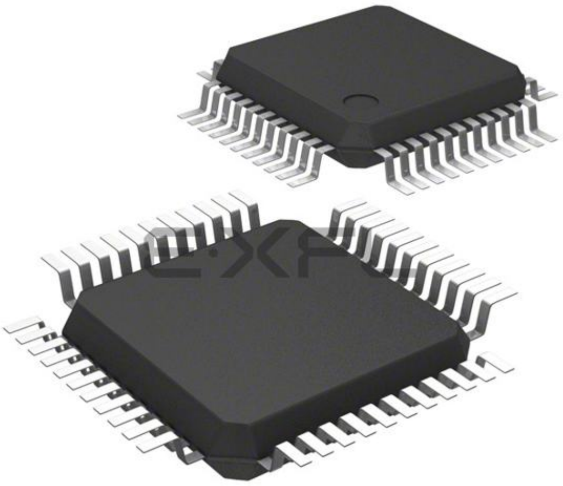


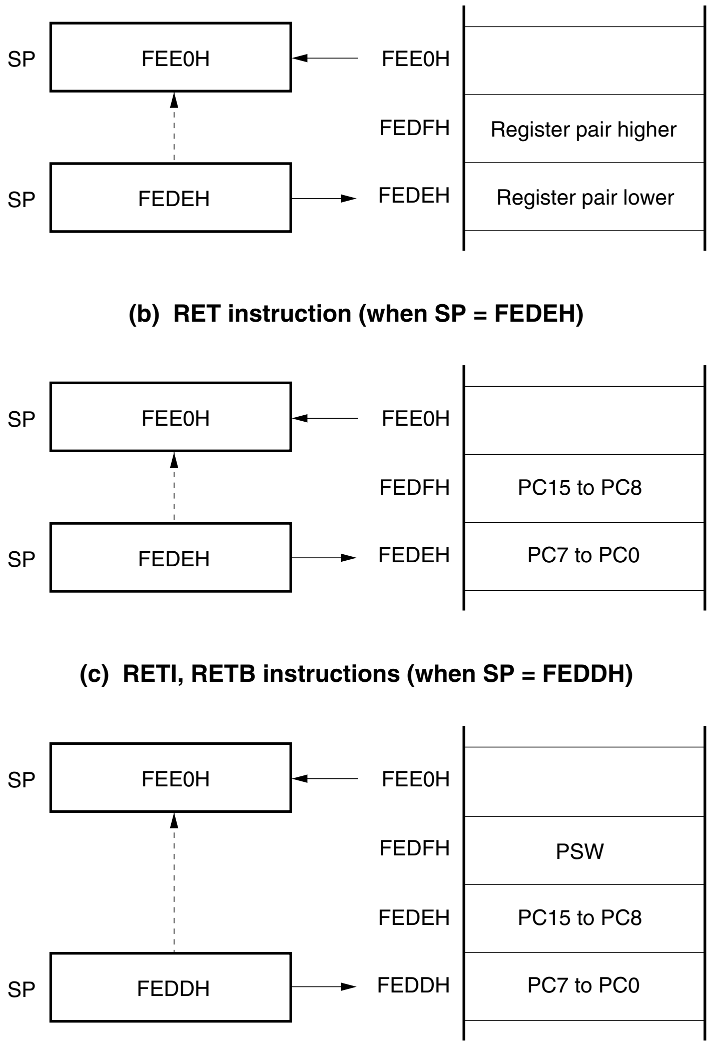
What is "Embedded - Microcontrollers"?

"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/0
Core Size	8-Bit
Speed	10MHz
Connectivity	3-Wire SIO, I²C, LINbus, UART/USART
Peripherals	LVD, POR, PWM, WDT
Number of I/O	33
Program Memory Size	32KB (32K x 8)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	1K x 8
Voltage - Supply (Vcc/Vdd)	1.8V ~ 5.5V
Data Converters	A/D 11x10b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	44-LQFP
Supplier Device Package	-
Purchase URL	https://www.e-xfl.com/product-detail/renesas-electronics-america/upd78f0583gb-gaf-ax

Figure 3-13. Data to Be Restored from Stack Memory



(a) POP rp instruction (when SP = FEDEH)

(b) RET instruction (when SP = FEDEH)

(c) RETI, RETB instructions (when SP = FEDDH)

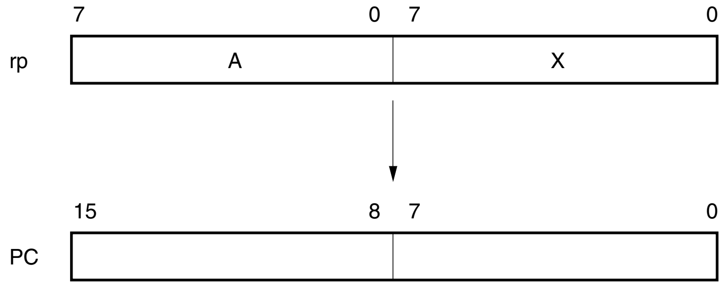
3.3.4 Register addressing

[Function]

Register pair (AX) contents to be specified with an instruction word are transferred to the program counter (PC) and branched.

This function is carried out when the BR AX instruction is executed.

[Illustration]



3.4 Operand Address Addressing

The following methods are available to specify the register and memory (addressing) to undergo manipulation during instruction execution.

3.4.1 Implied addressing

[Function]

The register that functions as an accumulator (A and AX) among the general-purpose registers is automatically (implicitly) addressed.

Of the 78K0/Kx2-L microcontroller instruction words, the following instructions employ implied addressing.

