Renesas Electronics America Inc - UPD78F9234MC-5A4-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	Obsolete
Core Processor	78K0S
Core Size	8-Bit
Speed	10MHz
Connectivity	LINbus, UART/USART
Peripherals	LVD, POR, PWM, WDT
Number of I/O	24
Program Memory Size	8KB (8K x 8)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	256 x 8
Voltage - Supply (Vcc/Vdd)	2V ~ 5.5V
Data Converters	A/D 4x10b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	30-LSSOP (0.240", 6.10mm Width)
Supplier Device Package	-
Purchase URL	https://www.e-xfl.com/product-detail/renesas-electronics-america/upd78f9234mc-5a4-a

NOTES FOR CMOS DEVICES

(1) VOLTAGE APPLICATION WAVEFORM AT INPUT PIN: Waveform distortion due to input noise or a reflected wave may cause malfunction. If the input of the CMOS device stays in the area between VIL (MAX) and VIH (MIN) due to noise, etc., the device may malfunction. Take care to prevent chattering noise from entering the device when the input level is fixed, and also in the transition period when the input level passes through the area between VIL (MAX) and VIH (MIN).

(2) HANDLING OF UNUSED INPUT PINS: Unconnected CMOS device inputs can be cause of malfunction. If an input pin is unconnected, it is possible that an internal input level may be generated due to noise, etc., causing malfunction. CMOS devices behave differently than Bipolar or NMOS devices. Input levels of CMOS devices must be fixed high or low by using pull-up or pull-down circuitry. Each unused pin should be connected to VDD or GND via a resistor if there is a possibility that it will be an output pin. All handling related to unused pins must be judged separately for each device and according to related specifications governing the device.

(3) PRECAUTION AGAINST ESD: A 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 when it has occurred. Environmental control must be adequate. When it is dry, a humidifier should be used. It is recommended to avoid using insulators that easily build up 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 benches and floors should be grounded. The operator should be grounded using a wrist strap. Semiconductor devices must not be touched with bare hands. Similar precautions need to be taken for PW boards with mounted semiconductor devices.

(4) STATUS BEFORE INITIALIZATION: Power-on does not necessarily define the initial status of a MOS device. Immediately after the power source is turned ON, devices with reset functions have not yet been initialized. Hence, power-on does not guarantee output pin levels, I/O settings or contents of registers. A device is not initialized until the reset signal is received. A reset operation must be executed immediately after power-on for devices with reset functions.

(5) POWER ON/OFF SEQUENCE: In the case of a device that uses different power supplies for the internal operation and external interface, as a rule, switch on the external power supply after switching on the internal power supply. When switching the power supply off, as a rule, switch off the external power supply and then the internal power supply. Use of the reverse power on/off sequences may result in the application of an overvoltage to the internal elements of the device, causing malfunction and degradation of internal elements due to the passage of an abnormal current. The correct power on/off sequence must be judged separately for each device and according to related specifications governing the device.

(6) INPUT OF SIGNAL DURING POWER OFF STATE : Do not input signals or an I/O pull-up power supply while the device is not powered. The current injection that results from input of such a signal or I/O pull-up power supply may cause malfunction and the abnormal current that passes in the device at this time may cause degradation of internal elements. Input of signals during the power off state must be judged separately for each device and according to related specifications governing the device.

4.2.3 Port 3

Pins P30 to P33 constitute a 4-bit I/O port with an output latch. Each bit of this port can be set to the input or output mode by using port mode register 3 (PM3). When the P30 to P33 pins are used as an input port, an on-chip pull-up resistor can be connected in 1-bit units by using pull-up resistor option register 3 (PU3).

The P30 and P31 pins are also used for both timer I/O and external interrupt request input pin functions.

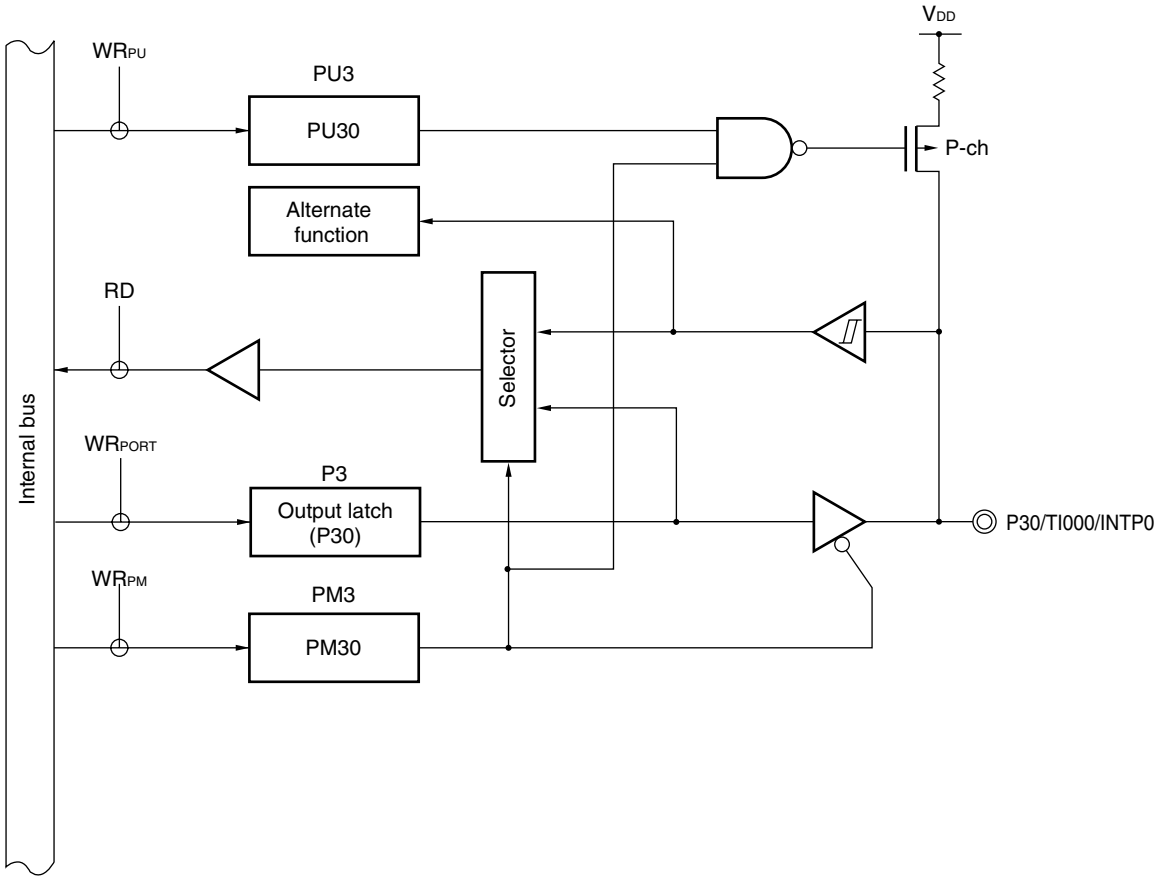
Generation of reset signal sets port 3 to the input mode.

