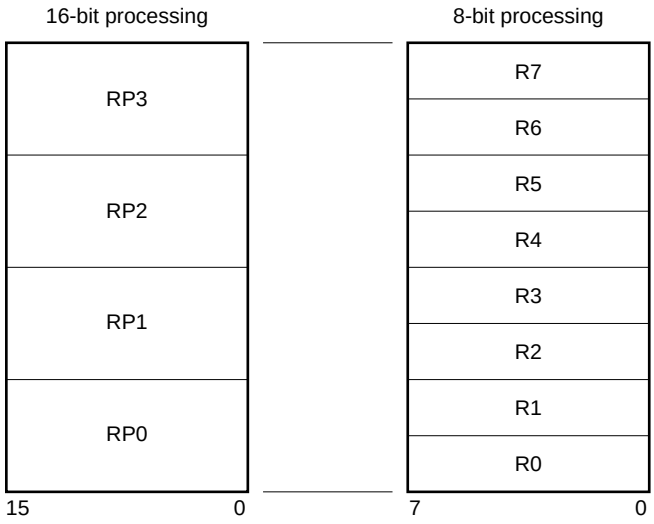
General-purpose registers consist of eight 8-bit registers (X, A, C, B, E, D, L, and H).

In addition to using each register as an 8-bit register, two 8-bit registers can be paired and used as 16-bit registers (AX, BC, DE, and HL).

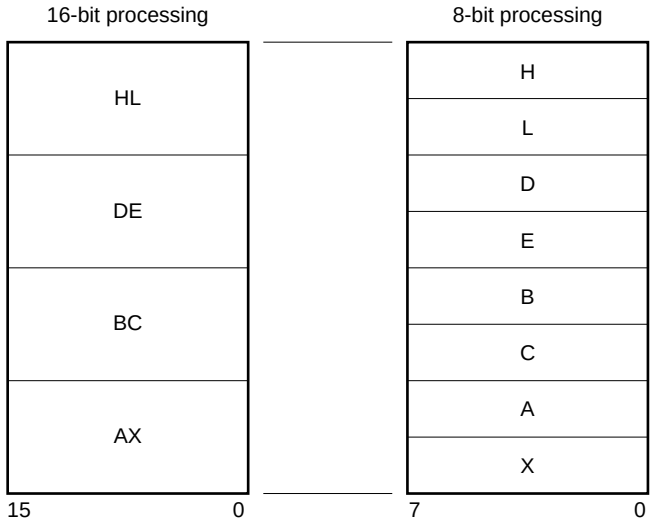
Registers can be described using function names (X, A, C, B, E, D, L, H, AX, BC, DE, and HL) or using absolute names (R0 to R7 or RP0 to RP3).

Figure 4-6. Configuration of General-purpose Registers

(a) Absolute names



(b) Function names



4.3.3 Special function registers (SFRs)

SFRs are registers such as peripheral hardware mode registers and control registers that provide various special functions. SFRs are mapped to the 256-byte space between addresses FF00H and FFFFH.

Bit symbols that are defined either as a reserved word by the RA78K0S or within the “sfrbit.h” header file by the CC78K0S are enclosed with brackets in each register format. For details, see **5. PERIPHERAL HARDWARE FUNCTIONS**.

Table 4-1. Special Function Register List (1/2)

Address	Special Function Register (SFR) Name	Symbol		R/W	Bit Manipulation Unit(s)			After Reset
					1 Bit	8 Bits	16 Bits	
FF00H	Port 0	P0		R/W	√	√	–	00H
FF01H	Port 1	P1			√	√	–	
FF02H	Port 2	P2			√	√	–	
FF03H	Port 3	P3			√	√	–	
FF16H	16-bit compare register 90	CR90 Note 1		W	–	–	√ Notes 2, 3	FFFFH
FF17H								
FF18H	16-bit timer counter 90	TM90 Note 1		R	–	–	√ Notes 2, 3	0000H
FF19H								
FF1AH	16-bit capture register 90	TCP90 Note 1			–	–	√ Notes 2, 3	Undefined
FF1BH								
FF20H	Port mode register 0	PM0		R/W	√	√	–	FFH
FF21H	Port mode register 1	PM1			√	√	–	
FF22H	Port mode register 2	PM2			√	√	–	
FF23H	Port mode register 3	PM3			√	√	–	
FF32H	Pull-up resistor option register B2	PUB2			√	√	–	00H
FF42H	Watchdog timer clock select register	WDCS			–	√	–	
FF48H	16-bit timer mode control register 90	TMC90			√	√	–	
FF49H	Buzzer output control register 90	BZC90			√	√	–	
FF50H	8-bit compare register 80	CR80		W	–	√	–	Undefined
FF51H	8-bit timer counter 80	TM80		R	–	√	–	00H
FF53H	8-bit timer mode control register 80	TMC80		R/W	√	√	–	
FF70H	Asynchronous serial interface mode register 20	ASIM20			√	√	–	
FF71H	Asynchronous serial interface status register 20	ASIS20		R	√	√	–	
FF72H	Serial operation mode register 20	CSIM20		R/W	√	√	–	
FF73H	Baud rate generator control register 20	BRGC20			–	√	–	
FF74H	Transmit shift register 20	TXS20	SIO20	W	–	√	–	FFH
	Receive buffer register 20	RXB20		R	–	√	–	Undefined

Notes 1. These SFR names are for 16-bit access only.

2. 16-bit access is enabled for short direct addressing only.

3. 8-bit access is also enabled in addition to 16-bit access. In such cases, use direct addressing.

Table 4-1. Special Function Register List (2/2)

Address	Special Function Register (SFR) Name	Symbol	R/W	Bit Manipulation Unit(s)			After Reset
				1 Bit	8 Bits	16 Bits	
FFE0H	Interrupt request flag register 0	IF0	R/W	√	√	–	00H
FFE1H	Interrupt request flag register 1	IF1		√	√	–	
FFE4H	Interrupt mask flag register 0	MK0		√	√	–	FFH
FFE5H	Interrupt mask flag register 1	MK1		√	√	–	
FFECH	External interrupt mode register 0	INTM0		–	√	–	00H
FFF7H	Pull-up resistor option register 0	PU0		√	√	–	
FFF9H	Watchdog timer mode register	WDTM		√	√	–	
FFFAH	Oscillation stabilization time select register	OSTS		–	√	–	04H
FFFBH	Processor clock control register	PCC		√	√	–	02H

5. PERIPHERAL HARDWARE FUNCTIONS

5.1 Ports

5.1.1 Port functions

The μPD78F9076 includes the ports shown in Figure 5-1, which can be used for a variety of control operations. The port functions are listed in Table 5-1.

In addition to functions that use the ports as digital I/O ports, some alternate functions are also provided. For details of alternate functions, see **3. PIN FUNCTIONS**.

Figure 5-1. Port Types

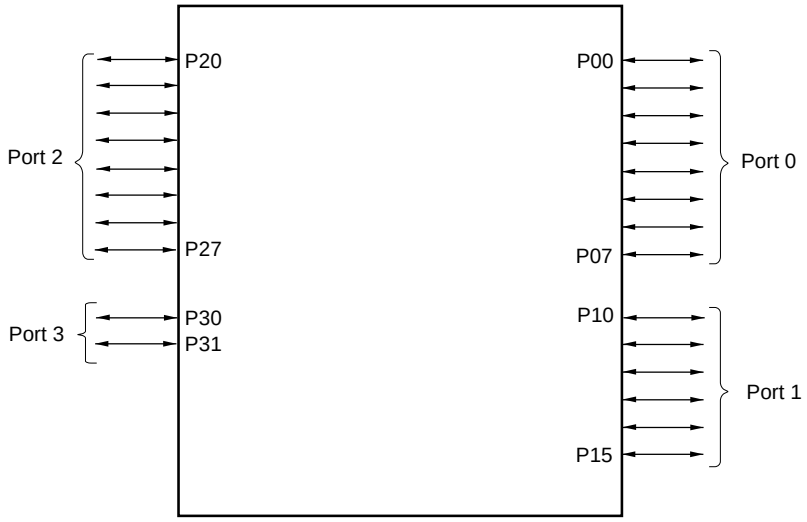


Table 5-1. Port Functions

Port Name	Pin Name	Function
Port 0	P00 to P07	Input/output port. Input/output can be specified in 1-bit units. When used as an input port, an on-chip pull-up resistor can be used by pull-up resistor option register 0 (PU0).
Port 1	P10 to P15	Input/output port. Input/output can be specified in 1-bit units. When used as an input port, an on-chip pull-up resistor can be used by pull-up resistor option register 0 (PU0).
Port 2	P20 to P27	Input/output port. Input/output can be specified in 1-bit units. An on-chip pull-up resistor can be used by pull-up resistor option register B2 (PUB2).
Port 3	P30, P31	Input/output port. Input/output can be specified in 1-bit units. When used as an input port, an on-chip pull-up resistor can be used by pull-up resistor option register 0 (PU0).

5.1.2 Port configuration

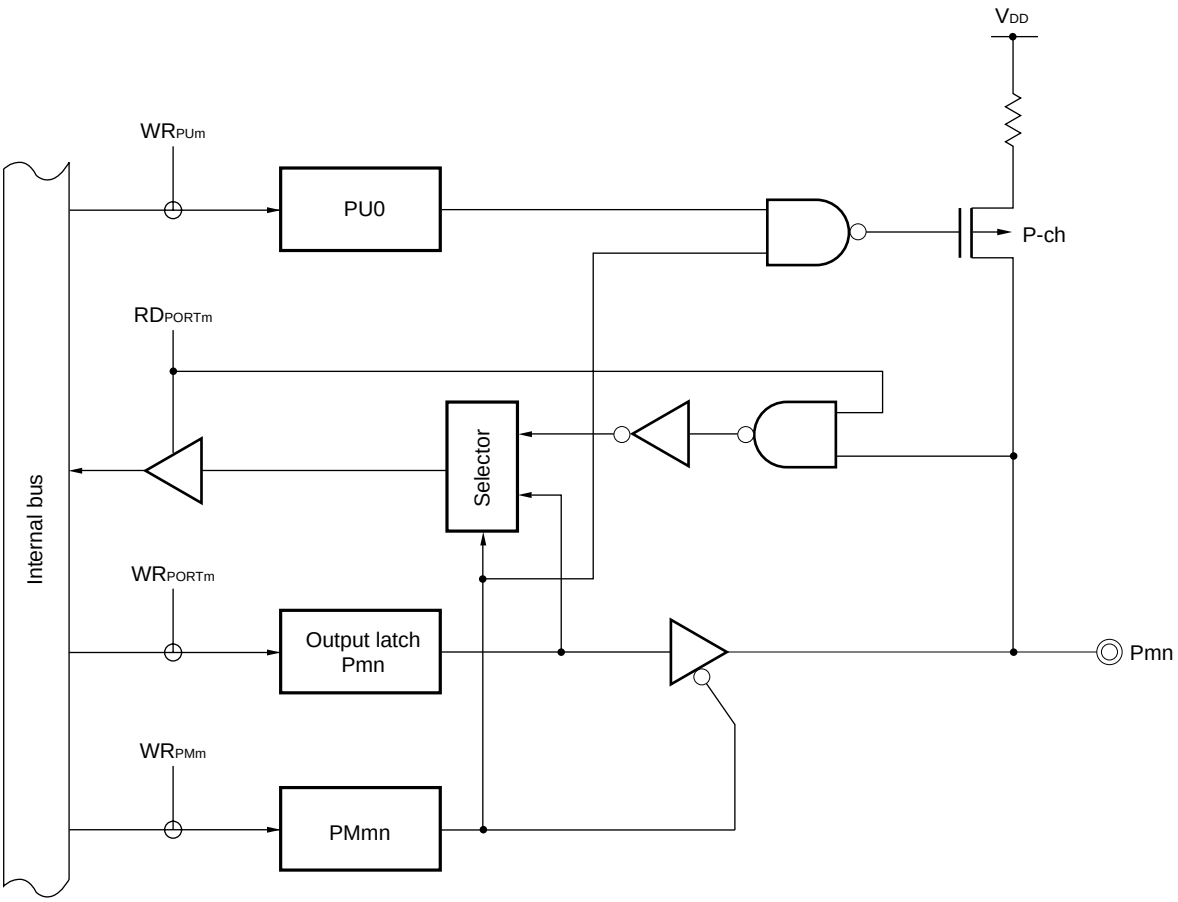
The ports include the following hardware.

Table 5-2. Port Configuration

Item	Configuration
Control registers	Port mode register (PMm: m = 0 to 3) Pull-up resistor option registers (PU0, PUB2)
Ports	Total: 24
Pull-up resistors	Total: 24 (24 with software control)

Figure 5-2. Basic Configuration of CMOS Ports (1/2)

(1) Basic Configuration of Ports 0, 1 and 3

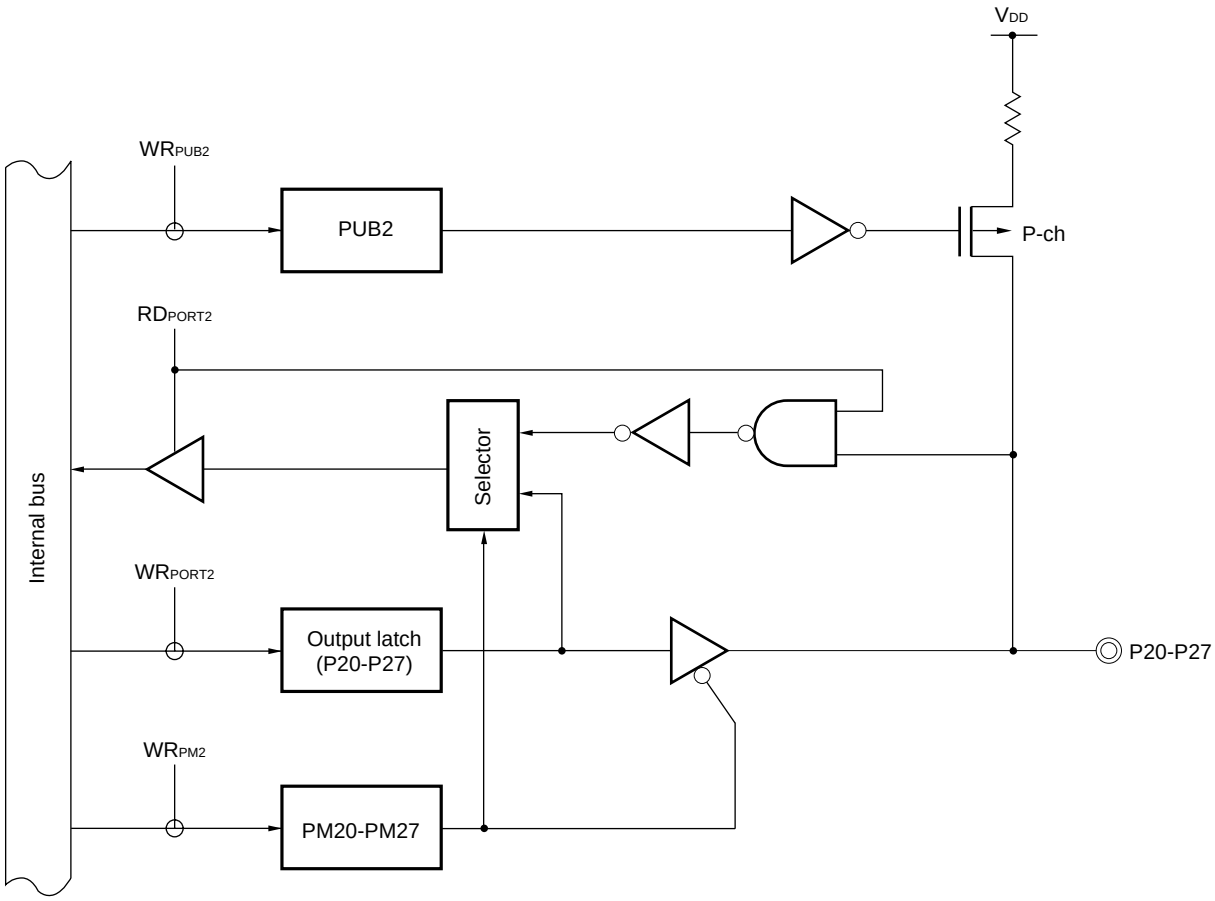


Caution Figure 5-2 shows the basic configuration of ports 0, 1 and 3. This configuration varies depending on the alternate pin functions.