Instruction	Register to Be Specified by Implied Addressing
MULU	A register for multiplicand and AX register for product storage
DIVUW	AX register for dividend and quotient storage
ADJBA/ADJBS	A register for storage of numeric values that become decimal correction targets
ROR4/ROL4	A register for storage of digit data that undergoes digit rotation

[Operand format]

Because implied addressing can be automatically determined with an instruction, no particular operand format is necessary.

[Description example]

In the case of MULU X

With an 8-bit × 8-bit multiply instruction, the product of the A register and X register is stored in AX. In this example, the A and AX registers are specified by implied addressing.

3.4.9 Stack addressing

[Function]

The stack area is indirectly addressed with the stack pointer (SP) contents.

This addressing method is automatically employed when the PUSH, POP, subroutine call and return instructions are executed or the register is saved/reset upon generation of an interrupt request.

With stack addressing, only the internal high-speed RAM area can be accessed.

[Description example]

PUSH DE; when saving DE register

Operation code

1	0	1	1	0	1	0	1

[Illustration]

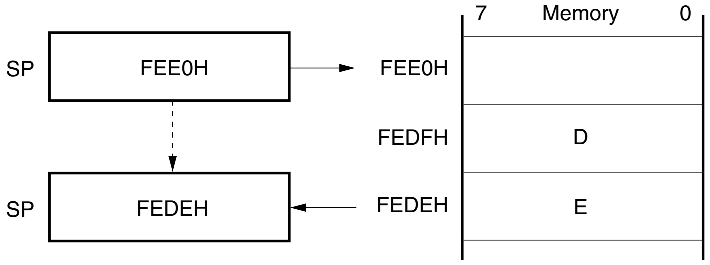


Table 4-9. Setting Functions of P11/ANI9/AMP1OUT Pin

ADPC1 Register	PM1 Register	OPAMP1E bit	ADS Register	P11/ANI9/AMP1OUT Pin
Digital I/O selection	Input mode	0	Selects ANI9.	Setting prohibited
			Does not select ANI9.	Digital input
		1	–	Setting prohibited
	Output mode	0	Selects ANI9.	Setting prohibited
			Does not select ANI9.	Digital output
		1	–	Setting prohibited
Analog I/O selection	Input mode	0	Selects ANI9.	Analog input (to be converted into digital signals)
			Does not select ANI9.	Analog input (not to be converted into digital signals)
		1	Selects ANI9.	Operational amplifier 1 output (to be converted into digital signals)
			Does not select ANI9.	Operational amplifier 1 output (not to be converted into digital signals)
	Output mode	–	–	Setting prohibited

Remark ADPC1: A/D port configuration register 1
PM1: Port mode register 1
OPAMP1E: Bit 7 of operational amplifier 1 control register (AMP1M)
ADS: Analog input channel specification register

Reset signal generation sets port 1 to digital input.
Figures 4-4 to 4-10 show block diagrams of port 1.

Cautions 1. To use P10/$\overline{\text{SCK10}}$ and P12/SO10 of 78K0/KB2-L and 78K0/KC2-L as general-purpose ports, set serial operation mode register 10 (CSIM10) and serial clock selection register 10 (CSIC10) to the default status (00H).

2. To use P13/TxD6 of 78K0/KB2-L and 78K0/KC2-L as general-purpose port, clear bit 0 (TXDLV6) of asynchronous serial interface control register 6 (ASICL6) to 0 (normal output of TxD6).

<R>

Figure 4-37. Format of Port Register (78K0/KA2-L (25-pin and 32-pin products))

Symbol	7	6	5	4	3	2	1	0	Address	After reset	R/W
P0	0	0	0	0	0	P02	P01[Note 2]	P00[Note 1]	FF00H	00H (output latch)	R/W
P2	P27[Note 2, 3]	P26[Note 3]	P25[Note 3]	P24[Note 3]	P23[Note 3]	P22[Note 3]	P21[Note 3]	P20[Note 3]	FF02H	00H (output latch)	R/W
P3	P37	P36	P35	P34	P33	P32	P31	0	FF00H	00H (output latch)	R/W
P6	0	0	0	0	0	0	P61	P60	FF00H	00H (output latch)	R/W
P7[Note 2]	0	0	0	0	0	P72[Note 2, 3]	P71[Note 2, 3]	P70[Note 2, 3]	FF07H	00H (output latch)	R/W
P12	0	0	P125	0	0	P122[Note 4]	P121[Note 4]	0	FF0CH	00H	R

Pmn	m = 0, 2, 3, 6, 7, 12; n = 0 to 7	
	Output data control (in output mode)	Input data read (in input mode)
0	Output 0	Input low level
1	Output 1	Input high level

Notes 1. 25-pin products only

2. 32-pin products only

3. If this pin is set as an analog input and to input mode, do not access the output latch.

4. “0” is always read from the output latch of the pin in the X1 oscillation mode or external clock input mode.

7.4.3 Square-wave output operation

A square wave with any selected frequency is output at intervals determined by the value preset to 8-bit timer compare register 5n (CR5n).

The TO5n pin output status is inverted at intervals determined by the count value preset to CR5n by setting bit 0 (TOE5n) of 8-bit timer mode control register 5n (TMC5n) to 1. This enables a square wave with any selected frequency to be output (duty = 50%).

Remark Square-wave output is operable only in the 78K0/KB2-L and 78K0/KC2-L.

Setting

<1> Set each register.

- Clear the port output latch (P17 or P33)[Note] and port mode register (PM17 or PM33)[Note] to 0.
- TCL5n: Select the count clock.
- CR5n: Compare value
- TMC5n: Stop the count operation, select the mode in which clear & start occurs on a match of TM5n and CR5n.