P34 is a 1-bit input-only port. This pin is also used as a $\overline{\text{RESET}}$ pin, and when the power is turned on, this is the reset function.

For settings of alternate function, refer to **CHAPTER 18 OPTION BYTE**. When using P34 as input port, pull up the P34 pin by using external resistor.

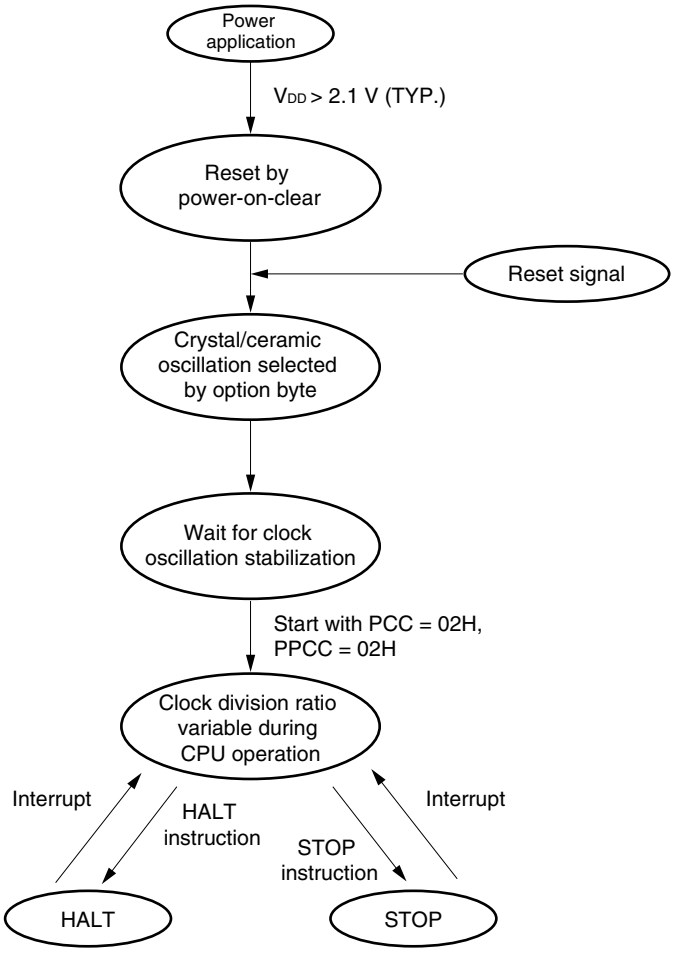
Figures 4-4 to 4-7 show the block diagrams of port 3.

Figure 4-4. Block Diagram of P30



PU3: Pull-up resistor option register 3
P3: Port register 3
PM3: Port mode register 3
RD: Read signal
WR××: Write signal

Figure 5-11. Status Transition of Default Start by Crystal/Ceramic Oscillation



Remark PCC: Processor clock control register
PPCC: Preprocessor clock control register

(3) External clock input circuit

If external clock input is selected by the option byte, the following is possible.

- High-speed operation
  The accuracy of processing is improved as compared with high-speed internal oscillation (8 MHz (TYP.)) because an oscillation frequency of 2 to 10 MHz can be selected and an external clock with a small frequency deviation can be supplied.
- Improvement of expandability
  If the external clock input circuit is selected as the oscillator, the X2 pin can be used as an I/O port pin. For details, refer to **CHAPTER 4 PORT FUNCTIONS**.

Figures 5-12 and 5-13 show the timing chart and status transition diagram of default start by external clock input.

(4) Prescaler mode register 00 (PRM00)

This register is used to set the 16-bit timer counter 00 (TM00) count clock and the TI000, TI010 pin input valid edges.

PRM00 is set by a 1-bit or 8-bit memory manipulation instruction.

Generation of reset signal sets the value of PRM00 to 00H.

Figure 6-8. Format of Prescaler Mode Register 00 (PRM00)

Address: FF61H After reset: 00H R/W

Symbol	7	6	5	4	3	2	1	0
PRM00	ES110	ES100	ES010	ES000	0	0	PRM001	PRM000

ES110	ES100	TI010 pin valid edge selection
0	0	Falling edge
0	1	Rising edge
1	0	Setting prohibited
1	1	Both falling and rising edges

ES010	ES000	TI000 pin valid edge selection
0	0	Falling edge
0	1	Rising edge
1	0	Setting prohibited
1	1	Both falling and rising edges

PRM001	PRM000	Count clock selection
0	0	f_{XP} (10 MHz)
0	1	$f_{XP}/2^2$ (2.5 MHz)
1	0	$f_{XP}/2^8$ (39.06 kHz)
1	1	TI000 pin valid edge[Note]

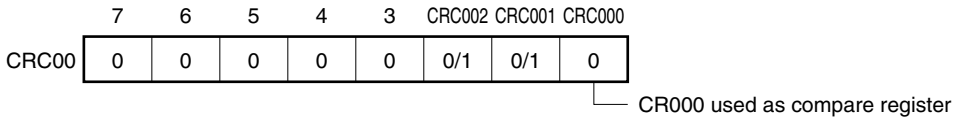
Remarks 1. f_{XP}: Oscillation frequency of clock supplied to peripheral hardware

2. ( ): f_{XP} = 10 MHz

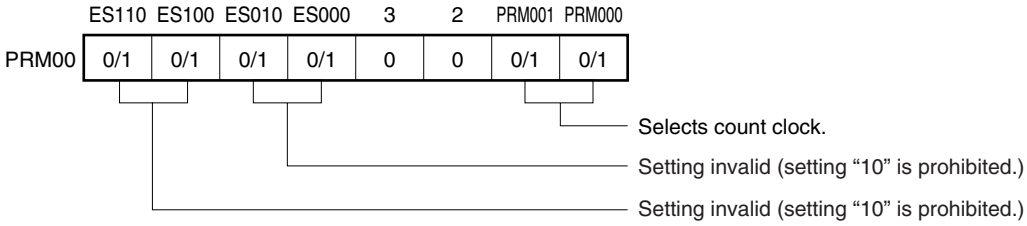
Note The external clock requires a pulse longer than two cycles of the internal count clock (f_{XP}).