Remark PU0: Pull-up resistor option register 0
PMmn: Bit n of port mode register m (m = 0, 1 and 3; n = 0 to 7)
Pmn: Bit n of port m
RD: Port read signal
WR: Port write signal

Figure 5-2. Basic Configuration of CMOS Ports (2/2)

(2) Basic Configuration of Port 2



Caution Figure 5-2 shows the basic configuration of port 2. This configuration varies depending on the alternate pin functions.

Remark PUB2: Pull-up resistor option register B2
PM2: Bit n of port mode register 2
RD: Port read signal
WR: Port write signal

5.1.3 Control registers for port functions

Ports are controlled by the following two types of registers.

- Port mode registers (PM0 to PM3)
- Pull-up resistor option registers (PU0, PUB2)

(1) Port mode registers (PM0 to PM3)

These registers set the port I/O mode in 1-bit units.

The port mode registers are set via a 1-bit memory manipulation instruction or an 8-bit memory manipulation instruction.

After the $\overline{\text{RESET}}$ signal is input, this register value is FFH.

When port pins are used as alternate function pins, set the port mode register and output latch as shown in Table 5-3.

Caution Since port 2 is used as an external interrupt input, the interrupt request flag becomes set whenever the port function's output mode is specified and the output level is changed. Therefore, be sure to set 1 to the interrupt mask flag before using the output mode.

Figure 5-3. Format of Port Mode Registers

Symbol	7	6	5	4	3	2	1	0	Address	After reset	R/W
PM0	PM07	PM06	PM05	PM04	PM03	PM02	PM01	PM00	FF20H	FFH	R/W
PM1	1	1	PM15	PM14	PM13	PM12	PM11	PM10	FF21H	FFH	R/W
PM2	PM27	PM26	PM25	PM24	PM23	PM22	PM21	PM20	FF22H	FFH	R/W
PM3	1	1	1	1	1	1	PM31	PM30	FF23H	FFH	R/W

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

Table 5-3. Port Mode Register and Output Latch Settings when Using Alternate Functions

Pin Name	Alternate Function		PM××	P××
	Name	I/O		
P24	INTP0	Input	1	×
P25	INTP1	Input	1	×
P26	INTP2	Input	1	×
	CPT90	Input	1	×
P27	TI80	Input	1	×
	TO80	Output	0	0
P30	TO90	Output	0	0
P31	BZO90	Output	0	0

Caution When using port 2 as a serial interface pin, it must be set either as an I/O or an output latch, according to the function. For details of these settings, see 5.6 Serial Interface 20.

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

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

This register sets whether or not to use on-chip pull-up resistor for ports 0, 1 and 3. At the port where use of a pull-up resistor is specified via PU0, an on-chip pull-up resistor can be used in port units only for bits that have been set to input mode. An on-chip pull-up resistor cannot be used for any bit that has been set to output mode, regardless of the PU0 setting. This is also true when it is used as an output pin for an alternate function.

The PU0 is set via a 1-bit memory manipulation instruction or an 8-bit memory manipulation instruction.

After the $\overline{\text{RESET}}$ signal is input, this register value is 00H.

Figure 5-4. Format of Pull-up Resistor Option Register 0

Symbol	7	6	5	4	<3>	2	<1>	<0>	Address	After reset	R/W
PU0	0	0	0	0	PU03	0	PU01	PU00	FFF7H	00H	R/W

PU0m	Selection of On-chip Pull-up Resistor for Port m (m = 0, 1, 3)
0	Do not connect to on-chip pull-up resistor
1	Connect to on-chip pull-up resistor

Caution Bits 2 and bits 4 to 7 must be set to 0.

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

This register sets whether or not to use on-chip pull-up resistor for P20 to P27. The pin so specified by PUB2 is connected to on-chip pull-up registor regardless of the setting of the port mode register.

The PUB2 is set via a 1-bit memory manipulation instruction or an 8-bit memory manipulation instruction.

After the $\overline{\text{RESET}}$ signal is input, this register value is 00H.

Figure 5-5. Format of Pull-up Resistor Option Register B2

Symbol	<7>	<6>	<5>	<4>	<3>	<2>	<1>	<0>	Address	After reset	R/W
PUB2	PUB27	PUB26	PUB25	PUB24	PUB23	PUB22	PUB21	PUB20	FF32H	00H	R/W

PUB2n	Selection of On-chip Pull-up Resistor for P2n (n = 0 to 7)
0	Do not connect to on-chip pull-up resistor
1	Connect to on-chip pull-up resistor

5.2 Clock Generator

5.2.1 Function of clock generator

The clock generator generates the clock to be supplied to the CPU and the peripheral hardware.

The system clock oscillator is the following type.

- System clock oscillator
  This circuit's oscillation frequency range is 1.0 to 5.0 MHz. Oscillation can be stopped by executing the STOP instruction

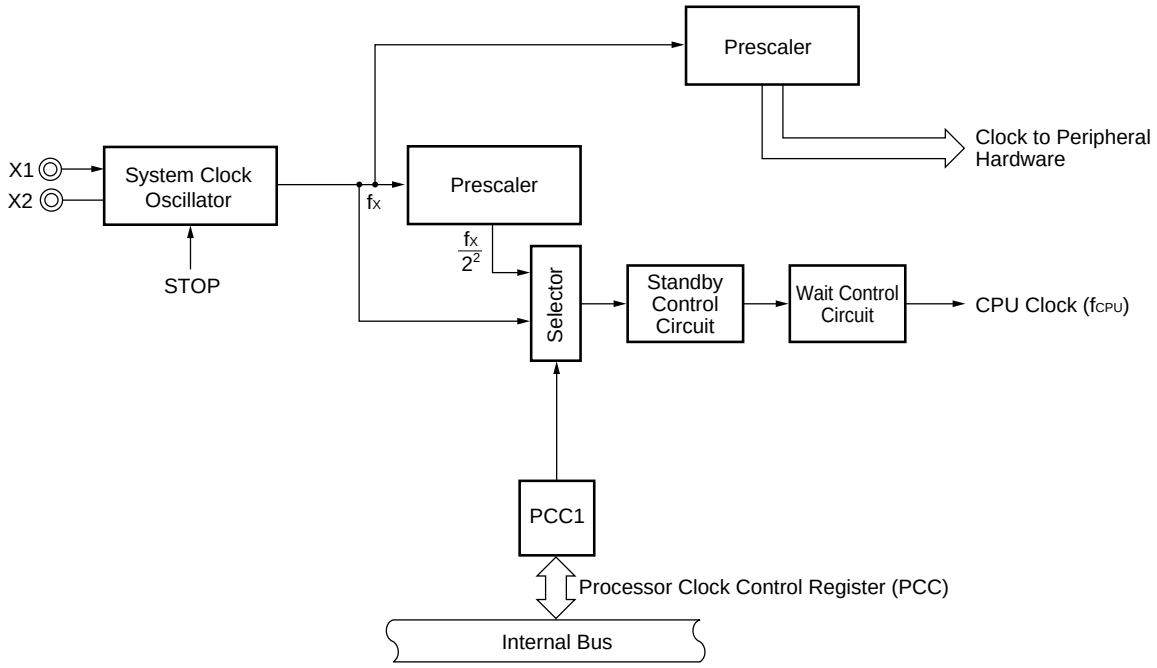
5.2.2 Configuration of clock generator

The clock generator includes the following hardware.

Table 5-4. Configuration of Clock Generator

Item	Configuration
Control registers	Processor clock control register (PCC)

Figure 5-6. Block Diagram of Clock Generator



5.2.3 Control register for clock generator

The clock generator is controlled by the following register:

- Processor clock control register (PCC)

(1) Processor clock control register (PCC)

PCC sets the CPU clock selection and the ratio of division.

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

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

Figure 5-7. Format of Processor Clock Control Register

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

PCC1	Selection of CPU Clock (f_{CPU})
0	f_X (0.2 μs)
1	$f_X/2^2$ (0.8 μs)

Caution Bit 0 and bits 2 to 7 must be set to 0.

Remarks 1. f_X: System clock oscillation frequency

2. Values in parentheses are when operating at f_X = 5.0-MHz.

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

- When f_{CPU} = 0.2 μs, this value is 0.4 μs
- When f_{CPU} = 0.8 μs, this value is 1.6 μs

5.3 16-Bit Timer 90

5.3.1 Function of 16-bit timer 90

The 16-bit timer 90 has the following functions.

- Timer interrupt
- Timer output
- Buzzer output
- Count value capture

(1) Timer interrupt

An interrupt is generated when a count value matches.

(2) Timer output

Timer output can be controlled when a count value and compare value matches.

(3) Buzzer output

Buzzer output can be controlled by software.

(4) Count value capture

A count value of 16-bit timer counter 90 is latched into a capture register synchronizing with the capture trigger and retained.

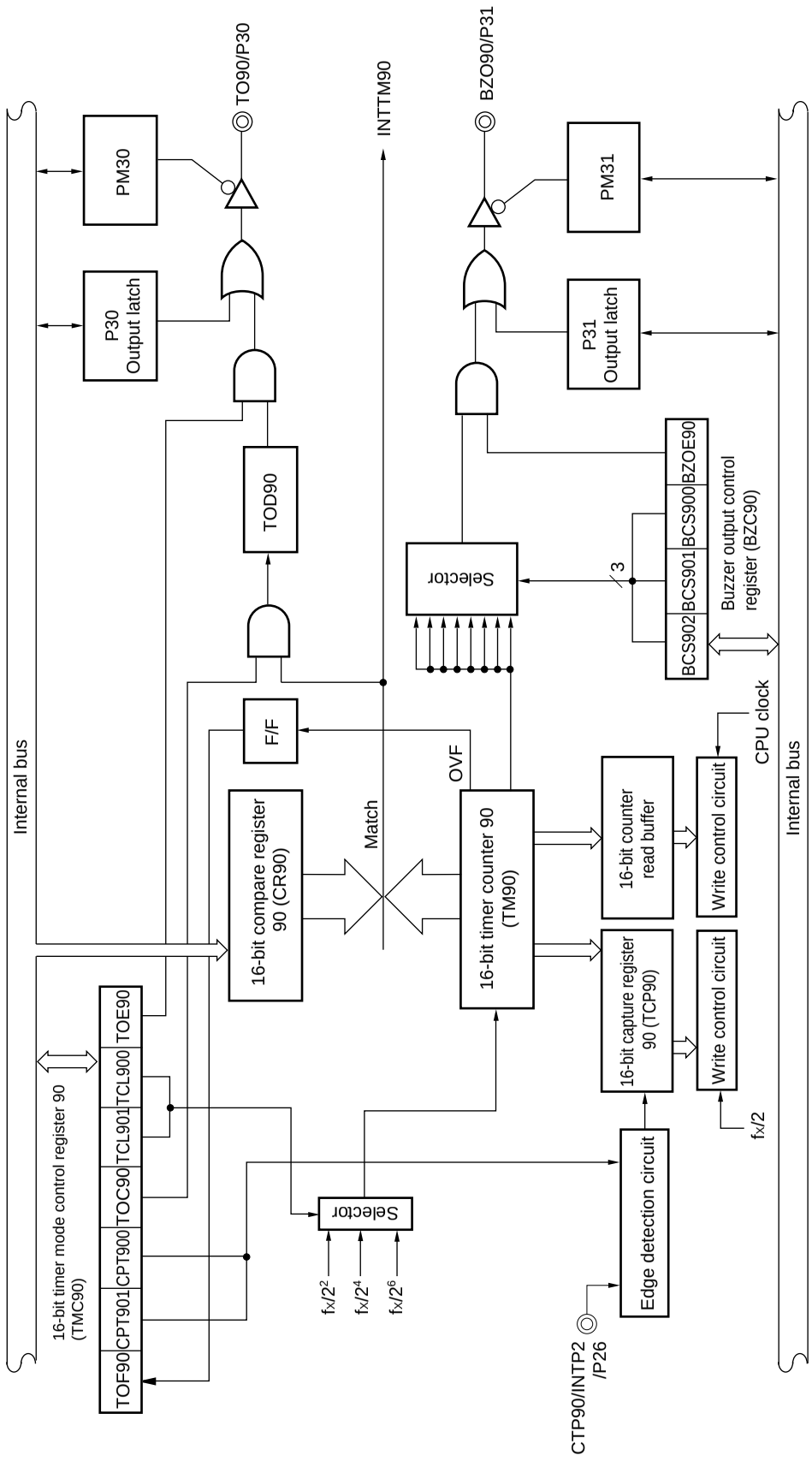
5.3.2 Configuration of 16-bit timer 90

The 16-bit timer (TM90) includes the following hardware.

Table 5-5. Configuration of 16-Bit Timer 90

Item	Configuration
Timer counter	16 bits × 1 (TM90)
Registers	Compare register: 16 bits × 1 (CR90) Capture register: 16 bits × 1 (TCP90)
Timer output	1 (TO90)
Control registers	16-bit timer mode control register 90 (TMC90) Buzzer output control register 90 (BZC90) Port mode register 2 (PM2)

Figure 5-8. Block Diagram of 16-Bit Timer



(1) 16-bit compare register 90 (CR90)

A value specified in CR90 is compared with the count in 16-bit timer register 90 (TM90). If they match, an interrupt request (INTTM90) is issued by CR90.

CR90 is set with an 8-bit or 16-bit memory manipulation instruction. Any value from 0000H to FFFFH can be set.

$\overline{\text{RESET}}$ input sets CR90 to FFFFH.

Cautions 1. CR90 is designed to be manipulated with a 16-bit memory manipulation instruction. It can also be manipulated with 8-bit memory manipulation instructions, however. When an 8-bit memory manipulation instruction is used to set CR90, it must be accessed in direct addressing.

2. To re-set CR90 during count operation, it is necessary to disable interrupts in advance, using 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 90 (TMC90).
If the value in CR90 is rewritten in the interrupt-enabled state, an interrupt request may occur at the moment of rewrite.

(2) 16-bit timer counter 90 (TM90)

TM90 is used to count the number of pulses.

The contents of TM90 are read with an 8-bit or 16-bit memory manipulation instruction.

This register is in free running during count clock input.

$\overline{\text{RESET}}$ input clears TM90 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. TM90 is designed to be manipulated with a 16-bit memory manipulation instruction. It can also be manipulated with 8-bit memory manipulation instructions, however. When an 8-bit memory instruction is used to manipulate TM90, it must be accessed in direct addressing.

3. When an 8-bit memory manipulation instruction is used to manipulate TM90, the lower and upper bytes must be read as a pair, in this order.

(3) 16-bit capture register 90 (TCP90)

TCP90 captures the contents of TM90.

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

$\overline{\text{RESET}}$ input makes TCP90 undefined.

Caution TCP90 is designed to be manipulated with a 16-bit memory manipulation instruction. It can also be manipulated with 8-bit memory manipulation instructions, however. When an 8-bit memory manipulation instruction is used to manipulate TCP90, it must be accessed in direct addressing.

(4) 16-bit counter read buffer 90

This buffer is used to latch and hold the count for TM90.

5.3.3 Control registers for 16-bit timer 90

The following three types of registers control the 16-bit timer.

- 16-bit timer mode control register 90 (TMC90)
- Buzzer output control register 90 (BZC90)
- Port mode register 3 (PM3)

(1) 16-bit timer mode control register 90 (TMC90)

16-bit timer mode control register 90 (TMC90) controls the setting of a count clock, capture edge, etc.

TMC90 is set with a 1-bit or 8-bit memory manipulation instruction.

$\overline{\text{RESET}}$ input clears TMC90 to 00H.

Figure 5-9. Format of 16-Bit Timer Mode Control Register 90

Symbol	7	<6>	5	4	3	2	1	<0>	Address	After reset	R/W
TMC90	TOD90	TOF90	CPT901	CPT900	TOC90	TCL901	TCL900	TOE90	FF48H	00H	R/W Note

TOD90	Timer output data
0	Timer output of 0
1	Timer output of 1

TOF90	Overflow flag control
0	Reset or cleared by software
1	Set when the 16-bit timer overflows

CPT901	CPT900	Capture edge selection
0	0	Capture operation disabled
0	1	Captured at the rising edge at the CPT90 pin
1	0	Captured at the falling edge at the CPT90 pin
1	1	Captured at both the rising and falling edges at the CPT90 pin

TOC90	Timer output data inversion control
0	Inversion disabled
1	Inversion enabled

TCL901	TCL900	16-bit timer counter 90 count clock selection
0	0	$f_X/2^2$ (1.25 MHz)
0	1	$f_X/2^6$ (78.1 kHz)
1	0	$f_X/2^4$ (313 kHz)
1	1	Setting prohibited

TOE90	16-bit timer counter 90 output control
0	Output disabled (port mode)
1	Output enabled

Note Bit 7 is read-only.

