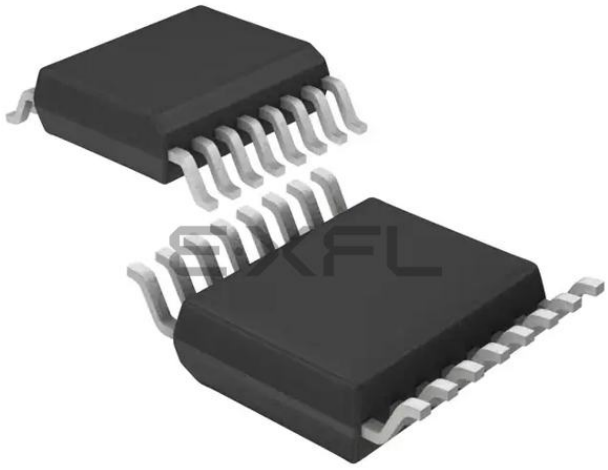
Renesas Electronics America Inc - UPD78F0750MA-FAA-AX Datasheet



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/0
Core Size	8-Bit
Speed	10MHz
Connectivity	-
Peripherals	LVD, POR, PWM, WDT
Number of I/O	9
Program Memory Size	4KB (4K x 8)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	384 x 8
Voltage - Supply (Vcc/Vdd)	2.7V ~ 5.5V
Data Converters	A/D 5x10b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 105°C (TA)
Mounting Type	Surface Mount
Package / Case	16-SSOP (0.173", 4.40mm Width)
Supplier Device Package	-
Purchase URL	https://www.e-xfl.com/product-detail/renesas-electronics-america/upd78f0750ma-faa-ax

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.

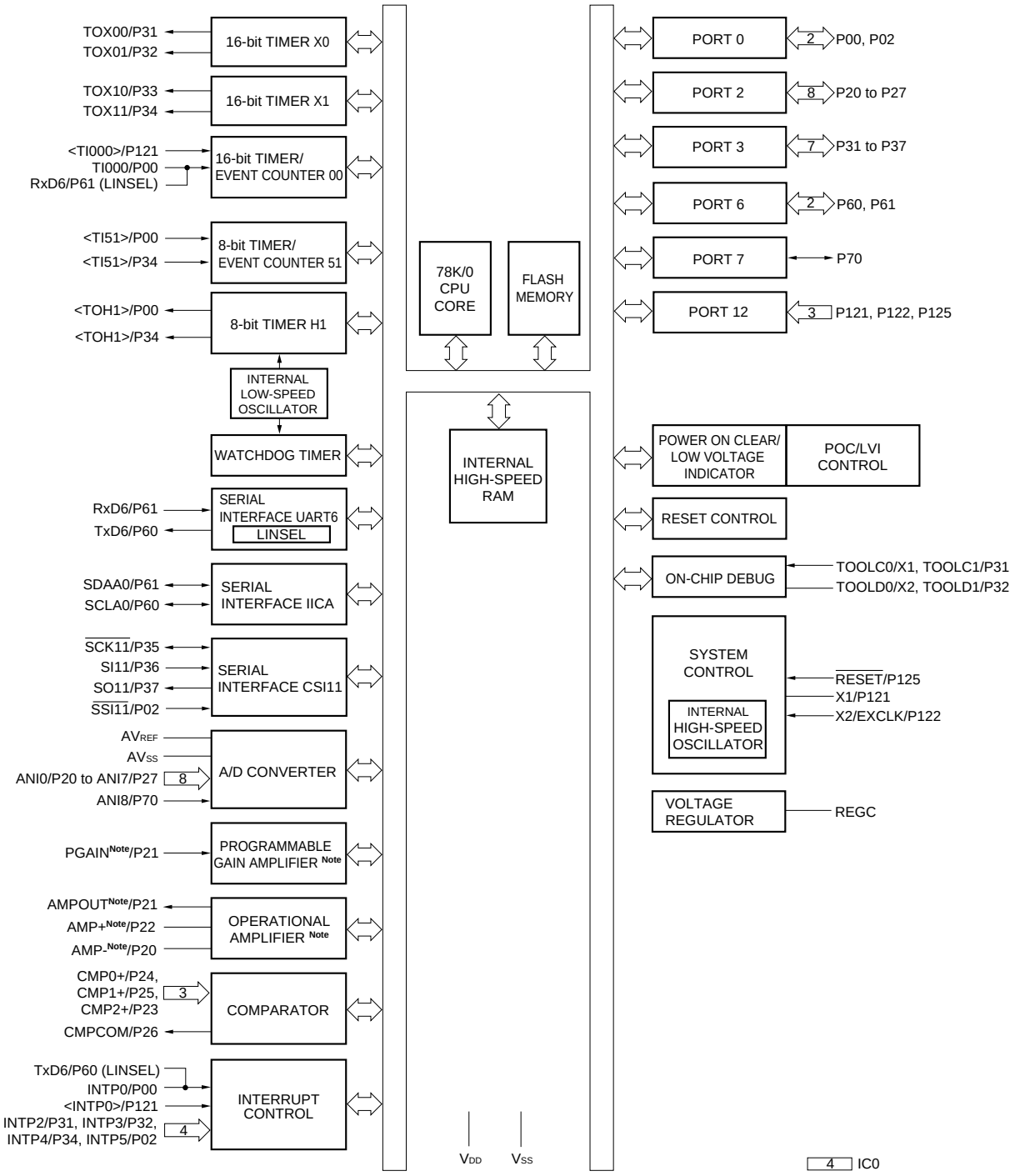
(3) PRECAUTION AGAINST ESD: A strong electric field, when exposed to a MOS device, can cause destruction of the gate oxide and ultimately degrade the device operation. Steps must be taken to stop generation of static electricity as much as possible, and quickly dissipate it when it has occurred. Environmental 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.

