Renesas Electronics America Inc - UPD78F0464GB-GAH-AX Datasheet

Welcome to E-XFL.COM

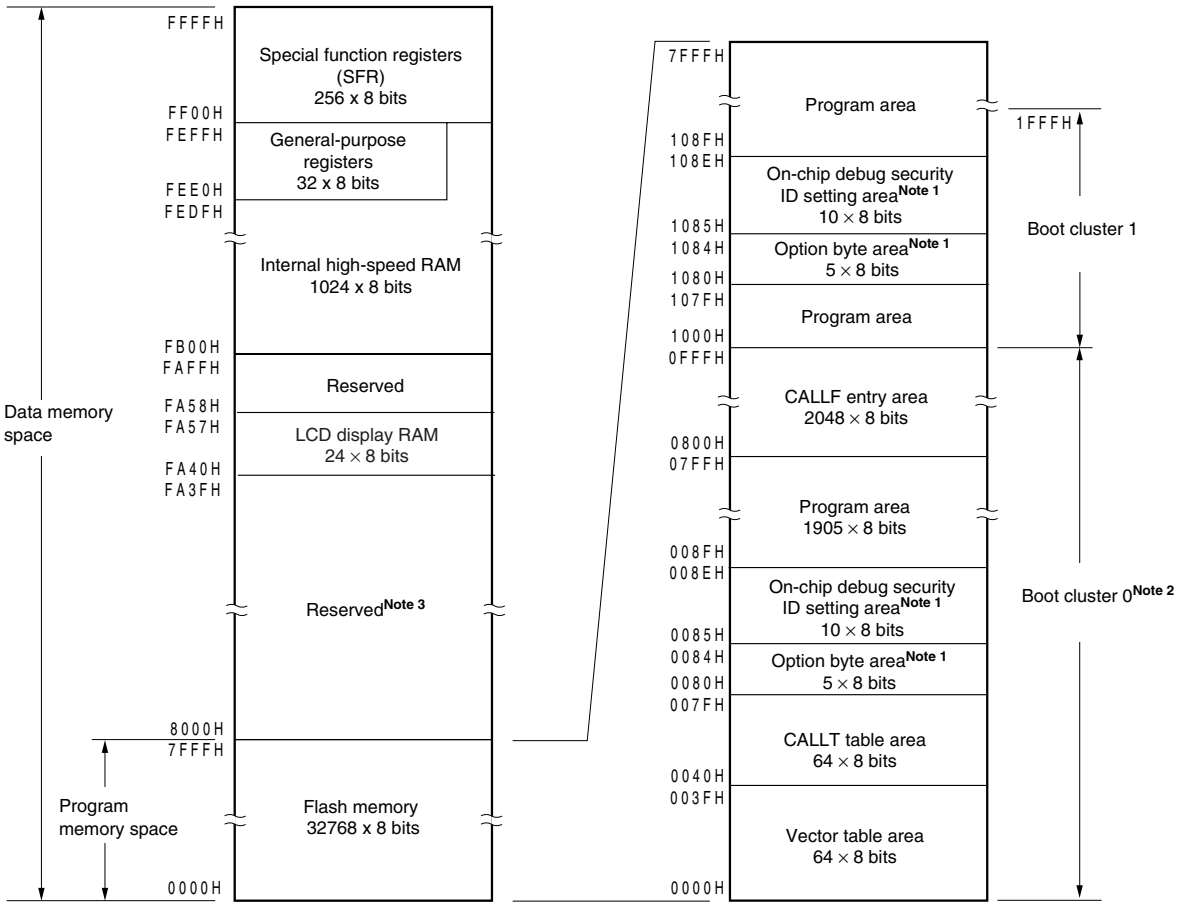
What is "Embedded - Microcontrollers"?

"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	3-Wire SIO, LINbus, UART/USART
Peripherals	LCD, LVD, POR, PWM, WDT
Number of I/O	46
Program Memory Size	48KB (48K x 8)
Program Memory Type	FLASH
EEPROM Size	-
RAM Size	2K x 8
Voltage - Supply (Vcc/Vdd)	1.8V ~ 5.5V
Data Converters	A/D 8x10b, 3x16b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	64-LQFP
Supplier Device Package	-
Purchase URL	https://www.e-xfl.com/product-detail/renesas-electronics-america/upd78f0464gb-gah-ax

Figure 3-6. Memory Map (μPD78F0463)



Notes 1. When boot swap is not used: Set the option bytes to 0080H to 0084H, and the on-chip debug security IDs to 0085H to 008EH.

When boot swap is used: Set the option bytes to 0080H to 0084H and 1080H to 1084H, and the on-chip debug security IDs to 0085H to 008EH and 1085H to 108EH.

2. Writing boot cluster 0 can be prohibited depending on the setting of security (see **27.8 Security Setting**).

<R> **3.** However, FA26H and FA27H can be used (See **13.3 Registers Used in 16-Bit ΔΣ Type A/D Converter**).