Caution Disable the interrupt in advance by using the interrupt mask flag register (MK1) to change the data of TCL901 and TCL900. Also, prevent the timer output data from being inverted by setting TOC90 to1.

Remarks 1. f_X:System clock oscillation frequency

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

(2) Buzzer output control register 90 (BZC90)

This register selects a buzzer frequency based on fcl selected with the count clock select bits (TCL901 and TCL900), and controls the output of a square wave.

BZC90 is set with a 1-bit or 8-bit memory manipulation instruction.

$\overline{\text{RESET}}$ input clears BZC90 to 00H.

Figure 5-10. Format of Buzzer Output Control Register 90

Symbol	7	6	5	4	3	2	1	<0>	Address	After reset	R/W
BZC90	0	0	0	0	BCS902	BCS901	BCS900	BZOE90	FF49H	00H	R/W Note

BCS902	BCS901	BCS900	Buzzer frequency		
			fcl = $f_X/2^2$	fcl = $f_X/2^6$	fcl = $f_X/2^4$
0	0	0	fcl/2^4 (78.1 kHz)	fcl/2^4 (4.88 kHz)	fcl/2^4 (19.5 kHz)
0	0	1	fcl/2^5 (39.01 kHz)	fcl/2^5 (2.44 kHz)	fcl/2^5 (9.77 kHz)
0	1	0	fcl/2^8 (4.88 kHz)	fcl/2^8 (305 Hz)	fcl/2^8 (1.22 kHz)
0	1	1	fcl/2^9 (2.44 kHz)	fcl/2^9 (153 Hz)	fcl/2^9 (610 Hz)
1	0	0	fcl/2^{10} (1.22 kHz)	fcl/2^{10} (76 Hz)	fcl/2^{10} (305 Hz)
1	0	1	fcl/2^{11} (610 Hz)	fcl/2^{11} (38 Hz)	fcl/2^{11} (153 Hz)
1	1	0	fcl/2^{12} (305 kHz)	fcl/2^{12} (19 Hz)	fcl/2^{12} (76.3 Hz)
1	1	1	fcl/2^{13} (153 kHz)	fcl/2^{13} (10 Hz)	fcl/2^{13} (38.1 Hz)

BZOE90	Buzzer port output control
0	Disables buzzer port output.
1	Enables buzzer port output. Note 2

Notes Bits 4 to 7 must all be set to 0.

Remarks 1. f_X: System clock oscillation frequency

2. f_{cl}: Count clock frequency of 16-bit timer 90

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

(3) Port mode register 3 (PM3)

PM3 is used to set each bit of port 3 to input or output.

When pin P30/TO90 is used for timer output, reset the output latch of P30 and PM30 to 0, when pin P31/BZO90 is used for buzzer output, reset the output latch of P31 and PM31 to 0.

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

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

Figure 5-11. Format of Port Mode Register 3

Symbol	7	6	5	4	3	2	1	0	Address	After reset	R/W
PM3	1	1	1	1	1	1	PM31	PM30	FF23H	00H	R/W

PM3n	P3n pin I/O mode (n = 0 or 1)
0	Output mode (output buffer ON)
1	Input mode (output buffer OFF)

5.4 8-Bit Timer/Event Counter

5.4.1 Functions of 8-bit timer/event counter

The 8-bit timer/event counters 80 (TM80) has the following functions:

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

(1) 8-bit interval timer

When an 8-bit timer/event counter is used as an interval timer, it generates an interrupt at any time intervals set in advance.

Table 5-6. Interval Time of 8-Bit Timer/Event Counter 80

Minimum Interval Time	Maximum Interval Time	Resolution
$1/f_X$ (200 ns)	$2^8/f_X$ (51.2 μs)	$1/f_X$ (200 ns)
$2^8/f_X$ (51.2 μs)	$2^{16}/f_X$ (13.1 ms)	$2^8/f_X$ (51.2 μs)

Remarks 1. f_X: System clock oscillation frequency

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

(2) External event counter

The number of pulses of an externally input signal can be counted.

(3) Square wave output

A square wave of arbitrary frequency can be output.

Table 5-7. Square Wave Output Range of 8-Bit Timer/Event Counter 80

Minimum Pulse Width	Maximum Pulse Width	Resolution
$1/f_X$ (200 ns)	$2^8/f_X$ (51.2 μs)	$1/f_X$ (200 ns)
$2^8/f_X$ (51.2 μs)	$2^{16}/f_X$ (13.1 ms)	$2^8/f_X$ (51.2 μs)

Remarks 1. f_X: System clock oscillation frequency

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

(4) PWM output

8-bit resolution PWM output can be produced.

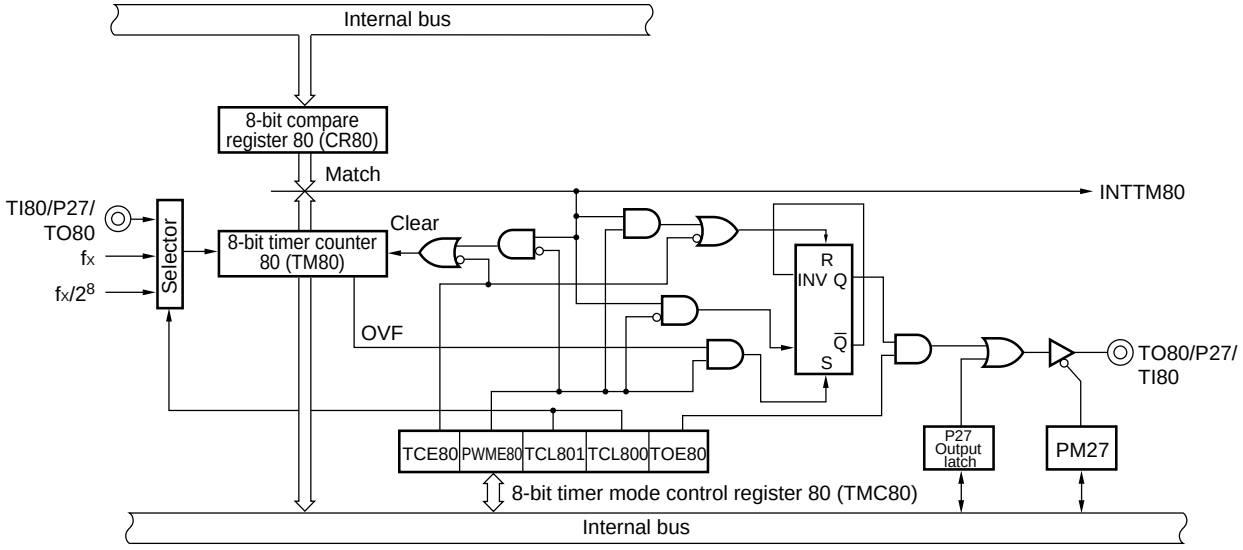
5.4.2 Configuration of 8-bit timer/event counter 80

The 8-bit timer/event counter 80 consists of the following hardware.

Table 5-8. 8-Bit Timer/Event Counter 80 Configuration

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

Figure 5-12. Block Diagram of 8-Bit Timer/Event Counter 80



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

This is an 8-bit register that compares the value set to CR80 with the 8-bit timer register 80 (TM80) count value, and if they match, generates an interrupt request (INTTM80).

CR80 is set with an 8-bit memory manipulation instruction. The values 00H to FFH can be set.

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

Cautions 1. Before rewriting CR80, stop the timer operation once. If CR80 is rewritten in the timer operation-enabled state, a match interrupt request signal may occur at the moment of rewrite.

2. Do not set CR80 to 00H in the 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 counter 80 (TM80)

This is an 8-bit register to count pulses.

TM80 is read with an 8-bit memory manipulation instruction.

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

5.4.3 Control registers for 8-bit timer/event counter 80

The 8-bit timer/event counter 80 is controlled by the following two types of registers.

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

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

This register enables/stops operation of 8-bit timer register 80 (TM80) , sets the count clock of TM80, and controls the operation of the output control circuit of 8-bit timer/event counter 80.

The TMC80 is set via a 1-bit memory manipulation instruction or an 8-bit memory manipulation instruction.

After the $\overline{\text{RESET}}$ signal is input, this register value is 00H.

Figure 5-13. Format of 8-Bit Timer Mode Control Register 80

Symbol	<7>	<6>	5	4	3	2	1	<0>	Address	After 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 stop (TM80 cleared to 0)
1	Operation enable

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

TCL801	TCL800	8-bit timer register 80 count clock selection
0	0	f_X (5.0 MHz)
0	1	$f_X/2^8$ (19.5 kHz)
1	0	Rising edge of TI80 Note
1	1	Falling edge of TI80 Note

TOE80	8-bit timer/event counter 80 output control
0	Output disable (port mode)
1	Output enable

Note When clock is externally input, timer output cannot be used.

Cautions 1. Be sure to set TMC80 after stopping timer operation.

2. For PWM mode operation, the interrupt mask flag (TMMK80) must be set.

Remarks 1. f_X: System clock oscillation frequency

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

(2) Port mode register 2 (PM2)

PM2 is used to set each bit of port 2 to input or output.

When pin TO80/P27/TI80 is used for timer output, reset the output latch of P27 and PM27 to 0.

PM2 is set with a 1-bit or 8-bit memory manipulation instruction.

$\overline{\text{RESET}}$ input sets PM2 to FFH.

Figure 5-14. Format of Port Mode Register 2

Symbol	7	6	5	4	3	2	1	0	Address	After reset	R/W
PM2	PM27	PM26	PM25	PM24	PM23	PM22	PM21	PM20	FF22H	FFH	R/W

PM27	P27 pin I/O mode selection
0	Output mode (output buffer ON)
1	Input mode (output buffer OFF)

5.5 Watchdog Timer

5.5.1 Functions of watchdog timer

The watchdog timer has the following functions.

(1) Watchdog timer

The watchdog timer detects program runaway. When runaway is detected, a non-maskable interrupt or a $\overline{\text{RESET}}$ signal can be issued.

(2) Interval timer

An interrupt is issued at a preset interval (any interval time can be set).

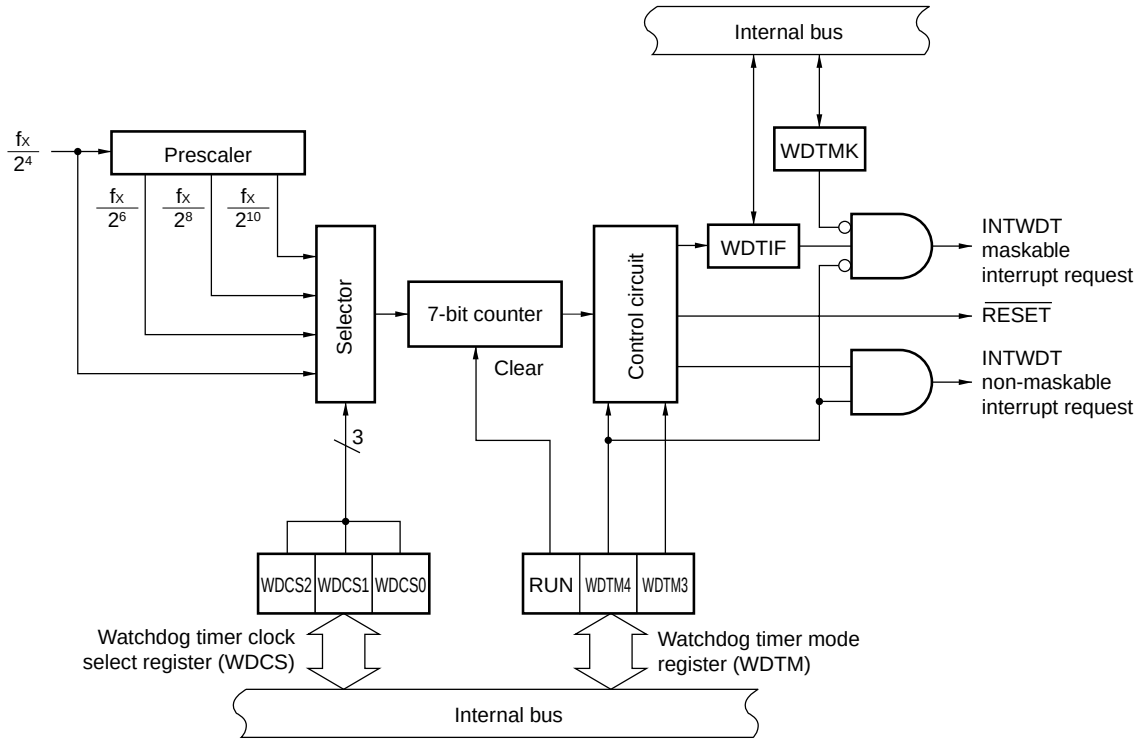
5.5.2 Configuration of watchdog timer

The watchdog timer includes the following hardware.

Table 5-9. Configuration of Watchdog Timer

Item	Configuration
Control registers	Watchdog timer clock select register (WDCS) Watchdog timer mode register (WDTM)

Figure 5-15. Block Diagram of Watchdog Timer



5.5.3 Control registers for watchdog timer

The watchdog timer is controlled by the following two types of registers.

- Watchdog timer clock select register (WDCS)
- Watchdog timer mode register (WDTM)

(1) Watchdog timer clock select register (WDCS)

This register is used to set the watchdog timer count clock.

The WDCS is set via an 8-bit memory manipulation instruction.

After the $\overline{\text{RESET}}$ signal is input, this register value is 00H.

Figure 5-16. Format of Watchdog Timer Clock Select Register

Symbol	7	6	5	4	3	2	1	0	Address	After reset	R/W
WDCS	0	0	0	0	0	WDCS2	WDCS1	WDCS0	FF42H	00H	R/W

WDCS2	WDCS1	WDCS0	Selection of Count Clock	Interval Time
0	0	0	$f_X/2^4$ (313 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 than above			Setting prohibited	

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)

This register is used to set the watchdog timer operation mode and count enable/disable status.

The WDTM is set via a 1-bit memory manipulation instruction or an 8-bit memory manipulation instruction.

After the $\overline{\text{RESET}}$ signal is input, this register value is 00H.

Figure 5-17. Format of Watchdog Timer Mode Register

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

RUN	Selection of Watchdog Timer Operation[Note 1]
0	Stops count operation
1	Clears count and starts count operation

WDTM4	WDTM3	Selection of Watchdog Timer Operation Mode[Note 2]
0	0	Operation is stopped
0	1	Interval timer mode (a maskable interrupt is issued in response to an overflow)[Note 3]
1	0	Watchdog timer mode 1 (a non-maskable interrupt is issued in response to an overflow)
1	1	Watchdog timer mode 2 (a reset operation is started in response to an overflow)

Notes 1. Once the RUN bit has been set (to 1), the count cannot be cleared (to 0) by software. Therefore, when a count has been started it cannot be stopped by anything other than input of the $\overline{\text{RESET}}$ signal.

2. Once WDTM3 and WDTM4 are set (to 1), the count cannot be cleared (to 0) by software.

3. Once 1 is set to the RUN bit, the timer starts operating as an interval timer.

Cautions 1. Once 1 is set to the RUN bit and the watchdog timer has been cleared, the actual overflow time becomes up to 0.8% shorter than the time that was set via the watchdog timer clock select register.

2. When using watchdog timer mode 1 or 2, confirm that the WDTIF (bit 0 in interrupt request flag register 0 (IF0)) value is 0, then set 1 to WDTM4. If watchdog timer mode 1 or 2 is selected and overwritten while the WDTIF value is 1, a non-maskable interrupt will occur.

5.6 Serial Interface 20

5.6.1 Functions of Serial interface 20

Serial interface 20 has the following three modes.

- Operation stop mode
- Asynchronous serial interface (UART) mode
- 3-wire serial I/O mode

(1) Operation stop mode

This mode is used when serial transfer is not 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 channel 0 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.