Figure 6-10. Control Register Settings for Interval Timer Operation

(a) Capture/compare control register 00 (CRC00)



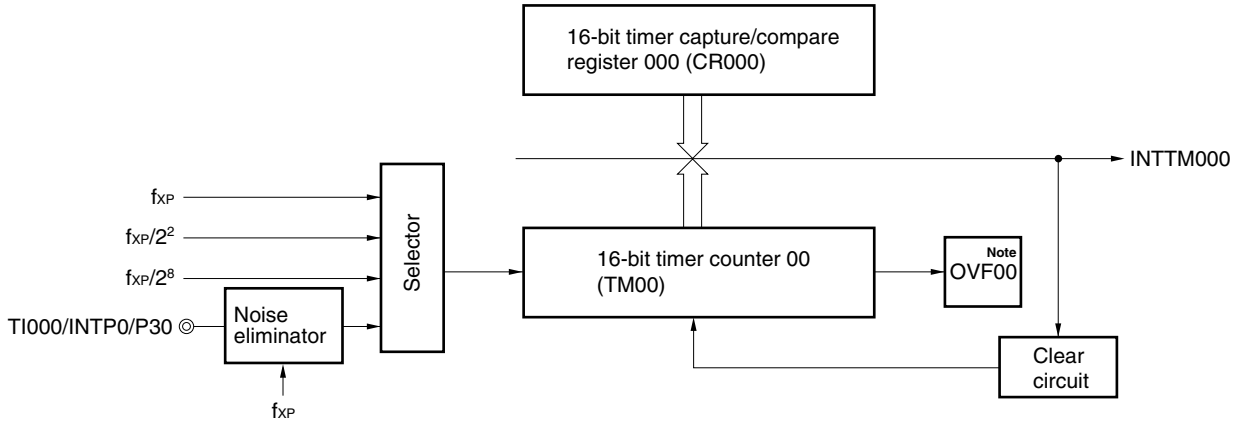
(b) Prescaler mode register 00 (PRM00)



(c) 16-bit timer mode control register 00 (TMC00)

Remark 0/1: Setting 0 or 1 allows another function to be used simultaneously with the interval timer. See the description of the respective control registers for details.

Figure 6-11. Interval Timer Configuration Diagram



Note OVF00 is set to 1 only when 16-bit timer capture/compare register 000 (CR000) is set to FFFFH.

6.4.5 PPG output operations

Setting 16-bit timer mode control register 00 (TMC00) and capture/compare control register 00 (CRC00) as shown in Figure 6-29 allows operation as PPG (Programmable Pulse Generator) output.

Setting

The basic operation setting procedure is as follows.

<1> Set the CRC00 register (see **Figure 6-29** for the set value).
<2> Set any value to the CR000 register as the cycle.
<3> Set any value to the CR010 register as the duty factor.
<4> Set the TOC00 register (see **Figure 6-29** for the set value).
<5> Set the count clock by using the PRM00 register.
<6> Set the TMC00 register to start the operation (see **Figure 6-29** for the set value).

Caution Changing the CRC0n0 setting during TM00 operation may cause a malfunction. To change the setting, refer to 6.5 Cautions Related to 16-bit Timer/Event Counter 00 (17) Changing compare register during timer operation.

Remarks 1. For the setting of the TO00 pin, see **6.3 (5) Port mode register 3 (PM3)**.
2. For how to enable the INTTM000 interrupt, see **CHAPTER 13 INTERRUPT FUNCTIONS**.
3. n = 0 or 1

In the PPG output operation, rectangular waves are output from the TO00 pin with the pulse width and the cycle that correspond to the count values preset in 16-bit timer capture/compare register 010 (CR010) and in 16-bit timer capture/compare register 000 (CR000), respectively.

(18) Edge detection

<1> In the following cases, note with caution that the valid edge of the TI0n0 pin is detected.

(a) Immediately after a system reset, if a high level is input to the TI0n0 pin, the operation of the 16-bit timer counter 00 (TM00) is enabled

→ If the rising edge or both rising and falling edges are specified as the valid edge of the TI0n0 pin, a rising edge is detected immediately after the TM00 operation is enabled.

(b) If the TM00 operation is stopped while the TI0n0 pin is high level, TM00 operation is then enabled after a low level is input to the TI0n0 pin

→ If the falling edge or both rising and falling edges are specified as the valid edge of the TI0n0 pin, a falling edge is detected immediately after the TM00 operation is enabled.

(c) If the TM00 operation is stopped while the TI0n0 pin is low level, TM00 operation is then enabled after a high level is input to the TI0n0 pin

→ If the rising edge or both rising and falling edges are specified as the valid edge, of the TI0n0 pin, a rising edge is detected immediately after the TM00 operation is enabled.

Remark n = 0, 1

<2> The sampling clock used to remove noise differs when a TI000 valid edge is used as the count clock and when it is used as a capture trigger. In the former case, the count clock is f_{XP}, and in the latter case the count clock is selected by prescaler mode register 00 (PRM00). The capture operation is not performed until the valid edge is sampled and the valid level is detected twice, thus eliminating, noise with a short pulse width.

(19) External event counter

<1> The timing of the count start is after two valid edge detections.

<2> When reading the external event counter count value, TM00 should be read.

(20) PPG output

<1> Values in the following range should be set in CR000 and CR010:
0000H < CR010 < CR000 ≤ FFFFH

<2> The cycle of the pulse generated through PPG output (CR000 setting value + 1) has a duty of (CR010 setting value + 1)/(CR000 setting value + 1).

(21) STOP mode or system clock stop mode setting

Except when the valid edge of the TI000 pin is selected as the count clock, stop the timer operation before setting STOP mode or system clock stop mode; otherwise the timer may malfunction when the system clock starts.

(22) P31/TI010/TO00 pin

When using P31 as the input pin (TI010) of the valid edge, it cannot be used as a timer output pin (TO00). When using P31 as the timer output pin (TO00), it cannot be used as the input pin (TI010) of the valid edge.

(2) Timing chart

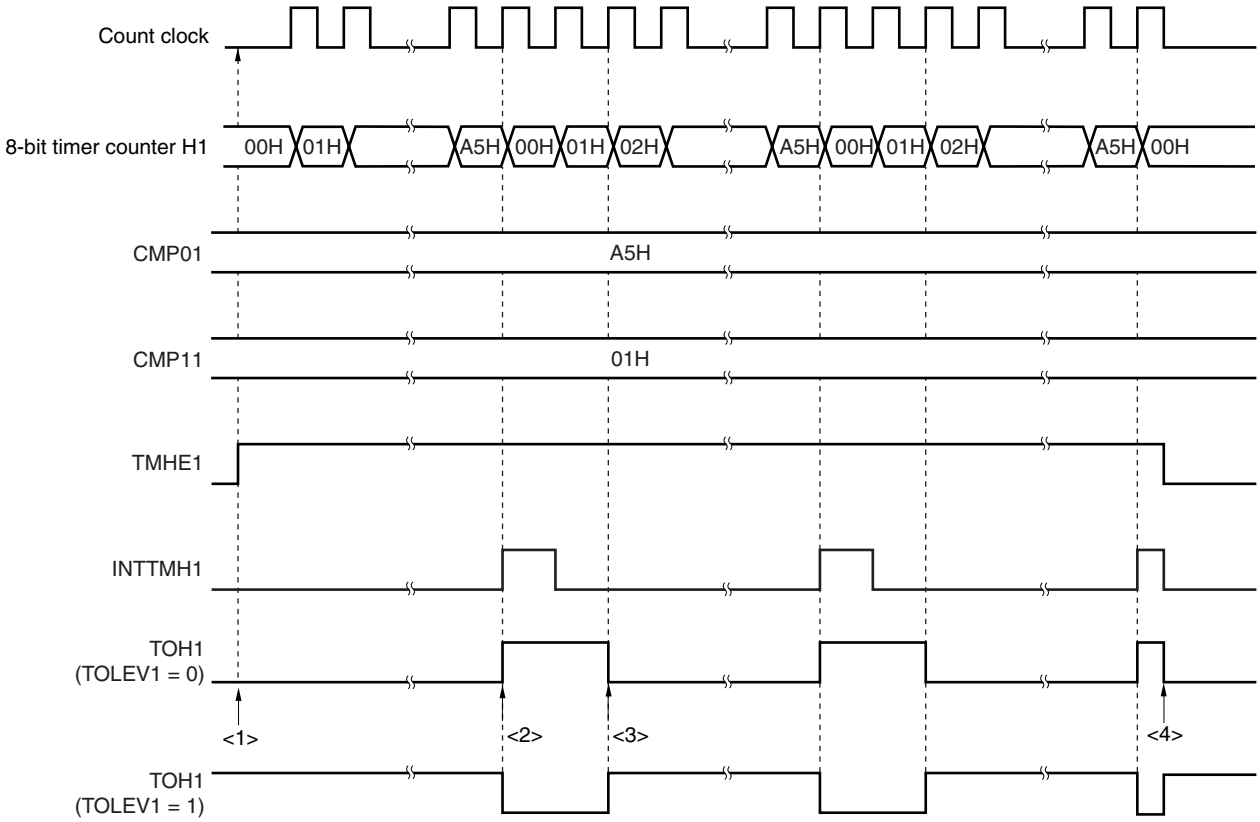
The operation timing in PWM output mode is shown below.

