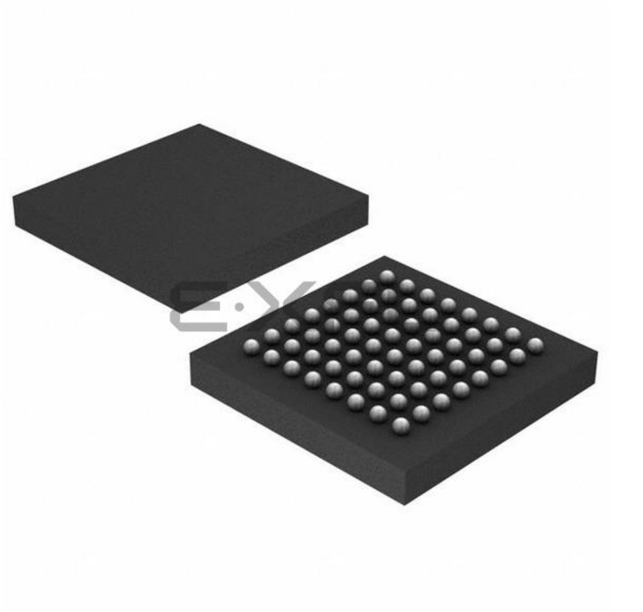
Renesas Electronics America Inc - UPD78F1142AF1-AN1-A Datasheet



Welcome to E-XFL.COM

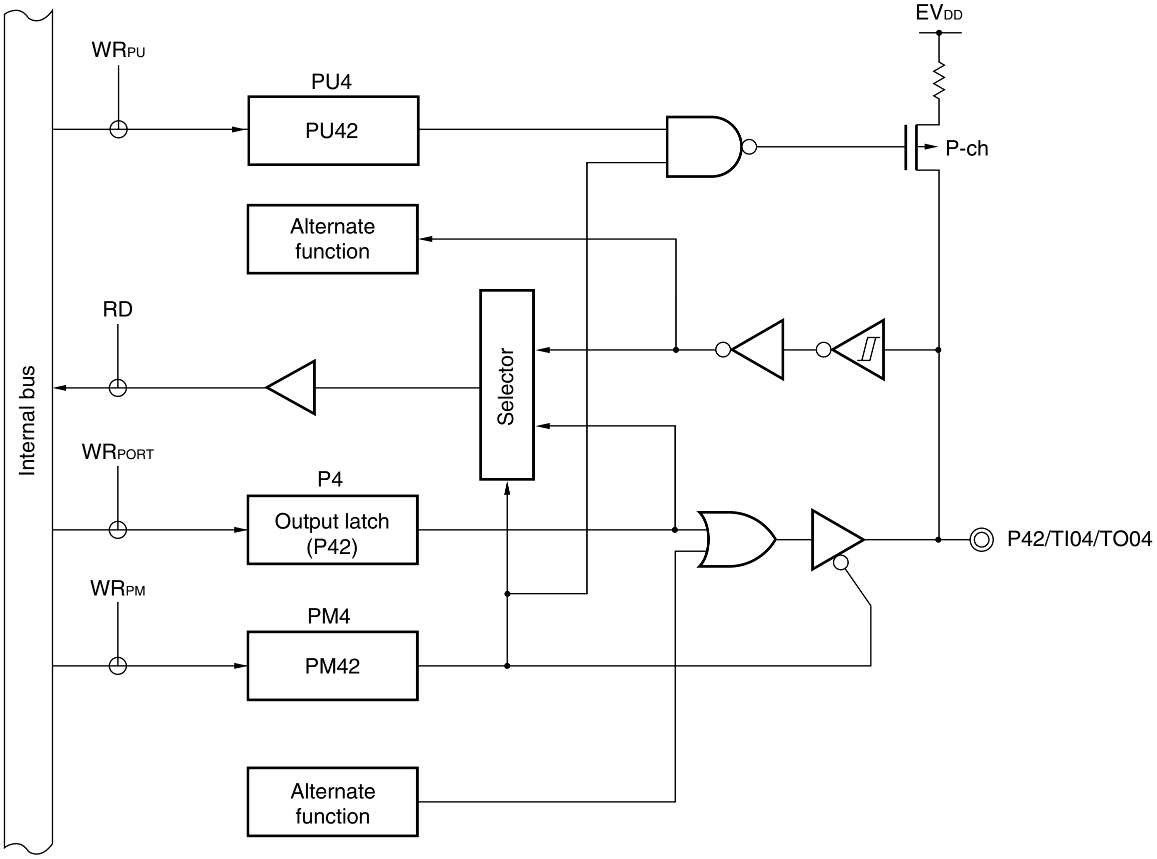
What is "Embedded - Microcontrollers"?

"Embedded - Microcontrollers" refer to small, integrated circuits designed to perform specific tasks within larger systems. These microcontrollers are essentially compact computers on a single chip, containing a processor core, memory, and programmable input/output peripherals. They are called "embedded" because they are embedded within electronic devices to control various functions, rather than serving as standalone computers. Microcontrollers are crucial in modern electronics, providing the intelligence and control needed for a wide range of applications.

Applications of "Embedded - Microcontrollers"

Details	
Product Status	Active
Core Processor	78K/0R
Core Size	16-Bit
Speed	20MHz
Connectivity	3-Wire SIO, I²C, LINbus, UART/USART
Peripherals	DMA, LVD, POR, PWM, WDT
Number of I/O	50
Program Memory Size	64KB (64K x 8)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	4K x 8
Voltage - Supply (Vcc/Vdd)	1.8V ~ 5.5V
Data Converters	A/D 8x10b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	64-VFBGA
Supplier Device Package	-
Purchase URL	https://www.e-xfl.com/product-detail/renesas-electronics-america/upd78f1142af1-an1-a

Figure 4-16. Block Diagram of P42



P4:	Port register 4
PU4:	Pull-up resistor option register 4
PM4:	Port mode register 4
RD:	Read signal
WR××:	Write signal

(3) Pull-up resistor option registers (PU0, PU1, PU3 to PU5, PU7, PU12, PU14)

These registers specify whether the on-chip pull-up resistors of P00 to P06, P10 to P17, P30, P31, P40 to P43, P50 to P55, P70 to P77, P120, P140, or P141 are to be used or not. On-chip pull-up resistors can be used in 1-bit units only for the bits set to input mode of the pins to which the use of an on-chip pull-up resistor has been specified in PU0, PU1, PU3 to PU5, PU7, PU12, and PU14. On-chip pull-up resistors cannot be connected to bits set to output mode and bits used as alternate-function output pins, regardless of the settings of PU0, PU1, PU3 to PU5, PU7, PU12, and PU14.

These registers can be set by a 1-bit or 8-bit memory manipulation instruction.

Reset signal generation clears these registers to 00H.