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

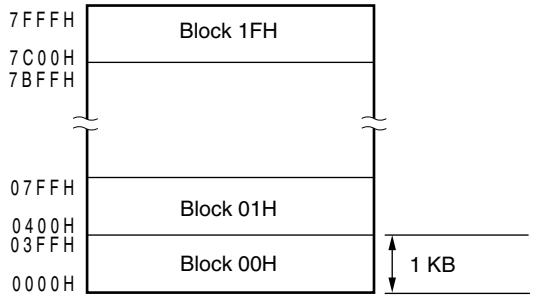
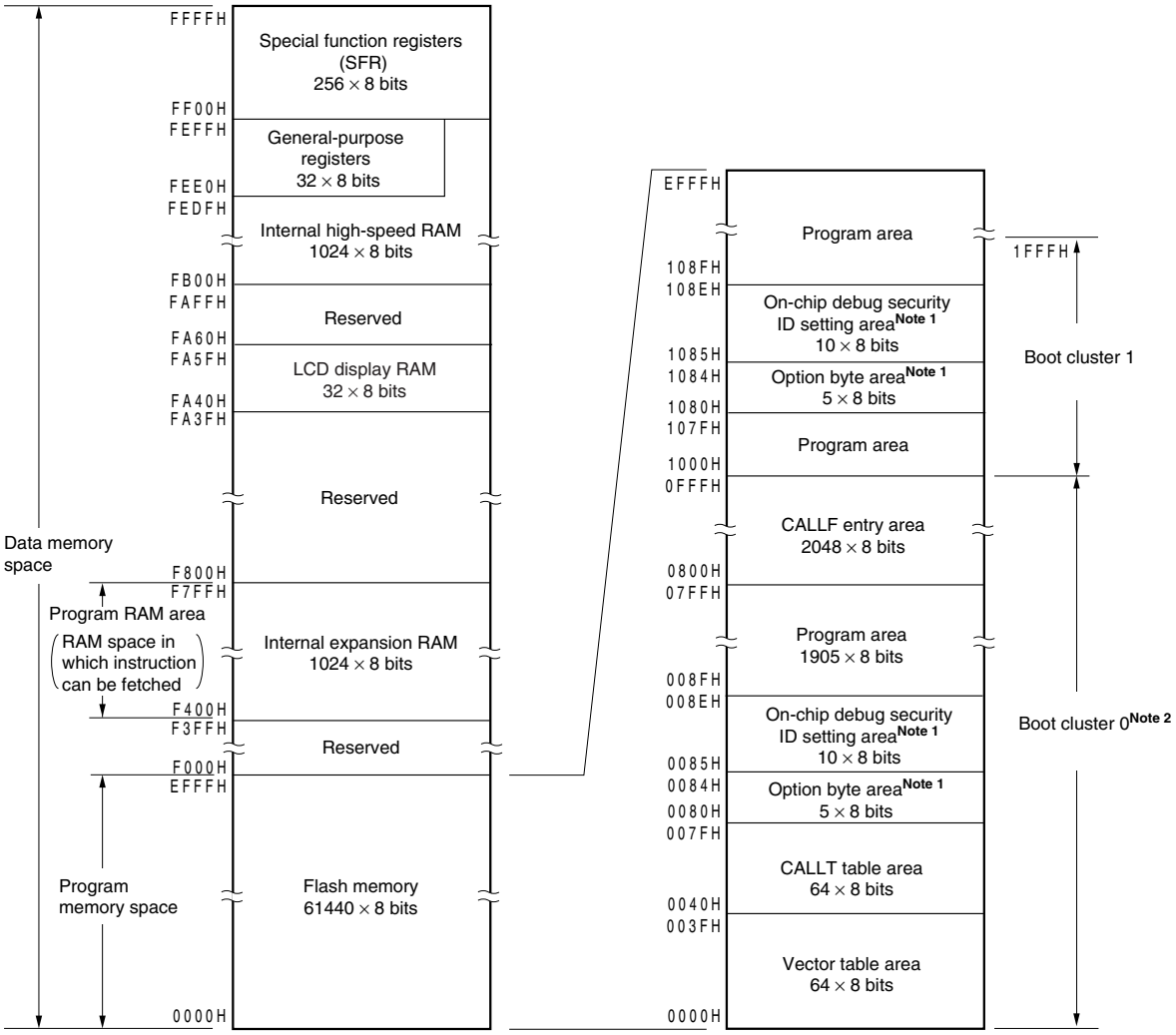


Figure 3-9. Memory Map (μPD78F0445, 78F0455)



Notes 1. When boot swap is not used: Set the option bytes to 0080H to 0084H, and the on-chip debug security IDs to 0085H to 008EH.

When boot swap is used: Set the option bytes to 0080H to 0084H and 1080H to 1084H, and the on-chip debug security IDs to 0085H to 008EH and 1085H to 108EH.

2. Writing boot cluster 0 can be prohibited depending on the setting of security (see **27.8 Security Setting**).

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

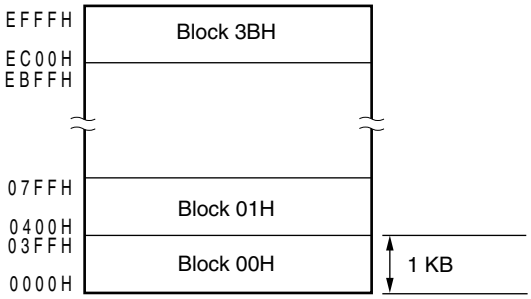


Table 4-2. Port Functions (1/2)

Function Name	I/O	Function	After Reset	Alternate Function
P11	I/O	Port 1. 4-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	$\overline{\text{SCK10}}$
P12				SI10/RxD0/<RxD6>
P13				SO10/TxD0/<TxD6>
P14				INTP4
P20	I/O	Port 2. 8-bit I/O port. Input/output can be specified in 1-bit units.	Digital input port	SEG31[Note1]/ANI0[Note2]/DS0−[Note3]
P21				SEG30[Note1]/ANI1[Note2]/DS0+[Note3]
P22				SEG29[Note1]/ANI2[Note2]/DS1−[Note3]
P23				SEG28[Note1]/ANI3[Note2]/DS1+[Note3]
P24				SEG27[Note1]/ANI4[Note2]/DS2−[Note3]
P25				SEG26[Note1]/ANI5[Note2]/DS2+[Note3]
P26				SEG25[Note1]/ANI6[Note2]/REF−[Note3]
P27				SEG24[Note1]/ANI7[Note2]/REF+[Note3]
P31	I/O	Port 3. 4-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	TOH1/INTP3
P32				TOH0/MCGO
P33				TI000/RTCDIV/RTCCL/BUZ/INTP2
P34				TI52/TI010/TO00/RTC1HZ/INTP1
P40	I/O	Port 4. 5-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	V_{LC3}/KR0
P41				RIN/KR1
P42				KR2
P43				TO51/TI51/KR3
P44				TO50/TI50/KR4
P80 to P83	I/O	Port 8. 4-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	SEG4 to SEG7

Notes **1.** μPD78F044x and 78F045x only.
2. μPD78F045x and 78F046x only.
3. μPD78F046x only.

Remark The functions within arrowheads (< >) can be assigned by setting the input switch control register (ISC).

Figure 5-2. Format of Clock Operation Mode Select Register (OSCCTL)

Address: FF9FH After reset: 00H R/W

Symbol	<7>	<6>	5	<4>	3	2	1	0
OSCCTL	EXCLK	OSCSEL	0	OSCSELS	0	0	0	0

EXCLK	OSCSEL	High-speed system clock pin operation mode	P121/X1 pin	P122/X2/EXCLK pin
0	0	Input port mode	Input port	
0	1	X1 oscillation mode	Crystal/ceramic resonator connection	
1	0	Input port mode	Input port	
1	1	External clock input mode	Input port	External clock input

Caution To change the value of EXCLK and OSCSEL, be sure to confirm that bit 7 (MSTOP) of the main OSC control register (MOC) is 1 (the X1 oscillator stops or the external clock from the EXCLK pin is disabled).
Be sure to clear bits 0 to 3, and 5 to "0".

Remark f_{XH}: High-speed system clock oscillation frequency

5.5 Clock Generator Operation

The clock generator generates the following clocks and controls the operation modes of the CPU, such as standby mode (see **Figure 5-1**).

- Main system clock f_{XP}
  - High-speed system clock f_{XH}
    X1 clock f_X
    External main system clock f_{EXCLK}
  - Internal high-speed oscillation clock f_{RH}
- Subsystem clock f_{SUB}
  - XT1 clock f_{XT}
- Internal low-speed oscillation clock f_{RL}
- CPU clock f_{CPU}
- Peripheral hardware clock f_{PRS}

The CPU starts operation when the internal high-speed oscillator starts outputting after a reset release in the 78K0/LE3, thus enabling the following.