Caution Make sure that the CMP11 register setting value (M) and CMP01 register setting value (N) are within the following range.

00H ≤ CMP11 (M) < CMP01 (N) ≤ FFH

Figure 8-9. Operation Timing in PWM Output Mode (1/4)

(a) Basic operation (00H < CMP11 < CMP01 < FFH)



<1> The count operation is enabled by setting the TMHE1 bit to 1. Start 8-bit timer counter H1 by masking one count clock to count up. At this time, TOH1 output remains inactive (when TOLEV1 = 0).

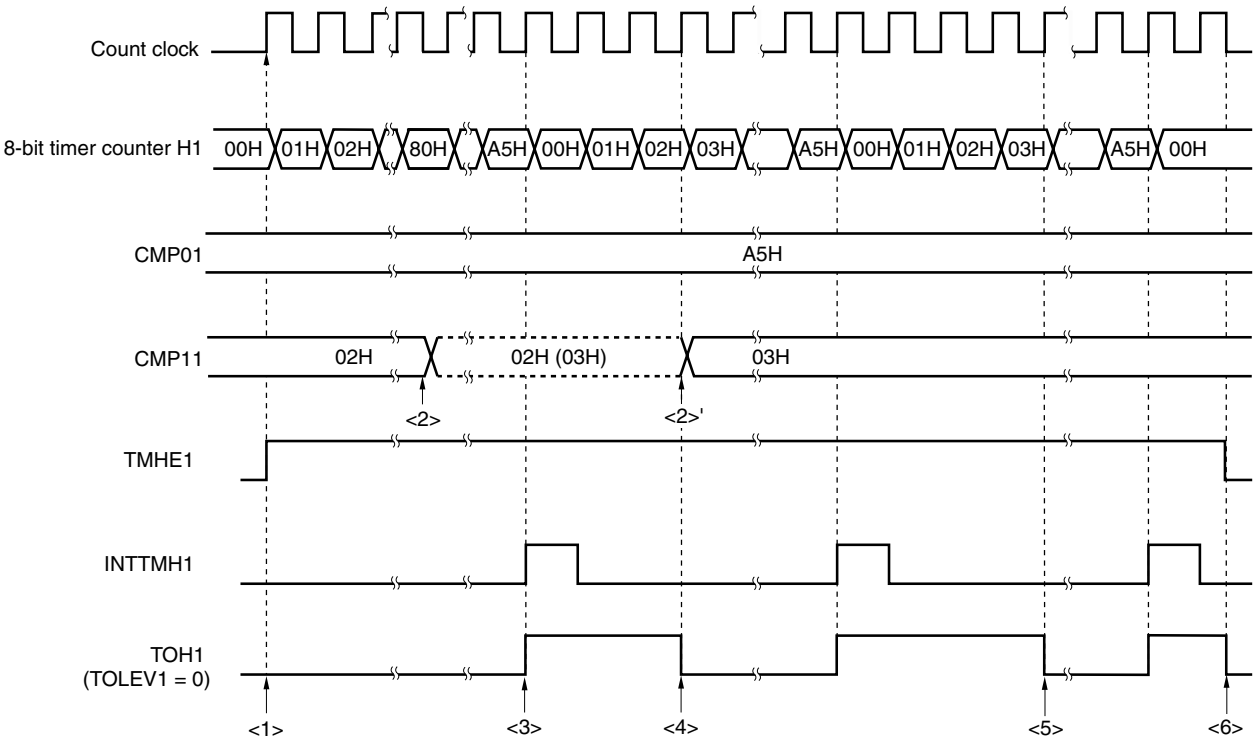
<2> When the values of 8-bit timer counter H1 and the CMP01 register match, the TOH1 output level is inverted, the value of 8-bit timer counter H1 is cleared, and the INTTMH1 signal is output.

<3> When the values of 8-bit timer counter H1 and the CMP11 register match, the level of the TOH1 output is returned. At this time, the 8-bit timer counter value is not cleared and the INTTMH1 signal is not output.

<4> Clearing the TMHE1 bit to 0 during timer H1 operation makes the INTTMH1 signal and TOH1 output inactive.

Figure 8-9. Operation Timing in PWM Output Mode (4/4)

(e) Operation by changing CMP11 (CMP11 = 02H → 03H, CMP01 = A5H)



<1> The count operation is enabled by setting TMHE1 = 1. Start 8-bit timer counter H1 by masking one count clock to count up. At this time, the TOH1 output remains inactive (when TOLEV1 = 0).

<2> The CMP11 register value can be changed during timer counter operation. This operation is asynchronous to the count clock.

<3> When the values of 8-bit timer counter H1 and the CMP01 register match, the value of 8-bit timer counter H1 is cleared, the TOH1 output becomes active, and the INTTMH1 signal is output.

<4> If the CMP11 register value is changed, the value is latched and not transferred to the register. When the values of 8-bit timer counter H1 and the CMP11 register before the change match, the value is transferred to the CMP11 register and the CMP11 register value is changed (<2>').

However, three count clocks or more are required from when the CMP11 register value is changed to when the value is transferred to the register. If a match signal is generated within three count clocks, the changed value cannot be transferred to the register.

<5> When the values of 8-bit timer counter H1 and the CMP11 register after the change match, the TOH1 output becomes inactive. 8-bit timer counter H1 is not cleared and the INTTMH1 signal is not generated.

<6> Clearing the TMHE1 bit to 0 during timer H1 operation makes the INTTMH1 signal and TOH1 output inactive.

10.5 How to Read A/D Converter Characteristics Table

Here, special terms unique to the A/D converter are explained.

(1) Resolution

This is the minimum analog input voltage that can be identified. That is, the percentage of the analog input voltage per bit of digital output is called 1LSB (Least Significant Bit). The percentage of 1LSB with respect to the full scale is expressed by %FSR (Full Scale Range).

1LSB is as follows when the resolution is 10 bits.