Figure 4-30. Format of Pull-up Resistor Option Register

Symbol	7	6	5	4	3	2	1	0	Address	After reset	R/W
PU0	0	PU06	PU05	PU04	PU03	PU02	PU01	PU00	F0030H	00H	R/W
PU1	PU17	PU16	PU15	PU14	PU13	PU12	PU11	PU10	F0031H	00H	R/W
PU3	0	0	0	0	0	0	PU31	PU30	F0033H	00H	R/W
PU4	0	0	0	0	PU43	PU42	PU41	PU40	F0034H	00H	R/W
PU5	0	0	PU55	PU54	PU53	PU52	PU51	PU50	F0035H	00H	R/W
PU7	PU77	PU76	PU75	PU74	PU73	PU72	PU71	PU70	F0037H	00H	R/W
PU12	0	0	0	0	0	0	0	PU120	F003CH	00H	R/W
PU14	0	0	0	0	0	0	PU141	PU140	F003EH	00H	R/W

PUmn	Pmn pin on-chip pull-up resistor selection (m = 0, 1, 3 to 5, 7, 12, 14; n = 0 to 7)
0	On-chip pull-up resistor not connected
1	On-chip pull-up resistor connected

(4) Port input mode registers (PIM0)

This register sets the input buffer of P03 or P04 in 1-bit units.

TTL input buffer can be selected during serial communication with an external device of the different potential.

This register can be set by a 1-bit or 8-bit memory manipulation instruction.

Reset signal generation clears this register to 00H.

Figure 4-31. Format of Port Input Mode Register

Address: F0040H After reset: 00H R/W

Symbol	7	6	5	4	3	2	1	0
PIM0	0	0	0	PIM04	PIM03	0	0	0

PIM0n	P0n pin input buffer selection (n = 3, 4)
0	Normal input buffer
1	TTL input buffer

(5) Port output mode registers (POM0)

This register sets the output mode of P02 to P04 in 1-bit units.

N-ch open drain output (V_{DD} tolerance) mode can be selected during serial communication with an external device of the different potential, and for the SDA10 and SDA20 pins during simplified I^2C communication with an external device of the same potential.

This register can be set by a 1-bit or 8-bit memory manipulation instruction.

Reset signal generation clears this register to 00H.

Figure 4-32. Format of Port Input Mode Register

Address: F0050H After reset: 00H R/W

Symbol	7	6	5	4	3	2	1	0
POM0	0	0	0	POM04	POM03	POM02	0	0

POMmn	Pmn pin output mode selection (n = 2 to 4)
0	Normal output mode
1	N-ch open-drain output (V_{DD} tolerance) mode

Table 4-5. Settings of Port Mode Register and Output Latch When Using Alternate Function (2/2)

Pin Name	Alternate Function		PM××	P××
	Function Name	I/O		
P20 to P27[Note]	ANI0 to ANI7[Note]	Input	1	×
P30	RTC1HZ	Output	0	0
	INTP3	Input	1	×
P31	TI03	Input	1	×
	TO03	Output	0	0
	INTP4	Input	1	×
P40	TOOL0	I/O	×	×
P41	TOOL1	Output	×	×
P42	TI04	Input	1	×
	TO04	Output	0	0
P50	INTP1	Input	1	×
P51	INTP2	Input	1	×
P60	SCL0	I/O	0	0
P61	SDA0	I/O	0	0
P70 to P73	KR0 to KR3	Input	1	×
P74 to P77	INTP8 to INTP11	Input	1	×
	KR4 to KR7	Input	1	×
P120	INTP0	Input	1	×
	EXLVI	Input	1	×
P140	PCLBUZ0	Output	0	0
	INTP6	Input	1	×
P141	PCLBUZ1	Output	0	0
	INTP7	Input	1	×

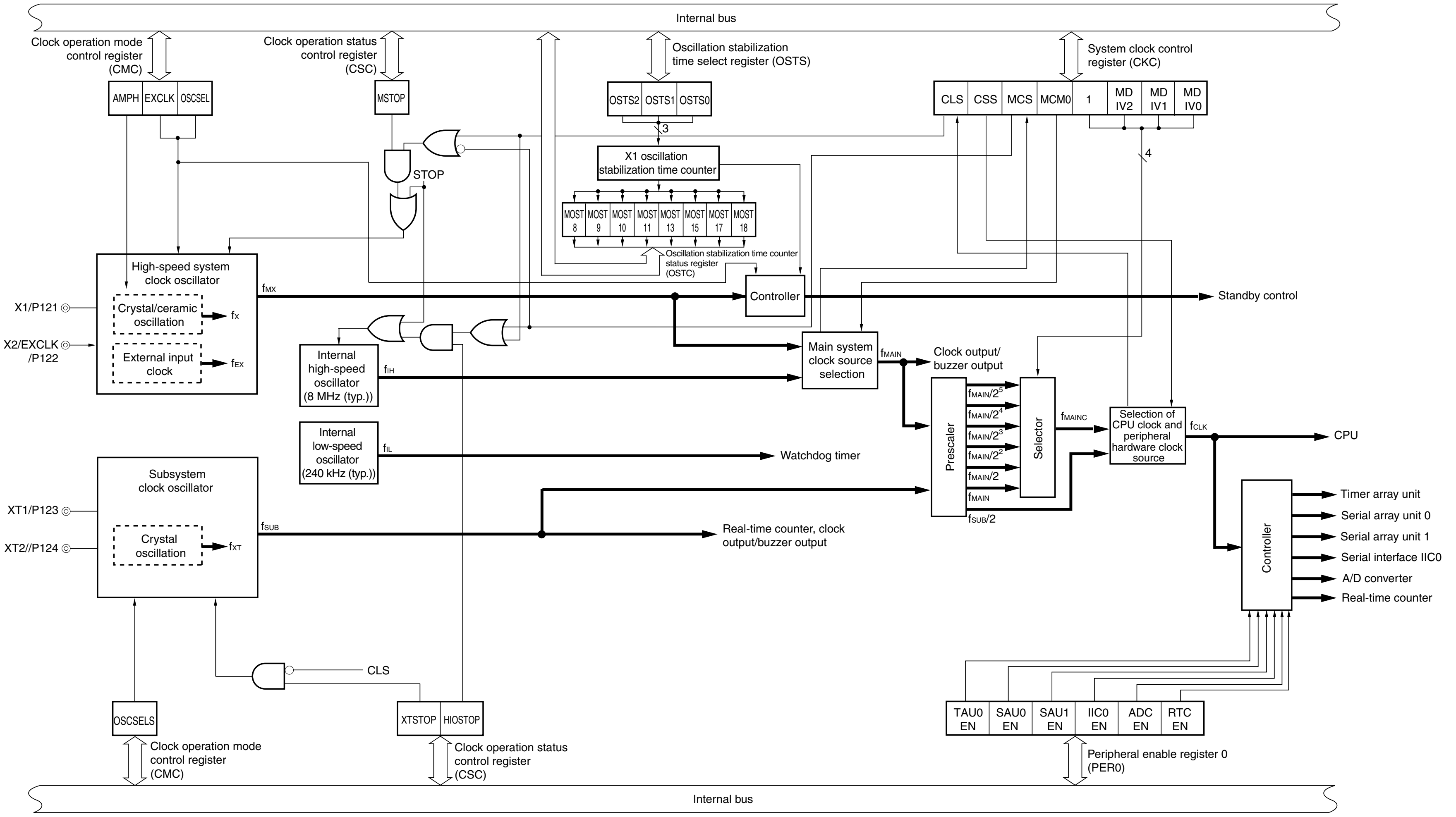
Remark ×: don't care
PM××: Port mode register
P××: Port output latch

Note The function of the ANI0/P20 to ANI7/P27 pins can be selected by using the A/D port configuration register (ADPC), the analog input channel specification register (ADS), and PM2.