(1) Enhancement of security function

When the X1 clock is set as the CPU clock by the default setting, the device cannot operate if the X1 clock is damaged or badly connected and therefore does not operate after reset is released. However, the start clock of the CPU is the internal high-speed oscillation clock, so the device can be started by the internal high-speed oscillation clock after a reset release. Consequently, the system can be safely shut down by performing a minimum operation, such as acknowledging a reset source by software or performing safety processing when there is a malfunction.

(2) Improvement of performance

Because the CPU can be started without waiting for the X1 clock oscillation stabilization time, the total performance can be improved.

When the power supply voltage is turned on, the clock generator operation is shown in Figure 5-13.

6.4 Operation of 16-Bit Timer/Event Counter 00

6.4.1 Interval timer operation

If bits 3 and 2 (TMC003 and TMC002) of the 16-bit timer mode control register (TMC00) are set to 11 (clear & start mode entered upon a match between TM00 and CR000), the count operation is started in synchronization with the count clock.

When the value of TM00 later matches the value of CR000, TM00 is cleared to 0000H and a match interrupt signal (INTTM000) is generated. This INTTM000 signal enables TM00 to operate as an interval timer.

Remarks 1. For the setting of I/O pins, see **6.3 (6) Port mode register 3 (PM3)**.

2. For how to enable the INTTM000 interrupt, see **CHAPTER 20 INTERRUPT FUNCTIONS**.

Figure 6-12. Block Diagram of Interval Timer Operation

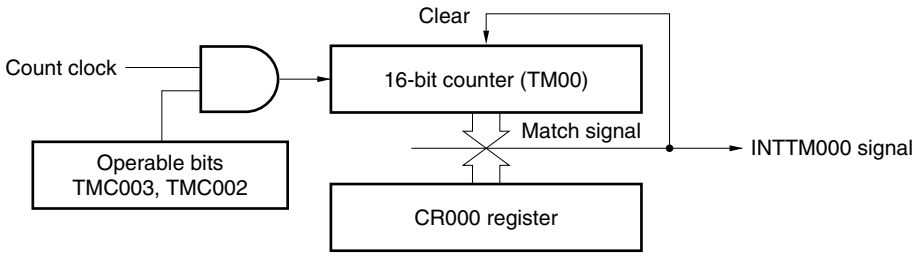


Figure 6-13. Basic Timing Example of Interval Timer Operation

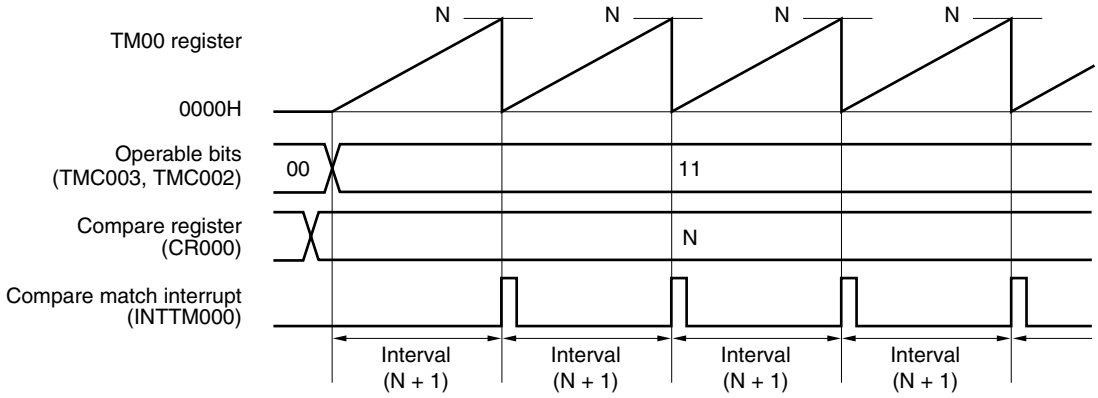
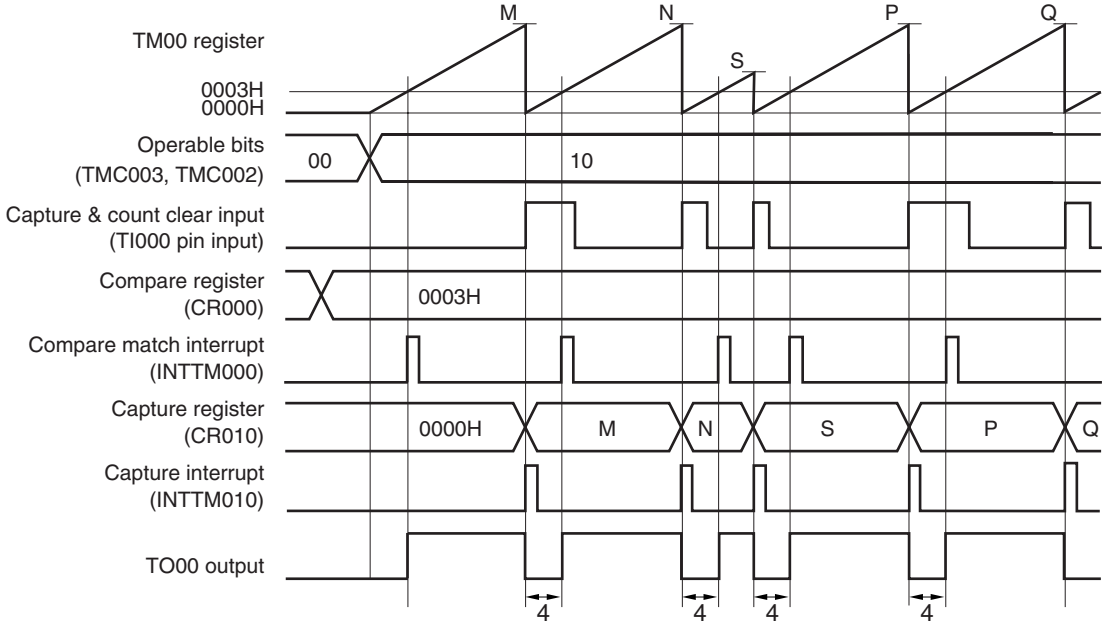


Figure 6-26. Timing Example of Clear & Start Mode Entered by TI000 Pin Valid Edge Input (CR000: Compare Register, CR010: Capture Register) (2/2)

(b) TOC00 = 13H, PRM00 = 10H, CRC00, = 04H, TMC00 = 0AH, CR000 = 0003H



This is an application example where the width set to CR000 (4 clocks in this example) is to be output from the TO00 pin when the count value has been captured & cleared.