LVS5n	LVR5n	Timer Output F/F Status Setting
0	1	Timer output F/F clear (0) (default value of TO5n output: low level)
1	0	Timer output F/F set (1) (default value of TO5n output: high level)

Timer output enabled
(TMC5n = 00001011B or 00000111B)

<2> After TCE5n = 1 is set, the count operation starts.

<3> The timer output F/F is inverted by a match of TM5n and CR5n. After INTTM5n is generated, TM5n is cleared to 00H.

<4> After these settings, the timer output F/F is inverted at the same interval and a square wave is output from TO5n. The frequency is as follows.

- Frequency = 1/2t (N + 1)
  (N: 00H to FFH)

Note 8-bit timer/event counter 50: P17, PM17
8-bit timer/event counter 51: P33, PM33

Caution Do not write other values to CR5n during operation.

Remarks 1. For how to enable the INTTM5n signal interrupt, refer to **CHAPTER 17 INTERRUPT FUNCTIONS**.

2. 78K0/KB2-L, 78K0/KC2-L: n = 0, 1

8.4 Operation of 8-Bit Timers H0 and H1

8.4.1 Operation as interval timer/square-wave output

When the 8-bit timer counter Hn and compare register 0n (CMP0n) match, an interrupt request signal (INTTMHn) is generated and the 8-bit timer counter Hn is cleared to 00H.

Compare register 1n (CMP1n) is not used in interval timer mode. Since a match of the 8-bit timer counter Hn and the CMP1n register is not detected even if the CMP1n register is set, timer output is not affected.

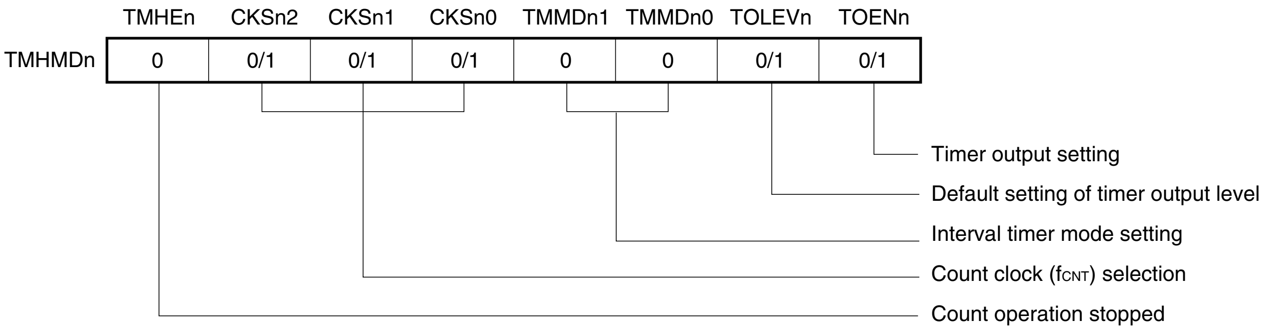
By setting bit 0 (TOENn) of timer H mode register n (TMHMDn) to 1, a square wave of any frequency (duty = 50%) is output from TOHn.

Setting

<1> Set each register.

Figure 8-12. Register Setting During Interval Timer/Square-Wave Output Operation

(i) Setting timer H mode register n (TMHMDn)



(ii) CMP0n register setting

The interval time is as follows if N is set as a comparison value.

- Interval time = (N +1)/f_{CNT}

<2> Count operation starts when TMHEn = 1.

<3> When the values of the 8-bit timer counter Hn and the CMP0n register match, the INTTMHn signal is generated and the 8-bit timer counter Hn is cleared to 00H.

<4> Subsequently, the INTTMHn signal is generated at the same interval. To stop the count operation, clear TMHEn to 0.

Remarks 1. For the setting of the output pin, refer to **8.3 (4) Port mode register 0 (PM0), port mode register 1 (PM1), port mode register 3 (PM3)**.

2. For how to enable the INTTMHn signal interrupt, refer to **CHAPTER 17 INTERRUPT FUNCTIONS**.

3. 78K0/KY2-L, 78K0/KA2-L: n = 1
78K0/KB2-L, 78K0/KC2-L: n = 0, 1

CHAPTER 12 A/D CONVERTER

<R>

Item	78K0/KY2-L (μPD78F055x)	78K0/KA2-L (μPD78F056x)			78K0/KB2-L (μPD78F057x)	78K0/KC2-L (μPD78F058x)		
	16 Pins	20 Pins	25 Pins	32 Pins	30 Pins	40 Pins	44 Pins	48 Pins
10-bit A/D converter	4 ch	6 ch	7 ch	11 ch	7 ch	10 ch	11 ch	11 ch

12.1 Function of A/D Converter

The A/D converter converts an analog input signal into a digital value, and consists of up to 11 channels (ANI0 to ANI10) with a resolution of 10 bits.

In products with operational amplifier, ANI1 function alternately as operational amplifier 0 output (AMP0OUT) and ANI9 function alternately as operational amplifier 1 output (AMP1OUT). This enables using operational amplifiers 0 and 1 output or PGA output as an analog input source.

The A/D converter has the following function.

- **10-bit resolution A/D conversion**
  10-bit resolution A/D conversion is carried out repeatedly for one analog input channel selected from ANI0 to ANI10, operational amplifiers 0 and 1 output, and PGA output. Each time an A/D conversion operation ends, an interrupt request (INTAD) is generated.

12.3 Registers Used in A/D Converter

The A/D converter uses the following seven registers.