- 32-pin



Note μPD78F0755, 78F0756 (products with operational amplifier) only

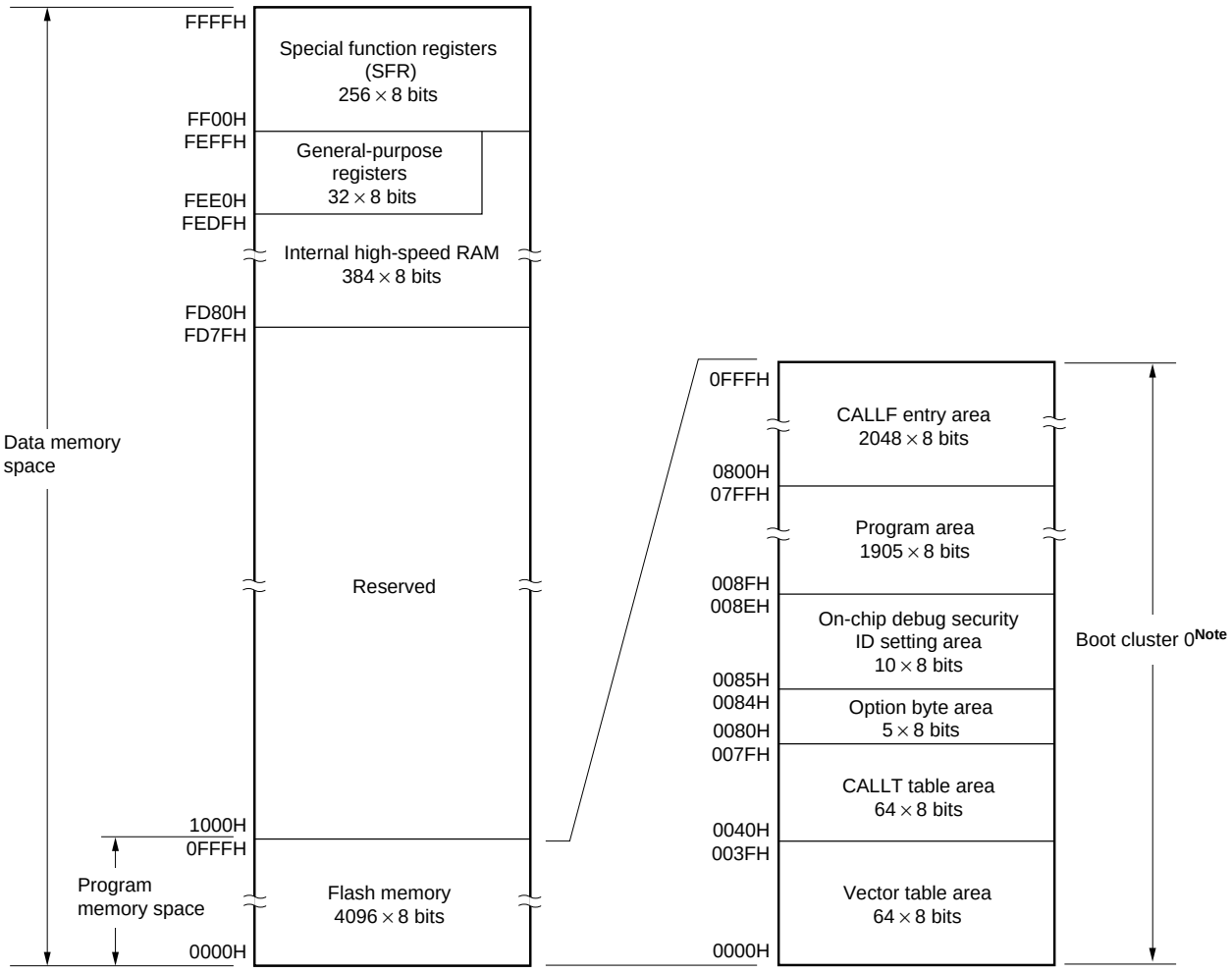
Cautions 1. Connect the REGC pin to VSS via a capacitor (0.47 to 1 μF).

2. ANI0/P20/AMP-, ANI1/P21/AMPOUT/PGAIN, ANI2/P22/AMP+, ANI3/P23/CMP2+, ANI4/P24/CMP0+, ANI5/P25/CMP1+, ANI6/P26/CMPCOM, ANI7/P27, and ANI8/P70 are set in the analog input mode after release of reset.

3. $\overline{\text{RESET}}$/P125 immediately after release of reset is set in the external reset input.

Remark Functions in angle brackets < > can be assigned by setting the input switch control register (MUXSEL).

Figure 3-1. Memory Map (μPD78F0740, 78F0750)



Note Writing boot cluster 0 can be prohibited depending on the setting of security (refer to **25.6 Security Settings**).

Remark The flash memory is divided into blocks (one block = 1 KB). For the address values and block numbers, refer to **Table 3-2 Correspondence Between Address Values and Block Numbers in Flash Memory**.

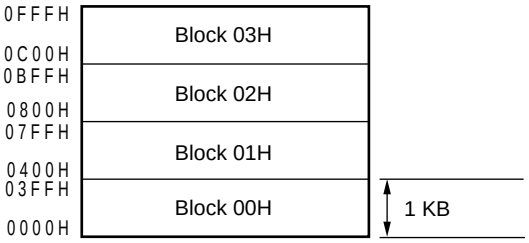


Table 4-5. Port Functions (78K0/IB2 (32 Pins))