The count value is captured to CR010, a capture interrupt signal (INTTM010) is generated, TM00 is cleared (to 0000H), and the TO00 output is inverted when the valid edge of the TI000 pin is detected. When the count value of TM00 is 0003H (four clocks have been counted), a compare match interrupt signal (INTTM000) is generated and the TO00 output level is inverted.

CHAPTER 7 8-BIT TIMER/EVENT COUNTERS 50, 51, AND 52

7.1 Functions of 8-Bit Timer/Event Counters 50, 51, and 52

8-bit timer/event counters 50 and 51 have the following functions.

- Interval timer
- External event counter[Note 1]
- Square-wave output[Note 2]
- PWM output[Note 2]

Notes 1. TM52 and TM00 can be connected in cascade to be used as an external 24-bit event counter. Also, the external event input of TM52 can be input enable-controlled via TMH2. For details, see **CHAPTER 6 16-BIT TIMER/EVENT COUNTER 00**.

2. TM50 and TM51 only.

7.2 Configuration of 8-Bit Timer/Event Counters 50, 51, and 52

8-bit timer/event counters 50, 51, and 52 include the following hardware.

Table 7-1. Configuration of 8-Bit Timer/Event Counters 50, 51, and 52

Item	Configuration
Timer register	8-bit timer counter 5n (TM5n)
Register	8-bit timer compare register 5n (CR5n)
Timer input	TI5n
Timer output	TO50, TO51
Control registers	Timer clock selection register 5n (TCL5n) 8-bit timer mode control register 5n (TMC5n) Input switch control register (ISC) Port mode register 3 (PM3) or port mode register 4 (PM4) Port register 3 (P3) or port register 4 (P4)

Remark n = 0 to 2

Figures 7-1 to 7-3 show the block diagrams of 8-bit timer/event counters 50, 51, and 52.

(1) 8-bit timer H compare register 0n (CMP0n)

This register can be read or written by an 8-bit memory manipulation instruction. This register is used in all of the timer operation modes.

This register constantly compares the value set to CMP0n with the count value of the 8-bit timer counter Hn and, when the two values match, generates an interrupt request signal (INTTMHn) and inverts the output level of TOHn.

Rewrite the value of CMP0n while the timer is stopped (TMHEn = 0).

A reset signal generation sets this register to 00H.

Figure 8-4. Format of 8-Bit Timer H Compare Register 0n (CMP0n)

Address: FF18H (CMP00), FF1AH (CMP01), FF44H (CMP02) After reset: 00H R/W

Symbol	7	6	5	4	3	2	1	0
CMP0n (n = 0 to 2)								

Caution CMP0n cannot be rewritten during timer count operation. CMP0n can be refreshed (the same value is written) during timer count operation.

(2) 8-bit timer H compare register 1n (CMP1n)

This register can be read or written by an 8-bit memory manipulation instruction. This register is used in the PWM output mode and carrier generator mode.

In the PWM output mode, this register constantly compares the value set to CMP1n with the count value of the 8-bit timer counter Hn and, when the two values match, inverts the output level of TOHn. No interrupt request signal is generated.

In the carrier generator mode, the CMP1n register always compares the value set to CMP1n with the count value of the 8-bit timer counter Hn and, when the two values match, generates an interrupt request signal (INTTMHn). At the same time, the count value is cleared.

CMP1n can be rewritten during timer count operation.

If the value of CMP1n is rewritten while the timer is operating, the new value is latched and transferred to CMP1n when the count value of the timer matches the old value of CMP1n, and then the value of CMP1n is changed to the new value. If matching of the count value and the CMP1n value and writing a value to CMP1n conflict, the value of CMP1n is not changed.

A reset signal generation sets this register to 00H.

Figure 8-5. Format of 8-Bit Timer H Compare Register 1n (CMP1n)

Address: FF19H (CMP10), FF1BH (CMP11), FF45H (CMP12) After reset: 00H R/W

Symbol	7	6	5	4	3	2	1	0
CMP1n (n = 0 to 2)								

Caution In the PWM output mode and carrier generator mode, be sure to set CMP1n when starting the timer count operation (TMHEn = 1) after the timer count operation was stopped (TMHEn = 0) (be sure to set again even if setting the same value to CMP1n).

Remark n = 0 to 2, however, TOH0 and TOH1 only for TOHn

Figure 8-8. Format of 8-Bit Timer H Mode Register 2 (TMHMD2)

Address: FF42H After reset: 00H R/W

	<7>	6	5	4	3	2	<1>	<0>
TMHMD2	TMHE2	CKS22	CKS21	CKS20	TMMD21	TMMD20	TOLEV2	TOEN2

TMHE2	Timer operation enable
0	Stops timer count operation (counter is cleared to 0)
1	Enables timer count operation (count operation started by inputting clock)

CKS22	CKS21	CKS20	Count clock selection[Note 1]			
				f_{PRS} = 2 MHz	f_{PRS} = 5 MHz	f_{PRS} = 10 MHz
0	0	0	f_{PRS}[Note 2]	2 MHz	5 MHz	10 MHz
0	0	1	$f_{PRS}/2$	1 MHz	2.5 MHz	5 MHz
0	1	0	$f_{PRS}/2^2$	500 kHz	1.25 MHz	2.5 MHz
0	1	1	$f_{PRS}/2^4$	125 kHz	312.5 kHz	625 kHz
1	0	0	$f_{PRS}/2^6$	31.25 kHz	78.13 kHz	156.25 kHz
1	0	1	$f_{PRS}/2^{10}$	1.95 kHz	4.88 kHz	9.77 kHz
1	1	0	$f_{PRS}/2^{12}$	0.49 kHz	1.22 kHz	2.44 kHz
Other than above			Setting prohibited			

TMMD21	TMMD20	Timer operation mode
0	0	Interval timer mode
1	0	Input enable width adjust mode for pins (PWM mode)
Other than above		Setting prohibited

TOLEV2	Timer output level control (in default mode)
0	Low level
1	High level

TOEN2	Timer output control
0	Disables output
1	Enables output[Note 3]

Notes 1. If the peripheral hardware clock (f_{PRS}) operates on the high-speed system clock (f_{XH}) (XSEL = 1), the f_{PRS} operating frequency varies depending on the supply voltage.
- V_{DD} = 2.7 to 5.5 V: $f_{PRS} \leq$ 10 MHz
- V_{DD} = 1.8 to 2.7 V: $f_{PRS} \leq$ 5 MHz

2. If the peripheral hardware clock (f_{PRS}) operates on the internal high-speed oscillation clock (f_{RH}) (XSEL = 0), when 1.8 V $\leq V_{DD} <$ 2.7 V, the setting of CKS22 = CKS21 = CKS20 = 0 (count clock: f_{PRS}) is prohibited.