(3) 3-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 ($\overline{SCK20}$) line and two serial data lines (SI20 and SO20).

As it supports simultaneous transmission and reception, 3-wire serial I/O mode requires less processing time for data transmission than asynchronous serial interface mode.

Because, in 3-wire serial I/O mode, it is possible to select whether 8-bit data transmission begins with the MSB or LSB, channel 0 can be connected to any device regardless of whether that device is designed for MSB-first or LSB-first transmission.

3-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 75XL, 78K, and 17K Series devices.

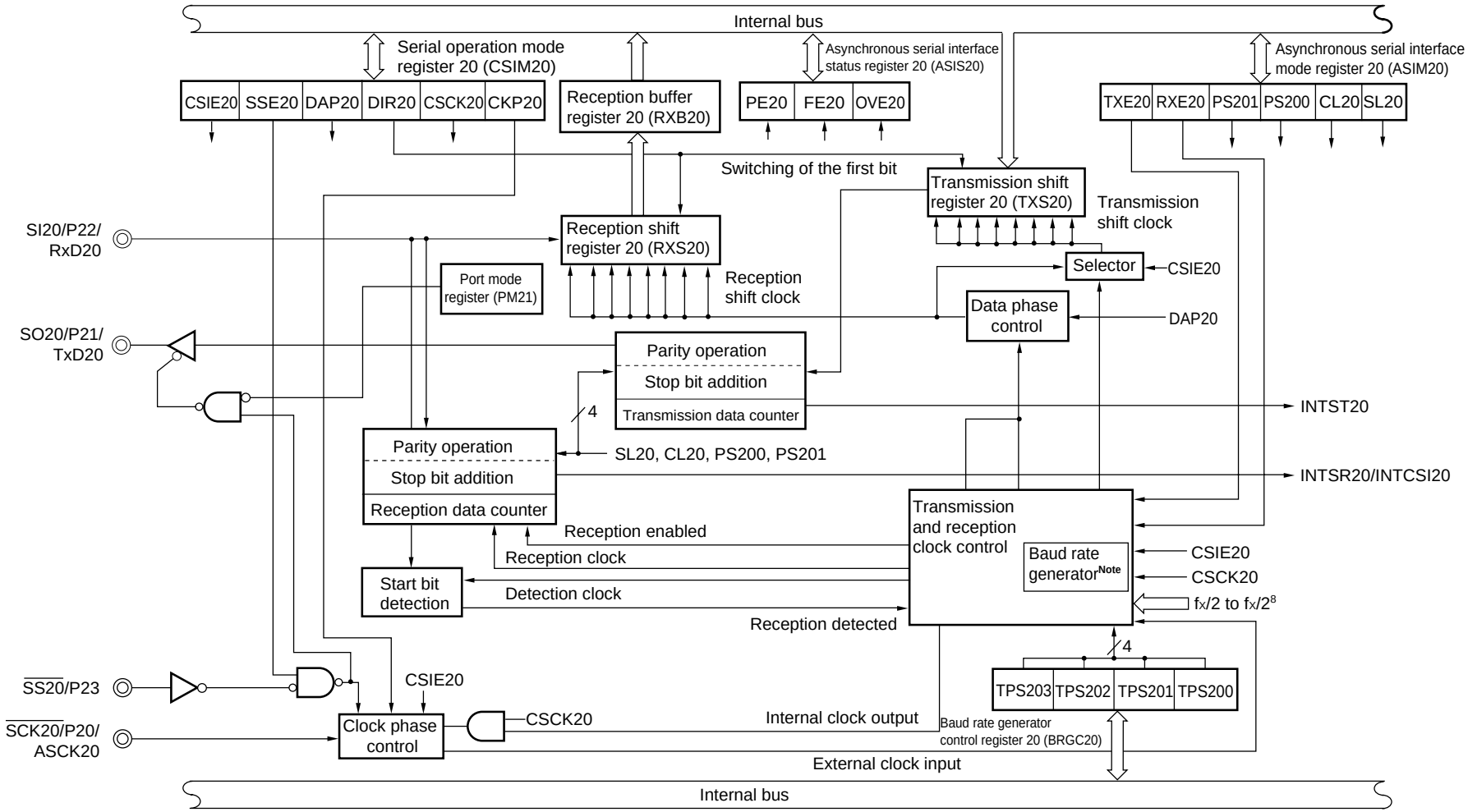
5.6.2 Serial interface 20 configuration

Serial interface 20 consists of the following hardware.

Table 5-10. Configuration of Serial Interface 20

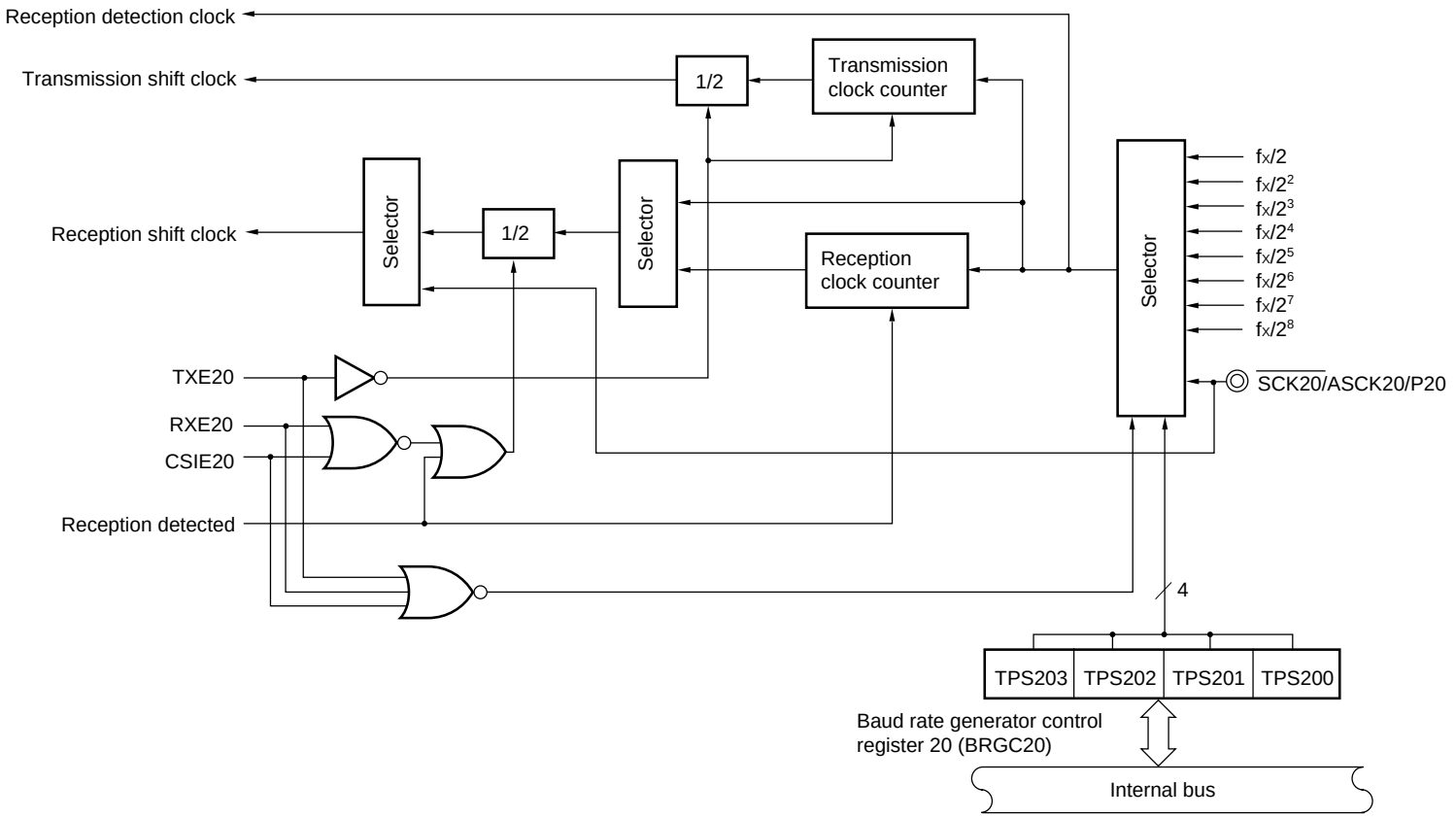
Item	Configuration
Register	Transmit shift register 20 (TXS20) Receive shift register 20 (RXS20) Receive buffer register 20 (RXB20)
Control register	Serial operating mode register 20 (CSIM20) Asynchronous serial interface mode register 20 (ASIM20) Asynchronous serial interface status register 20 (ASIS20)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

Figure 5-18. Block Diagram of Serial Interface 20



Note See **Figure 5-19** for the configuration of the baud rate generator.

Figure 5-19. Block Diagram of Baud Rate Generator 20



(1) Transmit 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 written with an 8-bit memory manipulation instruction, but cannot be read.

$\overline{\text{RESET}}$ input sets TXS20 to FFH.

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

(2) Receive 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 transfers the reception data to receive buffer register 20 (RXB20).

RXS20 cannot be manipulated directly by a program.

(3) Receive buffer register 20 (RXB20)

RXB20 holds receive data. New receive data is transferred from receive shift register 0 (RXS20) per 1 byte of data received.

When the data length is specified as seven bits, the receive data is sent to bits 0 to 6 of RXB20, in which the MSB is always fixed to 0.

RXB20 can be read with an 8-bit memory manipulation instruction, but cannot be written to.

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

Caution RXB20 and transmit shift register 20 (TXS20) are mapped at the same address, so 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 transmit shift register 20 (TXS20), according to the setting of asynchronous serial interface mode register 20 (ASIM20).

(5) Reception control circuit

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

5.6.3 Control registers for serial interface 20

Serial interface 20 is controlled by the following four registers.

- 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 operating mode register 20 (CSIM20)

CSIM20 is used to make the settings related to 3-wire serial I/O mode.

CSIM20 is set with a 1-bit or 8-bit memory manipulation instruction.

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

Figure 5-20. Format of Serial Operating Mode Register 20

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

CSIE20	Control of Operation during 3-wire Serial I/O Mode
0	Operation disabled
1	Operation enabled

SSE20	$\overline{\text{SS20}}$-pin selection	Function 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	3-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	3-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	3-wire serial I/O mode clock phase selection
0	Clock is low active, and $\overline{\text{SCK20}}$ is at high level in the idle state
1	Clock is high active, and $\overline{\text{SCK20}}$ is at low level 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 asynchronous serial interface mode.

ASIM20 is set with a 1-bit or 8-bit memory manipulation instruction.

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

Figure 5-21. Format of Asynchronous Serial Interface Mode Register 20

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

TXE20	Control of Transmit Operation
0	Transmit operation stopped
1	Transmit operation enabled

RXE20	Control of Receive Operation
0	Receive operation stopped
1	Receive operation enabled

PS201	PS200	Specification of Parity Bit
0	0	No parity
0	1	During transmission, always add zero parity. During reception, parity is not detected (parity errors do not occur).
1	0	Odd parity
1	1	Even parity

CL20	Specification of Transmit Data Character Length
0	7 bits
1	8 bits

SL20	Specification of Transmit Data Stop Bit(s)
0	1 bit
1	2 bits

Cautions 1. Be sure to set 0 to bits 0 and 1.

2. If 3-wire serial I/O mode has been selected, set 00H to the ASIM20.

3. Set the operation mode after the serial transmission or reception operation has been stopped.

Table 5-11. List of Operation Mode Settings for Serial Interface 20

(1) Operation stop mode

ASIM20		CSIM20			PM22	P22	PM21	P21	PM20	P20	First Bit	Shift Clock	P22/SI20/ RxD20 Pin Function	P21/SO20/ TxD20 Pin Function	P20/$\overline{SCK20}$/ 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 than above											Setting prohibited				

(2) 3-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/$\overline{SCK20}$/ ASCK20 Pin Function
TXE20	RXE20	CSIE20	DIR20	CSCK20											
0	0	1	0	0	×Note 2	×Note 2	0	1	1	×	MSB	External clock	SI20Note 2	SO20 (CMOS output)	$\overline{SCK20}$ input
				1					0	1		Internal clock			$\overline{SCK20}$ output
		1	1	0					1	×	LSB	External clock			$\overline{SCK20}$ input
				1					0	1		Internal clock			$\overline{SCK20}$ output
Other than above											Setting prohibited				

(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/$\overline{SCK20}$/ 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 than above											Setting prohibited				

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

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

Remark ×: Don't care.

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

When a receive error occurs while in asynchronous serial interface mode, this register indicates the type of error.

The ASIS20 is read via a 1-bit memory manipulation instruction or an 8-bit memory manipulation instruction.

The contents of the ASIS20 are undefined during 3-wire serial I/O mode.

After the $\overline{\text{RESET}}$ signal is input, this register value is 00H.

Figure 5-22. Format of Asynchronous Serial Interface Status Register 20

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

PE20	Parity Error Flag
0	No parity error
1	Parity error has occurred (due to mismatch with parity of transmit data)

FE20	Framing Error Flag
0	No framing error
1	Framing error has occurred (no stop bit detected)[Note 1]

OVE20	Overrun Error Flag
0	No overrun error
1	Overrun error has occurred[Note 2]. (Before data was read from the reception buffer register, the subsequent reception sequence was completed.)

Notes 1. Even if a stop bit length of two bits has been set to bit 2 (SL20) of asynchronous serial interface mode register 20 (ASIM20), only one stop bit is detected during reception.

2. Whenever an overrun error has occurred, be sure to read the contents of the receive buffer register 20 (RXB20). Until the RXB20 has been read, the overrun error will continue to occur each time data is received.

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

This register is used to set the serial clock for serial interface 20.

The BRGC20 is set via an 8-bit memory manipulation instruction.

After the $\overline{\text{RESET}}$ signal is input, this register value is 00H.

Figure 5-23. Format of Baud Rate Generator Control Register 20

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

TPS203	TPS202	TPS201	TPS200	Selection of Source Clock for 3-bit Counter	n
0	0	0	0	$f_X/2$ (2.5 MHz)	1
0	0	0	1	$f_X/2^2$ (1.25 MHz)	2
0	0	1	0	$f_X/2^3$ (625 kHz)	3
0	0	1	1	$f_X/2^4$ (313 kHz)	4
0	1	0	0	$f_X/2^5$ (156 kHz)	5
0	1	0	1	$f_X/2^6$ (78.1 kHz)	6
0	1	1	0	$f_X/2^7$ (39.1 kHz)	7
0	1	1	1	$f_X/2^8$ (19.5 kHz)	8
1	0	0	0	Inputs a clock from an external source to the ASCK20 pin[Note]	–
Other than above				Setting prohibited	

Note This can be used only during UART mode.

Cautions 1. When data is written to BRGC20 during a communication operation, the output from the baud rate generator is undefined and it becomes impossible to perform normal communication operations. Therefore, never write to the BRGC20 during a communication operation.

2. When f_X = 5.0 MHz, do not specify n = 1 since it may exceed the maximum rated value of baud rate.

3. When an input clock from an external source has been selected, set port mode register 2 (PM2) to input mode.

Remarks 1. f_X: System clock oscillation frequency

2. n: Value determined by the TPS200 to TPS203 setting ($1 \leq n \leq 8$).

3. Values in parentheses are when f_X = 5.0 MHz.

The transmit/receive clock for the generated baud rate is either a signal that is divided from the system clock or a signal that is divided from the clock that is input via the ASCK20 pin.

(a) Generation of transmit/receive clock for baud rate based on system clock

A transmit/receive clock is generated when it is divided from the system clock. The baud rate that is generated from the system clock can be determined via the following formula.

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

f_X: System clock oscillation frequency

n: Value from Figure 5-23, determined by the TPS200 to TPS203 setting ($2 \leq n \leq 8$).

Table 5-12. Relation between System Clock and Baud Rate

Baud Rate (bps)	n	Setting in BRGC20	Error (%)	
			f_X = 5.0 MHz	f_X = 4.9152 MHz
1200	8	70H	1.73	0
2400	7	60H		
4800	6	50H		
9600	5	40H		
19200	4	30H		
38400	3	20H		
76800	2	10H		

Caution When f_X = 5.0 MHz, do not specify n = 1 since it may exceed the maximum rated value of baud rate.