Table 4-6. Setting Functions of ANI0/P20 to ANI7/P27 Pins

ADPC	PM2	ADS	ANI0/P20 to ANI7/P27 Pins
Digital I/O selection	Input mode	–	Digital input
	Output mode	–	Digital output
Analog input selection	Input mode	Selects ANI.	Analog input (to be converted)
		Does not select ANI.	Analog input (not to be converted)
	Output mode	Selects ANI.	Setting prohibited
		Does not select ANI.	

<R> Figure 5-1. Block Diagram of Clock Generator


Internal bus
Clock operation mode control register (CMC)
AMPH
EXCLK
OSCSEL
Clock operation status control register (CSC)
MSTOP
STOP
Oscillation stabilization time select register (OSTS)
OSTS2
OSTS1
OSTS0
3
X1 oscillation stabilization time counter
MOST 8
MOST 9
MOST 10
MOST 11
MOST 13
MOST 15
MOST 17
MOST 18
Oscillation stabilization time counter status register (OSTC)
System clock control register (CKC)
CLS
CSS
MCS
MCM0
1
MDIV2
MDIV1
MDIV0
4
High-speed system clock oscillator
Crystal/ceramic oscillation
fX
External input clock
fEX
X1/P121
X2/EXCLK /P122
fMX
Controller
Standby control
Internal high-speed oscillator (8 MHz (typ.))
fIH
Main system clock source selection
fMAIN
Clock output/ buzzer output
Internal low-speed oscillator (240 kHz (typ.))
fIL
Watchdog timer
Prescaler
fMAIN/2^5
fMAIN/2^4
fMAIN/2^3
fMAIN/2^2
fMAIN/2
fMAIN
fSUB/2
Selector
fMAINC
Selection of CPU clock and peripheral hardware clock source
fCLK
CPU
Subsystem clock oscillator
Crystal oscillation
fXT
XT1/P123
XT2//P124
fSUB
Real-time counter, clock output/buzzer output
Controller
Timer array unit
Serial array unit 0
Serial array unit 1
Serial interface IIC0
A/D converter
Real-time counter
CLS
OSCSELS
Clock operation mode control register (CMC)
XTSTOP
HIOSTOP
Clock operation status control register (CSC)
TAU0 EN
SAU0 EN
SAU1 EN
IIC0 EN
ADC EN
RTC EN
Peripheral enable register 0 (PER0)
Internal bus

(2) Example of setting procedure when using the external main system clock

<1> Setting P121/X1 and P122/X2/EXCLK pins (CMC register)

EXCLK	OSCSEL	0	OSCSELS	0	0	0	AMPH
1	1	0	0/1	0	0	0	×

Remarks 1. ×: don't care

2. For setting of the P123/XT1 and P124/XT2 pins, see **5.6.3 (1) Example of setting procedure when oscillating the subsystem clock**.

<2> Controlling external main system clock input (CSC register)
When MSTOP is cleared to 0, the input of the external main system clock is enabled.

Cautions 1. The CMC register can be written only once after reset release, by an 8-bit memory manipulation instruction.
Therefore, it is necessary to also set the value of the OSCSELS bits at the same time. For OSCSELS bits, see 5.6.3 Example of controlling subsystem clock.

2. Set the external main system clock after the supply voltage has reached the operable voltage of the clock to be used (see CHAPTER 27 ELECTRICAL SPECIFICATIONS (STANDARD PRODUCTS) and CHAPTER 28 ELECTRICAL SPECIFICATIONS ((A) GRADE PRODUCTS)).

(3) Example of setting procedure when using high-speed system clock as CPU/peripheral hardware clock

<1> Setting high-speed system clock oscillation[Note]
(See **5.6.1 (1) Example of setting procedure when oscillating the X1 clock** and **(2) Example of setting procedure when using the external main system clock.**)

Note The setting of <1> is not necessary when high-speed system clock is already operating.

<2> Setting the high-speed system clock as the source clock of the CPU/peripheral hardware clock and setting the division ratio of the set clock (CKC register)

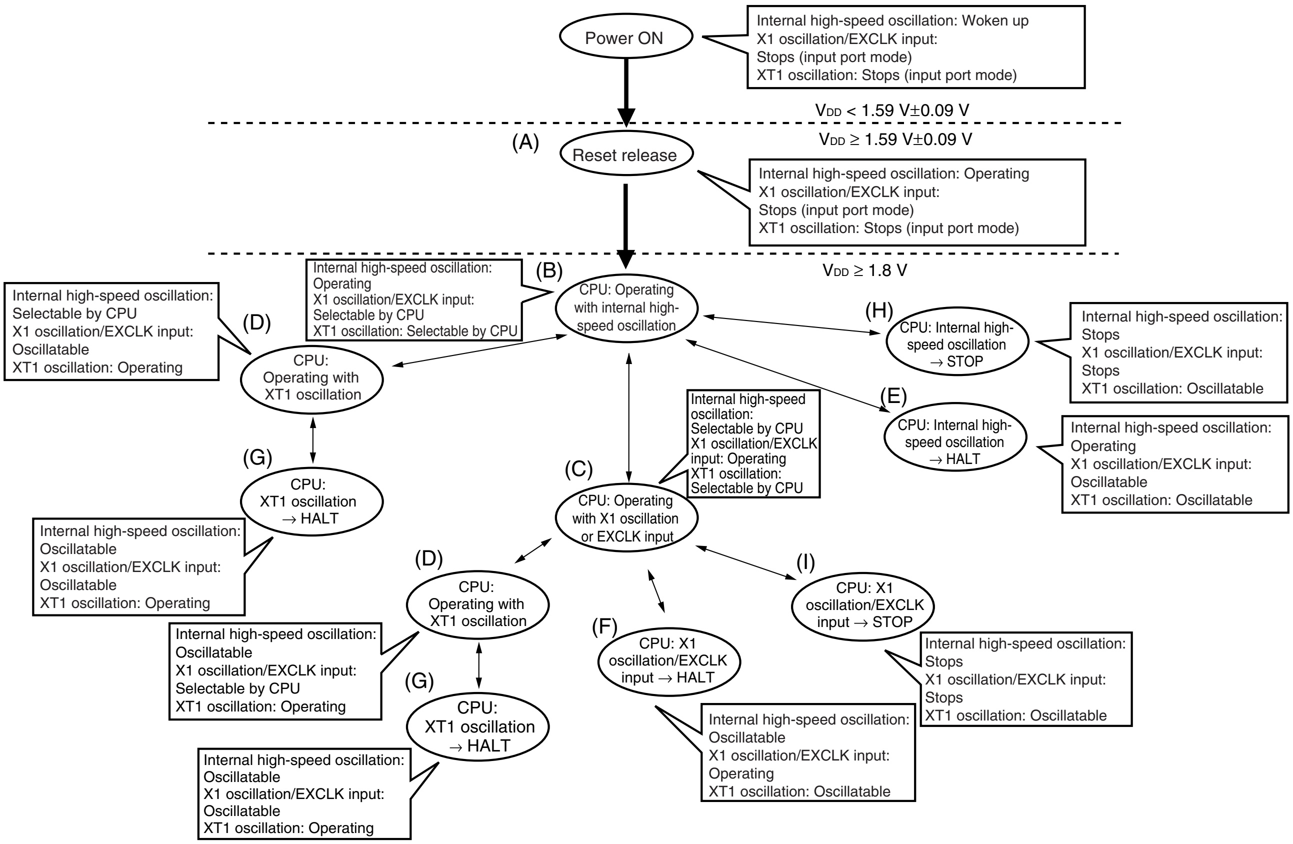
MCM0	MDIV2	MDIV1	MDIV0	Selection of CPU/Peripheral Hardware Clock (f_{CLK})
1	0	0	0	f_{MX}
	0	0	1	$f_{MX}/2$
	0	1	0	$f_{MX}/2^2$
	0	1	1	$f_{MX}/2^3$
	1	0	0	$f_{MX}/2^4$
	1	0	1	$f_{MX}/2^5$ [Note]

Note Setting is prohibited when f_{MX} < 4 MHz.