3. The timer output of TMH2 can only be used as an external event input enable signal of TM52. No pins for external output are available.

Caution When TMHE2 = 1, setting the other bits of TMHMD2 is prohibited.

Remark f_{PRS}: Peripheral hardware clock frequency

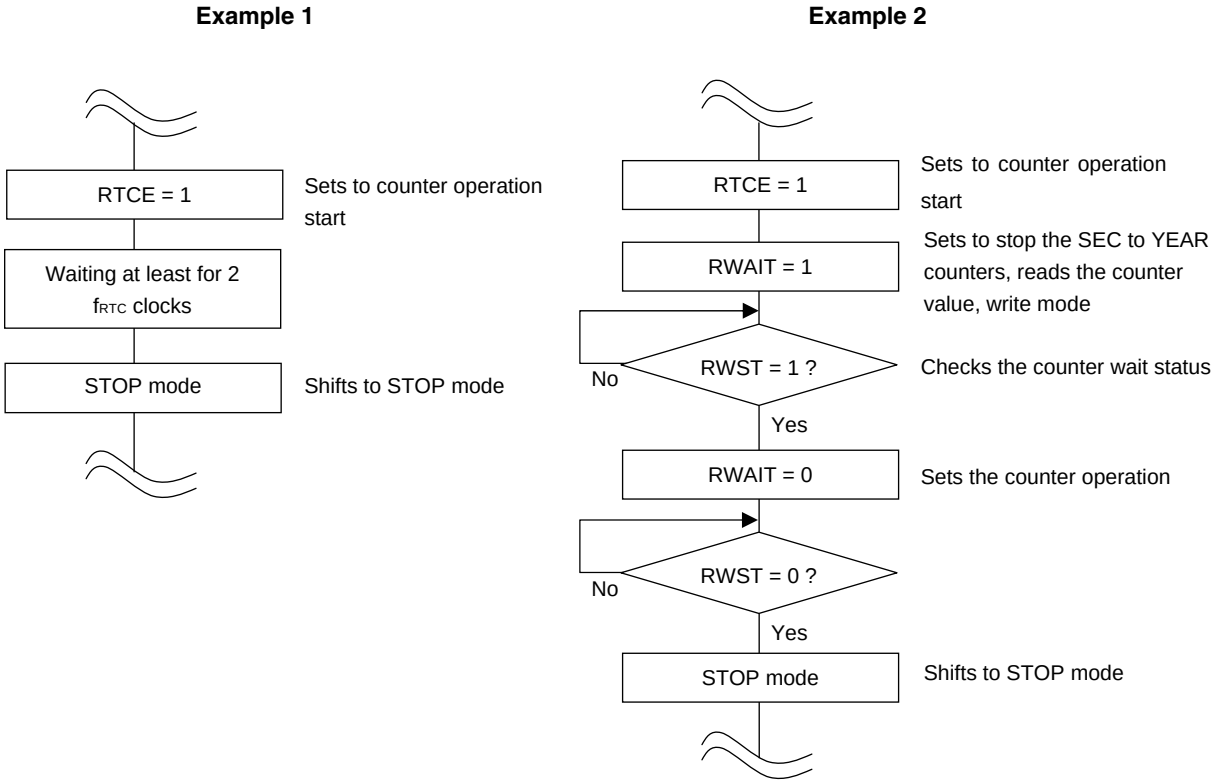
<R> **9.4.2 Shifting to STOP mode after starting operation**

Perform one of the following processing when shifting to STOP mode immediately after setting RTCE to 1.

However, after setting RTCE to 1, this processing is not required when shifting to STOP mode after the first INTRTC interrupt has occurred.

- Shifting to STOP mode when at least two input clocks (f_{RTC}) have elapsed after setting RTCE to 1 (see **Figure 9-19**, **Example 1**).
- Checking by polling RWST to become 1, after setting RTCE to 1 and then setting RWAIT to 1. Afterward, setting RWAIT to 0 and shifting to STOP mode after checking again by polling that RWST has become 0 (see **Figure 9-19**, **Example 2**).

Figure 9-19. Procedure for Shifting to STOP Mode After Setting RTCE to 1



CHAPTER 12 10-BIT SUCCESSIVE APPROXIMATION TYPE A/D CONVERTER (μPD78F045x and 78F046x only)

12.1 Function of 10-Bit Successive Approximation Type A/D Converter

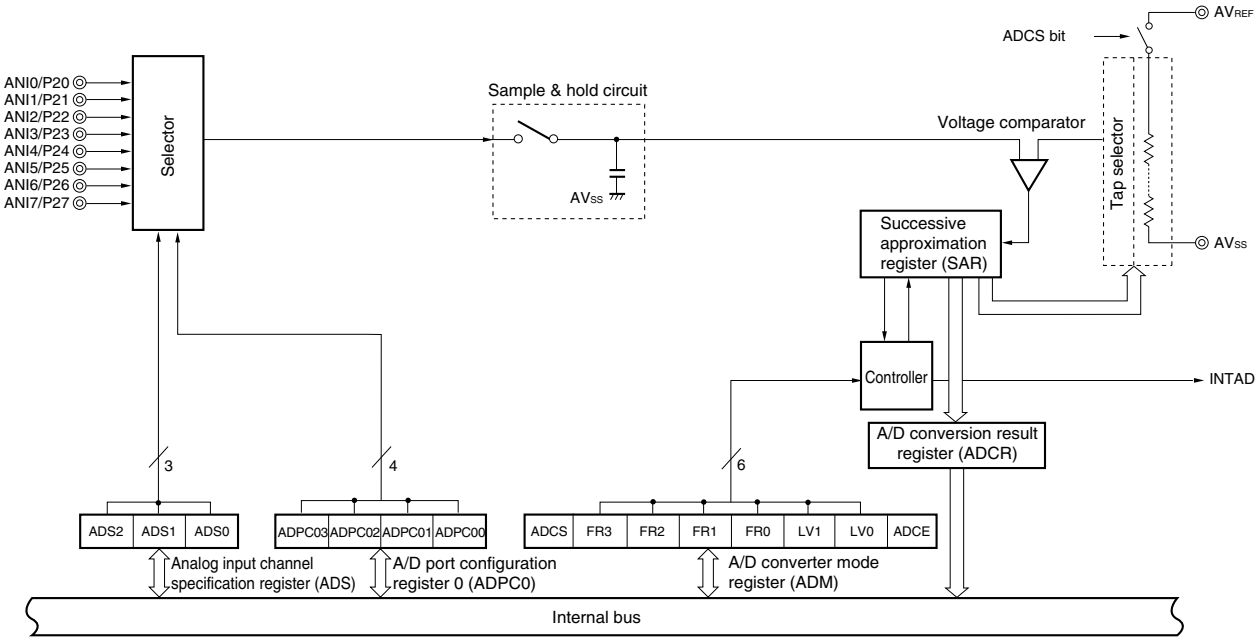
The 10-bit successive approximation type A/D converter converts an analog input signal into a digital value, and consists of up to eight channels (ANI0 to ANI7) with a resolution of 10 bits.