$1LSB = 1/2^{10} = 1/1024$
$= 0.098\%FSR$

Accuracy has no relation to resolution, but is determined by overall error.

(2) Overall error

This shows the maximum error value between the actual measured value and the theoretical value.

Zero-scale error, full-scale error, integral linearity error, and differential linearity errors that are combinations of these express the overall error.

Note that the quantization error is not included in the overall error in the characteristics table.

(3) Quantization error

When analog values are converted to digital values, a ±1/2LSB error naturally occurs. In an A/D converter, an analog input voltage in a range of ±1/2LSB is converted to the same digital code, so a quantization error cannot be avoided.

Note that the quantization error is not included in the overall error, zero-scale error, full-scale error, integral linearity error, and differential linearity error in the characteristics table.

Figure 10-13. Overall Error

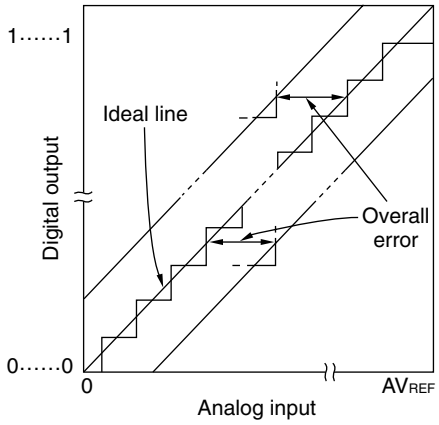
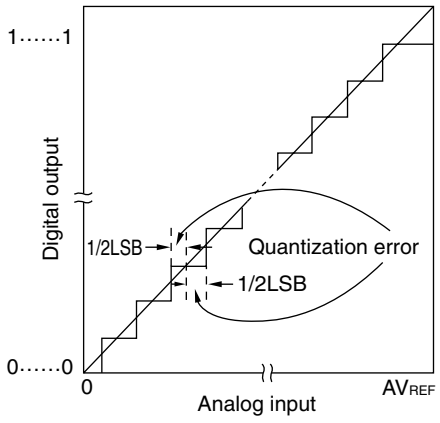


Figure 10-14. Quantization Error



(4) Zero-scale error

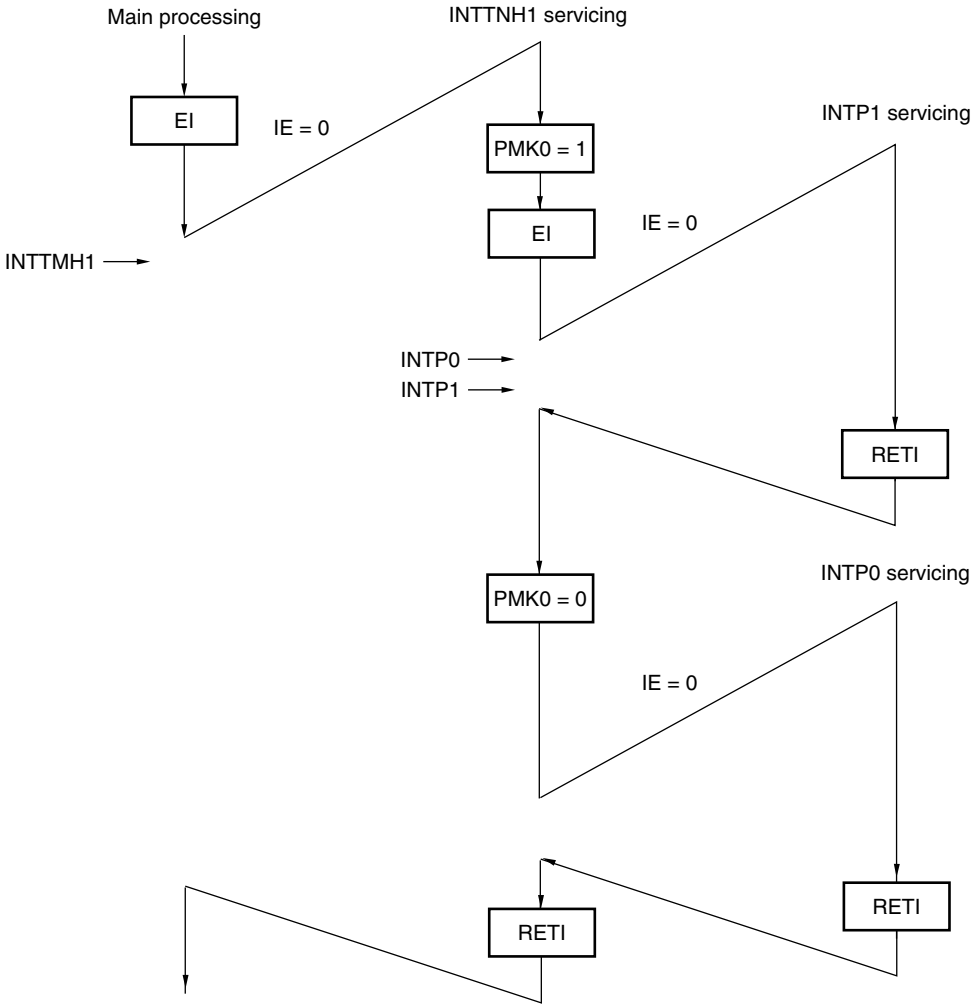
This shows the difference between the actual measurement value of the analog input voltage and the theoretical value (1/2LSB) when the digital output changes from 0.....000 to 0.....001.

If the actual measurement value is greater than the theoretical value, it shows the difference between the actual measurement value of the analog input voltage and the theoretical value (3/2LSB) when the digital output changes from 0……001 to 0……010.

Figure 13-10. Example of Multiple Interrupts (2/2)

Example 3. A priority is controlled by the multiple interrupts

The vector interrupt enable state is set for INTP0, INTP1, and INTTMH1.
(Interrupt priority INTP0 > INTP1 > INTTMH1 (refer to **Table13-1**))



In the interrupt INTTMH1 servicing, servicing is performed such that the INTP1 interrupt is given priority, since the INTP0 interrupt was first masked.

Afterwards, once the interrupt mask for INTP0 is released, INTP0 processing through multiple interrupts is performed.