5.6.5 CPU clock status transition diagram

Figure 5-15 shows the CPU clock status transition diagram of this product.

Figure 5-15. CPU Clock Status Transition Diagram


Power ON
Internal high-speed oscillation: Woken up
X1 oscillation/EXCLK input:
Stops (input port mode)
XT1 oscillation: Stops (input port mode)
VDD < 1.59 V±0.09 V
VDD ≥ 1.59 V±0.09 V
(A)
Reset release
Internal high-speed oscillation: Operating
X1 oscillation/EXCLK input:
Stops (input port mode)
XT1 oscillation: Stops (input port mode)
VDD ≥ 1.8 V
Internal high-speed oscillation:
Operating
X1 oscillation/EXCLK input:
Selectable by CPU
XT1 oscillation: Selectable by CPU
(B)
CPU: Operating with internal high-speed oscillation
Internal high-speed oscillation:
Selectable by CPU
X1 oscillation/EXCLK input:
Oscillatable
XT1 oscillation: Operating
(D)
CPU: Operating with XT1 oscillation
(H)
CPU: Internal high-speed oscillation → STOP
Internal high-speed oscillation:
Stops
X1 oscillation/EXCLK input:
Stops
XT1 oscillation: Oscillatable
(E)
CPU: Internal high-speed oscillation → HALT
Internal high-speed oscillation:
Operating
X1 oscillation/EXCLK input:
Oscillatable
XT1 oscillation: Oscillatable
Internal high-speed oscillation:
Selectable by CPU
X1 oscillation/EXCLK input: Operating
XT1 oscillation:
Selectable by CPU
(G)
CPU: XT1 oscillation → HALT
Internal high-speed oscillation:
Oscillatable
X1 oscillation/EXCLK input:
Oscillatable
XT1 oscillation: Operating
(C)
CPU: Operating with X1 oscillation or EXCLK input
(D)
CPU: Operating with XT1 oscillation
Internal high-speed oscillation:
Oscillatable
X1 oscillation/EXCLK input:
Selectable by CPU
XT1 oscillation: Operating
(I)
CPU: X1 oscillation/EXCLK input → STOP
Internal high-speed oscillation:
Stops
X1 oscillation/EXCLK input:
Stops
XT1 oscillation: Oscillatable
(F)
CPU: X1 oscillation/EXCLK input → HALT
Internal high-speed oscillation:
Oscillatable
X1 oscillation/EXCLK input:
Operating
XT1 oscillation: Oscillatable
(G)
CPU: XT1 oscillation → HALT
Internal high-speed oscillation:
Oscillatable
X1 oscillation/EXCLK input:
Oscillatable
XT1 oscillation: Operating

Remark If the low-power-supply detector (LVI) is set to ON by default by the option bytes, the reset will not be released until the power supply voltage (V_{DD}) exceeds 2.07 V±0.2 V.

After the reset operation, the status will shift to (B) in the above figure.

Table 5-4 shows transition of the CPU clock and examples of setting the SFR registers.

Table 5-4. CPU Clock Transition and SFR Register Setting Examples (1/4)

(1) CPU operating with internal high-speed oscillation clock (B) after reset release (A)

Status Transition	SFR Register Setting
(A) → (B)	SFR registers do not have to be set (default status after reset release).

(2) CPU operating with high-speed system clock (C) after reset release (A)

(The CPU operates with the internal high-speed oscillation clock immediately after a reset release (B).)

(Setting sequence of SFR registers) →

Setting Flag of SFR Register Status Transition	CMC Register[Note 1]			CSC Register	OSMC Register	OSTC Register	CKC Register
	EXCLK	OSCSEL	AMPH	MSTOP	FSEL		MCM0
(A) → (B) → (C) (X1 clock: 2 MHz ≤ f_X ≤ 10 MHz)	0	1	0	0	0	Must be checked	1
(A) → (B) → (C) (X1 clock: 10 MHz < f_X ≤ 20 MHz)	0	1	1	0	1[Note 2]	Must be checked	1
<R> (A) → (B) → (C) (external main clock)	1	1	×	0	0/1	Must not be checked	1

Notes 1. The CMC and OSMC registers can be written only once by an 8-bit memory manipulation instruction after reset release.

2. FSEL = 1 when f_{CLK} > 10 MHz

If a divided clock is selected and f_{CLK} ≤ 10 MHz, use with FSEL = 0 is possible even if f_X > 10 MHz.

Caution Set the clock after the supply voltage has reached the operable voltage of the clock to be set (see CHAPTER 27 ELECTRICAL SPECIFICATIONS (STANDARD PRODUCTS) and CHAPTER 28 ELECTRICAL SPECIFICATIONS ((A) GRADE PRODUCTS)).

<R> **Remark** ×: don't care

(3) CPU operating with subsystem clock (D) after reset release (A)

(The CPU operates with the internal high-speed oscillation clock immediately after a reset release (B).)