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 ANI7. Each time an A/D conversion operation ends, an interrupt request (INTAD) is generated.

Figure 12-1. Block Diagram of 10-Bit Successive Approximation type A/D Converter



12.4 10-Bit Successive Approximation Type A/D Converter Operations

12.4.1 Basic operations of A/D converter

<1> Set bit 0 (ADCE) of the A/D converter mode register (ADM) to 1 to start the operation of the comparator.

<2> Set channels for A/D conversion to analog input by using the A/D port configuration register (ADPC0) and set to input mode by using port mode register 2 (PM2).

<3> Set A/D conversion time by using bits 6 to 1 (FR3 to FR0, LV1, and LV0) of ADM.

<4> Select one channel for A/D conversion using the analog input channel specification register (ADS).

<5> Start the conversion operation by setting bit 7 (ADCS) of ADM to 1.
(<6> to <12> are operations performed by hardware.)

<6> The voltage input to the selected analog input channel is sampled by the sample & hold circuit.

<7> When sampling has been done for a certain time, the sample & hold circuit is placed in the hold state and the sampled voltage is held until the A/D conversion operation has ended.

<8> Bit 9 of the successive approximation register (SAR) is set. The series resistor string voltage tap is set to (1/2) AV_{REF} by the tap selector.

<9> The voltage difference between the series resistor string voltage tap and sampled voltage is compared by the voltage comparator. If the analog input is greater than (1/2) AV_{REF}, the MSB of SAR remains set to 1. If the analog input is smaller than (1/2) AV_{REF}, the MSB is reset to 0.

<10> Next, bit 8 of SAR is automatically set to 1, and the operation proceeds to the next comparison. The series resistor string voltage tap is selected according to the preset value of bit 9, as described below.

- Bit 9 = 1: (3/4) AV_{REF}
- Bit 9 = 0: (1/4) AV_{REF}

The voltage tap and sampled voltage are compared and bit 8 of SAR is manipulated as follows.

- Analog input voltage ≥ Voltage tap: Bit 8 = 1
- Analog input voltage < Voltage tap: Bit 8 = 0

<11> Comparison is continued in this way up to bit 0 of SAR.

<12> Upon completion of the comparison of 10 bits, an effective digital result value remains in SAR, and the result value is transferred to the A/D conversion result register (ADCR, ADCRH) and then latched.
At the same time, the A/D conversion end interrupt request (INTAD) can also be generated.

<13> Repeat steps <6> to <12>, until ADCS is cleared to 0.
To stop the A/D converter, clear ADCS to 0.
To restart A/D conversion from the status of ADCE = 1, start from <5>. To start A/D conversion again when ADCE = 0, set ADCE to 1, wait for 1 μs or longer, and start <5>. To change a channel of A/D conversion, start from <4>.

Caution Make sure the period of <1> to <5> is 1 μs or more.

Remark Two types of A/D conversion result registers are available.
- ADCR (16 bits): Store 10-bit A/D conversion value
- ADCRH (8 bits): Store 8-bit A/D conversion value

<R> **(11) Internal equivalent circuit**

The equivalent circuit of the analog input block is shown below.

Figure 12-22. Internal Equivalent Circuit of ANIn Pin

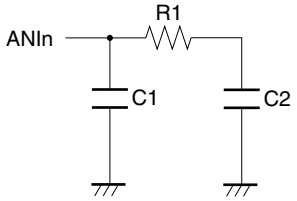


Table 12-4. Resistance and Capacitance Values of Equivalent Circuit (Reference Values)

AV_{REF}	R1	C1	C2
$4.0\ V \le AV_{REF} \le 5.5\ V$	8.1 kΩ	8 pF	5 pF
$2.7\ V \le AV_{REF} < 4.0\ V$	31 kΩ	8 pF	5 pF
$2.3\ V \le AV_{REF} < 2.7\ V$	381 kΩ	8 pF	5 pF

Remarks 1. The resistance and capacitance values shown in Table 12-4 are not guaranteed values.

2. n = 0 to 7

(12) Simultaneous use of the 10-bit successive approximation type A/D converter and the 16-bit ΔΣ type A/D converter (μPD78F046x only)

The A/D conversion accuracy may deteriorate when the 10-bit successive approximation type A/D converter and the 16-bit ΔΣ type A/D converter are used at the same time.

Stop the 16-bit ΔΣ type A/D converter during 10-bit successive approximation type A/D converter operation, because the accuracy cannot be guaranteed. Also, stop the 10-bit successive approximation type A/D converter during 16-bit ΔΣ type A/D converter operation. (Do not operate them simultaneously.)

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

<1> Set the CSIC10 register (see **Figures 16-3**).
<2> Set bits 4 and 6 (DIR10 and TRMD10) of the CSIM10 register (see **Figures 16-2**).
<3> Set bit 7 (CSIE10) of the CSIM10 register to 1. → Transmission/reception is enabled.
<4> Write data to transmit buffer register 10 (SOTB10). → Data transmission/reception is started.
Read data from serial I/O shift register 10 (SIO10). → Data reception is started.

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

The relationship between the register settings and pins is shown below.

Table 16-2. Relationship Between Register Settings and Pins

CSIE10	TRMD10	PM12	P12	PM13	P13	PM11	P11	CSI10 Operation	Pin Function		
									SI10/RxD0/ <RxD6>/P12	SO10/TxD0/ <TxD6>/P13	$\overline{\text{SCK10}}$/ P11
0	×	×Note 1	×Note 1	×Note 1	×Note 1	×Note 1	×Note 1	Stop	RxD0/ <RxD6>/P12	TxD0/ <TxD6>/P13	P11Note 2
1	0	1	×	×Note 1	×Note 1	1	×	Slave receptionNote 3	SI10	TxD0/ <TxD6>/P13	$\overline{\text{SCK10}}$ (input)Note 3
1	1	×Note 1	×Note 1	0	0	1	×	Slave transmissionNote 3	RxD0/ <RxD6>/P12	SO10	$\overline{\text{SCK10}}$ (input)Note 3
1	1	1	×	0	0	1	×	Slave transmission/ receptionNote 3	SI10	SO10	$\overline{\text{SCK10}}$ (input)Note 3
1	0	1	×	×Note 1	×Note 1	0	1	Master reception	SI10	TxD0/ <TxD6>/P13	$\overline{\text{SCK10}}$ (output)
1	1	×Note 1	×Note 1	0	0	0	1	Master transmission	RxD0/ <RxD6>/P12	SO10	$\overline{\text{SCK10}}$ (output)
1	1	1	×	0	0	0	1	Master transmission/ reception	SI10	SO10	$\overline{\text{SCK10}}$ (output)

Notes 1. Can be set as port function.
2. To use P11/$\overline{\text{SCK10}}$ as port pins, clear CKP10 to 0.
3. To use the slave mode, set CKS102, CKS101, and CKS100 to 1, 1, 1.

