RL78/I1D

R01AN3997EJ0100
Rev.1.00
Nov. 9, 2017

Implementation of Highly Accurate Interval Timer by Low-Speed On-Chip Oscillator Clock CC-RL

Introduction

This application note describes how to improve the accuracy of the interval timer interval using the low-speed on-chip oscillator. For this purpose, the high-speed on-chip oscillator clock, which has high frequency accuracy, is used.

Target Device

RL78/I1D

RL78/G11

When applying the sample program covered in this application note to another microcomputer, modify the program according to the specifications for the target microcomputer and conduct an extensive evaluation of the modified program.

目次

1. Specifications

The oscillation period of the low-speed on-chip oscillator clock is measured by using the high-speed on-chip oscillator clock, which has high frequency accuracy.

The oscillation period of the low-speed on-chip oscillator clock is measured by using the input pulse interval measurement function of the timer array unit channel 1. The compare value of the 8-bit interval timer 0 is calculated based on the measurement result of the input pulse interval.

By setting the calculated compare value to the 8-bit interval timer compare register 0 to correct the interval time, the accuracy of the interval time is improved.

With this application, a compare match interrupt request of the 8-bit interval timer 0 is generated at approximately 20-second intervals to return from STOP mode, and A/D conversion is performed. The A/D conversion result is checked, and if it is equal to or greater than the reference value, buzzer output is provided. Buzzer output is stopped by pressing the switch. In addition, the interval time of the 8-bit interval timer 0 is corrected to maintain the accuracy of the interval time.

Table 1.1 shows the peripheral functions and applications, and Figure 1.1 shows how the interval time of the 8-bit interval timer 0 is corrected.

Table 1.1 Peripheral Functions and Applications

Peripheral Function	Application
Timer array unit	Measuring the oscillation period of the low-speed on-chip oscillator clock
8-bit interval timer 0	Counting the interval time
A/D converter	Acquiring the thermistor voltage
Buzzer output	Outputting to the piezoelectric buzzer