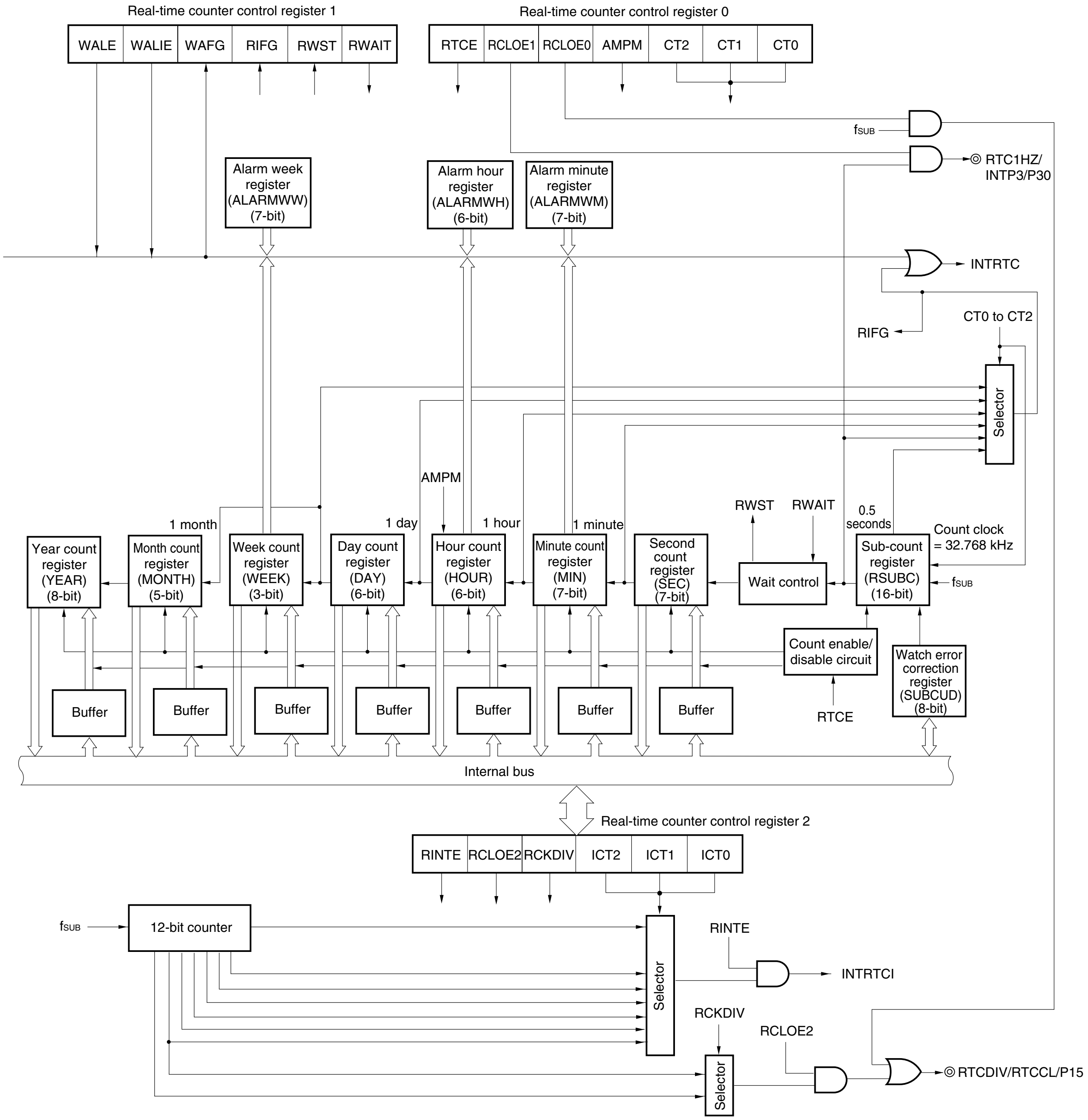
(Setting sequence of SFR registers) →

Setting Flag of SFR Register Status Transition	CMC Register[Note]	CSC Register	Waiting for Oscillation Stabilization	CKC Register
	OSCSELS	XTSTOP		CSS
(A) → (B) → (D)	1	0	Necessary	1

Note The CMC register can be written only once by an 8-bit memory manipulation instruction after reset release.

Remark (A) to (I) in Table 5-4 correspond to (A) to (I) in Figure 5-15.

Figure 7-1. Block Diagram of Real-Time Counter


Real-time counter control register 1
WALE
WALIE
WAFG
RIFG
RWST
RWAIT
Real-time counter control register 0
RTCE
RCLOE1
RCLOE0
AMPM
CT2
CT1
CT0
fSUB
RTC1HZ/
INTP3/P30
Alarm week register (ALARMWW) (7-bit)
Alarm hour register (ALARMWH) (6-bit)
Alarm minute register (ALARMWM) (7-bit)
INTRTC
RIFG
CT0 to CT2
Selector
AMPM
RWST
RWAIT
0.5 seconds
Count clock = 32.768 kHz
1 month
1 day
1 hour
1 minute
Year count register (YEAR) (8-bit)
Month count register (MONTH) (5-bit)
Week count register (WEEK) (3-bit)
Day count register (DAY) (6-bit)
Hour count register (HOUR) (6-bit)
Minute count register (MIN) (7-bit)
Second count register (SEC) (7-bit)
Wait control
Sub-count register (RSUBC) (16-bit)
fSUB
Count enable/ disable circuit
Watch error correction register (SUBCUD) (8-bit)
RTCE
Buffer
Buffer
Buffer
Buffer
Buffer
Buffer
Buffer
Internal bus
Real-time counter control register 2
RINTE
RCLOE2
RCKDIV
ICT2
ICT1
ICT0
fSUB
12-bit counter
Selector
RINTE
INTRTCI
RCKDIV
RCLOE2
Selector
RTCDIV/RTCCL/P15

(11) Month count register (MONTH)

The MONTH register is an 8-bit register that takes a value of 1 to 12 (decimal) and indicates the count value of months.

It counts up when the day counter overflows.

When data is written to this register, it is written to a buffer and then to the counter up to 2 clocks (32.768 kHz) later. Even if the day count register overflows while this register is being written, this register ignores the overflow and is set to the value written. Set a decimal value of 01 to 12 to this register in BCD code. If a value outside the range is set, the register value returns to the normal value after 1 period.

MONTH can be set by an 8-bit memory manipulation instruction.

Reset signal generation clears this register to 01H.

Figure 7-12. Format of Month Count Register (MONTH)

Address: FFF97H After reset: 01H R/W

Symbol	7	6	5	4	3	2	1	0
MONTH	0	0	0	MONTH10	MONTH8	MONTH4	MONTH2	MONTH1

(12) Year count register (YEAR)

The YEAR register is an 8-bit register that takes a value of 0 to 99 (decimal) and indicates the count value of years.

It counts up when the month counter overflows.

Values 00, 04, 08, …, 92, and 96 indicate a leap year.

When data is written to this register, it is written to a buffer and then to the counter up to 2 clocks (32.768 kHz) later. Even if the month count register overflows while this register is being written, this register ignores the overflow and is set to the value written. Set a decimal value of 00 to 99 to this register in BCD code. If a value outside the range is set, the register value returns to the normal value after 1 period.

YEAR can be set by an 8-bit memory manipulation instruction.

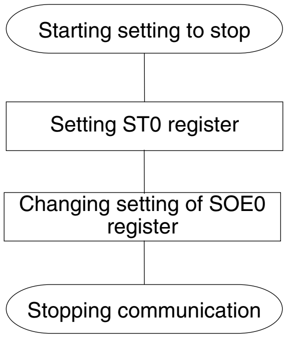
Reset signal generation clears this register to 00H.

Figure 7-13. Format of Year Count Register (YEAR)

Address: FFF98H After reset: 00H R/W

Symbol	7	6	5	4	3	2	1	0
YEAR	YEAR80	YEAR40	YEAR20	YEAR10	YEAR8	YEAR4	YEAR2	YEAR1

Figure 11-26. Procedure for Stopping Master Transmission



Write 1 to the ST0n bit of the target channel.

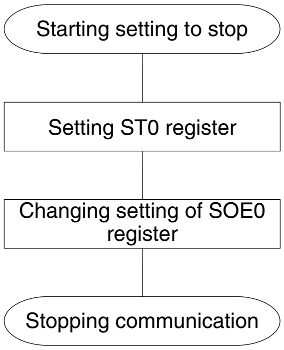
Set the SOE0 register and stop the output of the target channel

Stop communication in midway.

Remarks 1. Even after communication is stopped, the pin level is retained. To resume the operation, re-set the SO0 register (see **Figure 11-27 Procedure for Resuming Master Transmission**).

2. p: CSI number (p = 00, 10)

Figure 11-64. Procedure for Stopping Slave Transmission/Reception



Write 1 to the ST0n bit of the target channel.

Set the SOE0 register and stop the output of the target channel.

Stop communication in midway.

Remark Even after communication is stopped, the pin level is retained. To resume the operation, re-set the SO0 register (see **Figure 11-65 Procedure for Resuming Slave Transmission/Reception**).