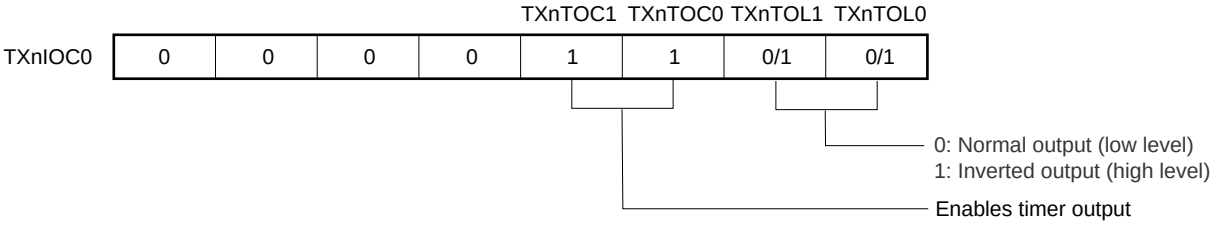
Function Name	I/O	Function	After Reset	Alternate Function
P00	I/O	Port 0. 3-bit I/O port. Input/output can be specified in 1-bit units. Use of an on-chip pull-up resistor can be specified by a software setting.	Input port	TI000/INTP0/<TOH1>/<TI51>
P02				$\overline{\text{SSI11}}$/INTP5
P20	I/O	Port 2. 8-bit I/O port. Input/output can be specified in 1-bit units.	Analog input	ANI0/AMP-Note
P21				ANI1/AMPOUTNote/PGAINNote
P22				ANI2/AMP+Note
P23				ANI3/CMP2+
P24				ANI4/CMP0+
P25				ANI5/CMP1+
P26				ANI6/CMPCOM
P27				ANI7
P31	I/O	Port 3. 8-bit I/O port. Input/output can be specified in 1-bit units. Use of an on-chip pull-up resistor can be specified by a software setting.	Input port	TOX00/INTP2/TOOLC1
P32				TOX01/INTP3/TOOLD1
P33				TOX10
P34				TOX11/INTP4/<TOH1>/<TI51>
P35				$\overline{\text{SCK11}}$
P36				SI11
P37				SO11
P60	I/O	Port 6. 2-bit I/O port. Input/output can be specified in 1-bit units. Input can be set to SMBus input buffer in 1-bit units. Output can be set to N-ch open-drain output (V_{DD} tolerance). Use of an on-chip pull-up resistor can be specified by a software setting.	Input port	SCLA0/TxD6
P61				SDAA0/RxD6
P70	I/O	Port 7. 1-bit I/O port. Input/output can be specified in 1-bit units.	Analog input	ANI8
P121	Input	Port 12. 3-bit input port. For only P125, use of an on-chip pull-up resistor can be specified by a software setting.	Input port	X1/TOOLC0/<TI000>/<INTP0>
P122				X2/EXCLK/TOOLD0
P125				$\overline{\text{RESET}}$

Note μPD78F0755, 78F0756 (products with operational amplifier) only

Remark Functions in angle brackets < > can be assigned by setting the input switch control register (MUXSEL).

Figure 6-41. Example of Register Settings for PWM Output Operation (TMX0 and TMX1 synchronous start/clear mode, PWM Output from TOX00, TOX01, TOX10, and TOX11 When TOH1 Output Is at High Level) (2/2)

(c) 16-bit timer Xn output control register 0



Remark n = 0, 1

Figure 6-42. PWM Output Timing (TMX0 and TMX1 synchronous start/clear mode, PWM Output from TOX00, TOX01, TOX10, and TOX11 When TOH1 Output Is at High Level)

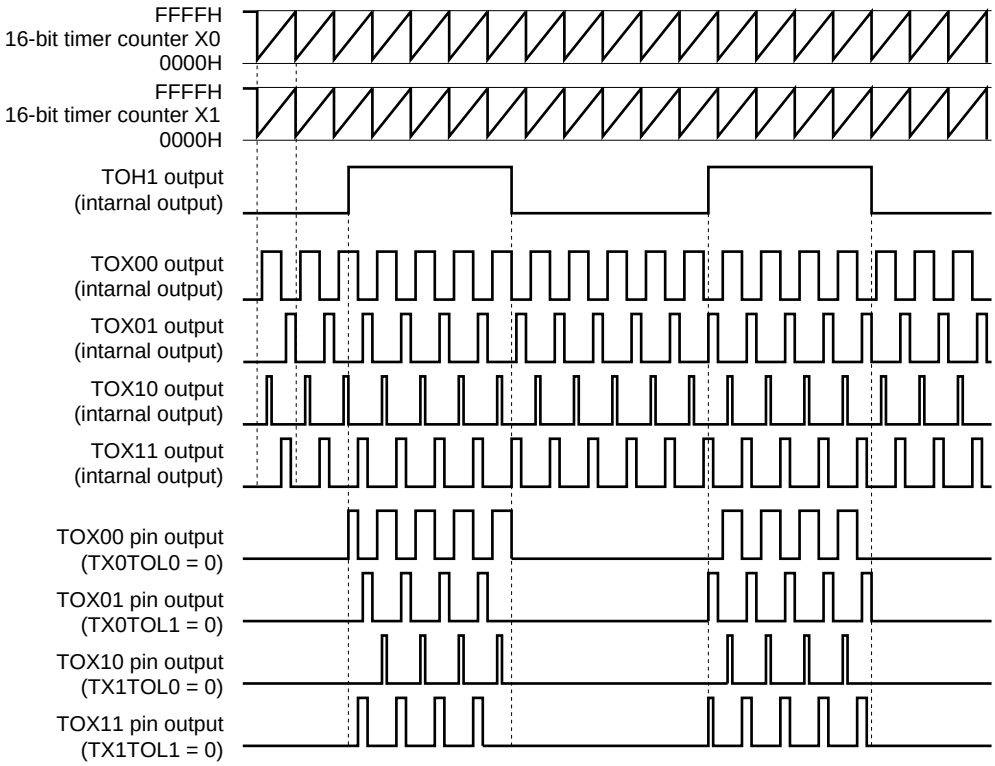


Figure 7-8. Format of 16-Bit Timer Output Control Register 00 (TOC00)

Address: FFBDH After reset: 00H R/W

Symbol	7	<6>	<5>	4	<3>	<2>	1	<0>
TOC00	0	OSPT00	OSPE00	TOC004	LVS00	LVR00	TOC001	TOE00

OSPT00	One-shot pulse output trigger via software
0	–
1	One-shot pulse output
The value of this bit is always “0” when it is read. Do not set this bit to 1 in a mode other than the one-shot pulse output mode. If it is set to 1, TM00 is cleared and started.	

OSPE00	One-shot pulse output operation control
0	Successive pulse output
1	One-shot pulse output
One-shot pulse output operates correctly in the free-running timer mode or clear & start mode entered by TI000 pin valid edge input. The one-shot pulse cannot be output in the clear & start mode entered upon a match between TM00 and CR000.	

TOC004	TO00 output control on match between CR010 and TM00
0	Disables inversion operation
1	Enables inversion operation
The interrupt signal (INTTM010) is generated even when TOC004 = 0.	