(b) Generation of transmit/receive clock for baud rate based on external clock from ASCK20 pin

A transmit/receive clock is generated when it is divided from the clock that is input via the ASCK20 pin. The baud rate that is generated from the clock input via the ASCK20 pin can be determined via the following formula.

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

f_{ASCK}: Frequency of clock input to ASCK20 pin

Table 5-13. Relation between ASCK20 Pin Input Frequency and Baud Rate (BRGC20 = 80H)

Baud Rate (bps)	ASCK20 Pin Input Frequency (kHz)
75	1.2
150	2.4
300	4.8
600	9.6
1200	19.2
2400	38.4
4800	76.8
9600	153.6
19200	307.2
31250	500.0
38400	614.4

6. INTERRUPT FUNCTIONS

6.1 Types of Interrupt Functions

There are two types of interrupt functions.

(1) Non-maskable interrupt

This type of interrupt is received unconditionally, even during interrupt disable mode. Moreover, this type of interrupt is not subject to interrupt prioritization control, it automatically takes priority over all other interrupt requests.

A standby release signal is issued for this interrupt.

Interrupts from the watchdog timer provide one source for non-maskable interrupts.

(2) Maskable interrupt

This type of interrupt is subject to mask control. When several interrupt requests have occurred at the same time, the prioritization of interrupts is determined as shown in Table 6-1.

A standby release signal is issued for this interrupt.

External interrupts provide three sources and internal interrupts provide four sources for maskable interrupts.

6.2 Interrupt Sources and Configuration

There are a total of eight sources for non-maskable and maskable interrupts combined (see Table 6-1).

Table 6-1. Interrupt Source List

Interrupt Type	Priority[Note 1]	Interrupt Source		Internal/ External	Vector Table Address	Basic Configuration Type[Note 2]
		Name	Trigger			
Non-maskable	–	INTWDT	Watchdog timer overflow (with watchdog timer mode 1 selected)	Internal	0004H	(A)
Maskable	0	INTWDT	Watchdog timer overflow (with interval timer mode selected)			(B)
	1	INTP0	Pin input edge detection	External	0006H	(C)
	2	INTP1			0008H	
	3	INTP2			000AH	
	4	INTSR20	End of serial interface 20 UART reception	Internal	000CH	(B)
		INTCSI20	End of serial interface 20 3-wire SIO transfer reception			
	5	INTST20	End of serial interface 20 UART transmission		000EH	
	6	INTTM80	Generation of match signal of 8-bit timer 80		0014H	
	7	INTTM90	Generation of match signal of 16-bit timer/event counter 90		0016H	

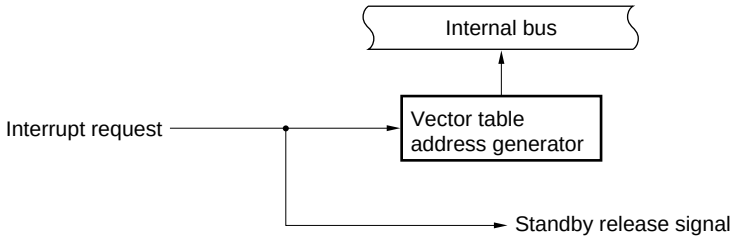
Notes 1. Priority is the priority order when several maskable interrupts are generated at the same time. 0 is the highest order and 7 is the lowest order.

2. Basic configuration types (A) to (C) correspond to (A) to (C) in Figure 6-1.

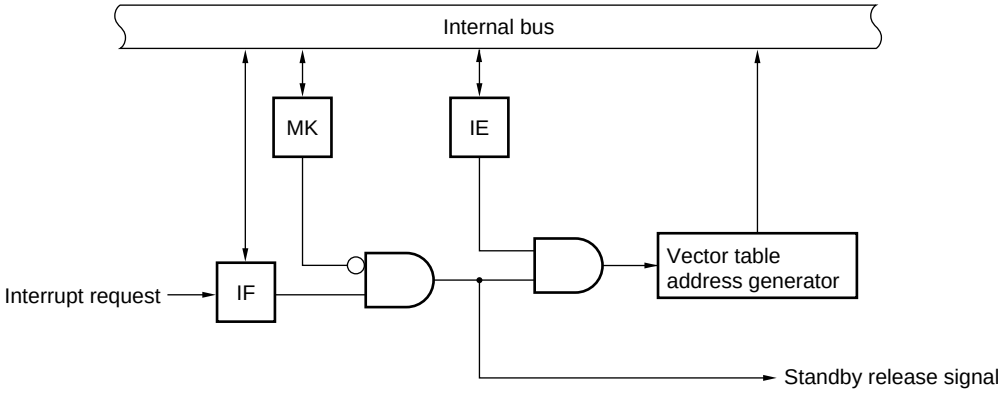
Remark As the interrupt source of the watchdog timer (INTWDT), either a non-maskable interrupt or a maskable interrupt (internal) can be selected.

Figure 6-1. Basic Configuration of Interrupt Function

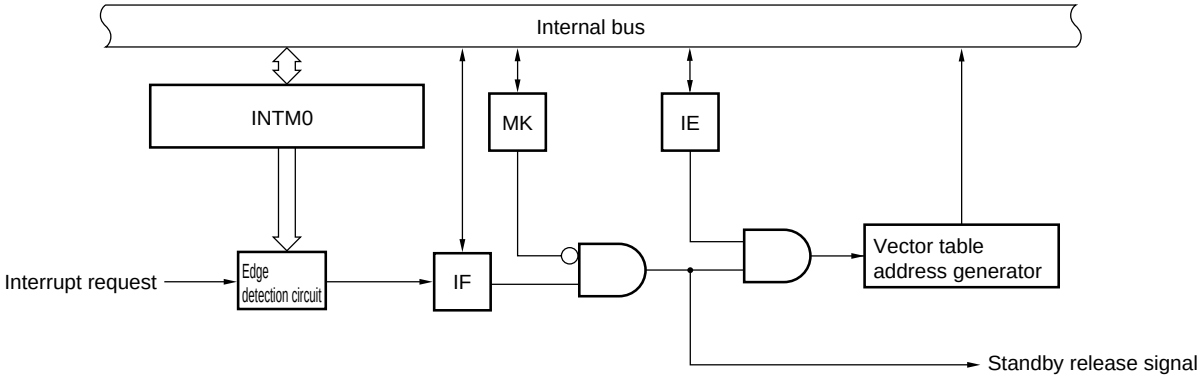
(A) Internal non-maskable interrupt



(B) Internal maskable interrupt



(C) External maskable interrupt



INTM0: External interrupt mode register 0
IF: Interrupt request flag
IE: Interrupt enable flag
MK: Interrupt mask flag

6.3 Control Registers for Interrupt Functions

Interrupt functions are controlled by the following five types of registers.

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

The names of the interrupt request flags and interrupt mask flags for each type of interrupt request are listed in Table 6-2.

Table 6-2. Flags Corresponding to Interrupt Request Signal Names

Interrupt Request Signal Name	Interrupt Request Flag	Interrupt Mask Flag
INTWDT	WDTIF	WDTMK
INTP0	PIF0	PMK0
INTP1	PIF1	PMK1
INTP2	PIF2	PMK2
INTSR20/INTCSI20	SRIF20	SRMK20
INTST20	STIF20	STMK20
INTTM80	TMIF80	TMMK80
INTTM90	TMIF90	TMMK90

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

An interrupt request flag is set (to 1) when a corresponding interrupt request occurs or when an instruction is executed. It is cleared (to 0) when the interrupt request has been received, when $\overline{\text{RESET}}$ signal is input, or when an instruction is executed.

The IF0 and IF1 are set via a 1-bit memory manipulation instruction or an 8-bit memory manipulation instruction.

After the $\overline{\text{RESET}}$ signal is input, the register value is 00H.

Figure 6-2. Format of Interrupt Request Flag Registers

Symbol	7	6	<5>	<4>	<3>	<2>	<1>	<0>	Address	After reset	R/W
IF0	0	0	STIF20	SRIF20	PIF2	PIF1	PIF0	WDTIF	FFE0H	00H	R/W
	7	6	5	4	3	2	<1>	<0>			
IF1	0	0	0	0	0	0	TMIF90	TMIF80	FFE1H	00H	R/W

××IF×	Interrupt Request Flag
0	Interrupt request signal has not occurred
1	Interrupt request signal has occurred, interrupt request status

Cautions 1. Be sure to set 0 to bits 6 and 7 in IF0 and bits 2 to 7 in IF7.

2. The WDTIF flag is R/W-accessible only when the watchdog timer is used as an interval timer. When using watchdog timer mode 1 or 2, set 0 to the WDTIF flag.

3. Since port 2 is also used as an external interrupt input, when the port function output mode has been specified and the output level has been changed, the interrupt request flag becomes set. Therefore, before using output mode, be sure to set 1 to the interrupt mask flag.

(2) Interrupt mask flag registers (MK0, MK1)

The interrupt mask flag is used to enable/disable the corresponding maskable interrupt service.

The MK0 and MK1 are set via a 1-bit memory manipulation instruction or an 8-bit memory manipulation instruction.

After the $\overline{\text{RESET}}$ signal is input, the register value is FFH.

Figure 6-3. Format of Interrupt Mask Flag Registers

Symbol	7	6	<5>	<4>	<3>	<2>	<1>	<0>	Address	After reset	R/W
MK0	1	1	STMK20	SRMK20	PMK2	PMK1	PMK0	WDTMK	FFE4H	FFH	R/W
	7	6	5	4	3	2	<1>	<0>			
MK1	1	1	1	1	1	1	TMMK90	TMMK80	FFE5H	FFH	R/W

××MK	Interrupt Servicing Control
0	Enable interrupt servicing
1	Disable interrupt servicing

Cautions 1. Be sure to set 1 to bits 6 and 7 in MK0 and bits 2 to 7 in MK1.

2. If the WDTMK flag is read when the watchdog timer is used in watchdog timer mode 1 or 2, its value becomes undefined.

3. Since port 2 is also used as an external interrupt input, when the port function output mode has been specified and the output level has been changed, the interrupt request flag becomes set. Therefore, before using output mode, be sure to set 1 to the interrupt mask flag.

(3) External interrupt mode register 0 (INTM0)

This register is used to set the valid edge of INTP0 to INTP2.

The INTM0 is set via an 8-bit memory manipulation instruction.

After the $\overline{\text{RESET}}$ signal is input, the register value is 00H.

Figure 6-4. Format of External Interrupt Mode Register 0

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

ES21	ES20	Selection of INTP2 Valid Edge
0	0	Falling edge
0	1	Rising edge
1	0	Setting prohibited
1	1	Both rising and falling edges

ES11	ES10	Selection of INTP1 Valid Edge
0	0	Falling edge
0	1	Rising edge
1	0	Setting prohibited
1	1	Both rising and falling edges

ES01	ES00	Selection of INTP0 Valid Edge
0	0	Falling edge
0	1	Rising edge
1	0	Setting prohibited
1	1	Both rising and falling edges

Cautions 1. Be sure to set 0 to bits 0 and 1.

2. Before setting the INTM0 register, be sure to set 1 to the corresponding interrupt mask flag to disable interrupts.

Afterward, clear (to 0) the interrupt request flag, then set 0 to the interrupt mask flag to enable interrupts.

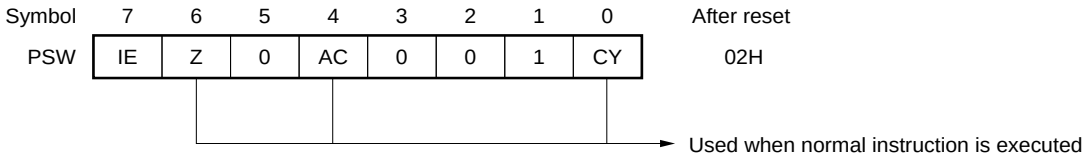
(4) Program status word (PSW)

The program status word is a register that is used to hold instruction execution results and the current status for interrupt requests. The IE flag, used to set maskable interrupt enable/disable status, is mapped.

In addition to read/write operations in 8-bit units, this register can carry out operations with a bit manipulation instruction and dedicated instructions (EI, DI). When a vectored interrupt request is acknowledged, PSW is automatically saved to a stack, and 0 is set to the IE flag.

After the $\overline{\text{RESET}}$ signal is input, the PSW value is 02H.

Figure 6-5. Configuration of Program Status Word



IE	Enable/Disable Interrupt Reception
0	Disable
1	Enable

7. STANDBY FUNCTION

7.1 Standby Function

The standby function is used to reduce the system power consumption via either the HALT mode or STOP mode. The HALT instruction is used to set HALT mode and the STOP instruction is used to set STOP mode.

(1) HALT mode

This mode stops the CPU operating clock. The average power consumption can be reduced by intermittent operation by combining this mode with the normal operation mode.

(2) STOP mode

This mode stops the oscillation of the main system clock. All operations that depend on the main system clock are stopped to minimize power consumption.

Caution Before switching to STOP mode, be sure to stop the operation of peripheral hardware, then execute the STOP instruction.

Table 7-1. Operation Status during HALT Mode

Item	HALT Mode Operating Status
Clock generator	System clock can be oscillated. Clock supply to CPU stops.
CPU	Operation stopped
Port (output latch)	Status prior to setting HALT mode is retained
16-bit timer 90	Operation enabled
8-bit timer/event counter 80	Operation enabled
Watchdog timer	Operation enabled
Serial interface 20	Operation enabled
External interrupt	Operation enabled[Note]

Note Non-masked maskable interrupt

Table 7-2. Operation Status during STOP Mode

Item	STOP Mode Operating Status
Clock generator	System clock oscillation stopped.
CPU	Operation stopped
Port (output latch)	Status prior to setting STOP mode is retained
16-bit timer 90	Operation stopped
8-bit timer/event counter 80	Operation is possible only when TI80 is selected as the count clock.
Watchdog timer	Operation stopped
Serial interface 20	Operation is possible in both 3-wire serial I/O and UART modes while an external clock is being used.
External interrupt	Operation enabled[Note]

Note Non-masked maskable interrupt

7.2 Control Register for Standby Function

The wait time from when STOP mode is canceled by an interrupt request until oscillation has become stabilized is controlled by the oscillation stabilization time select register (OSTS).

The OSTS is set via an 8-bit memory manipulation instruction.

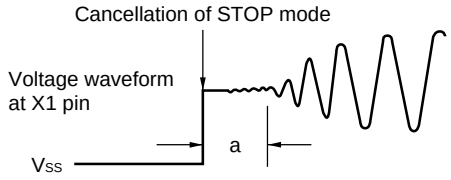
After the $\overline{\text{RESET}}$ signal is input, the register value is 04H. However, the oscillation stabilization time following input of the $\overline{\text{RESET}}$ signal is $2^{15}/f_X$, not $2^{17}/f_X$.

Figure 7-1. Format of Oscillation Stabilization Time Select Register

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

OSTS2	OSTS1	OSTS0	Selection of Oscillation Stabilization Time
0	0	0	$2^{12}/f_X$ (819 μs)
0	1	0	$2^{15}/f_X$ (6.55 ms)
1	0	0	$2^{17}/f_X$ (26.2 ms)
Other than above			Setting prohibited

Caution The wait time following cancellation of STOP mode, input of the $\overline{\text{RESET}}$ signal, or occurrence of an interrupt does not include the time between cancellation of STOP mode and the start of clock oscillation (see "a" in the following figure).



Remarks 1. f_X: System clock oscillation frequency

2. Values in parentheses are during f_X = 5.0-MHz operation.

8. RESET FUNCTIONS

There are two ways to issue a reset signal.

(1) External reset input by $\overline{\text{RESET}}$ signal input
(2) Internal reset by watchdog timer runaway time detection

There are no functional differences between external and internal resets, since in both cases program execution starts at the address written to 0000H and 0001H after the $\overline{\text{RESET}}$ signal is input.

When a low level is input to the $\overline{\text{RESET}}$ pin or when the watchdog timer overflows, a reset is triggered and all hardware is set to the status shown in Table 8-1. Each pin has high impedance during reset input or during oscillation stabilization time immediately after a reset is canceled.

When a high level is input to the $\overline{\text{RESET}}$ pin, the reset is canceled and program execution is started after the oscillation stabilization time ($2^{15}/f_X$) has elapsed. The reset triggered by a watchdog timer overflow is automatically canceled after a reset, and program execution is started after the oscillation stabilization time has elapsed.