13.1 Functions of Multiplier

The multiplier has the following functions.

- Can execute calculation of 16 bits × 16 bits = 32 bits.

Figure 13-1 shows the block diagram of the multiplier.

Figure 13-1. Block Diagram of Multiplier

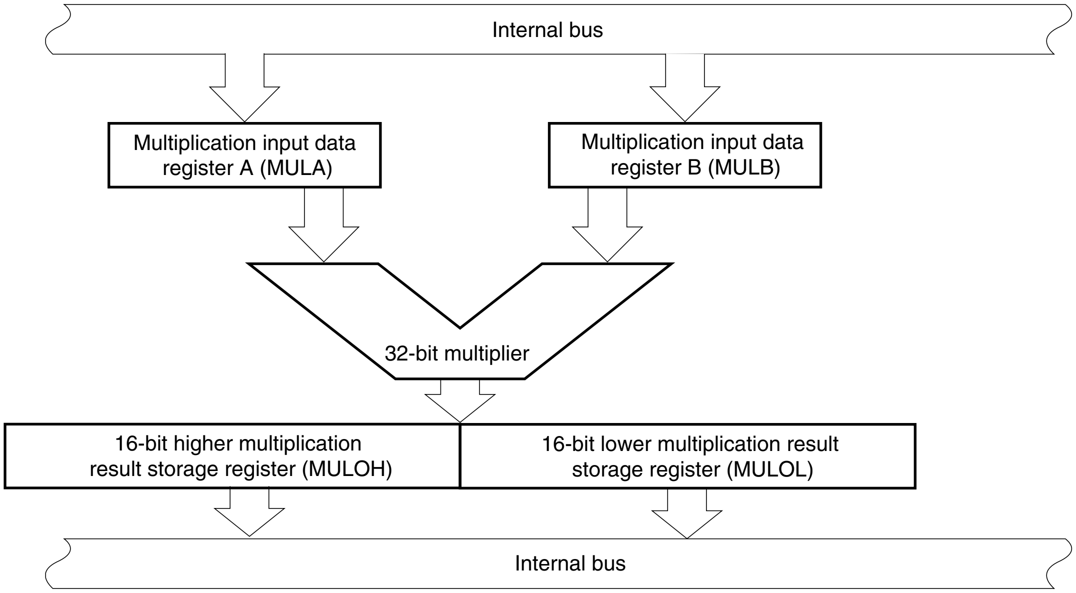
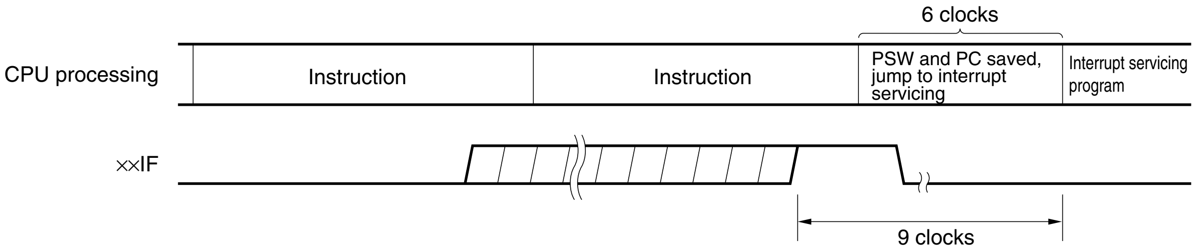
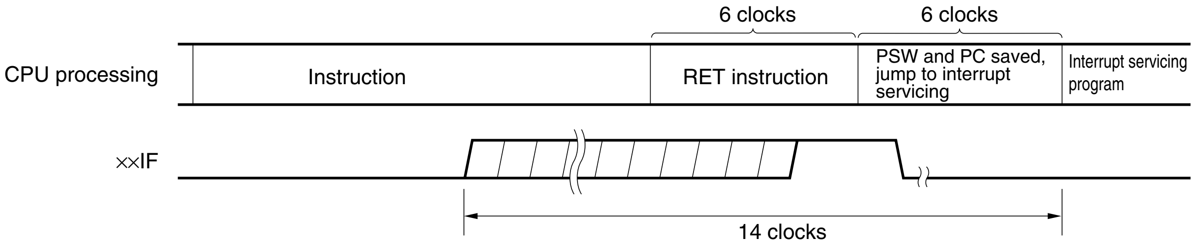

Internal bus
Multiplication input data register A (MULA)
Multiplication input data register B (MULB)
32-bit multiplier
16-bit higher multiplication result storage register (MULOH)
16-bit lower multiplication result storage register (MULOL)
Internal bus

Figure 15-8. Interrupt Request Acknowledgment Timing (Minimum Time)


6 clocks
CPU processing
Instruction
Instruction
PSW and PC saved, jump to interrupt servicing
Interrupt servicing program
××IF
9 clocks

Remark 1 clock: 1/f_{CLK} (f_{CLK}: CPU clock)

Figure 15-9. Interrupt Request Acknowledgment Timing (Maximum Time)


6 clocks
6 clocks
CPU processing
Instruction
RET instruction
PSW and PC saved, jump to interrupt servicing
Interrupt servicing program
××IF
14 clocks

Remark 1 clock: 1/f_{CLK} (f_{CLK}: CPU clock)

15.4.2 Software interrupt request acknowledgment

A software interrupt acknowledge is acknowledged by BRK instruction execution. Software interrupts cannot be disabled.

If a software interrupt request is acknowledged, the contents are saved into the stacks in the order of the program status word (PSW), then program counter (PC), the IE flag is reset (0), and the contents of the vector table (0007EH, 0007FH) are loaded into the PC and branched.

Restoring from a software interrupt is possible by using the RETB instruction.

Caution Do not use the RETI instruction for restoring from the software interrupt.

CHAPTER 17 STANDBY FUNCTION

17.1 Standby Function and Configuration

17.1.1 Standby function

The standby function reduces the operating current of the system, and the following two modes are available.

(1) HALT mode

HALT instruction execution sets the HALT mode. In the HALT mode, the CPU operation clock is stopped. If the high-speed system clock oscillator, internal high-speed oscillator, or subsystem clock oscillator is operating before the HALT mode is set, oscillation of each clock continues. In this mode, the operating current is not decreased as much as in the STOP mode, but the HALT mode is effective for restarting operation immediately upon interrupt request generation and carrying out intermittent operations frequently.

(2) STOP mode

STOP instruction execution sets the STOP mode. In the STOP mode, the high-speed system clock oscillator and internal high-speed oscillator stop, stopping the whole system, thereby considerably reducing the CPU operating current.

Because this mode can be cleared by an interrupt request, it enables intermittent operations to be carried out. However, because a wait time is required to secure the oscillation stabilization time after the STOP mode is released when the X1 clock is selected, select the HALT mode if it is necessary to start processing immediately upon interrupt request generation.

In either of these two modes, all the contents of registers, flags and data memory just before the standby mode is set are held. The I/O port output latches and output buffer statuses are also held.

Cautions 1. The STOP mode can be used only when the CPU is operating on the main system clock. The STOP mode cannot be set while the CPU operates with the subsystem clock. The HALT mode can be used when the CPU is operating on either the main system clock or the subsystem clock.

2. When shifting to the STOP mode, be sure to stop the peripheral hardware operation operating with main system clock before executing STOP instruction.