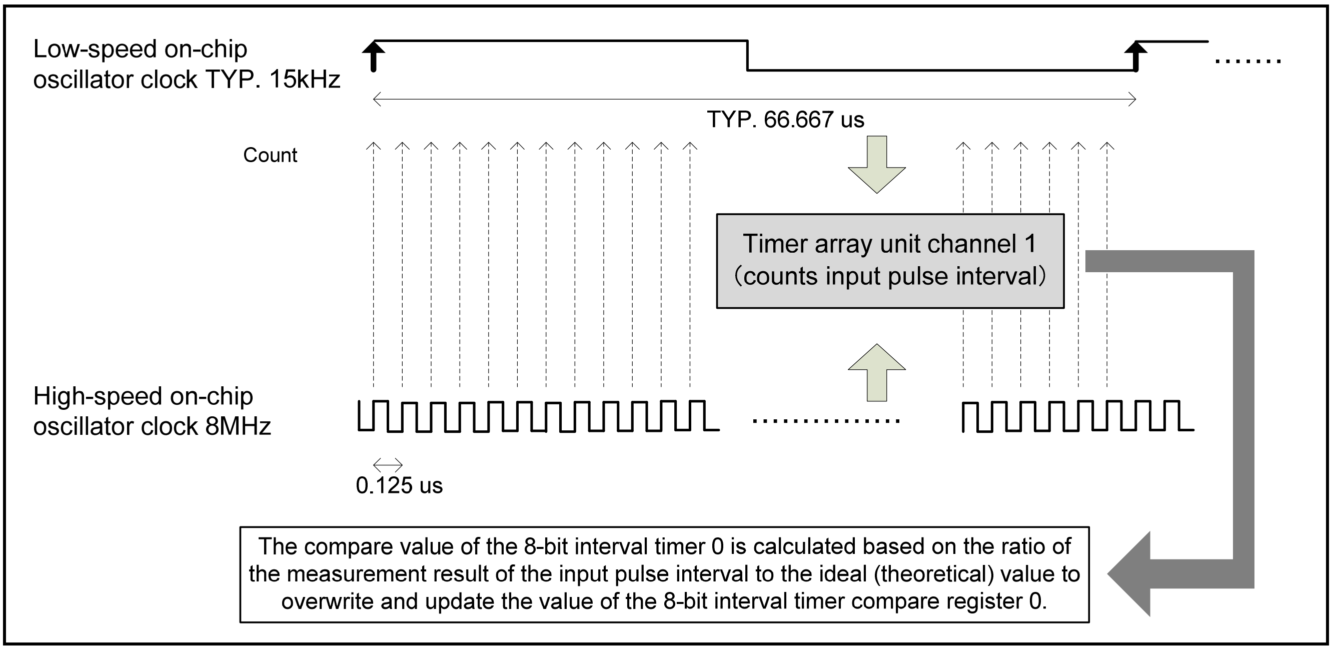

Figure 1.1 How The Interval Time of The 8-bit Interval Timer 0 is Corrected

2. Operation Confirmation Conditions

The sample code accompanying this application note has been run and confirmed under the conditions below.

Table 2.1 Operation Confirmation Conditions

Item	Contents
MCU used	RL78/I1D (R5F117GC)
Operating frequencies	● High-speed on-chip oscillator clock: 8MHz
Operating voltage	3.3V (operating range 2.0V to 3.6V) LVD operations (V_{LVD}): reset mode 1.88V (1.84V to 1.91V)
Integrated development environment (CS+)	CS+ for CC V6.00.00 from Renesas Electronics Corp.
C compiler (CS+)	CC-RL V1.05.00 from Renesas Electronics Corp.
Integrated development environment (e^2 studio)	e^2 studio V5.4.0.018 from Renesas Electronics Corp.
C compiler (e^2 studio)	CC-RL V1.05.00 from Renesas Electronics Corp.

3. Related Application Notes

Application notes related to this document are shown below. Please refer to these as needed.

RL78/I1D CPU Clock Changing and Standby Settings (C Language) CC-RL (R01AN3528E)
RL78/G13 Timer Array Unit (Pulse Interval Measurement) CC-RL (R01AN2702E)

4. Hardware Explanation

4.1 Hardware Configuration Example

Figure 4.1 shows an example of the hardware configuration used in this application note.