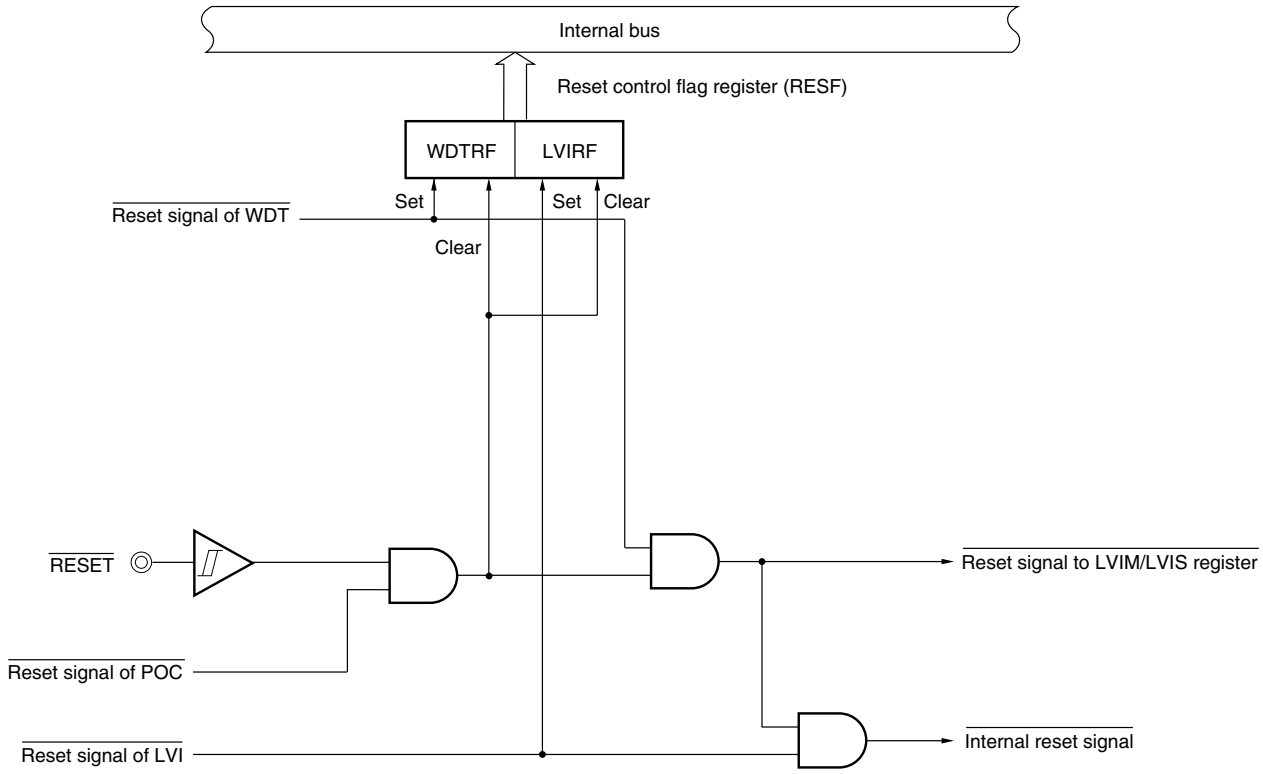
IE = 0: Interrupt request acknowledgment disabled

13.4.3 Interrupt request pending

Some instructions may keep pending the acknowledgment of an instruction request until the completion of the execution of the next instruction even if the interrupt request (maskable interrupt and external interrupt) is generated during the execution. The following shows such instructions (interrupt request pending instruction).

- Manipulation instruction for interrupt request flag registers 0, 1 (IF0, IF1)
- Manipulation instruction for interrupt mask flag registers 0, 1 (MK0, MK1)

Figure 15-1. Block Diagram of Reset Function



Caution The LVI circuit is not reset by the internal reset signal of the LVI circuit.

Remarks 1. LVIM: Low-voltage detect register

2. LVIS: Low-voltage detection level select register

Example programs that perform internal verify 1 and 2 in self programming mode are shown below.

- Internal verify 1

```
;----------------------------
;START
;----------------------------
FlashVerify:
       MOV      FLCMD,#01H      ; Sets flash control command (internal verify 1)
       MOV      FLAPH,#07H      ; Sets block number for which internal verify is performed,
                                ; to FLAPH (Example: Block 7 is specified here)
       MOV      FLAPL,#00H      ; Sets 00H
       MOV      FLAPHC,#07H
       MOV      FLAPLC,#FFH     ; Sets FFH

       MOV      PFS,#00H        ; Clears flash status register
       MOV      WDTE,#0ACH      ; Clears & restarts WDT
       HALT                     ; Self programming is started
       MOV      A,PFS
       MOV      CmdStatus,A     ; Execution result is stored in variable
                                ; (CmdStatus = 0: normal termination, other than 0: abnormal
                                ; termination)

;----------------------------
;END
;----------------------------
```

- Internal verify 2

```
;----------------------------
;START
;----------------------------
FlashVerify:
       MOV      FLCMD,#02H      ; Sets flash control command (internal verify 2)
       MOV      FLAPH,#07H      ; Sets block number for which internal verify is
                                ; performed, to FLAPH (Example: Block 7 is specified here)
       MOV      FLAPL,#00H      ; Sets FLAPL to the start address for verify (Example: Address
                                ; 00H is specified here)
       MOV      FLAPHC,#07H
       MOV      FLAPLC,#20H     ; Sets FLAPLC to the end address for verify (Example: Address
                                ; 20H is specified here)

       MOV      PFS,#00H        ; Clears flash status register
       MOV      WDTE,#0ACH      ; Clears & restarts WDT
       HALT                     ; Self programming is started
       MOV      A,PFS
       MOV      CmdStatus,A     ; Execution result is stored in variable
                                ; (CmdStatus = 0: normal termination, other than 0: abnormal
                                ; termination)

;----------------------------
;END
;----------------------------
```

CHAPTER 21 INSTRUCTION SET OVERVIEW

This chapter lists the instruction set of the 78K0S/KB1+. For details of the operation and machine language (instruction code) of each instruction, refer to **78K/0S Series Instructions User's Manual (U11047E)**.

21.1 Operation

21.1.1 Operand identifiers and description methods

Operands are described in "Operand" column of each instruction in accordance with the description method of the instruction operand identifier (refer to the assembler specifications for details). When there are two or more description methods, select one of them. Uppercase letters and the symbols #, !, $, and [ ] are key words and are 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 #, !, $ and [ ] 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 21-1. 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 labels FE20H to FF1FH Immediate data or labels (even addresses only)
addr16 addr5	0000H to FFFFH Immediate data or labels (only even addresses for 16-bit data transfer instructions) 0040H to 007FH Immediate data or labels (even addresses only)
word byte bit	16-bit immediate data or label 8-bit immediate data or label 3-bit immediate data or label

Remark For symbols of special function registers, see **Table 3-3 Special Function Registers**.