LVS00	LVR00	Setting of TO00 output status
0	0	No change
0	1	Initial value of TO00 output is low level (TO00 output is cleared to 0).
1	0	Initial value of TO00 output is high level (TO00 output is set to 1).
1	1	Setting prohibited

- LVS00 and LVR00 can be used to set the initial value of the TO00 output level. If the initial value does not have to be set, leave LVS00 and LVR00 as 00.
- Be sure to set LVS00 and LVR00 when TOE00 = 1.
  LVS00, LVR00, and TOE00 being simultaneously set to 1 is prohibited.
- LVS00 and LVR00 are trigger bits. By setting these bits to 1, the initial value of the TO00 output level can be set. Even if these bits are cleared to 0, TO00 output is not affected.
- The values of LVS00 and LVR00 are always 0 when they are read.
- For how to set LVS00 and LVR00, refer to **7.5.2 Setting LVS00 and LVR00**.
- The actual TO00/TI010/P01 pin output is determined depending on PM01 and P01, besides TO00 output.

TOC001	TO00 output control on match between CR000 and TM00
0	Disables inversion operation
1	Enables inversion operation
The interrupt signal (INTTM000) is generated even when TOC001 = 0.	

TOE00	TO00 output control
0	Disables output (TO00 output fixed to low level)
1	Enables output

A pulse width can be measured in the following three ways.

- Measuring the pulse width by using two input signals of the TI000 and TI010 pins (free-running timer mode)
- Measuring the pulse width by using one input signal of the TI000 pin (free-running timer mode)
- Measuring the pulse width by using one input signal of the TI000 pin (clear & start mode entered by the TI000 pin valid edge input)

Remarks 1. For the setting of the I/O pins, refer to **7.3 (6) Port mode register 0 (PM0)**.

2. For how to enable the INTTM000 signal interrupt, refer to **CHAPTER 18 INTERRUPT FUNCTIONS**.

(1) Measuring the pulse width by using two input signals of the TI000 and TI010 pins (free-running timer mode)

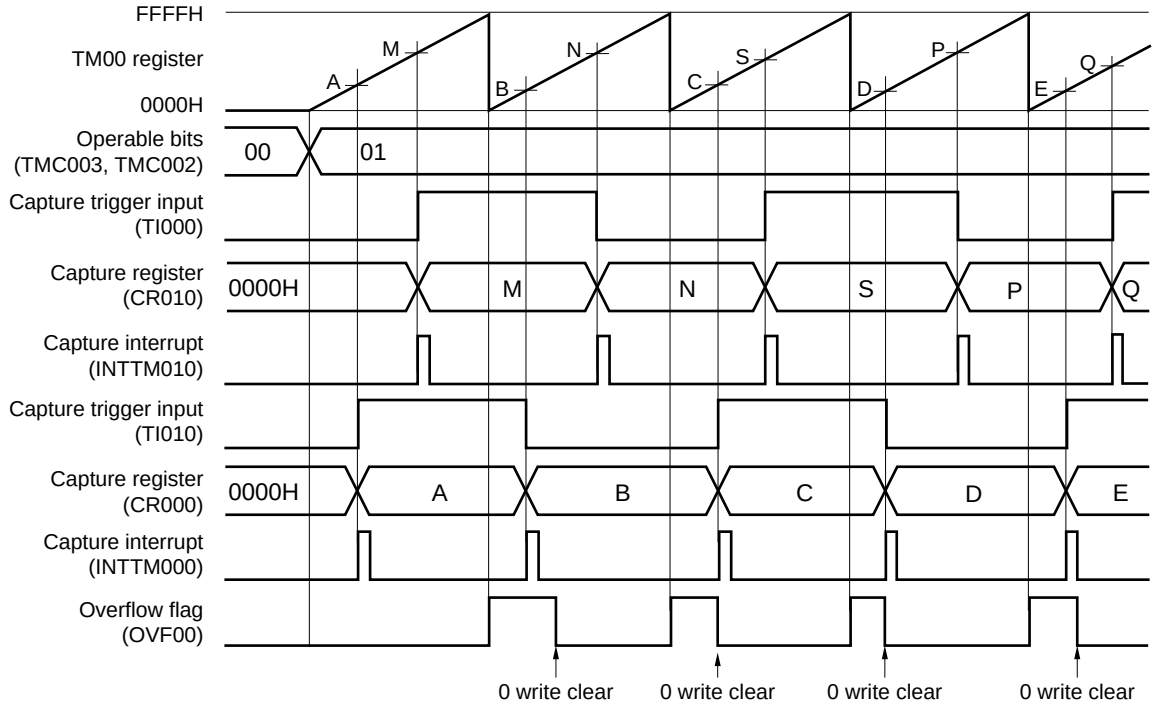
Set the free-running timer mode (TMC003 and TMC002 = 01). When the valid edge of the TI000 pin is detected, the count value of TM00 is captured to CR010. When the valid edge of the TI010 pin is detected, the count value of TM00 is captured to CR000. Specify detection of both the edges of the TI000 and TI010 pins.

By this measurement method, the previous count value is subtracted from the count value captured by the edge of each input signal. Therefore, save the previously captured value to a separate register in advance.

If an overflow occurs, the value becomes negative if the previously captured value is simply subtracted from the current captured value and, therefore, a borrow occurs (bit 0 (CY) of the program status word (PSW) is set to 1). If this happens, ignore CY and take the calculated value as the pulse width. In addition, clear bit 0 (OVF00) of 16-bit timer mode control register 00 (TMC00) to 0.

Figure 7-49. Timing Example of Pulse Width Measurement (1)

• TMC00 = 04H, PRM00 = F0H, CRC00 = 05H



- Timer trigger mode

<1> Set the A/D conversion time and the operation mode by using bits 5 to 1 (FR2 to FR0, LV1, and LV0) of the A/D converter mode register 0 (ADM0).

<2> Set bit 0 (ADCE) of ADM0 to 1.

<3> Set the channel to be used to analog input by using the A/D port configuration registers 0 and 1 (ADPC0, ADPC1) and port mode registers 2 and 7 (PM2, PM7).

<4> Set the PGA operation to set the PGA output and the single Amp operation to set the operational amplifier output for analog input. (refer to **CHAPTER 12 OPERATIONAL AMPLIFIER**).

<5> Select TMX0 or TMX1 synchronization by using bits 4 and 5 (ADTRG0, ADTRG1) of the analog input channel specification register (ADS).