3. The following sequence is recommended for operating current reduction of the A/D converter when the standby function is used: First clear bit 7 (ADCS) and bit 0 (ADCE) of the A/D converter mode register (ADM) to 0 to stop the A/D conversion operation, and then execute the STOP instruction.

4. It can be selected by the option byte whether the internal low-speed oscillator continues oscillating or stops in the HALT or STOP mode. For details, see CHAPTER 22 OPTION BYTE.

17.1.2 Registers controlling standby function

The standby function is controlled by the following two registers.

- Oscillation stabilization time counter status register (OSTC)
- Oscillation stabilization time select register (OSTS)

Remark For the registers that start, stop, or select the clock, see **CHAPTER 5 CLOCK GENERATOR**.

Table 26-5. Operation List (17/17)

	Instruction Group	Mnemonic	Operands	Bytes	Clocks		Operation	Flag		
					Note 1	Note 2		Z	AC	CY
	Conditional branch	BF	saddr.bit, $addr20	4	3/5$^{\text{Note 3}}$	–	PC ← PC + 4 + jdisp8 if (saddr).bit = 0			
			sfr.bit, $addr20	4	3/5$^{\text{Note 3}}$	–	PC ← PC + 4 + jdisp8 if sfr.bit = 0			
			A.bit, $addr20	3	3/5$^{\text{Note 3}}$	–	PC ← PC + 3 + jdisp8 if A.bit = 0			
			PSW.bit, $addr20	4	3/5$^{\text{Note 3}}$	–	PC ← PC + 4 + jdisp8 if PSW.bit = 0			
<R>			[HL].bit, $addr20	3	3/5$^{\text{Note 3}}$	6/7	PC ← PC + 3 + jdisp8 if (HL).bit = 0			
<R>			ES:[HL].bit, $addr20	4	4/6$^{\text{Note 3}}$	7/8	PC ← PC + 4 + jdisp8 if (ES, HL).bit = 0			
		BTCLR	saddr.bit, $addr20	4	3/5$^{\text{Note 3}}$	–	PC ← PC + 4 + jdisp8 if (saddr).bit = 1 then reset (saddr).bit			
			sfr.bit, $addr20	4	3/5$^{\text{Note 3}}$	–	PC ← PC + 4 + jdisp8 if sfr.bit = 1 then reset sfr.bit			
			A.bit, $addr20	3	3/5$^{\text{Note 3}}$	–	PC ← PC + 3 + jdisp8 if A.bit = 1 then reset A.bit			
			PSW.bit, $addr20	4	3/5$^{\text{Note 3}}$	–	PC ← PC + 4 + jdisp8 if PSW.bit = 1 then reset PSW.bit	×	×	×
			[HL].bit, $addr20	3	3/5$^{\text{Note 3}}$	–	PC ← PC + 3 + jdisp8 if (HL).bit = 1 then reset (HL).bit			
			ES:[HL].bit, $addr20	4	4/6$^{\text{Note 3}}$	–	PC ← PC + 4 + jdisp8 if (ES, HL).bit = 1 then reset (ES, HL).bit			
	Conditional skip	SKC	–	2	1	–	Next instruction skip if CY = 1			
		SKNC	–	2	1	–	Next instruction skip if CY = 0			
		SKZ	–	2	1	–	Next instruction skip if Z = 1			
		SKNZ	–	2	1	–	Next instruction skip if Z = 0			
		SKH	–	2	1	–	Next instruction skip if (Z ∨ CY) = 0			
		SKNH	–	2	1	–	Next instruction skip if (Z ∨ CY) = 1			
	CPU control	SEL	RBn	2	1	–	RBS[1:0] ← n			
		NOP	–	1	1	–	No Operation			
		EI	–	3	4	–	IE ← 1(Enable Interrupt)			
		DI	–	3	4	–	IE ← 0(Disable Interrupt)			
		HALT	–	2	3	–	Set HALT Mode			
		STOP	–	2	3	–	Set STOP Mode			

Notes 1. When the internal RAM area, SFR area, or extended SFR area is accessed, or for an instruction with no data access.

2. When the program memory area is accessed.

3. This indicates the number of clocks “when condition is not met/when condition is met”.

Remarks 1. One instruction clock cycle is one cycle of the CPU clock (f_{CLK}) selected by the system clock control register (CKC).

2. This number of clocks is for when the program is in the internal ROM (flash memory) area. When fetching an instruction from the internal RAM area, the number of clocks is twice the number of clocks plus 3, maximum.

3. n indicates the number of register banks (n = 0 to 3)

DC Characteristics (7/10)

(T_A = –40 to +85°C, 1.8 V ≤ V_{DD} = EV_{DD} ≤ 5.5 V, 1.8 V ≤ AV_{REF} ≤ V_{DD}, V_{SS} = EV_{SS} = AV_{SS} = 0 V)

Parameter	Symbol	Conditions				MIN.	TYP.	MAX.	Unit
Supply current	I_{DD1}[Note 1]	Operating mode	f_{MX} = 20 MHz[Note 2], V_{DD} = 5.0 V		Square wave input		7.0	12.2	mA
					Resonator connection		7.3	12.5	mA
			f_{MX} = 20 MHz[Note 2], V_{DD} = 3.0 V		Square wave input		7.0	12.2	mA
					Resonator connection		7.3	12.5	mA
			f_{MX} = 10 MHz[Notes 2, 3], V_{DD} = 5.0 V		Square wave input		3.8	6.2	mA
					Resonator connection		3.9	6.3	mA
			f_{MX} = 10 MHz[Notes 2, 3], V_{DD} = 3.0 V		Square wave input		3.8	6.2	mA
					Resonator connection		3.9	6.3	mA
			f_{MX} = 5 MHz[Notes 2, 3], V_{DD} = 3.0 V	Normal current mode	Square wave input		2.1	3.0	mA
					Resonator connection		2.2	3.1	mA
				Low consumption current mode[Note 4]	Square wave input		1.5	2.1	mA
					Resonator connection		1.5	2.1	mA
			f_{MX} = 5 MHz[Notes 2, 3], V_{DD} = 2.0 V	Normal current mode	Square wave input		1.4	2.1	mA
					Resonator connection		1.4	2.1	mA
				Low consumption current mode[Note 4]	Square wave input		1.4	2.0	mA
					Resonator connection		1.4	2.0	mA
			f_{IH} = 8 MHz[Note 5]		V_{DD} = 5.0 V		3.1	5.0	mA
					V_{DD} = 3.0 V		3.1	5.0	mA
			f_{SUB} = 32.768 kHz[Note 6], T_A = –40 to +70 °C		V_{DD} = 5.0 V		6.4	24.0	μA
					V_{DD} = 3.0 V		6.4	24.0	μA
					V_{DD} = 2.0 V		6.3	21.0	μA
			f_{SUB} = 32.768 kHz[Note 6], T_A = –40 to +85 °C		V_{DD} = 5.0 V		6.4	31.0	μA
					V_{DD} = 3.0 V		6.4	31.0	μA
					V_{DD} = 2.0 V		6.3	28.0	μA

Notes 1. Total current flowing into V_{DD}, EV_{DD}, and AV_{REF}, including the input leakage current flowing when the level of the input pin is fixed to V_{DD} or V_{SS}. The values below the MAX. column include the peripheral operation current. However, not including the current flowing into the A/D converter, LVI circuit, I/O port, and on-chip pull-up/pull-down resistors.