Mnemonic	Operand	Bytes	Clocks	Operation	Flag		
					Z	AC	CY
CALL	!addr16	3	6	$(SP-1) \leftarrow (PC+3)_H$, $(SP-2) \leftarrow (PC+3)_L$, $PC \leftarrow$ addr16, $SP \leftarrow SP-2$			
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), $SP \leftarrow SP-2$			
RET		1	6	$PC_H \leftarrow (SP+1)$, $PC_L \leftarrow (SP)$, $SP \leftarrow SP+2$			
RETI		1	8	$PC_H \leftarrow (SP+1)$, $PC_L \leftarrow (SP)$, $PSW \leftarrow (SP+2)$, $SP \leftarrow SP+3$	R	R	R
PUSH	PSW	1	2	$(SP-1) \leftarrow PSW$, $SP \leftarrow SP-1$			
	rp	1	4	$(SP-1) \leftarrow rp_H$, $(SP-2) \leftarrow rp_L$, $SP \leftarrow SP-2$			
POP	PSW	1	4	$PSW \leftarrow (SP)$, $SP \leftarrow SP+1$	R	R	R
	rp	1	6	$rp_H \leftarrow (SP+1)$, $rp_L \leftarrow (SP)$, $SP \leftarrow SP+2$			
MOVW	SP, AX	2	8	$SP \leftarrow AX$			
	AX, SP	2	6	$AX \leftarrow SP$			
BR	!addr16	3	6	$PC \leftarrow$ addr16			
	$addr16	2	6	$PC \leftarrow PC+2+$ jdisp8			
	AX	1	6	$PC_H \leftarrow A$, $PC_L \leftarrow X$			
BC	$saddr16	2	6	$PC \leftarrow PC+2+$ jdisp8 if CY = 1			
BNC	$saddr16	2	6	$PC \leftarrow PC+2+$ jdisp8 if CY = 0			
BZ	$saddr16	2	6	$PC \leftarrow PC+2+$ jdisp8 if Z = 1			
BNZ	$saddr16	2	6	$PC \leftarrow PC+2+$ jdisp8 if Z = 0			
BT	saddr.bit, $addr16	4	10	$PC \leftarrow PC+4+$ jdisp8 if (saddr.bit) = 1			
	sfr.bit, $addr16	4	10	$PC \leftarrow PC+4+$ jdisp8 if sfr.bit = 1			
	A.bit, $addr16	3	8	$PC \leftarrow PC+3+$ jdisp8 if A.bit = 1			
	PSW.bit, $addr16	4	10	$PC \leftarrow PC+4+$ jdisp8 if PSW.bit = 1			
BF	saddr.bit, $addr16	4	10	$PC \leftarrow PC+4+$ jdisp8 if (saddr.bit) = 0			
	sfr.bit, $addr16	4	10	$PC \leftarrow PC+4+$ jdisp8 if sfr.bit = 0			
	A.bit, $addr16	3	8	$PC \leftarrow PC+3+$ jdisp8 if A.bit = 0			
	PSW.bit, $addr16	4	10	$PC \leftarrow PC+4+$ jdisp8 if PSW.bit = 0			
DBNZ	B, $addr16	2	6	$B \leftarrow B-1$, then $PC \leftarrow PC+2+$ jdisp8 if $B \neq 0$			
	C, $addr16	2	6	$C \leftarrow C-1$, then $PC \leftarrow PC+2+$ jdisp8 if $C \neq 0$			
	saddr, $addr16	3	8	(saddr) $\leftarrow$ (saddr) -1, then $PC \leftarrow PC+3+$ jdisp8 if (saddr) $\neq 0$			
NOP		1	2	No Operation			
EI		3	6	IE $\leftarrow$ 1 (Enable Interrupt)			
DI		3	6	IE $\leftarrow$ 0 (Disable Interrupt)			
HALT		1	2	Set HALT Mode			
STOP		1	2	Set STOP Mode			

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

(A2) grade product T_A = –40 to +125°C

AC Characteristics

(1) Basic operation (T_A = –40 to +125°C, V_{DD} = 2.0 to 5.5 V[Note 1], V_{SS} = 0 V)

Parameter	Symbol	Conditions		MIN.	TYP.	MAX.	Unit
Cycle time (minimum instruction execution time)	T_{CY}	Crystal/ceramic oscillation clock, external clock input	4.0 V ≤ V_{DD} ≤ 5.5 V	0.25		16	μs
			3.0 V ≤ V_{DD} < 4.0 V	0.33		16	μs
			2.7 V ≤ V_{DD} < 3.0 V	0.4		16	μs
			2.0 V ≤ V_{DD} < 2.7 V	1		16	μs
		High-speed internal oscillation clock	4.0 V ≤ V_{DD} ≤ 5.5 V	0.23		4.22	μs
			2.7 V ≤ V_{DD} < 4.0 V	0.47		4.22	μs
			2.0 V ≤ V_{DD} < 2.7 V	0.95		4.22	μs
TI000 input high-level width, low-level width	t_{TIH}, t_{TIL}	4.0 V ≤ V_{DD} ≤ 5.5 V		2/fsam+ 0.1[Note 2]			μs
		2.0 V ≤ V_{DD} < 4.0 V		2/fsam+ 0.2[Note 2]			μs
Interrupt input high-level width, low-level width	t_{INTH}, t_{INTL}			1			μs
$\overline{\text{RESET}}$ input low-level width	t_{RSL}			2			μs

Notes 1. Use this product in a voltage range of 2.26 to 5.5 V because the detection voltage (V_{POC}) of the power-on-clear (POC) circuit is 2.26 V (MAX.).

2. Selection of fsam = f_{XP}, f_{XP}/4, or f_{XP}/256 is possible using bits 0 and 1 (PRM000, PRM001) of prescaler mode register 00 (PRM00). Note that when selecting the valid edge of the TI000 pin as the count clock, fsam = f_{XP}.

CPU Clock Frequency, Peripheral Clock Frequency

Parameter	Conditions	CPU Clock (f_{CPU})	Peripheral Clock (f_{XP})
Ceramic resonator, crystal resonator, external clock	4.0 to 5.5 V	125 kHz ≤ f_{CPU} ≤ 8 MHz	500 kHz ≤ f_{XP} ≤ 8 MHz
	3.0 to 4.0 V	125 kHz ≤ f_{CPU} ≤ 6 MHz	
	2.7 to 3.0 V	125 kHz ≤ f_{CPU} ≤ 5 MHz	
	2.0 to 2.7 V[Note]	125 kHz ≤ f_{CPU} ≤ 2 MHz	500 kHz ≤ f_{XP} ≤ 5 MHz
High-speed internal oscillator	4.0 to 5.5 V	500 kHz (TYP.) ≤ f_{CPU} ≤ 8 MHz (TYP.)	2 MHz (TYP.) ≤ f_{XP} ≤ 8 MHz (TYP.)
	2.7 to 4.0 V	500 kHz (TYP.) ≤ f_{CPU} ≤ 4 MHz (TYP.)	
	2.0 to 2.7 V[Note]	500 kHz (TYP.) ≤ f_{CPU} ≤ 2 MHz (TYP.)	2 MHz (TYP.) ≤ f_{XP} ≤ 4 MHz (TYP.)

Note Use this product in a voltage range of 2.26 to 5.5 V because the detection voltage (V_{POC}) of the power-on-clear (POC) circuit is 2.26 V (MAX.).

Table 25-1. Surface Mounting Type Soldering Conditions (2/2)

- **30-pin plastic SSOP (lead-free products)**

μPD78F9232MC(A)-CAB-AX, 78F9234MC(A)-CAB-AX, 78F9232MC(A2)-CAB-AX, 78F9234MC(A2)-CAB-AX

Soldering Method	Soldering Conditions	Recommended Condition Symbol
Infrared reflow	Package peak temperature: 260°C, Time: 60 seconds max. (at 220°C or higher), Count: 3 times or less, Exposure limit: 7 days[Note] (after that, prebake at 125°C for 10 to 72 hours)	IR60-107-3
Wave soldering	Solder bath temperature: 260°C max., Time: 10 seconds max., Count: Once, Preheating temperature: 120°C max. (package surface temperature), Exposure limit: 7 days[Note] (after that, prebake at 125°C for 10 to 72 hours)	WS60-107-1
Partial heating	Pin temperature: 350°C max., Time: 3 seconds max. (per pin row)	–

Note After opening the dry pack, store it at 25°C or less and 65% RH or less for the allowable storage period.