<6> Select a channel to be used by using the analog input channel specification register (ADS).

<7> Set the timer trigger wait state by setting (1) bit 7 (ADCS) of ADM0.

<8> A conversion operation is started when a trigger signal (TMX0 or TMX1 output) is detected.

<9> When one A/D conversion has been completed, an interrupt request signal (INTAD) is generated.

<10> Transfer the A/D conversion data to the A/D conversion result register (ADCRXn, ADCRXnL).

<Change the channel>

<11> Set bit 0 (ADMK) of the interrupt mask flag register 1L (MK1L) to 1**Note**.

<12> Change the channel by using ADS to start A/D conversion.

<13> Clear bit 0 (ADIF) of the interrupt request flag register 1L (IF1L) to 0.

<14> Clear ADMK to 0**Note**.

<15> When one A/D conversion has been completed, an interrupt request signal (INTAD) is generated.

<16> Transfer the A/D conversion data to the A/D conversion result register (ADCRXn, ADCRHXnL).

<Complete A/D conversion>

<17> Clear ADCS to 0.

<18> Clear ADCE to 0.

Note Execute this only if interrupt servicing is used for A/D conversion.

Cautions 1. Make sure the period of <2> to <7> is 1 μs or more.

2. If the timing of <2> is earlier than that of <6>, <2> may be performed any time.

3. <2> can be omitted. However, ignore data of the first conversion after <8> in this case.

4. The period from <9> to <15> differs from the conversion time set using bits 5 to 1 (FR2 to FR0, LV1, and LV0) of ADM0. The period from <12> to <15> is the conversion time set using FR2 to FR0, LV1, and LV0.

5. When switching from timer trigger mode to software trigger mode, switch the operation mode and input channel after stopping the A/D conversion operation (clearing (0) ADCS).

6. To select the internal voltage (1.2 V) as an analog input, set the ADCS bit to 1 when at least 10 μs have elapsed after having set the V12SEL bit to 1 while the A/D conversion operation was stopped (ADCS = 0).

14.3 Registers Controlling Serial Interface UART6/DALI

Serial interface UART6/DALI is controlled by the following nine registers.

○ In UART mode

- UART/DALI mode control register (UADLCTL)
- UART/DALI operation mode register 6 (ASIM6)
- UART/DALI reception error status register 6 (ASIS6)
- UART transmission status register 6 (ASIF6)
- Clock selection register 6 (CKSR6)
- Baud rate generator control register 6 (BRGC6)
- LIN operation control register 6 (ASICL6)
- Input switch control register (ISC)
- Port mode register 6 (PM6)
- Port register 6 (P6)
- Port output mode register 6 (POM6)

○ In DALI mode

- UART/DALI mode control register (UADLCTL)
- UART/DALI operation mode register 6 (ASIM6)
- UART/DALI reception error status register 6 (ASIS6)
- Clock selection register 6 (CKSR6)
- Baud rate generator control register 6 (BRGC6)
- Port mode register 6 (PM6)
- Port output mode register 6 (POM6)

(1) UART/DALI mode control register (UADLCTL)

This is an 8-bit register that controls whether to operate serial interface UART6/DALI in UART mode or DALI mode.

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

Reset signal generation sets this register to 00H.

Figure 14-5. Format of UART/DALI mode control register (UADLCTL)

Address: FF5BH After reset: 00H R/W

Symbol	7	6	5	4	3	2	1	<0>
UADLCTL	0	0	0	0	0	0	0	UADLSEL

UADLSEL	Operation mode
0	UART mode
1	DALI mode

Note Make sure that POWER6, TXE6 and RXE6 = 0 when rewriting the UADLSEL bit.

14.4.3 DALI mode

This mode is used to perform slave transmission/reception of DALI (Digital Addressable Lighting Interface).

DALI performs communication using the following protocol.

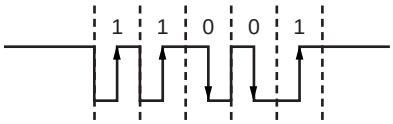
(1) Data structure

<1> Bit definition

A falling edge is bit-defined as “0” and a rising edge as “1”, because DALI communication uses Manchester code.

If no communication is performed, DALI communication is fixed to the high level.

Figure 14-26. Bit Definition

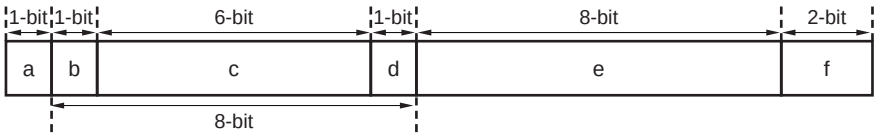


<2> Frame

- Forward frame

This is a frame used when transmitting from the master to a slave. A frame consists of 19 bits.

Figure 14-27. Forward-Frame Structure



a: Start bit

This indicates the start of the frame. It is always the same waveform as “1”.

b-d: Address byte

This specifies the transmission destination of the frame.

e: Data byte

This specifies a command.

f: Stop bit

This indicates the end of the frame. It is fixed to the high level.

(8) Port output mode register 6 (POM6)

This register sets the output mode of P60 and P61 in 1-bit units. During I^2C communication, set POM60 and POM61 to 1.

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

Reset signal generation clears this register to 00H.

Figure 15-12. Format of Port Output Mode Register 6 (POM6)

Address: FF2AH After reset: 00H R/W

Symbol	7	6	5	4	3	2	1	0
POM6	0	0	0	0	0	0	POM61	POM60

POM6n	P6n pin output mode selection (n = 0, 1)
0	Normal output (CMOS output) mode
1	N-ch open drain output (V_{DD} tolerance) mode

(9) Port mode register 6 (PM6)

This register sets the input/output of port 6 in 1-bit units.

When using the P60/SCLA0/TxD6 pin as clock I/O and the P61/SDAA0/RxD6 pin as serial data I/O, clear PM60 and PM61 to 0, and set the output latches of P60 and P61 to 1.

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

Reset signal generation sets this register to FFH.