- A/D converter mode register 0 (ADM0)
- 10-bit A/D conversion result register (ADCR)
- 8-bit A/D conversion result register L (ADCRL)
- 8-bit A/D conversion result register H (ADCRH)
- Analog input channel specification register (ADS)
- A/D port configuration registers 0, 1 (ADPC0, ADPC1)
- Port mode registers 1, 2, 7 (PM1, PM2, PM7)

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

This register sets the conversion time for analog input to be A/D converted, and starts/stops conversion.
ADM0 can be set by a 1-bit or 8-bit memory manipulation instruction.
Reset signal generation clears this register to 00H.

Figure 12-2. Format of A/D Converter Mode Register 0 (ADM0)

Address: FF28H After reset: 00H R/W

Symbol	<7>	6	5	4	3	2	1	<0>
ADM0	ADCS	0	FR2[Note 1]	FR1[Note 1]	FR0[Note 1]	LV1[Note 1]	LV0[Note 1]	ADCE

ADCS	A/D conversion operation control
0	Stops conversion operation
1	Enables conversion operation

ADCE	A/D voltage comparator operation control[Note 2]
0	Stops A/D voltage comparator operation
1	Enables A/D voltage comparator operation

Notes 1. For details of FR2 to FR0, LV1, LV0, and A/D conversion, refer to **Table 12-2 A/D Conversion Time Selection**.

2. The operation of the A/D voltage comparator is controlled by ADCS and ADCE, and it takes 1 μs from operation start to operation stabilization. Therefore, when ADCS is set to 1 after 1 μs or more has elapsed from the time ADCE is set to 1, the conversion result at that time has priority over the first conversion result. Otherwise, ignore data of the first conversion.

Table 12-1. Settings of ADCS and ADCE

ADCS	ADCE	A/D Conversion Operation
0	0	Stop status (DC power consumption path does not exist)
0	1	Conversion waiting mode (only A/D voltage comparator consumes power)
1	0	Setting prohibited
1	1	Conversion mode (A/D voltage comparator operation)

Table 12-2. A/D Conversion Time Selection (3/3)

(3) 1.8 V ≤ AV_{REF} < 2.7 V

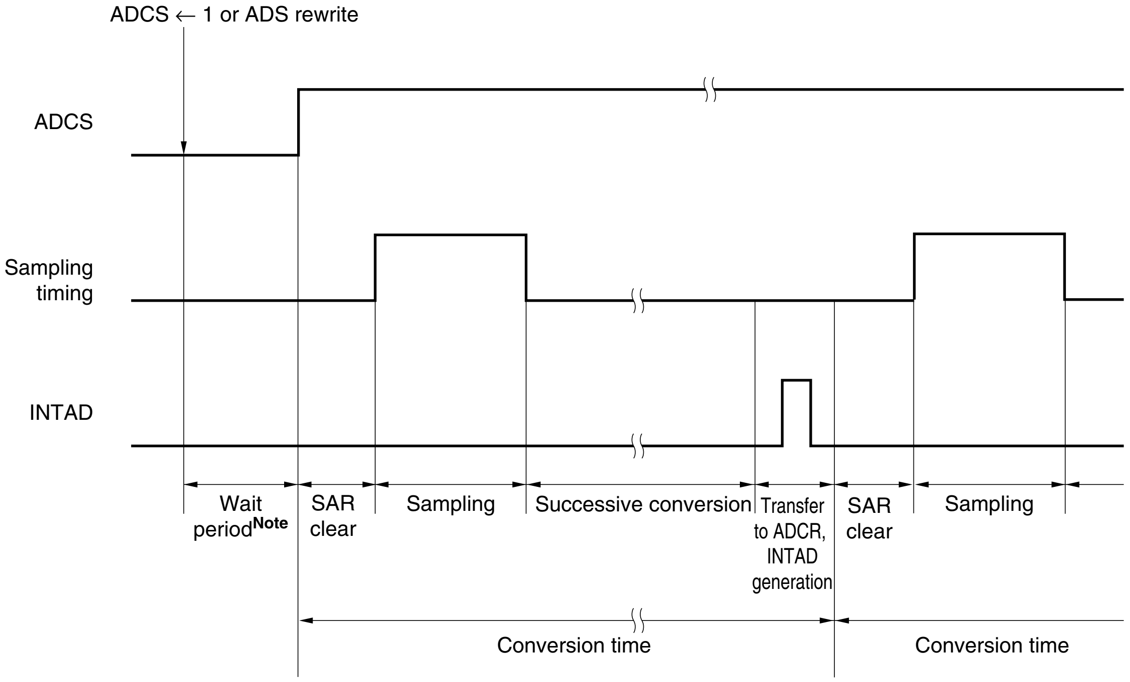
A/D Converter Mode Register 0 (ADM0)					Mode	Conversion Time Selection				Conversion Clock (f_{AD})
FR2	FR1	FR0	LV1	LV0			f_{PRS} = 4 MHz	f_{PRS} = 8 MHz	f_{PRS} = 10 MHz	
0	0	0	0	1	Low-voltage	528/f_{PRS}	Setting prohibited	66.0 μs	52.8 μs	f_{PRS}/12
0	0	1				352/f_{PRS}	Setting prohibited	44.0 μs	Setting prohibited	f_{PRS}/8
0	1	0				264/f_{PRS}	66.0 μs	Setting prohibited		f_{PRS}/6
0	1	1				176/f_{PRS}	44.0 μs	Setting prohibited		f_{PRS}/4
1	0	0				132/f_{PRS}	Setting prohibited			f_{PRS}/3
1	0	1				88/f_{PRS}	Setting prohibited			f_{PRS}/2
1	1	0				66/f_{PRS}	Setting prohibited			f_{PRS}/1.5
1	1	1				44/f_{PRS}	Setting prohibited			f_{PRS}
Other than above					Setting prohibited					

Cautions 1. When rewriting FR2 to FR0, LV1, and LV0 to other than the same data, stop A/D conversion once (ADCS = 0) beforehand.