Remark ×: don't care
CSIE10: Bit 7 of serial operation mode register 10 (CSIM10)
TRMD10: Bit 6 of CSIM10
CKP10: Bit 4 of serial clock selection register 10 (CSIC10)
CKS102, CKS101, CKS100: Bits 2 to 0 of CSIC10
PM1×: Port mode register
P1×: Port output latch

(a) When both diodes A and B are missing

When both diodes A and B are missing, the segment key scan function cannot be used, because of the following.

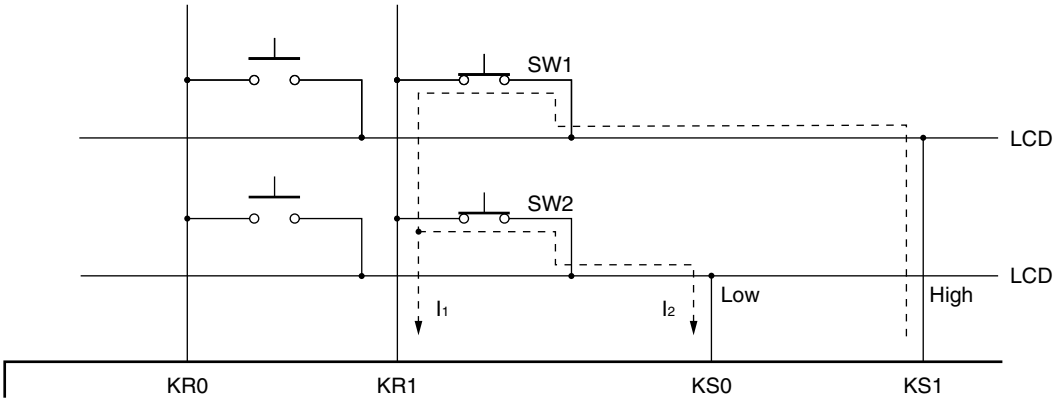
The following figure shows a circuit example when both diodes A and B are missing.

Assume, as shown in the figure below, that switches SW1 and SW2 are turned on, and a high level and a low level are output from the KS1 pin and KS0 pin, respectively.

When diode A is missing at this time, currents I_1 and I_2, shown as broken lines, will flow.

Consequently, the high level of KS1 and the low level of KS0 will not be output normally due to I_2, and the key input data of KR1 will become undefined.

Furthermore, the LCD display will not be turned on or off normally.



(b) When only diode A exists

When only diode A exists, whether switches are pressed simultaneously cannot be identified, due to the following.

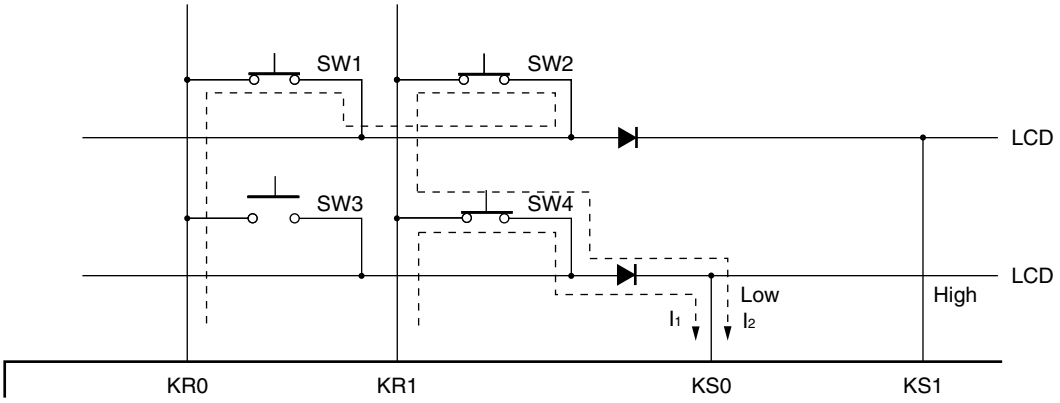
The following figure shows a circuit example when only diode A exists.

Assume, as shown in the figure below, that switches SW1, SW2, and SW4 are turned on, and a high level and a low level are output from the KS1 pin and KS0 pin, respectively.

At this time currents I_1 and I_2, shown as broken lines, will flow.

Consequently, even though SW3 is turned off, SW3 is identified to be turned on, because a low level is input to KR0 due to I_2.

There is no interference with the LCD display.



(c) When only diode B exists

Identification of at least three switches being pressed simultaneously can be performed.

There is no interference with the LCD display.

(2) When detecting level of input voltage from external input pin (EXLVI)

- When starting operation
  - <1> Mask the LVI interrupt (LVIMK = 1).
  - <2> Set bit 2 (LVISEL) of the low-voltage detection register (LVIM) to 1 (detects level of input voltage from external input pin (EXLVI)).
  - <3> Set bit 7 (LVION) of LVIM to 1 (enables LVI operation).
  - <4> Use software to wait for an operation stabilization time (10 μs (MAX.)).
  - <5> Confirm that "input voltage from external input pin (EXLVI) $\geq$ detection voltage (V_{EXLVI} = 1.21 V (TYP.)" when detecting the falling edge of EXLVI, or "input voltage from external input pin (EXLVI) < detection voltage (V_{EXLVI} = 1.21 V (TYP.)" when detecting the rising edge of EXLVI, at bit 0 (LVIF) of LVIM.
  - <6> Clear the interrupt request flag of LVI (LVIIF) to 0.
  - <7> Release the interrupt mask flag of LVI (LVIMK).
  - <8> Clear bit 1 (LVIMD) of LVIM to 0 (generates interrupt signal when the level is detected) (default value).
  - <9> Execute the EI instruction (when vector interrupts are used).

  Figure 25-8 shows the timing of the interrupt signal generated by the low-voltage detector. The numbers in this timing chart correspond to <1> to <8> above.