Figure 15-13. Format of Port Mode Register 6 (PM6)

Address: FF26H After reset: FFH R/W

Symbol	7	6	5	4	3	2	1	0
PM6	1	1	1	1	1	1	PM61	PM60

PM6n	P6n pin I/O mode selection (n = 0, 1)
0	Output mode (output buffer on)
1	Input mode (output buffer off)

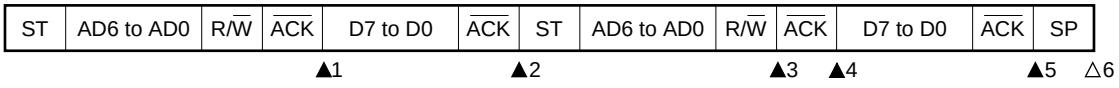
(c) Start ~ Address ~ Data ~ Start ~ Code ~ Data ~ Stop

(i) When WTIM0 = 0 (after restart, does not match address (= extension code))

▲1: IICAS0 = 0001×110B
▲2: IICAS0 = 0001×000B
▲3: IICAS0 = 0010×010B
▲4: IICAS0 = 0010×000B
△5: IICAS0 = 00000001B

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

(ii) When WTIM0 = 1 (after restart, does not match address (= extension code))



▲1: IICAS0 = 0001×110B
▲2: IICAS0 = 0001××00B
▲3: IICAS0 = 0010×010B
▲4: IICAS0 = 0010×110B
▲5: IICAS0 = 0010××00B
△6: IICAS0 = 00000001B

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

17.3 Operation of Multiplier

The result of the multiplication can be obtained by storing the values in the MULA and MULB registers and then reading the MUL0H and MUL0L registers after waiting for 1 clock. The result can also be obtained after 1 clock or more has elapsed, even when fixing either of MULA or MULB and rewrite the other of these. The result can be read without problem, regardless of whether MUL0H or MUL0L is read in first.

A multiplication source example is shown below.

Example 1: Multiplication of 8 bits by 8 bits

```
MOV     MULAL, #005H
MOV     MULBL, #022H
NOP                         ; 1 clock wait.  Doesn't have to be NOP
MOVW    AX, MUL0L           ; Acquire multiplication result
```

Caution In Example 1, set the multiplied data to MULAL and MULBL.

Example 2: Multiplication of 16 bits by 16 bits (using the MOVW instruction for setting the multiplication data)

```
MOVW    MULA, #1234H
MOVW    MULB, #5678H
NOP                         ; 1 clock wait.  Doesn't have to be NOP
MOVW    AX, MUL0H           ; The result obtained on upper side
PUSH    AX
MOVW    AX, MUL0L           ; The result obtained on lower side
```

Example 3: Multiplication of 16 bits by 16 bits (using the MOV instruction for setting the multiplication data)

```
MOV     MULAL, #034H
MOV     MULAH, #012H
MOV     MULBL, #078H
MOV     MULBH, #056H
NOP                         ; 1 clock wait.  Doesn't have to be NOP
MOVW    AX, MUL0H           ; The result obtained on upper side
PUSH    AX
MOVW    AX, MUL0L           ; The result obtained on lower side
```

Caution In Example 3, set the higher 8 bits of the multiplied data to MULAH/MULBH after setting the lower 8 bits to MULAL/MULBL. When the higher 8 bits of the multiplied data are the same value and only the lower 8 bits are to be changed, set the changed value to MULAL/MULBL and re-set the same value to MULAH/MULBH.

Table 20-2. Hardware Statuses After Reset Acknowledgment (1/4)

Hardware		After Reset Acknowledgment[Note 1]
Program counter (PC)		The contents of the reset vector table (0000H, 0001H) are set.
Stack pointer (SP)		Undefined
Program status word (PSW)		02H
RAM	Data memory	Undefined[Note 2]
	General-purpose registers	Undefined[Note 2]
Port registers 0, 2, 3, 6, 7, 12 (P0, P2, P3, P6, P7, P12) (output latches)		00H
Port mode registers 0, 2, 3, 6, 7 (PM0, PM2, PM3, PM6, PM7)		FFH
Pull-up resistor option registers 0, 3, 6 (PU0, PU3, PU6)		00H
Pull-up resistor option register 12 (PU12)		20H
Port input mode register 6 (PIM6)		00H
Port output mode register 6 (POM6)		00H
Reset pin mode register (RSTMASK)		00H
Port alternate switch control register (MUXSEL)		00H
Internal memory size switching register (IMS)		CFH[Note 3]

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.