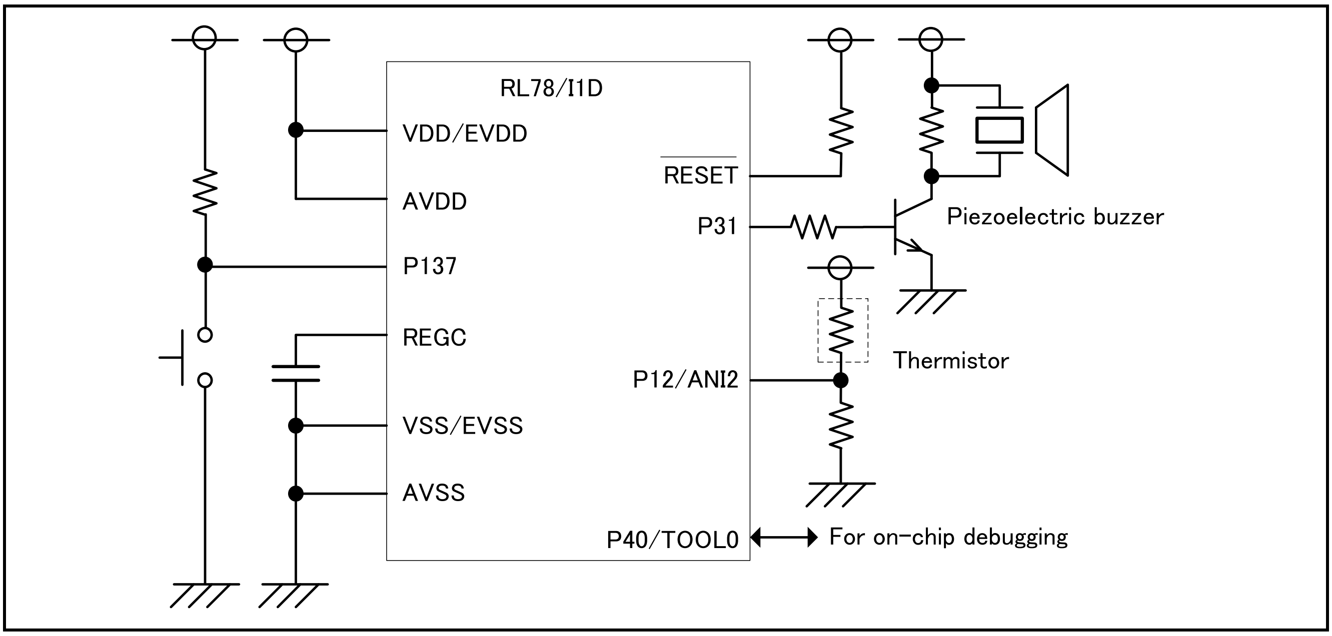

Figure 4.1 Hardware Configuration

Note: 1. This simplified circuit diagram was created to show an overview of connections only. When actually designing your circuit, make sure the design includes sufficient pin processing and meets electrical characteristic requirements. (Connect each input-only port to V_{DD} or V_{SS} through a resistor.)

2. Make V_{DD} higher than the RESET release voltage (V_{LVD}) set in LVD.

4.2 Used Pin List

Table 4.1 provides List of Pins and Functions.

Table 4.1 List of Pins and Functions

Pin Name	Input/Output	Function
P137/INTP0	Input	Switch input port
P12/ANI2	Input	Analog input port
P31/PCLBUZ0	output	Buzzer output

5. Software Explanation

5.1 Operation Outline

With this application, a compare match interrupt request of the 8-bit interval timer 0 is generated at approximately 20-second intervals to return from STOP mode, and A/D conversion is performed. The A/D conversion result is checked, and if it is equal to or greater than the reference value (0x0E00), buzzer output is provided. Buzzer output is stopped by pressing the switch. In addition, the interval time of the 8-bit interval timer 0 is corrected to maintain the accuracy of the interval time.

The specific operations are described in ① to ⑯ below.

① Perform the initial setting of the timer array unit.

<Channel 1 setting condition>

- Set the high-speed on-chip oscillator clock 8 MHz as the operating clock.
- Set the independent channel operation function.
- Set the low-speed on-chip oscillator clock as the start trigger and capture trigger.
- Set the falling edge of the start trigger and capture trigger as the valid edge.
- Set input pulse interval measurement mode.

② Perform the initial setting of the 8-bit interval timer 0.

<8-bit interval timer 0 setting condition>

- Set 16-bit count mode.
- Set $f_{IL}/8$ as the frequency division ratio.
- Set 0x927B as the compare value to generate a compare match interrupt request at approximately 20-second intervals.

③ Perform the initial setting of the clock output/buzzer output controller.

<Buzzer output setting condition>

- Set the low-speed on-chip oscillator clock as the PCLBUZ0 pin output clock.
- Set $f_{IL}/2^2$ as the frequency division ratio.

④ Perform the initial setting of the A/D converter.

<A/D converter setting condition>

- Set ANI2 as an analog input channel.
- Set select mode as the A/D conversion channel selection mode.
- Set one-shot conversion mode as the A/D conversion operation mode.
- Set a software trigger as the A/D conversion start condition.
- Set the A/D conversion time to 6.75 μs.

⑤ Perform the initial setting of the external interrupt.

<External interrupt setting condition>

- Use the P137/INTP0 pin.
- Set the falling edge of the INTP0 pin as the valid edge.

⑥ After completing the initial settings, wait for the oscillation accuracy of the low-speed on-chip oscillator clock to stabilize (210 us).