Cautions 1. For an external reset, input a low level for at least 10 μs to the $\overline{\text{RESET}}$ pin.
2. When the STOP mode is canceled by reset, the STOP mode contents are retained during reset input. However, the port pins are set to high impedance.

Figure 8-1. Block Diagram of Reset Function

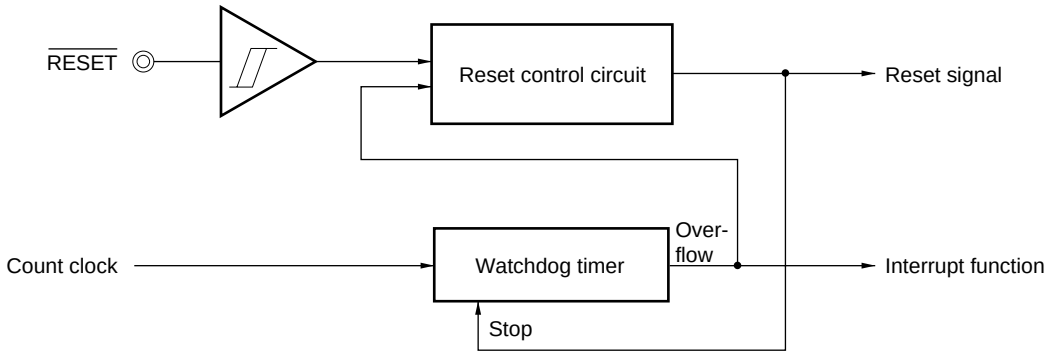


Table 8-1. Status of Each Hardware after Reset (1/2)

Hardware		Status after Reset
Program counter (PC)[Note 1]		Contents of reset vector table (0000H, 0001H) are set
Stack pointer (SP)		Undefined
Program status word (PSW)		02H
RAM	Data memory	Undefined[Note 2]
	General-purpose registers	Undefined[Note 2]
Ports (P0 to P3) (output latch)		00H
Port mode registers (PM0 to PM3)		FFH
Pull-up resistor option registers (PU0, PUB2)		00H
Processor clock control register (PCC)		02H
Oscillation stabilization time select register (OSTS)		04H
16-bit timer	Timer counter (TM90)	0000H
	Compare register (CR90)	FFFFH
	Control register (TMC90)	00H
	Capture register (TCP90)	Undefined
8-bit timer/event counter	Timer counter (TM80)	00H
	Compare register (CR80)	Undefined
	Mode control register (TMC80)	00H
Watchdog timer	Clock select register (WDCS)	00H
	Mode register (WDTM)	00H
Serial interface	Serial operation 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
	Transmit shift register (TXS20)	FFH
	Receive buffer register (RXB20)	Undefined
Interrupt	Request flag registers (IF0, IF1)	00H
	Mask flag registers (MK0, MK1)	FFH
	External interrupt mode register (INTM0)	00H

Notes 1. The status of all hardware during a reset input operation or during the wait time for stabilization of oscillation is undefined in the PC only. Otherwise, the status after the reset is the same as before the reset.

2. The status after a reset for standby mode is retained.

9. FLASH MEMORY PROGRAMMING

The on-chip program memory in the μPD78F9076 is flash memory.

The flash memory can be written with the μPD78F9076 mounted on the target system (on-board). Connect the dedicated flash programmer (Flashpro III (part number: FL-PR3, PG-FP3)) to the host machine and target system to write the flash memory.

Remark FL-PR3 is made by Naito Densei Machida Mfg. Co., Ltd.

9.1 Selecting Communication Mode

The flash memory is written by using Flashpro III and by means of serial communication. Select a communication mode from those listed in Table 9-1. To select a communication mode, the format shown in Figure 9-1 is used. Each communication mode is selected by the number of V_{PP} pulses shown in Table 9-1.

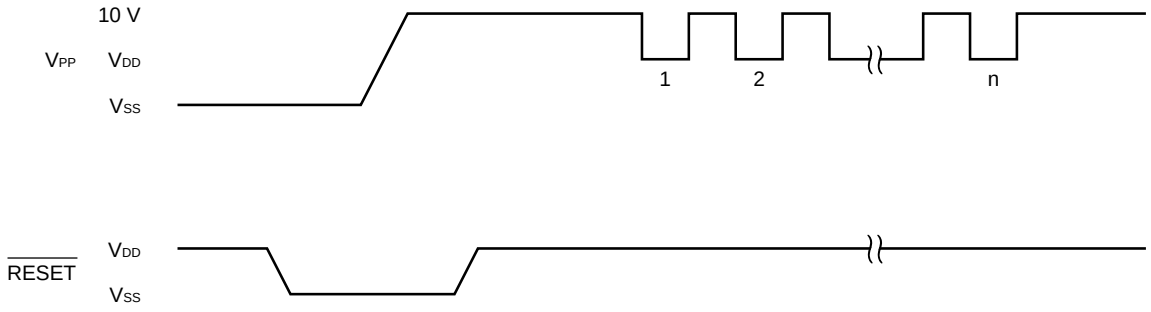
Table 9-1. Communication Mode

Communication Mode	Pins Used	Number of V_{PP} Pulses
3-wire serial I/O	$\overline{SCK20}$/ASCK20/P20 SO20/TxD20/P21 SI20/RxD20/P22	0
UART	TxD20/SO20/P21 RxD20/SI20/P22	8
Pseudo 3-wire mode [Note]	P00 (Serial clock input) 12 P01 (Serial data output) P02 (Serial data input)	12

Note Serial transfer is performed by controlling a port by software.

Caution Be sure to select a communication mode based on the V_{PP} pulse number shown in Table 9-1.

Figure 9-1. Communication Mode Selection Format



9.2 Function of Flash Memory Programming

By transmitting/receiving commands and data in the selected communication mode, operations such as writing to the flash memory are performed. Table 9-2 shows the major functions of flash memory programming.

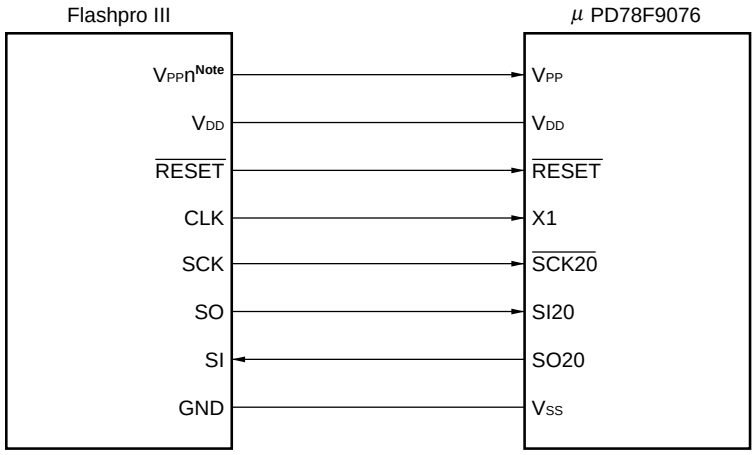
Table 9-2. Functions of Flash Memory Programming

Function	Description
Batch erase	Erases all contents of memory
Batch blank check	Checks erased state of entire memory
Data write	Write to flash memory based on write start address and number of data written (number of bytes)
Batch verify	Compares all contents of memory with input data

9.3 Flashpro III Connection Example

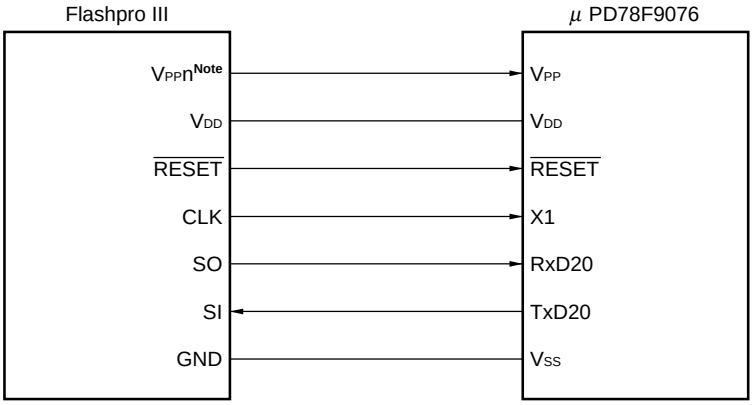
How the Flashpro III is connected to the μPD78F9076 differs depending on the communication mode (3-wired serial I/O, UART, or pseudo 3-wire mode). Figures 9-2 to 9-4 show the connection in the respective mode.

Figure 9-2. Flashpro III Connection in 3-wired Serial I/O Mode



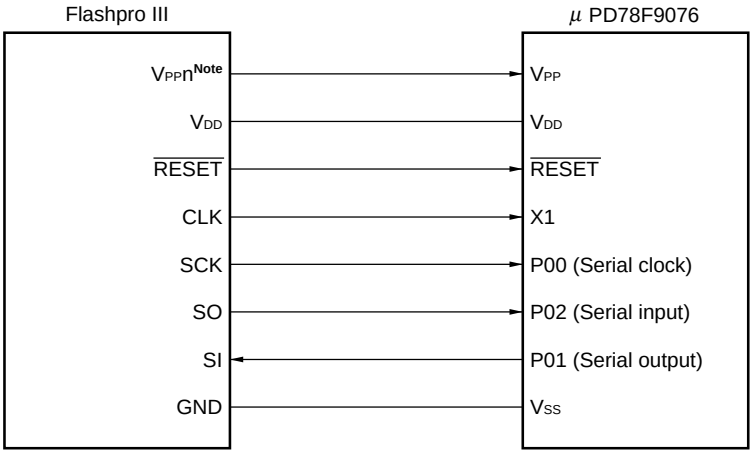
Note n = 1, 2

Figure 9-3. Flashpro III Connection in UART Mode



Note n= 1, 2

Figure 9-4. Flashpro III Connection in Pseudo 3-Wire Mode (When Port 0 is Used)



Note n= 1, 2

9.4 Example of Settings for Flashpro III (PG-FP3)

Set as follows when writing to flash memory using the Flashpro III (PG-FP3).

<1> Download the parameter file.

<2> Select the serial mode and the serial clock using the type command.

<3> The following is a setting example using the PG-FP3.

Table 9-3. Example Using PG-FP3

Communication mode	Setting example using PG-FP3		Number of V_{PP} pulses[Note1]
3-wired serial I/O mode	COMM PORT	SIO ch-0	0
	CPU CLK	On target board	
		In Flashpro	
	On target board	4.1943 MHz	
	SIO CLK	1.0 MHz	
	In Flashpro	4.0 MHz	
	SIO CLK	1.0 MHz	
UART	COMM PORT	UART-ch0	8
	CPU CLK	On target board	
	On target board	4.1943 MHz	
	UART BPS	9600 bps[Note2]	
Pseudo 3-wire mode	COMM PORT	Port A	12
	CPU CLK	On target board	
		In Flashpro	
	On target board	4.1943 MHz	
	SIO CLK	1 kHz	
	In Flashpro	4.0 MHz	
	SIO CLK	1 kHz	

Notes 1. The number of V_{PP} pulses supplied from the Flashpro III during serial communication initialization. The pins to be used in communication are determined by this number of pulses.

2. Select one of 9600 bps, 19200 bps, 38400 bps, or 76800 bps.

Remark COMM PORT: Selection of serial port
SIO CLK: Selection of serial clock frequency
CPU CLK: Selection of CPU clock source to be input

10. INSTRUCTION SET OVERVIEW

This section lists the μPD78F9076 instruction set.

10.1 Conventions

10.1.1 Operand identifiers and description methods

Operands are described in the "Operand" column of each instruction in accordance with the description method of the instruction operand identifier (refer to the assembler specifications for detail). When there are two or more description methods, select one of them. Alphabetic letters in capitals and the symbols, #, !, $, and [ ], are keywords and must be described as they are. Each symbol has the following meaning.

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

In the case of immediate data, describe an appropriate numeric value or a label. When using a label, be sure to describe the #,!, $, or [ ] symbols.

For operand register identifiers, r and rp, either function names (X, A, C, etc.) or absolute names (names in parentheses in the table below, R0, R1, R2, etc.) can be used for description.

Table 10-1. Operand Identifiers and Description Methods

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

Remark See **Table 4-1 Special Function Register List** for symbols of special function registers.

10.1.2 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:	Non-maskable interrupt servicing flag
( ):	Memory contents indicated by address or register contents in parentheses
X_H, X_L:	Higher 8 bits and lower 8 bits of 16-bit register
∧:	Logical product (AND)
∨:	Logical sum (OR)
∨:	Exclusive OR
——:	Inverted data
addr16:	16-bit immediate data or label
jdisp8:	Signed 8-bit data (displacement value)

10.1.3 Description of the flag operation field

(Blank):	Not affected
0:	Cleared to 0
1:	Set to 1
×:	Set/cleared according to the result
R:	Previously saved value is restored

10.2 Operations

Mnemonic	Operand	Bytes	Clock	Operation	Flags		
					Z	AC	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			

Notes 1. Except r = A

2. Except r = A, X

3. Only when rp = BC, DE, HL

Remark One clock of an instruction is one clock of the CPU clock (f_{CPU}) selected using the processor clock control register (PCC).

Mnemonic	Operand	Bytes	Clock	Operation	Flags		
					Z	AC	CY
XCHW	AX, rp[Note]	1	8	AX ←→ rp			
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	×	×	×

Note Only when rp = BC, DE, HL

Remark One clock of an instruction is one clock of the CPU clock (f_{CPU}) selected using the processor clock control register (PCC).

Mnemonic	Operand	Bytes	Clock	Operation	Flags		
					Z	AC	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)	×		
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	×	×	

Remark One clock of an instruction is one clock of the CPU clock (f_{CPU}) selected using the processor clock control register (PCC).

Mnemonic	Operand	Bytes	Clock	Operation	Flags		
					Z	AC	CY
INCW	rp	1	4	rp ← rp + 1			
DECW	rp	1	4	rp ← rp – 1			
ROR	A, 1	1	2	(CY, $A_7 \leftarrow A_0$, $A_{m-1} \leftarrow A_m$) × 1			×
ROL	A, 1	1	2	(CY, $A_0 \leftarrow A_7$, $A_{m+1} \leftarrow A_m$) × 1			×
RORC	A, 1	1	2	(CY ← A_0, A_7 ← CY, $A_{m-1} \leftarrow A_m$) × 1			×
ROLC	A, 1	1	2	(CY ← A_7, A_0 ← CY, $A_{m+1} \leftarrow A_m$) × 1			×
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			

Remark One clock of an instruction is one clock of the CPU clock (f_{CPU}) selected using the processor clock control register (PCC).

Mnemonic	Operand	Bytes	Clock	Operation	Flags
					Z AC CY
BR	!addr16	3	6	PC ← addr16	
	$addr16	2	6	PC ← PC + 2 + jdisp8	
	AX	1	6	PC_H ← A, PC_L ← X	
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, $saddr16	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, $saddr16	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 One clock of an instruction is one clock of the CPU clock (f_{CPU}) selected using the processor clock control register (PCC).

11. ELECTRICAL SPECIFICATIONS

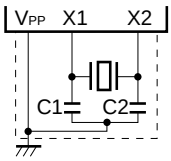
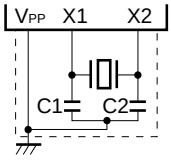
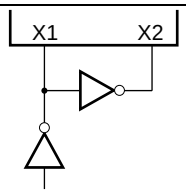
Absolute Maximum Ratings (T_A = 25°C)

Parameter	Symbol	Conditions	Ratings	Unit
Supply voltage	V_{DD}		–0.3 to +6.5	V
	V_{PP}		–0.3 to +10.5	V
Input voltage	V_I		–0.3 to V_{DD} + 0.3	V
Output voltage	V_O		–0.3 to V_{DD} + 0.3	V
Output current, high	I_{OH}	Per pin	–10	mA
		Total for all pins	–30	mA
Output current, low	I_{OL}	Per pin	30	mA
		Total for all pins	160	mA
Operating ambient temperature	T_A	In normal operation mode	–40 to +85	°C
		During flash memory programming	10 to 40	°C
Storage temperature	T_{stg}		–40 to +125	°C

Caution Product quality may suffer if the absolute maximum rating is exceeded even momentarily for any parameter. That is, the absolute maximum ratings are rated values at which the product is on the verge of suffering physical damage, and therefore the product must be used under conditions that ensure that the absolute maximum ratings are not exceeded.