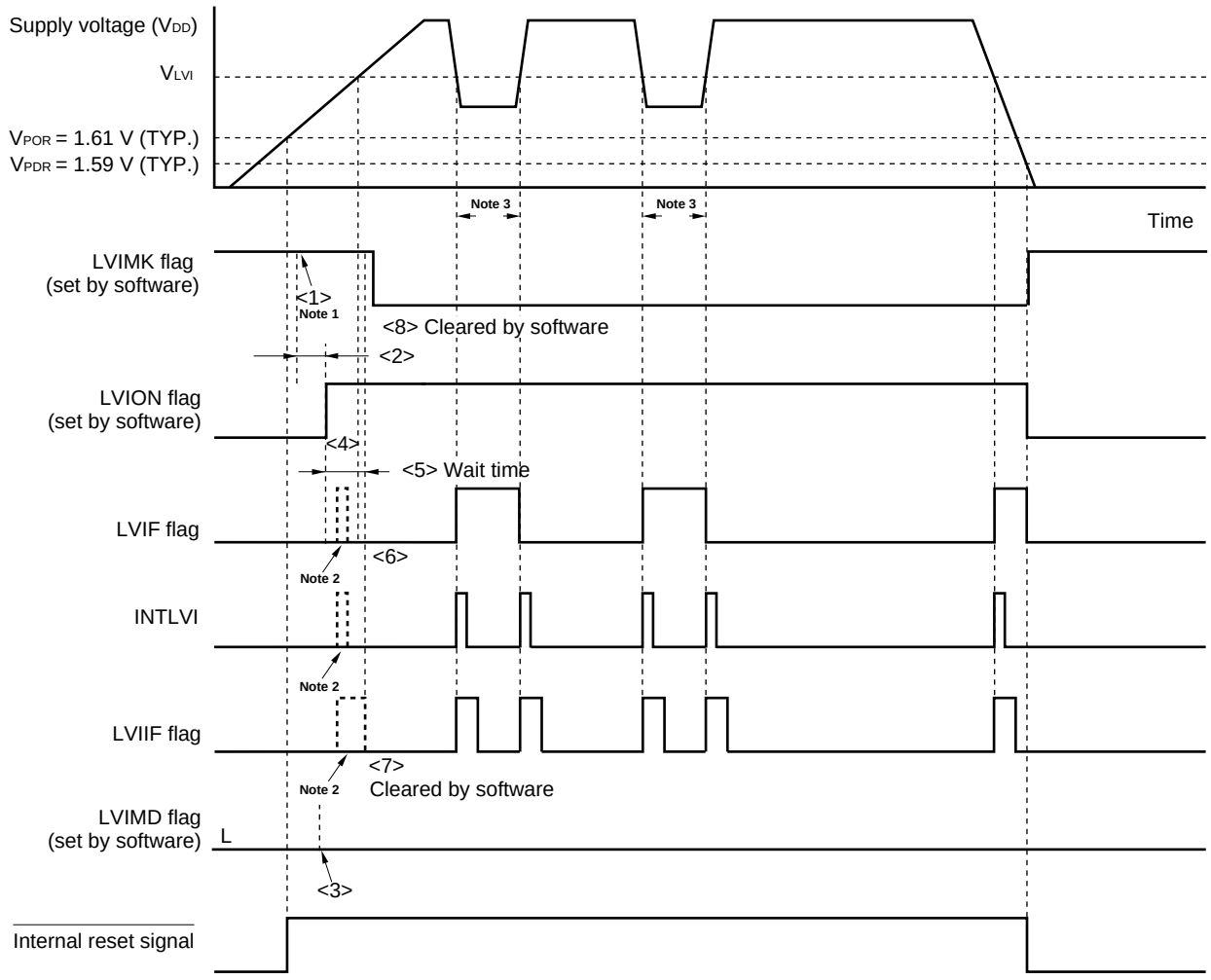
2. When a reset is executed in the standby mode, the pre-reset status is held even after reset.

3. Reset signal generation makes the setting of the ROM area undefined. Therefore, set the value corresponding to each product as indicated below after release of reset.

Products			IMS	ROM Capacity	Internal High-Speed RAM Capacity
78K0/IY2	78K0/IA2	78K0/IB2			
μPD78F0740, 78F0750	–	–	61H	4 KB	384 bytes
μPD78F0741, 78F0751	μPD78F0743, 78F0753	μPD78F0745, 78F0755	42H	8 KB	512 bytes
μPD78F0742, 78F0752	μPD78F0744, 78F0754	μPD78F0746, 78F0756	04H	16 KB	768 bytes

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

Figure 22-6. Timing of Low-Voltage Detector Interrupt Signal Generation (LVISTART = 0)



Notes 1. The LVIMK flag is set to “1” by reset signal generation.

2. The interrupt request signal (INTLVI) is generated and the LVIF and LVIIF flags may be set (1).

3. If LVI operation is disabled (clears LVION) when the supply voltage (V_{DD}) is less than or equal to the detection voltage (V_{LVI}), an interrupt request signal (INTLVI) is generated and LVIIF may be set to 1.

Remarks 1. <1> to <8> in Figure 22-6 above correspond to <1> to <8> in the description of “When starting operation” in **22.4.2 (1) When LVI default start function stopped is set (LVISTART = 0).**

2. V_{POR}: POC power supply rise detection voltage
V_{PDR}: POC power supply fall detection voltage

25.6 Security Settings

The 78K0/Ix2 microcontrollers support a security function that prohibits rewriting the user program written to the internal flash memory, so that the program cannot be changed by an unauthorized person.

The operations shown below can be performed using the Security Set command. The security setting is valid when the programming mode is set next.

- Disabling batch erase (chip erase)
  Execution of the block erase and batch erase (chip erase) commands for entire blocks in the flash memory is prohibited by this setting during on-board/off-board programming. Once execution of the batch erase (chip erase) command is prohibited, all of the prohibition settings (including prohibition of batch erase (chip erase)) can no longer be cancelled.

  Caution After the security setting for the batch erase is set, erasure cannot be performed for the device. In addition, even if a write command is executed, data different from that which has already been written to the flash memory cannot be written, because the erase command is disabled.

- Disabling block erase
  Execution of the block erase command for a specific block in the flash memory is prohibited during on-board/off-board programming. However, blocks can be erased by means of self programming.

- Disabling write
  Execution of the write and block erase commands for entire blocks in the flash memory is prohibited during on-board/off-board programming. However, blocks can be written by means of self programming.