2. The above conversion time does not include clock frequency errors. Select conversion time, taking clock frequency errors into consideration.

Remark f_{PRS}: Peripheral hardware clock frequency

Figure 12-4. A/D Converter Sampling and A/D Conversion Timing



Note For details of wait period, refer to **CHAPTER 31 CAUTIONS FOR WAIT**.

(2) 10-bit A/D conversion result register (ADCR)

This register is a 16-bit register that stores the A/D conversion result. The higher 6 bits are fixed to 0. Each time A/D conversion ends, the conversion result is loaded from the successive approximation register. The higher 2 bits of the conversion result are stored in FF09H and the lower 8 bits of the conversion result are stored in FF08H.

ADCR can be read by a 16-bit memory manipulation instruction.

Reset signal generation clears this register to 0000H.

Figure 12-5. Format of 10-Bit A/D Conversion Result Register (ADCR)

Address: FF08H, FF09H After reset: 0000H R

Symbol	FF09H								FF08H							
ADCR	0	0	0	0	0	0										

Cautions 1. When writing to the A/D converter mode register 0 (ADM0), analog input channel specification register (ADS), and A/D port configuration registers 0, 1 (ADPC0, ADPC1), the contents of ADCR may become undefined. Read the conversion result following conversion completion before writing to ADM0, ADS, ADPC0, and ADPC1. Using timing other than the above may cause an incorrect conversion result to be read.

2. If data is read from ADCR, a wait cycle is generated. Do not read data from ADCR when the peripheral hardware clock (f_{PRS}) is stopped. For details, refer to CHAPTER 31 CAUTIONS FOR WAIT.

15.5.4 Acknowledge ($\overline{\text{ACK}}$)

$\overline{\text{ACK}}$ is used to check the status of serial data at the transmission and reception sides.

The reception side returns $\overline{\text{ACK}}$ each time it has received 8-bit data.

The transmission side usually receives $\overline{\text{ACK}}$ after transmitting 8-bit data. When $\overline{\text{ACK}}$ is returned from the reception side, it is assumed that reception has been correctly performed and processing is continued. Whether $\overline{\text{ACK}}$ has been detected can be checked by using bit 2 (ACKD0) of the IICA status register 0 (IICAS0).

When the master receives the last data item, it does not return $\overline{\text{ACK}}$ and instead generates a stop condition. If a slave does not return $\overline{\text{ACK}}$ after receiving data, the master outputs a stop condition or restart condition and stops transmission. If $\overline{\text{ACK}}$ is not returned, the possible causes are as follows.

<1> Reception was not performed normally.
<2> The final data item was received.
<3> The reception side specified by the address does not exist.

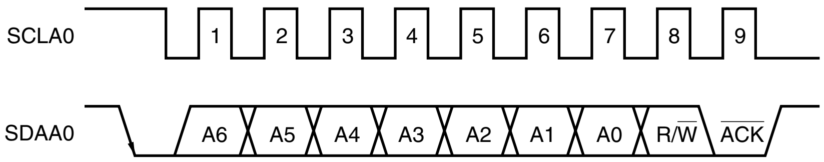
To generate $\overline{\text{ACK}}$, the reception side makes the SDAA0 line low at the ninth clock (indicating normal reception).

Automatic generation of $\overline{\text{ACK}}$ is enabled by setting bit 2 (ACKE0) of IICA control register 0 (IICACTL0) to 1. Bit 3 (TRC0) of the IICAS0 register is set by the data of the eighth bit that follows 7-bit address information. Usually, set ACKE0 to 1 for reception (TRC0 = 0).

If a slave can receive no more data during reception (TRC0 = 0) or does not require the next data item, then the slave must inform the master, by clearing ACKE0 to 0, that it will not receive any more data.

When the master does not require the next data item during reception (TRC0 = 0), it must clear ACKE0 to 0 so that $\overline{\text{ACK}}$ is not generated. In this way, the master informs a slave at the transmission side that it does not require any more data (transmission will be stopped).

Figure 15-19. $\overline{\text{ACK}}$



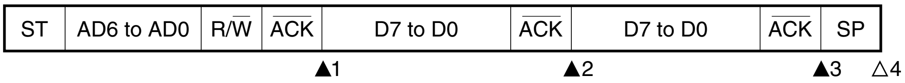
When the local address is received, $\overline{\text{ACK}}$ is automatically generated, regardless of the value of the ACKE0 bit. When an address other than that of the local address is received, $\overline{\text{ACK}}$ is not generated (NACK).

When an extension code is received, $\overline{\text{ACK}}$ is generated if the ACKE0 bit is set to 1 in advance.

How $\overline{\text{ACK}}$ is generated when data is received differs as follows depending on the setting of the wait timing.

- When 8-clock wait state is selected (bit 3 (WTIM0) of IICACTL0 register = 0):
  By setting the ACKE0 bit to 1 before releasing the wait state, $\overline{\text{ACK}}$ is generated at the falling edge of the eighth clock of the SCLA0 pin.
- When 9-clock wait state is selected (bit 3 (WTIM0) of IICACTL0 register = 1):
  $\overline{\text{ACK}}$ is generated by setting the ACKE0 bit to 1 in advance.