Remark Unless specified otherwise, the characteristics of alternate-function pins are the same as those of port pins.

System Clock Oscillator Characteristics (T_A = –40 to +85°C, V_{DD} = 1.8 to 5.5 V)

Resonator	Recommended Circuit	Parameter	Conditions	MIN.	TYP.	MAX.	Unit
Ceramic resonator		Oscillation frequency (f_X)Note 1	V_{DD} = oscillation voltage range	1.0		5.0	MHz
		Oscillation stabilization timeNote 2	After V_{DD} reaches oscillation voltage range MIN.			4	ms
Crystal resonator		Oscillation frequency (f_X)Note 1		1.0		5.0	MHz
		Oscillation stabilization timeNote 2	V_{DD} = 4.5 to 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. Indicates only oscillator characteristics. Refer to **AC characteristics** for instruction execution time.

2. Time required to stabilize oscillation after a reset or STOP mode release. Use the resonator that stabilizes oscillation during the oscillation wait time.

Caution When using the system clock oscillator, wire as follows in the area enclosed by the broken lines in the above figures to avoid an adverse effect from wiring capacitance.

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

Remark For the resonator selection and oscillator constant, customers are requested to either evaluate the oscillation themselves or apply to the resonator manufacturer for evaluation.

DC Characteristics (T_A = –40 to +85°C, V_{DD} = 1.8 to 5.5 V) (1/2)

Parameter	Symbol	Conditions		MIN.	TYP.	MAX.	Unit
Output current, low	I_{OL}	Per pin				10	mA
		Total for all pins				80	mA
Output current, high	I_{OH}	Per pin				–1	mA
		Total for all pins				–15	mA
Input voltage, high	V_{IH1}	P00 to P07, P10 to P15, P30, P31	V_{DD} = 2.7 to 5.5 V	0.7 V_{DD}		V_{DD}	V
				0.9 V_{DD}		V_{DD}	V
	V_{IH2}	$\overline{\text{RESET}}$, P20 to P27	V_{DD} = 2.7 to 5.5 V	0.8 V_{DD}		V_{DD}	V
				0.9 V_{DD}		V_{DD}	V
	V_{IH3}	X1, X2	V_{DD} = 4.5 to 5.5 V	V_{DD}–0.5		V_{DD}	V
				V_{DD}–0.1		V_{DD}	V
Input voltage, low	V_{IL1}	P00 to P07, P10 to P15, P30, P31	V_{DD} = 2.7 to 5.5 V	0		0.3 V_{DD}	V
				0		0.1 V_{DD}	V
	V_{IL2}	$\overline{\text{RESET}}$, P20 to P27	V_{DD} = 2.7 to 5.5 V	0		0.2 V_{DD}	V
				0		0.1 V_{DD}	V
	V_{IL3}	X1, X2	V_{DD} = 4.5 to 5.5 V	0		0.4	V
				0		0.1	V
Output voltage, high	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
Output voltage, low	V_{OL}	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
Input leakage current, high	I_{LIH1}	V_{IN} = V_{DD}	P00 to P07, P10 to P15, P20 to P27, P30, P31, $\overline{\text{RESET}}$			3	μA
	I_{LIH2}		X1, X2			20	μA
Input leakage current, low	I_{LIL1}	V_{IN} = 0 V	P00 to P07, P10 to P15, P20 to P27, P30, P31, $\overline{\text{RESET}}$			–3	μA
	I_{LIL2}		X1, X2			–20	μA
Output leakage current, high	I_{LOH}	V_{OUT} = V_{DD}				3	μA
Output leakage current, low	I_{LOL}	V_{OUT} = 0 V				–3	μA
Software pull-up resistor	R_1	V_{IN} = 0 V		50	100	200	kΩ

Remark Unless specified otherwise, the characteristics of alternate-function pins are the same as those of port pins.

DC Characteristics (T_A = –40 to +85°C, V_{DD} = 1.8 to 5.5 V) (2/2)

Parameter	Symbol	Conditions		MIN.	TYP.	MAX.	Unit
Power supply current[Note 1]	I_{DD1}	5.0-MHz crystal oscillation operating mode (C1 = C2 = 22pF)	V_{DD} = 5.0 V±10%[Note 2]		4.0	15.0	mA
			V_{DD} = 3.0 V±10%[Note 3]		1.0	5.0	mA
			V_{DD} = 2.0 V±10%[Note 3]		0.8	3.0	mA
	I_{DD2}	5.0-MHz crystal oscillation HALT mode (C1 = C2 = 22pF)	V_{DD} = 5.0 V±10%[Note 2]		0.8	5.0	mA
			V_{DD} = 3.0 V±10%[Note 3]		0.5	2.5	mA
			V_{DD} = 2.0 V±10%[Note 3]		0.3	1.0	mA
	I_{DD3}	STOP mode	V_{DD} = 5.0 V±10%		0.1	30	μA
			V_{DD} = 3.0 V±10%		0.05	10	μA
			V_{DD} = 2.0 V±10%		0.05	10	μA

Notes 1. The current flowing to the ports (including the current flowing through an on-chip pull-up resistor) is not included.

2. High-speed mode operation (when processor clock control register (PCC) is set to 00H.)

3. Low-speed mode operation (when PCC is set to 02H).

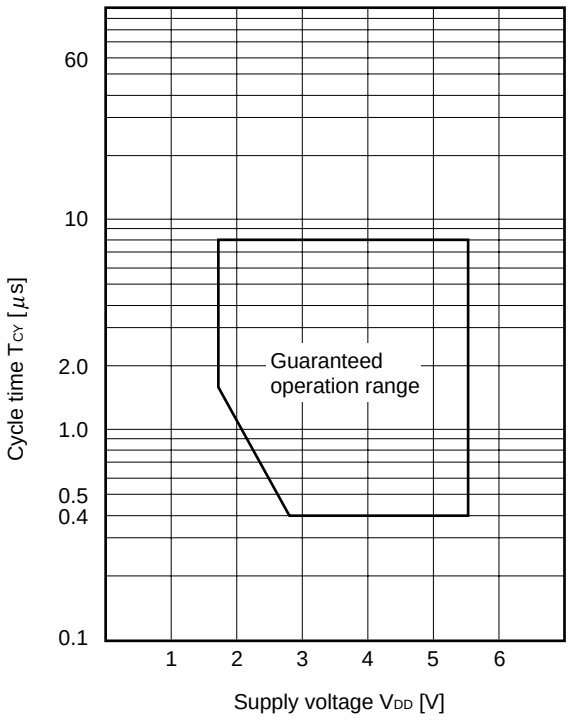
Remark Unless specified otherwise, the characteristics of alternate-function pins are the same as those of port pins.

AC Characteristics

(1) Basic operation (T_A = –40 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 frequency	f_{TI}	V_{DD} = 2.7 to 5.5 V	0		4	MHz
			0		275	kHz
TI80 input high-/low-level width	t_{TIH}, t_{TIL}	V_{DD} = 2.7 to 5.5 V	0.1			μs
			1.8			μs
CPT90 input high-/low-level width	t_{CPH}, t_{CPL}		10			μs
Interrupt input high-/low-level width	t_{INTH}, t_{INTL}	INTP0 to INTP2	10			μs
$\overline{\text{RESET}}$ low-level width	t_{RST}		10			μs

T_{CY} vs V_{DD}



(2) Serial interface 20 (T_A = –40 to +85°C, V_{DD} = 1.8 to 5.5 V)

(a) 3-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
				3200			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 (to $\overline{SCK20}$↑)	t_{SIK1}	V_{DD} = 2.7 to 5.5 V		150			ns
				500			ns
SI20 hold time (from $\overline{SCK20}$↑)	t_{KSI1}	V_{DD} = 2.7 to 5.5 V		400			ns
				600			ns
SO20 output delay time from $\overline{SCK20}$↓	t_{KSO1}	R = 1 kΩ, C = 100 pF[Note]	V_{DD} = 2.7 to 5.5 V	0		250	ns
				0		1000	ns

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

(b) 3-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		900			ns
				3500			ns
$\overline{SCK20}$ high-/low-level width	t_{KH2}, t_{KL2}	V_{DD} = 2.7 to 5.5 V		400			ns
				1600			ns
SI20 setup time (to $\overline{SCK20}$↑)	t_{SIK2}	V_{DD} = 2.7 to 5.5 V		100			ns
				150			ns
SI20 hold time (from $\overline{SCK20}$↑)	t_{KSI2}	V_{DD} = 2.7 to 5.5 V		400			ns
				600			ns
SO20 output delay time from $\overline{SCK20}$↓	t_{KSO2}	R = 1 kΩ, C = 100 pF[Note]	V_{DD} = 2.7 to 5.5 V	0		300	ns
				0		1000	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 load resistance and load capacitance of the SO20 output line.

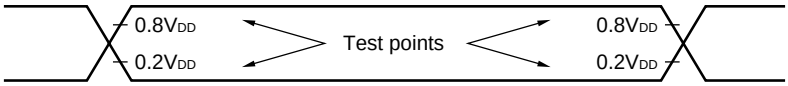
(c) UART mode (Dedicated baud rate generator output)

Parameter	Symbol	Conditions	MIN.	TYP.	MAX.	Unit
Transfer rate		V_{DD} = 2.7 to 5.5 V			78125	bps
					19531	bps

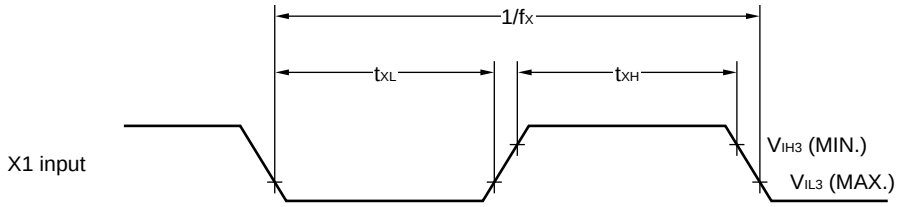
(d) UART mode (external clock input)

Parameter	Symbol	Conditions	MIN.	TYP.	MAX.	Unit
ASCK20 cycle time	t_{KCY3}	V_{DD} = 2.7 to 5.5 V	900			ns
			3500			ns
ASCK20 high-/low-level width	t_{KH3}, t_{KL3}	V_{DD} = 2.7 to 5.5 V	400			ns
			1600			ns
Transfer rate		V_{DD} = 2.7 to 5.5 V			39063	bps
					9766	bps
ASCK20 rise/fall time	t_R, t_F				1	μs

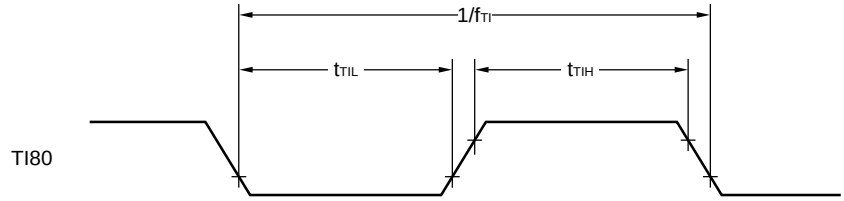
AC Timing Test Points (excluding X1 input)



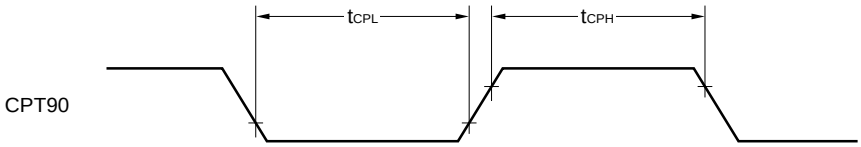
Clock Timing



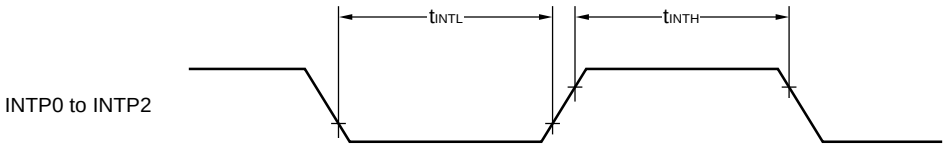
TI80 Timing



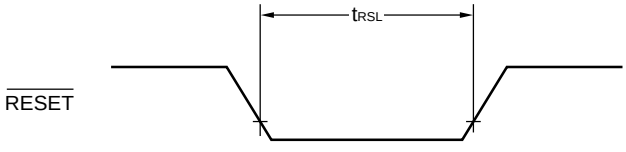
CPT90 Input Timing



Interrupt Input Timing

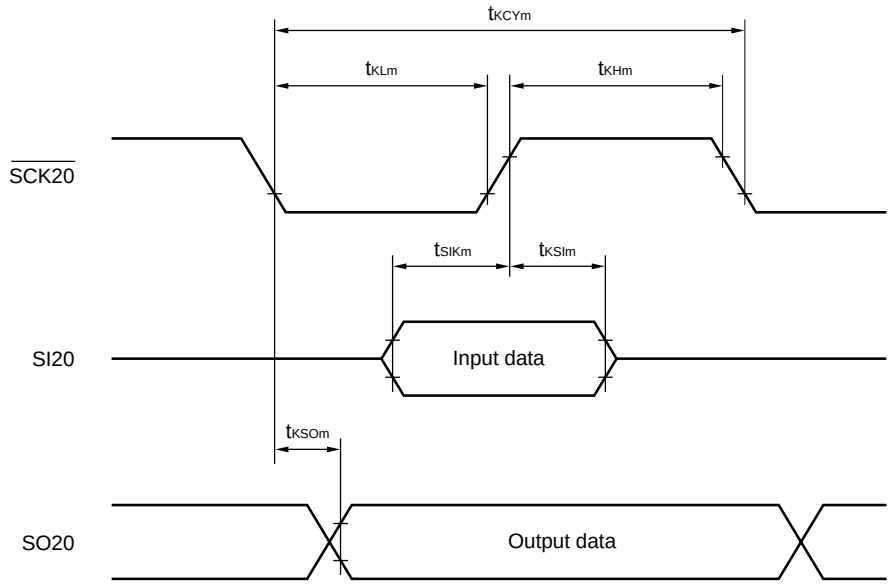


$\overline{\text{RESET}}$ Input Timing



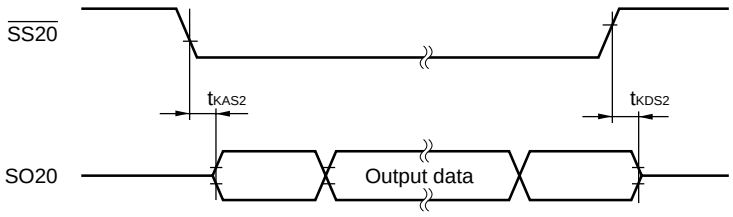
Serial Transfer Timing

3-wire serial I/O mode:

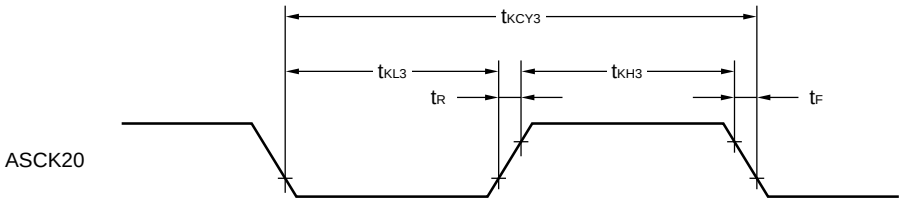


m = 1, 2

3-wire serial I/O mode (when $\overline{SS20}$ is used):



UART mode (external clock input):



Data Memory STOP Mode Low Supply Voltage Data Retention Characteristics (T_A = –40 to +85°C)

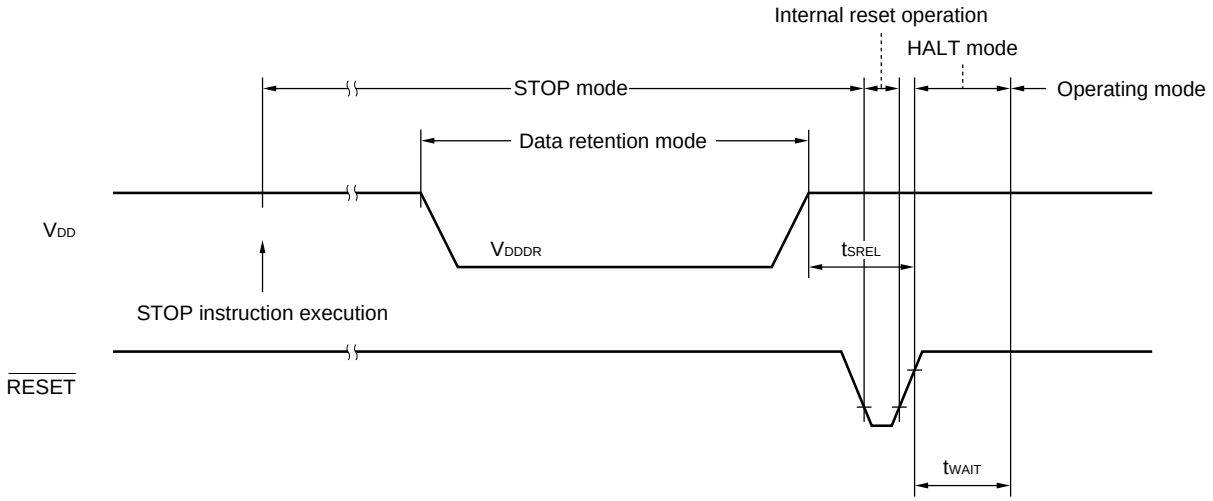
Parameter	Symbol	Conditions	MIN.	TYP.	MAX.	Unit
Data retention supply voltage	V_{DDDR}		1.8		5.5	V
Release signal set time	t_{SREL}		0			μs
Oscillation stabilization wait time[Note 1]	t_{WAIT}	Release by $\overline{\text{RESET}}$		$2^{15}/f_X$		s
		Release by interrupt request		**Note 2**		s

Notes 1. The oscillation stabilization wait time is the period during which the CPU operation is stopped to avoid unstable operation at the beginning of oscillation.