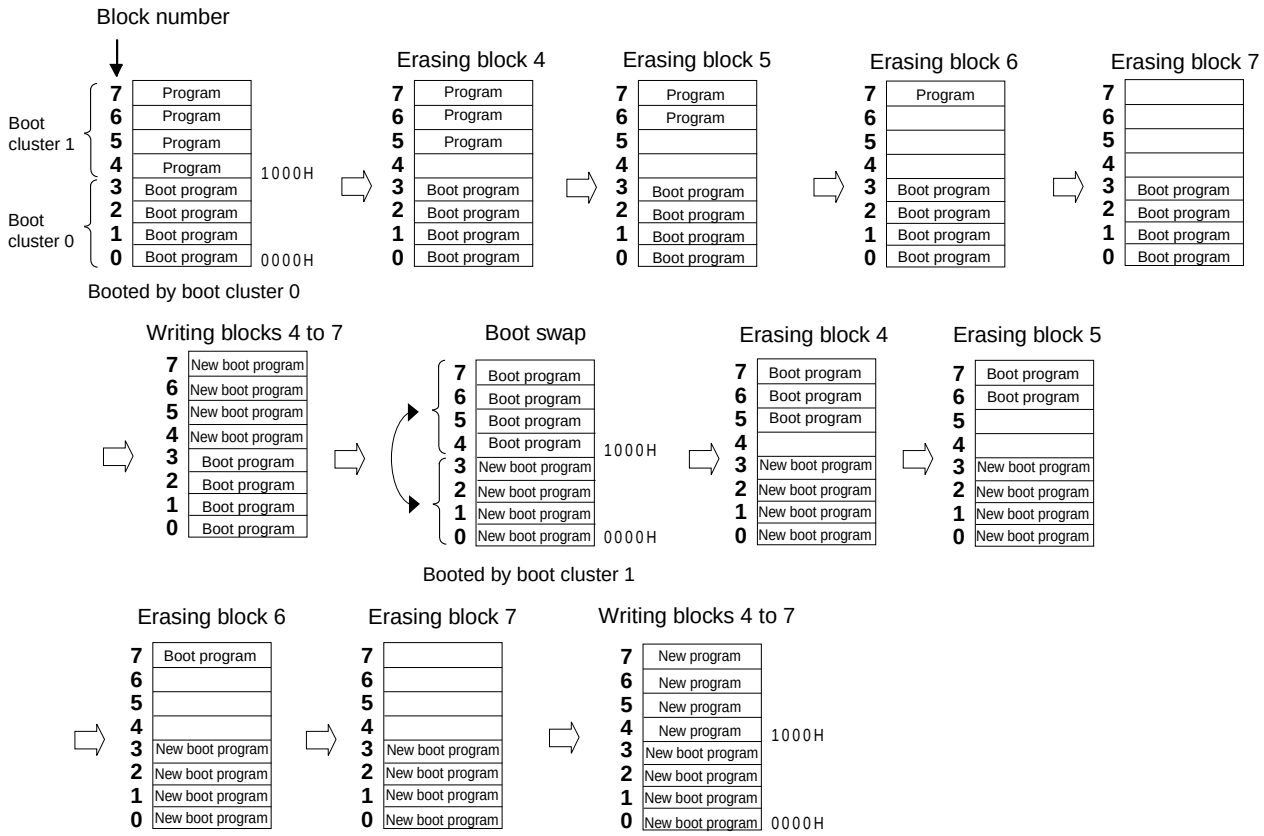
- Disabling rewriting boot cluster 0
  Execution of the block erase command and write command on boot cluster 0 (0000H to 0FFFH) in the flash memory is prohibited by this setting. Execution of the batch erase (chip erase) command is also prohibited by this setting.

The batch erase (chip erase), block erase, write commands, and rewriting boot cluster 0 are enabled by the default setting when the flash memory is shipped. Security can be set by on-board/off-board programming and self programming. Each security setting can be used in combination.

All the security settings are cleared by executing the batch erase (chip erase) command.

Table 25-7 shows the relationship between the erase and write commands when the 78K0/Ix2 microcontroller security function is enabled.

Figure 25-11. Example of Executing Boot Swapping



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

(T_A = –40 to +105°C, 2.7 V ≤ V_{DD} ≤ 5.5 V, AV_{REF} ≤ V_{DD}, V_{SS} = AV_{SS} = 0 V)

Parameter	Symbol	Conditions		MIN.	TYP.	MAX.	Unit
Input voltage, high	V_{IH1}	P37, P121, P122, P125		0.7V_{DD}		V_{DD}	V
	V_{IH2}	P20 to P27, P70	AV_{REF} = V_{DD}	0.7AV_{REF}		AV_{REF}	V
	V_{IH3}	P60, P61 (I/O port mode)		0.7V_{DD}		V_{DD}	V
	V_{IH4}	P00 to P02, P30 to P36, $\overline{\text{RESET}}$, EXCLK		0.8V_{DD}		V_{DD}	V
	V_{IH5}	P60, P61 (SMBus input mode)	3.4 V ≤ V_{DD} ≤ 5.5 V	0.8V_{DD}			V
			2.7 V ≤ V_{DD} < 3.4 V	2.1			V
	V_{IH6}	X1, X2		V_{DD} – 0.1		V_{DD}	V
Input voltage, low	V_{IL1}	P37, P121, P122, P125		0		0.3V_{DD}	V
	V_{IL2}	P20 to P27, P70	AV_{REF} = V_{DD}	0		0.3AV_{REF}	V
	V_{IL3}	P60, P61 (I/O port mode)		0		0.3V_{DD}	V
	V_{IL4}	P00 to P02, P30 to P36, $\overline{\text{RESET}}$, EXCLK		0		0.2V_{DD}	V
	V_{IL5}	P60, P61 (SMBus input mode)	3.4 V ≤ V_{DD} ≤ 5.5 V	0		0.2V_{DD}	V
			2.7 V ≤ V_{DD} < 3.4 V	0		0.8	V
	V_{IL6}	X1, X2		0		0.1	V
Output voltage, high	V_{OH1}	P00 to P02, P30 to P37, P60, P61	4.0 V ≤ V_{DD} ≤ 5.5 V, I_{OH1} = –3.0 mA	V_{DD} – 0.7			V
			2.7 V ≤ V_{DD} < 4.0 V, I_{OH1} = –2.5 mA	V_{DD} – 0.5			V
	V_{OH2}	P20 to P27, P70	AV_{REF} = V_{DD}, I_{OH2} = –100 μA	V_{DD} – 0.5			V

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

Caution 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.

A.1 Software Package

SP78K0 78K0 microcontroller software package	Development tools (software) common to the 78K0 microcontrollers are combined in this package.

A.2 Language Processing Software

RA78K0[Note 1] 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 (DF780756). **<Precaution when using RA78K0 in PC environment>** This assembler package is a DOS-based application. It can also be used in Windows, however, by using the Project Manager (PM+) on Windows. PM+ is included in assembler package.
CC78K0[Note 1] 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. **<Precaution when using CC78K0 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 (PM+) on Windows. PM+ is included in assembler package.
DF780756[Note 2] Device file	This file contains information peculiar to the device. This device file should be used in combination with a tool (RA78K0, CC78K0, ID78K0-QB, and SM+ for 78K0). The corresponding OS and host machine differ depending on the tool to be used.

Notes 1. If the versions of RA78K0 and CC78K0 are Ver.4.00 or later, different versions of RA78K0 and CC78K0 can be installed on the same machine.

2. The DF780756 can be used in common with the RA78K0, CC78K0, ID78K0-QB, and SM+ for 78K0. Download the DF780588 from the download site for development tools

<R> (https://secure-resource.renesas.com/micro/tool_reg/OdsListTop.do?lang=en).