(ii) When WTIM0 = 1

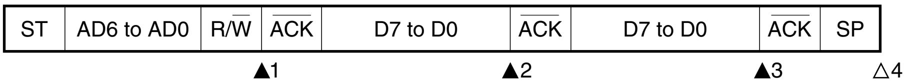


▲1: IICAS0 = 0101×110B
▲2: IICAS0 = 0001×100B
▲3: IICAS0 = 0001××00B
△4: IICAS0 = 00000001B

Remark ▲: Always generated
△: Generated only when SPIE0 = 1
×: Don't care

(b) When arbitration loss occurs during transmission of extension code

(i) When WTIM0 = 0

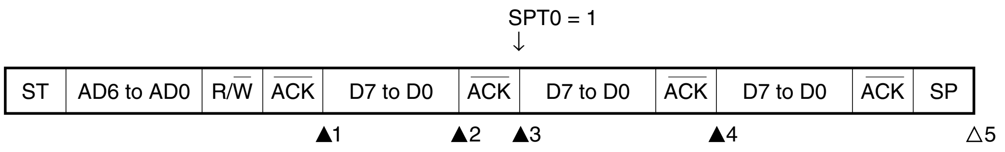


▲1: IICAS0 = 0110×010B
▲2: IICAS0 = 0010×000B
▲3: IICAS0 = 0010×000B
△4: IICAS0 = 00000001B

Remark ▲: Always generated
△: Generated only when SPIE0 = 1
×: Don't care

(h) When arbitration loss occurs due to low-level data when attempting to generate a stop condition

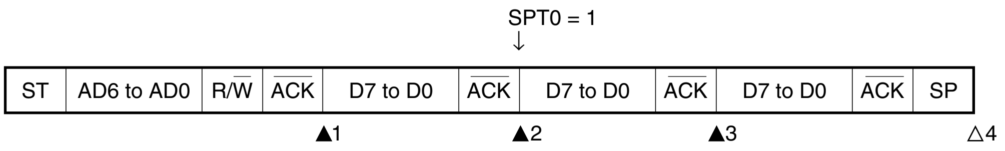
(i) When WTIM0 = 0



▲1: IICAS0 = 1000×110B
▲2: IICAS0 = 1000×000B (Sets WTIM0 to 1)
▲3: IICAS0 = 1000×100B (Clears WTIM0 to 0)
▲4: IICAS0 = 01000100B
△5: IICAS0 = 00000001B

Remark ▲: Always generated
△: Generated only when SPIE0 = 1
×: Don't care

(ii) When WTIM0 = 1



▲1: IICAS0 = 1000×110B
▲2: IICAS0 = 1000×100B (Sets SPT0 to 1)
▲3: IICAS0 = 01000100B
△4: IICAS0 = 00000001B

Remark ▲: Always generated
△: Generated only when SPIE0 = 1
×: Don't care

- Serial operation mode register 11 (CSIM11)

Address: FF88H After reset: 00H R/W

Symbol	<7>	6	5	4	3	2	1	0
CSIM11	CSIE11	TRMD11	SSE11	DIR11	0	0	0	CSOT11

CSIE11	Operation control in 3-wire serial I/O mode
0	Disables operation[Note 1] and asynchronously resets the internal circuit[Note 2].

Notes 1. To use P62/SO11, P60/$\overline{\text{SCK11}}$/SCLA0, and P42/$\overline{\text{SSI11}}$/PCL/INTP6 as general-purpose ports when CSISEL = 0, set CSIM11 in the default status (00H).
To use P120/SO11/INTP0/EXLVI, P40/$\overline{\text{SCK11}}$/RTCCL/RTCDIV, and P42/$\overline{\text{SSI11}}$/PCL/INTP6 as general-purpose ports when CSISEL = 1, set CSIM11 in the default status (00H).
To use P37/SO11, P35/$\overline{\text{SCK11}}$, and P02/$\overline{\text{SSI11}}$/INTP5 as general-purpose ports when CSISEL = 1, set CSIM11 in the default status (00H).

2. Bit 0 (CSOT11) of CSIM11 and serial I/O shift register 11 (SIO11) are reset.

16.4.2 3-wire serial I/O mode

The 3-wire serial I/O mode is used for connecting peripheral ICs and display controllers with a clocked serial interface.

In this mode, communication is executed by using three lines: the serial clock ($\overline{\text{SCK1n}}$), serial output (SO1n), and serial input (SI1n) lines.

(1) Registers used

- Serial operation mode register 1n (CSIM1n)
- Serial clock selection register 1n (CSIC1n)
- Port mode register x (PMx)
- Port register x (Px)

The basic procedure of setting an operation in the 3-wire serial I/O mode is as follows.

<1> Set the CSIC1n register (refer to **Figures 16-6** and **16-7**).
<2> Set bits 4 to 6 (DIR1n, SSE11 (serial interface CSI11 only), and TRMD1n) of the CSIM1n register (refer to **Figures 16-4** and **16-5**).
<3> Set bit 7 (CSIE1n) of the CSIM1n register to 1. → Transmission/reception is enabled.
<4> Write data to transmit buffer register 1n (SOTB1n). → Data transmission/reception is started.
Read data from serial I/O shift register 1n (SIO1n). → Data reception is started.

Caution Take relationship with the other party of communication when setting the port mode register and port register.

Remark

78K0/KA2-L (25, 32-pin products):	n = 0, x = 0, 3
78K0/KB2-L:	n = 0, x = 1
78K0/KC2-L:	n = 0, 1, x = 1, 4, 6, 12

Table 20-2. Hardware Statuses After Reset Acknowledgment (4/4)

Hardware		Status After Reset Acknowledgment[Note 1]
Reset function	Reset control flag register (RESF)	00H[Note 2]
Low-voltage detector	Low-voltage detection register (LVIM)	00H[Note 2]
	Low-voltage detection level selection register (LVIS)	00H[Note 2]
Interrupt	Request flag registers 0L, 0H, 1L, 1H (IF0L, IF0H, IF1L, IF1H)	00H
	Mask flag registers 0L, 0H, 1L, 1H (MK0L, MK0H, MK1L, MK1H)	FFH
	Priority specification flag registers 0L, 0H, 1L, 1H (PR0L, PR0H, PR1L, PR1H)	FFH
	External interrupt rising edge enable registers 0, 1 (EGPCTL0, EGPCTL1)	00H
	External interrupt falling edge enable registers 0, 1 (EGNCTL0, EGNCTL1)	00H
Regulator	Regulator mode control register (RMC)	00H

Notes 1. During reset signal generation or oscillation stabilization time wait, only the PC contents among the hardware statuses become undefined. All other hardware statuses remain unchanged after reset.