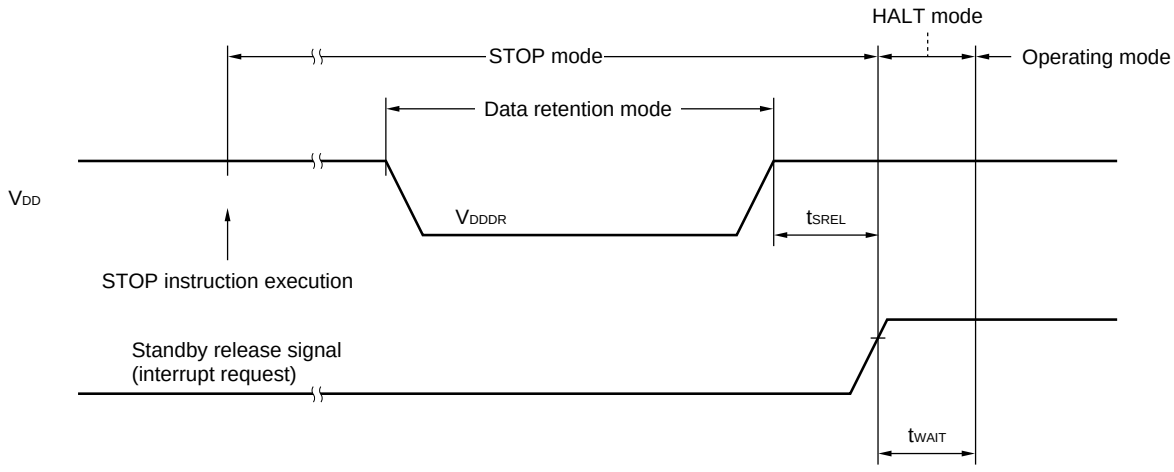
2. Selection of $2^{12}/f_X$, $2^{15}/f_X$, or $2^{17}/f_X$ is possible with bits 0 to 2 (OSTS0 to OSTS2) of the oscillation stabilization time select register.

Remark f_X: System clock oscillation frequency

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



Data Retention Timing (Standby release signal: STOP mode release by interrupt signal)



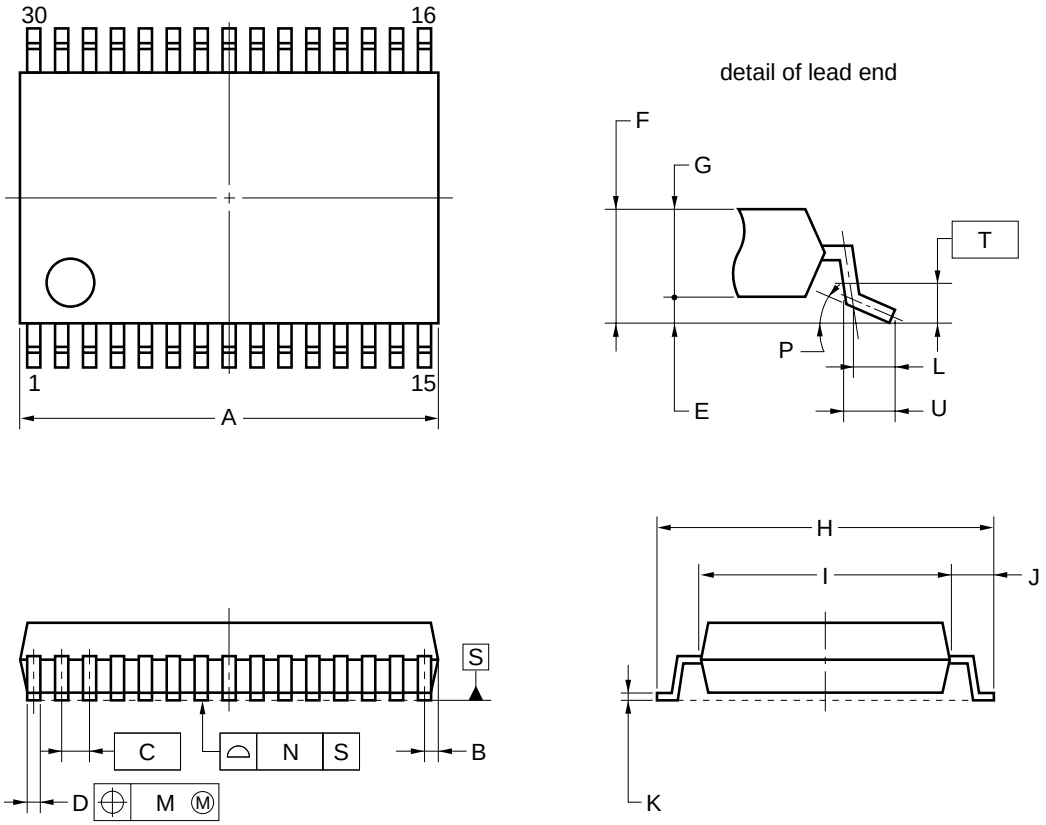
FLASH MEMORY WRITE/DELETE CHARACTERISTICS (T_A = 10°C to 40°C, V_{DD} = 1.8 to 5.5 V)

Parameter	Symbol	Conditions	MIN.	TYP.	MAX.	Unit
Write current (V_{DD} pin) Note	I_{DDW}	When V_{PP} supply voltage = V_{PP1} (5.0-MHz crystal oscillation operating mode)			18	mA
Write current (V_{PP} pin) Note	I_{PPW}	When V_{PP} supply voltage = V_{PP1}			7.5	mA
Delete current (V_{DD} pin) Note	I_{DDE}	When V_{PP} supply voltage = V_{PP1} (5.0-MHz crystal oscillation operating mode)			18	mA
Delete current (V_{PP} pin) Note	I_{PPE}	When V_{PP} supply voltage = V_{PP1}			100	mA
Unit delete time	t_{er}		0.5	1	1	s
Total delete time	t_{era}				20	s
Write count		Delete/write are regarded as 1 cycle			20	Times
V_{PP} supply voltage	V_{PP0}	In normal operation	0		$0.2V_{DD}$	V
	V_{PP1}	During flash memory programming	9.7	10.0	10.3	V
Operating frequency	f_X	V_{DD} = 1.8 to 5.5 V	1.0		1.25	MHz
		V_{DD} = 2.7 to 5.5 V	1.0		5.0	MHz

Note The current flowing to the ports (including the current flowing through an on-chip pull-up resistor) and AV_{DD} current are not included.

12. PACKAGE DRAWING

30-PIN PLASTIC SSOP (7.62 mm (300))



NOTE

Each lead centerline is located within 0.13 mm of its true position (T.P.) at maximum material condition.

ITEM	MILLIMETERS
A	9.85±0.15
B	0.45 MAX.
C	0.65 (T.P.)
D	$0.24^{+0.08}_{-0.07}$
E	0.1±0.05
F	1.3±0.1
G	1.2
H	8.1±0.2
I	6.1±0.2
J	1.0±0.2
K	0.17±0.03
L	0.5
M	0.13
N	0.10
P	$3^{\circ}{}^{+5^{\circ}}_{-3^{\circ}}$
T	0.25
U	0.6±0.15

S30MC-65-5A4-2

APPENDIX A DEVELOPMENT TOOLS

The following development tools are available for system development using the μPD78F9076.

Language Processing Software

RA78K0S[Notes 1, 2, 3]	Assembler package common to 78K/0S Series
CC78K0S[Notes 1, 2, 3]	C compiler package common to 78K/0S Series
DF789076[Notes 1, 2, 3]	Device file for μPD78F9076
CC78K/0S-L[Notes 1, 2, 3]	C compiler library source file common to 78K/0S Series

Flash Memory Writing Tools

Flashpro III (Model number: FL-PR3[Note 4], PG-FP3)	Dedicated flash programmer for on-chip flash memory
FA-30MC[Note 4]	Flash memory writing adapter

Debugging Tools (1/2)

IE-78K0S-NS In-circuit emulator	In-circuit emulator serves to debug hardware and software when developing application systems using a 78K/0S Series product. It supports the ID78K0S-NS integrated debugger. Used in combination with an AC adapter, emulation probe, and interface adapter connecting to the host machine.
IE-70000-MC-PS-B AC adapter	Adapter used to supply power from a power outlet of 100 V AC to 240 V AC.
IE-70000-98-IF-C Interface adapter	Adapter when PC-9800 series PC (except notebook type) is used as the IE-78K0S-NS host machine (C bus supported).
IE-70000-CD-IF-A PC card interface	PC card and interface cable when notebook PC is used as the IE-78K0S-NS host machine (PCMCIA socket supported).
IE-70000-PC-IF-C Interface adapter	Adapter when using an IBM PC/AT™ or compatible as the IE-78K0S-NS host machine.
IE-70000-PCI-IF Interface adapter	Adapter when using PC that includes a PCI bus as the IE-78K0S-NS host machine.
IE-789046-NS-EM1+ NP-K907[Note 4] Emulation board	Board for emulation of the peripheral hardware peculiar to a device. Used in combination with an in-circuit emulator.
NP-36GS[Note 4]	Board used to connect the in-circuit emulator to the target system. For a 30-pin plastic SSOP (MC-5A4 type), used in combination with NGS-30.
NGS-30[Note 4] Conversion socket	Conversion socket used to connect the NP-36GS to the target system board designed to mount a 30-pin plastic SSOP (MC-5A4 type).

Notes 1. PC-9800 series (Japanese Windows™) based

2. IBM PC/AT or compatibles (Japanese/English Windows) based

3. HP9000 series 700™ (HP-UX™), SPARCstation™ (SunOS™, Solaris™), or NEWS™ (NEWS-OS™) based.

4. Products made by Naito Densei Machida Mfg. Co., Ltd. (Phone: +81-44-822-3813).

Remark RA78K0S, CC78K0S, and SM78K0S are used in combination with the DF789076.

Debugging Tools (2/2)

SM78K0S[Notes 1, 2]	System simulator common to 78K/0S Series
ID78K0S-NS[Notes 1, 2]	Integrated debugger common to 78K/0S Series
DF789076[Notes 1, 2]	Device file for μPD78F9076

Real-time OS

MX78K0S[Notes 1, 2]	OS for 78K/0S Series

Notes 1. PC-9800 series (Japanese Windows) based.

2. IBM PC/AT or compatibles (Japanese/English Windows) based.

APPENDIX B RELATED DOCUMENTS

The related documents indicated in this publication may include preliminary versions. However, preliminary versions are not marked as such.

Documents Related to Devices

Document Name	Document No.	
	Japanese	English
μPD789071, 789072, 789074 Data Sheet	To be prepared	To be prepared
μPD78F9076 Preliminary Product Information	U14708J	This manual
μPD789074 Subseries User's Manual	U14801J	To be prepared
78K/0S Series User's Manual Instruction	U11047J	U11047E
78K/0, 78K/0S Series Application Note Flash Memory Write	U14458J	U14458E

Documents Related to Development Tools (User's Manuals)

Document Name		Document No.	
		Japanese	English
RA78K0S Assembler Package	Operation	U11622J	U11622E
	Assembly Language	U11599J	U11599E
	Structured Assembly Language	U11623J	U11623E
CC78K0S C Compiler	Operation	U11816J	U11816E
	Language	U11817J	U11817E
SM78K0S System Simulator Windows Based	Reference	U11489J	U11489E
SM78K Series System Simulator	External Parts User Open Interface Specifications	U10092J	U10092E
ID78K0S-NS Integrated Debugger Windows Based	Reference	U12901J	U12901E
IE-78K0S-NS In-circuit Emulator		U13549J	U13549E
IE-789046-NS-EM1 Emulation Board		U14433J	U14433E

Documents Related to Embedded Software (User's Manuals)

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

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

Other Related Documents

Document Name	Document No.	
	Japanese	English
SEMICONDUCTOR SELECTION GUIDE Products & Packages (CD-ROM)	X13769X	
Semiconductor Device Mounting Technology Manual	C10535J	C10535E
Quality Grades on NEC Semiconductor Devices	C11531J	C11531E
NEC Semiconductor Device Reliability/Quality Control System	C10983J	C10983E
Guide to Prevent Damage for Semiconductor Devices by Electrostatic Discharge (ESD)	C11892J	C11892E
Guide to Microcomputer-Related Products by Third Party	U11416J	–

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

[MEMO]

NOTES FOR CMOS DEVICES

① PRECAUTION AGAINST ESD FOR SEMICONDUCTORS

Note:

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

② HANDLING OF UNUSED INPUT PINS FOR CMOS

Note:

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

③ STATUS BEFORE INITIALIZATION OF MOS DEVICES

Note:

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

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.l.
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: 91-504-2787
Fax: 91-504-2860

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

J99.1

- **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.**
- No part of this document may be copied or reproduced in any form or by any means without the prior written consent of NEC Corporation. NEC Corporation assumes no responsibility for any errors which may appear in this document.
- NEC Corporation does not assume any liability for infringement of patents, copyrights or other intellectual property rights of third parties by or arising from use of a device described herein or any other liability arising from use of such device. No license, either express, implied or otherwise, is granted under any patents, copyrights or other intellectual property rights of NEC Corporation or others.
- Descriptions of circuits, software, and other related information in this document are provided for illustrative purposes in semiconductor product operation and application examples. The incorporation of these circuits, software, and information in the design of the customer's equipment shall be done under the full responsibility of the customer. NEC Corporation assumes no responsibility for any losses incurred by the customer or third parties arising from the use of these circuits, software, and information.
- While NEC Corporation has been making continuous effort to enhance the reliability of its semiconductor devices, the possibility of defects cannot be eliminated entirely. To minimize risks of damage or injury to persons or property arising from a defect in an NEC semiconductor device, customers must incorporate sufficient safety measures in its design, such as redundancy, fire-containment, and anti-failure features.
- NEC devices are classified into the following three quality grades:
  "Standard", "Special", and "Specific". The Specific quality grade applies only to devices developed based on a customer designated "quality assurance program" for a specific application. The recommended applications of a device depend on its quality grade, as indicated below. Customers must check the quality grade of each device before using it in a particular application.

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

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

  Specific: Aircraft, aerospace equipment, submersible repeaters, nuclear reactor control systems, life support systems or medical equipment for life support, etc.

  The quality grade of NEC devices is "Standard" unless otherwise specified in NEC's Data Sheets or Data Books. If customers intend to use NEC devices for applications other than those specified for Standard quality grade, they should contact an NEC sales representative in advance.

M5 98.8