Caution Do not use different soldering methods together (except for partial heating).

- **32-pin plastic SDIP (lead-free products)**

μPD78F9232CS-CAA-A, 78F9234CS-CAA-A

Soldering Method	Soldering Conditions	Recommended Condition Symbol
Wave soldering (only for pins)	Solder bath temperature: 260°C max., Time: 10 seconds max.	–
Partial heating	Pin temperature: 350°C max., Time: 3 seconds max. (per one pin)	–

Caution Only the pins of the THD are heated when performing wave soldering.
Make sure that flow solder does not come in contact with the package.

(18/20)

Chapter	Classification	Function	Details of Function	Cautions	Page
Chapter 19	Soft	Flash memory	Self programming function	Since the security function set via on-board/off-board programming is disabled in self programming mode, the self programming command can be executed regardless of the security function setting. To disable write or erase processing during self programming, set the protect byte.	p.285 ☐
				Be sure to clear bits 5 to 7 of flash address pointer H (FLAPH) and flash address pointer H compare register (FLAPHC) to 0 before executing the self programming command. If the self programming command is executed with these bits set to 1, the device may malfunction.	p.285 ☐
				Clear the value of the FLCMD register to 00H immediately before setting to self programming mode and normal mode.	p.285 ☐
			FLPMC: Flash programming mode control register	Cautions in the case of setting the self programming mode, refer to 19.8.2 Cautions on self programming function.	p.286 ☐
				Set the CPU clock beforehand so that it is 1 MHz or higher during self programming.	p.286 ☐
				Execute self programming after executing the NOP and HALT instructions immediately after executing a specific sequence to set self programming mode. At this time, the HALT instruction is automatically released after 10 μs (MAX.) + 2 CPU clocks (f_{CPU}).	p.286 ☐
				If the clock of the oscillator or an external clock is selected as the system clock, execute the NOP and HALT instructions immediately after executing a specific sequence to set self programming mode, wait for 8 μs after releasing the HALT status, and then execute self programming.	p.286 ☐
				Clear the value of the FLCMD register to 00H immediately before setting to self programming mode and normal mode.	p.286 ☐
			PFCMD: Flash protect command register	Interrupt servicing cannot be executed in self programming mode. Disable interrupt servicing (by executing the DI instruction while MK0 and MK1 = FFH) between the points before executing the specific sequence that sets self programming mode and after executing the specific sequence that changes the mode to the normal mode.	p.287 ☐
			PFS: Flash status register	Check FPRERR using a 1-bit memory manipulation instruction.	p.287 ☐
			FLAPH and FLAPL: Flash address pointers H and L	Be sure to clear bits 5 to 7 of FLAPH and FLAPHC to 0 before executing the self programming command. If the self programming command is executed with these bits set to 1, the device may malfunction.	p.290 ☐
			FLAPHC and FLAPLC: Flash address pointer H compare register and flash address pointer L compare register	Be sure to clear bits 5 to 7 of FLAPH and FLAPHC to 0 before executing the self programming command. If the self programming command is executed with these bits set to 1, the device may malfunction.	p.290 ☐
				Set the number of the block subject to a block erase, verify, or blank check (same value as FLAPH) to FLAPHC.	p.290 ☐
				Clear FLAPLC to 00H when a block erase is performed, and FFH when a blank check is performed.	p.290 ☐

E.2 Revision History up to Previous Editions

The following table shows the revision history up to this edition. The “Applied to:” column indicates the chapters of each edition in which the revision was applied.

(1/6)

Edition	Description	Applied to:
2nd edition	Addition of part number to **1.3 Ordering Information**	**CHAPTER 1 OVERVIEW**
	Modification of operating temperature range in **1.5 78K0S/Kx1+ Product Lineup**	
	Addition of **Note** to P34/$\overline{\text{RESET}}$, P121/X1 and P122/X2 in **2.1 Pin Function List**	**CHAPTER 2 PIN FUNCTIONS**
	Addition of description to **2.2.3 P30 to P34 (Port 3)**	
	Addition of description to **2.2.7 $\overline{\text{RESET}}$**	
	Addition of description to and modification of **Cautions** in **4.2.3 Port 3**	**CHAPTER 4 PORT FUNCTIONS**
	Modification of description in **(2) External event counter** in **6.1 Functions of 16-bit Timer/Event Counter 00**	**CHAPTER 6 16-BIT TIMER/EVENT COUNTER 00**
	Modification of **Caution 2** in **Figure 6-2 Format of 16-bit Timer Counter 00 (TM00)**	
	Modification of **Caution 2** in **Figure 6-5 Format of 16-bit Timer Mode Control Register 00 (TMC00)**	
	Addition of **Caution** to (1), (2), (3), and (4) in **6.4.3 Pulse width measurement operations**	
	Modification of **Figure 6-19 Configuration Diagram for Pulse Width Measurement by Free-Running Counter**	
	Modification of **Figure 6-20 Timing of Pulse Width Measurement Operation by Free-Running Counter and One Capture Register (with Both Edges Specified)** and **Note**	
	Modification of **Figure 6-22 Timing of Pulse Width Measurement Operation with Free-Running Counter (with Both Edges Specified)** and **Note**	
	Modification of **Figure 6-24 Timing of Pulse Width Measurement Operation by Free-Running Counter and Two Capture Registers (with Rising Edge Specified)** and **Note**	
	Modification of <3> and <4> in **(2) 16-bit timer counter 00 (TM00) operation** of **6.5 Cautions Related to 16-bit Timer/Event Counter 00**	
	Modification of **Caution** in **(1) 8-bit compare register 80 (CR80)** of **7.2 Configuration of 8-bit Timer 80**	**CHAPTER 7 8-BIT TIMER 80**
	Modification of description in **(2) 8-bit timer H compare register 11 (CMP11)** of **8.2 Configuration of 8-bit Timer H1**	**CHAPTER 8 8-BIT TIMER H1**
	Modification of **Caution 1** in **8.4.2 Operation as PWM output mode**	
	Modification of **(e) Operation by changing CMP11 (CMP11 = 02H → 03H, CMP01 = A5H)** in **Figure 8-9 Operation Timing in PWM Output Mode**	
	Addition of description to **Caution 2** in **Figure 9-2 Format of Watchdog Timer Mode Register (WDTM)**	**CHAPTER 9 WATCHDOG TIMER**
	Modification of **Table 10-1 Sampling Time and A/D Conversion Time** and **Note 1**	**CHAPTER 10 A/D CONVERTER**
	Modification of **Figure 10-3 Format of A/D Converter Mode Register (ADM)** and **Note 2**	
	Addition of description to **(7) Input switch control register (ISC)** in **11.3 Registers Controlling Serial Interface UART6**	**CHAPTER 11 SERIAL INTERFACE UART6**
	Modification of value in **Table 11-4 Set Data of Baud Rate Generator**	

Published by: NEC Electronics Corporation (http://www.necel.com/)
Contact: http://www.necel.com/support/