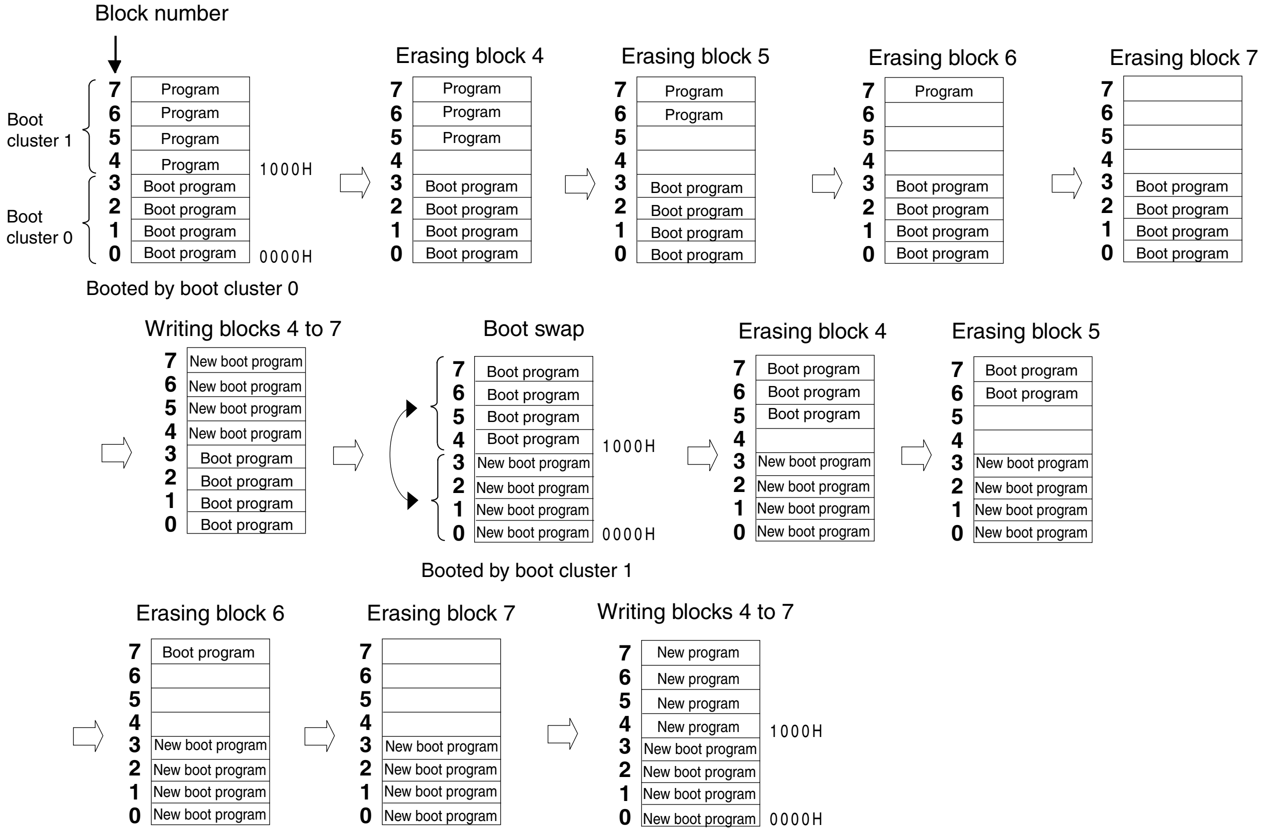
2. These values vary depending on the reset source.

<R>

Register \ Reset Source		$\overline{\text{RESET}}$ Input	Reset by POC	Reset by WDT	Reset by LVI (Except Reset by LVI Default Start Function)	Reset by LVI Default Start Function
RESF	WDTRF flag	Cleared (0)	Cleared (0)	Set (1)	Held	Cleared (0)
	LVIRF flag			Held	Set (1)	
LVIM		Cleared (00H)	Cleared (00H)	Cleared (00H)	Held	Cleared (00H)
LVIS						

Remark The special function registers (SFRs) mounted depend on the product. Refer to **3.2.3 Special function registers (SFRs)**.