2. When internal high-speed oscillator and subsystem clock are stopped.

3. When AMPH (bit 0 of clock operation mode control register (CMC)) = 0 and FSEL (bit 0 of operation speed mode control register (OSMC)) = 0.

4. When the RMC register is set to 5AH.

5. When high-speed system clock and subsystem clock are stopped. When FSEL (bit 0 of operation speed mode control register (OSMC)) = 0 is set.

6. When internal high-speed oscillator and high-speed system clock are stopped. When watchdog timer is stopped.

Remarks 1. f_{MX}: High-speed system clock frequency (X1 clock oscillation frequency or external main system clock frequency)

f_{IH}: Internal high-speed oscillation clock frequency

f_{SUB}: Subsystem clock frequency (XT1 clock oscillation frequency)

2. For details on the normal current mode and low consumption current mode according to the regulator output voltage, refer to **CHAPTER 21 REGULATOR**.

<R> **3.** Temperature condition of the TYP. value is T_A = 25°C

(2) Serial interface: Serial array unit (5/18)
(T_A = –40 to +85°C, 1.8 V ≤ V_{DD} = EV_{DD} ≤ 5.5 V, V_{SS} = EV_{SS} = AV_{SS} = 0 V)

(d) During communication at same potential (simplified I^2C mode)

Parameter	Symbol	Conditions	MIN.	MAX.	Unit
<R> SCL10 clock frequency	f_{SCL}	2.7 V ≤ V_{DD} ≤ 5.5 V, C_b = 100 pF, R_b = 3 kΩ		400 Note	kHz
		1.8 V ≤ V_{DD} < 2.7 V C_b = 100 pF, R_b = 5 kΩ		300 Note	kHz
Hold time when SCL10 = “L”	t_{LOW}	2.7 V ≤ V_{DD} ≤ 5.5 V, C_b = 100 pF, R_b = 3 kΩ	995		ns
		1.8 V ≤ V_{DD} < 2.7 V C_b = 100 pF, R_b = 5 kΩ	1500		ns
Hold time when SCL10 = “H”	t_{HIGH}	2.7 V ≤ V_{DD} ≤ 5.5 V, C_b = 100 pF, R_b = 3 kΩ	995		ns
		1.8 V ≤ V_{DD} < 2.7 V C_b = 100 pF, R_b = 5 kΩ	1500		ns
Data setup time (reception)	$t_{SU:DAT}$	2.7 V ≤ V_{DD} ≤ 5.5 V, C_b = 100 pF, R_b = 3 kΩ	$1/f_{MCK}$ + 120		ns
		1.8 V ≤ V_{DD} < 2.7 V C_b = 100 pF, R_b = 5 kΩ	$1/f_{MCK}$ + 230		ns
Data hold time (transmission)	$t_{HD:DAT}$	2.7 V ≤ V_{DD} ≤ 5.5 V, C_b = 100 pF, R_b = 3 kΩ	0	160	ns
		1.8 V ≤ V_{DD} < 2.7 V C_b = 100 pF, R_b = 5 kΩ	0	210	ns

<R> **Note** The value must also be f_{MCK}/4 or less.

(Remarks are given on the next page.)

A.1 Software Package

SP78K0R 78K0R Series software package	Development tools (software) common to the 78K0R microcontrollers are combined in this package.
	Part number: μS××××SP78K0R

Remark ×××× in the part number differs depending on the host machine and OS used.

μS××××SP78K0R

××××	Host Machine	OS	Supply Medium
AB17	PC-9800 series, IBM PC/AT compatibles	Windows (Japanese version)	CD-ROM
BB17		Windows (English version)	

A.2 Language Processing Software

RA78K0R Assembler package	This assembler converts programs written in mnemonics into object codes executable with a microcontroller. This assembler is also provided with functions capable of automatically creating symbol tables and branch instruction optimization. This assembler should be used in combination with a device file (DF781188). **<Precaution when using RA78K0R in PC environment>** This assembler package is a DOS-based application. It can also be used in Windows, however, by using the Project Manager (included in assembler package) on Windows.
	Part number: μS××××RA78K0R
CC78K0R C compiler package	This compiler converts programs written in C language into object codes executable with a microcontroller. This compiler should be used in combination with an assembler package and device file (both sold separately). **<Precaution when using CC78K0R in PC environment>** This C compiler package is a DOS-based application. It can also be used in Windows, however, by using the Project Manager (included in assembler package) on Windows.
	Part number: μS××××CC78K0R
DF781188[Note] Device file	This file contains information peculiar to the device. This device file should be used in combination with a tool (RA78K0R, CC78K0R, SM+ for 78K0R, and ID78K0R-QB) (all sold separately). The corresponding OS and host machine differ depending on the tool to be used.
	Part number: μS××××DF781188

Note The DF781188 can be used in common with the RA78K0R, CC78K0R, SM+ for 78K0R, and ID78K0R-QB. Download the DF781188 from the download site for development tools (http://www.necel.com/micro/ods/eng/index.html).

(29/33)

Chapter	Classification	Function	Details of Function	Cautions	Page
Chapter 27	Hard	Electrical specifications (standard products)	Recommended oscillator constants	The oscillator constants shown above are reference values based on evaluation in a specific environment by the resonator manufacturer. If it is necessary to optimize the oscillator characteristics in the actual application, apply to the resonator manufacturer for evaluation on the implementation circuit. When doing so, check the conditions for using the RMC register, and whether to enter or exit the STOP mode. The oscillation voltage and oscillation frequency only indicate the oscillator characteristic. Use the 78K0R/KE3 so that the internal operation conditions are within the specifications of the DC and AC characteristics.	p.714 □
			DC characteristics	P02 to P04 do not output high level in N-ch open-drain mode.	p.715 □
				The maximum value of V_{IH} of pins P02 to P04 is V_{DD}, even in the N-ch open-drain mode.	p.717 □
				For P122/EXCLK, the value of V_{IH} and V_{IL} differs according to the input port mode or external clock mode. Make sure to satisfy the DC characteristics of EXCLK in external clock input mode.	p.717 □
	Soft		During communication at same potential (UART mode) (dedicated baud rate generator output)	When using UART1, select the normal input buffer for RxD1 and the normal output mode for TxD1 by using the PIM0 and POM0 registers.	p.731 □
			During communication at same potential (CSI mode) (master mode, $\overline{\text{SCKp}}$... internal clock output)	When using CSI10, select the normal input buffer for SI10 and the normal output mode for SO10 and $\overline{\text{SCK10}}$ by using the PIM0 and POM0 registers.	p.732 □
			During communication at same potential (CSI mode) (slave mode, $\overline{\text{SCKp}}$... external clock input)	When using CSI10, select the normal input buffer for SI10 and $\overline{\text{SCK10}}$ and the normal output mode for SO10 by using the PIM0 and POM0 registers.	p.733 □
			During communication at same potential (simplified I^{2}C mode)	Select the normal input buffer and the N-ch open drain output (V_{DD} tolerance) mode for SDA10 and the normal output mode for SCL10 by using the PIM0 and POM0 registers.	p.736 □
			During communication at different potential (2.5 V, 3 V) (UART mode) (dedicated baud rate generator output)	Select the TTL input buffer for RxD1 and the N-ch open drain output (V_{DD} tolerance) mode for TxD1 by using the PIM0 and POM0 registers.	pp.737, 738, 740 □