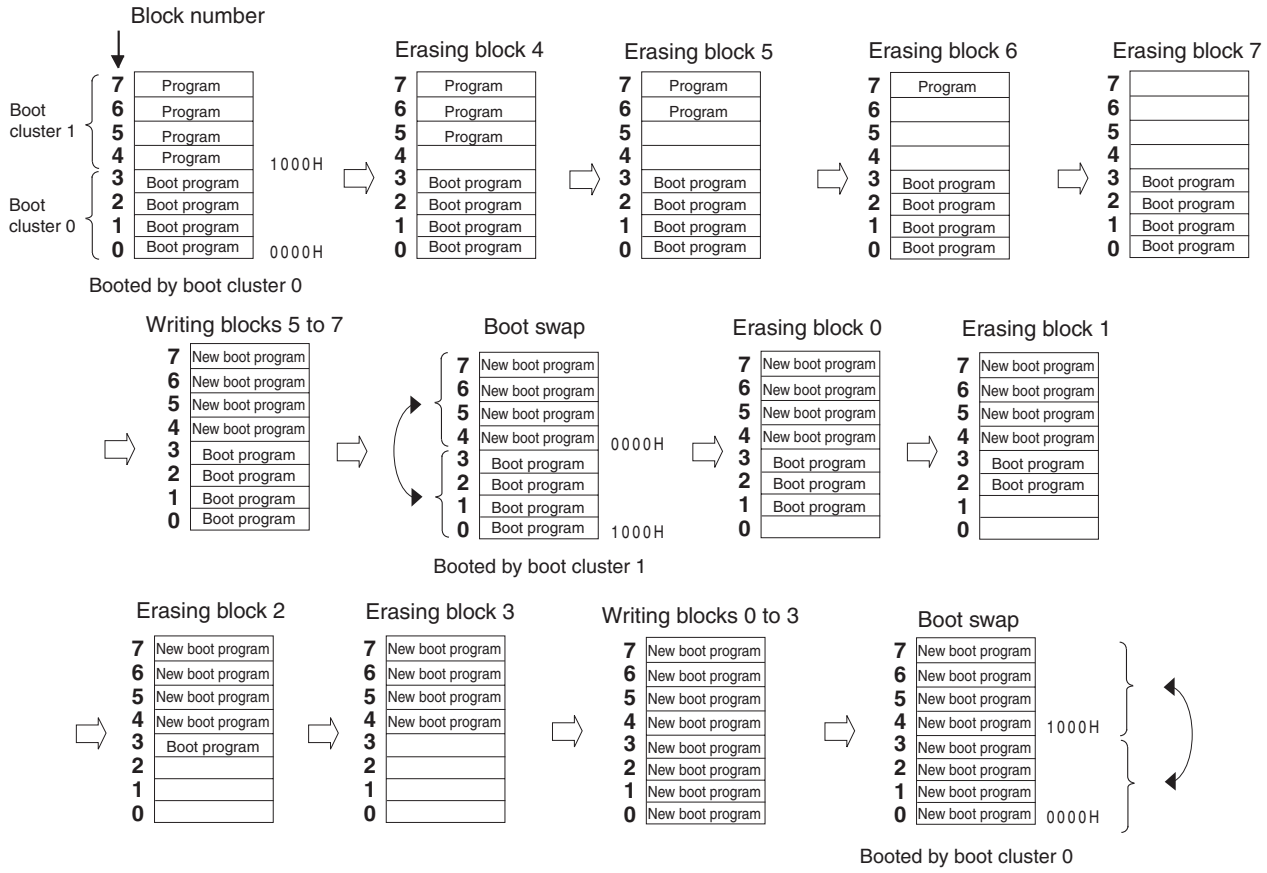
  Caution Input voltage from external input pin (EXLVI) must be EXLVI < V_{DD}.

- When stopping operation
  Either of the following procedures must be executed.

  - When using 8-bit memory manipulation instruction:
    Write 00H to LVIM.

  - When using 1-bit memory manipulation instruction:
    Clear LVION to 0.

Figure 27-18. Example of Executing Boot Swapping



<R> **16-bit ΔΣ type A/D Converter Characteristics (μPD78F046x only)**

(T_A = –40 to +85°C, 2.7 V ≤ AV_{REF} ≤ V_{DD} ≤ 5.5 V, V_{SS} = AV_{SS} = 0 V)

Parameter	Symbol	Conditions			MIN.	TYP.	MAX.	Unit
Resolution	R_{ES2}				8		16	bit
Sampling clock[Note 1]	f_{VP}	At differential input		3.5 V ≤ AV_{REF} ≤ 5.5 V	0.016		1.25	MHz
				2.7 V ≤ AV_{REF} < 3.5 V	0.016		0.625	MHz
		At single input		2.85 V ≤ AV_{REF} ≤ 5.5 V	0.016		0.625	MHz
				2.7 V ≤ AV_{REF} < 2.85 V	0.016		0.525	MHz
Integral non-linearity error (relative accuracy)	I_{LE2}	At differential input[Note 2]	14-bit resolution[Note 3]	AV_{REF} = 5.0 V		±1.0		LSB
				3.5 V ≤ AV_{REF} ≤ 5.5 V		±1.7		LSB
				2.7 V ≤ AV_{REF} < 3.5 V		±2.6		LSB
		At single input[Note 2]	12-bit resolution[Note 3]			±2.8		LSB
Differential non-linearity error (relative accuracy)	D_{LE2}	At differential input[Note 2]	14-bit resolution[Note 3]	AV_{REF} = 5.0 V		±1.0		LSB
				3.5 V ≤ AV_{REF} ≤ 5.5 V		±1.7		LSB
				2.7 V ≤ AV_{REF} < 3.5 V		±2.6		LSB
		At single input[Note 2]	12-bit resolution[Note 3]			±2.8		LSB
Offset	EOS	At differential input				±0.032		%FSR
		At single input				±0.16		%FSR
Gain error	GE	At differential input				±0.09		%
		At single input				±0.1		%
Reference voltage	REF+					AV_{REF}		V
	REF–					AV_{SS}		V
Analog input voltage	V_{AIN2}	In high-accuracy mode OFF			0		REF+	V
		In high-accuracy mode ON			0.1REF+		0.9REF+	V

Notes 1. The conversion time can be calculated by using the following expression, based on the sampling clock (f_{VP}) and set resolution (N bits).

Conversion time = 2^N / f_{VP}

2. These values apply when the high-accuracy mode is set to be on during differential input, or when the high-accuracy mode is set to be off during single input.

3. The characteristics of resolutions (N bits) other than those stated as conditions in the integral linearity error (I_{LE2}) and differential linearity error (D_{LE2}) columns can be calculated by using the following expressions.

- During differential input

  I_{LE2} in N-bit resolution = I_{LE2} in 14-bit resolution × $2^{(N-14)}$

  D_{LE2} in N-bit resolution = D_{LE2} in 14-bit resolution × $2^{(N-14)}$

- During single input

  I_{LE2} in N-bit resolution = I_{LE2} in 12-bit resolution × $2^{(N-12)}$

  D_{LE2} in N-bit resolution = D_{LE2} in 12-bit resolution × $2^{(N-12)}$

Remark In the 16-bit ΔΣ type A/D converter characteristics, the approximation line is defined by the least-squares method.