Figure 25-11. Example of Executing Boot Swapping



Instruction Group	Mnemonic	Operands	Bytes	Clocks		Operation	Flag		
				Note 1	**Note 2**		Z	AC	CY
16-bit operation	**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	×	×	×
Multiply/ divide	**MULU**	X	2	16	–	AX ← A × X			
	DIVUW	C	2	25	–	AX (Quotient), C (Remainder) ← AX ÷ C			
Increment/ decrement	**INC**	r	1	2	–	r ← r + 1	×	×	
		saddr	2	4	6	(saddr) ← (saddr) + 1	×	×	
	DEC	r	1	2	–	r ← r – 1	×	×	
		saddr	2	4	6	(saddr) ← (saddr) – 1	×	×	
	INCW	rp	1	4	–	rp ← rp + 1			
	DECW	rp	1	4	–	rp ← rp – 1			
Rotate	**ROR**	A, 1	1	2	–	(CY, A_7 ← A_0, A_{m-1} ← A_m) × 1 time			×
	ROL	A, 1	1	2	–	(CY, A_0 ← A_7, A_{m+1} ← A_m) × 1 time			×
	RORC	A, 1	1	2	–	(CY ← A_0, A_7 ← CY, A_{m-1} ← A_m) × 1 time			×
	ROLC	A, 1	1	2	–	(CY ← A_7, A_0 ← CY, A_{m+1} ← A_m) × 1 time			×
	ROR4	[HL]	2	10	12	A_{3-0} ← $(HL)_{3-0}$, $(HL)_{7-4}$ ← A_{3-0}, $(HL)_{3-0}$ ← $(HL)_{7-4}$			
	ROL4	[HL]	2	10	12	A_{3-0} ← $(HL)_{7-4}$, $(HL)_{3-0}$ ← A_{3-0}, $(HL)_{7-4}$ ← $(HL)_{3-0}$			
BCD adjustment	**ADJBA**		2	4	–	Decimal Adjust Accumulator after Addition	×	×	×
	ADJBS		2	4	–	Decimal Adjust Accumulator after Subtract	×	×	×
Bit manipulate	**MOV1**	CY, saddr.bit	3	6	7	CY ← (saddr.bit)			×
		CY, sfr.bit	3	–	7	CY ← sfr.bit			×
		CY, A.bit	2	4	–	CY ← A.bit			×
		CY, PSW.bit	3	–	7	CY ← PSW.bit			×
		CY, [HL].bit	2	6	7	CY ← (HL).bit			×
		saddr.bit, CY	3	6	8	(saddr.bit) ← CY			
		sfr.bit, CY	3	–	8	sfr.bit ← CY			
		A.bit, CY	2	4	–	A.bit ← CY			
		PSW.bit, CY	3	–	8	PSW.bit ← CY	×	×	
		[HL].bit, CY	2	6	8	(HL).bit ← CY			

Notes 1. When the internal high-speed RAM area is accessed or for an instruction with no data access

2. When an area except the internal high-speed RAM area is accessed

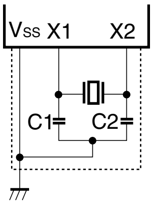
Remarks 1. One instruction clock cycle is one cycle of the CPU clock (f_{CPU}) selected by the processor clock control register (PCC).

2. This clock cycle applies to the internal ROM program.

Caution The pins mounted depend on the product. Refer to Caution 2 at the beginning of this chapter.

X1 Oscillator Characteristics

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

Resonator	Recommended Circuit	Parameter	Conditions	MIN.	TYP.	MAX.	Unit
Ceramic resonator, crystal resonator		X1 clock oscillation frequency (fx)[Note]	2.7 V ≤ V_{DD} ≤ 5.5 V	1.0		10.0	MHz
			1.8 V ≤ V_{DD} < 2.7 V	1.0		5.0	

Note Indicates only oscillator characteristics. Refer to **AC Characteristics** for instruction execution time.

Cautions 1. When using the X1 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.**

2. Since the CPU is started by the internal high-speed oscillation clock after a reset release, check the X1 clock oscillation stabilization time using the oscillation stabilization time counter status register (OSTC) by the user. Determine the oscillation stabilization time of the OSTC register and oscillation stabilization time select register (OSTS) after sufficiently evaluating the oscillation stabilization time with the resonator to be used.

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.

Caution The pins mounted depend on the product. Refer to Caution 2 at the beginning of this chapter.

(4) Operational amplifier 1

(T_A = –40 to +85°C, 2.2 V ≤ V_{DD} ≤ 5.5 V, V_{SS} = AV_{SS} = 0 V, Output load: R_L = 47 kΩ, C_L = 50 pF)

Parameter	Symbol	Conditions	MIN.	TYP.	MAX.	Unit
Input offset voltage	V_{IOP1}	V_{DD} = 3.0 V			±10	mV
Power supply voltage rejection ratio	$PSRR_{OP1}$	V_{DD} = 3.0 V		70		dB
Output voltage, high	V_{OHOP1}	V_{DD} = 3.0 V/2.2 V, I_{OH} = –500 μA	V_{DD} – 0.2			V
Output voltage, low	V_{OLOP1}	V_{DD} = 3.0 V/2.2 V, I_{OL} = 500 μA			0.1	V
Common-mode input voltage	V_{ICMOP1}	V_{DD} = 3.0 V/2.2 V	0		V_{DD} – 0.6	V
Slew rate	SR_{OP1}	V_{DD} = 3.0 V		1.8		V/μs
		V_{DD} = 5.0 V		2.0		V/μs
Input noise spectral density (Inoise)		V_{DD} = 3.0 V, V_{IN} = 0.1 V, f = 1 kHz		73		nV / √Hz
		V_{DD} = 3.0 V, V_{IN} = V_{DD}/2 V, f = 1 kHz		60		
		V_{DD} = 3.0 V, V_{IN} = V_{DD}–0.6 V, f = 1 kHz		55		
Phase margin		V_{DD} = 3.0 V		40		deg
Large-amplitude voltage gain	AV_{OP1}	V_{DD} = 3.0 V		100		dB
Gain-bandwidth product	GBW_{OP1}	V_{DD} = 5.0 V/3.0 V/2.2 V		3.0		MHz
Operation stabilization wait time[Note]	t_{OP1}	V_{DD} = 3.0 V		10		μs

Note Time required until a state is entered where the DC and AC specifications of the operational amplifier 1 are satisfied after the operational amplifier 1 operation has been enabled (OPAMP1E = 1).