⑦ Start the count operation of the timer array unit channel 1 to measure the input pulse interval.

⑧ Acquire the measurement result of the input pulse interval upon generation of the second capture end interrupt request of the timer array unit.

⑨ Based on the measurement result of the input pulse interval, calculate the compare value of the 8-bit interval timer 0, and set it to the 8-bit interval timer compare register 0.

⑩ Stop the count operation of the timer array unit channel 1.

⑪ Start the count operation of the 8-bit interval timer 0.

⑫ Causes a transition to STOP mode.

⑬ Release the STOP mode by generation of the compare match interrupt request of the 8-bit interval timer 0.

⑭ Start A/D conversion and cause a transition to HALT mode to wait for the A/D conversion end interrupt to be requested.

⑮ When the A/D conversion result is smaller than the reference value (0x0E00), proceed to (16). When equal to or greater than the reference value (0x0E00), execute the following operations.

- Enable buzzer output.
- Stop the 8-bit interval timer 0.
- Enable external interrupt processing.

  When the external interrupt is generated (switch is pressed), buzzer output is stopped.

⑯ Execute the correction processing (⑦ to ⑩) of the interval time of the 8-bit interval timer 0, resume the count operation of the 8-bit interval timer 0, and return to ⑫.

5.2 Correction

The following gives a concrete explanation of the correction method in detail.

① Measuring the oscillation period of the low-speed on-chip oscillator clock

As shown in Figure 1.1, the oscillation period of the low-speed on-chip oscillator clock is measured by using the input pulse interval measurement function of the timer array unit channel 1, which operates on the high-speed on-chip oscillator clock.

After the timer array unit channel 1 starts count operation, when the first capture end interrupt request is generated, the measurement result of the input pulse interval is discarded since it is smaller than the count value corresponding to a single period. When the second capture end interrupt request is generated, the measurement result of the input pulse interval is acquired.

When the high-speed on-chip oscillator clock frequency is 8 MHz, the measurement result of the input pulse interval can be calculated by the following equation. Table 5.1 shows the range of the measurement result of the input pulse interval. Since the frequency accuracy of the high-speed on-chip oscillator clock is ±1%, there is a margin of measurement error as shown in Table 5.1.

Measurement result of input pulse interval = (input pulse interval/period of count clock)

= ((1 / 15kHz) / (1 / 8MHz))

Note Measurement result of the input pulse interval = TDR01 + 1

Table 5.1 Range of Measurement Result of Input Pulse Interval

		Low-speed on-chip oscillator clock (15kHz±15%)		
		min: 12.75 kHz	Typ: 15.00 kHz	Max: 17.25 kHz
High-speed on-chip oscillator clock (8MHz±1%)	Max: 8.08 MHz	634	538	468
	Typ: 8.00 MHz	627	533	463
	min: 7.92 MHz	621	528	459

② Correcting the compare value of the 8-bit interval timer 0

With this application, the 8-bit interval timer 0 is used to count 20 seconds. Therefore, the following equation is used to calculate the compare value to be set to the 8-bit interval timer compare register 0 (TRTCMP0) for counting 20 seconds.

*(TRTCMP0 + 1) = 20s / ((1 / 15kHz) * frequency division ratio)*

In addition, the above equation can be rearranged as shown below, if the frequency accuracy of the low-speed on-chip oscillator clock is considered.

In the equation, the ideal value is 533 according to table 5.1, and the frequency division is the division ratio of the count source of the 8-bit interval timer.

*(TRTCMP0 + 1) = 20s / ((1 / 15kHz) * (measurement result of the input pulse interval) / ideal value) * frequency division ratio))*

*= (20s * 15kHz * ideal value/frequency division ratio) / measurement result of the input pulse interval*

*= (20 * 15000 * 533 / 8) / measurement result of the input pulse interval*

= 19987500 / measurement result of the input pulse interval

Accordingly, the compare value to be set to the TRTCMP0 register is the result of the following equation.

TRTCMP0 = (19987500 / measurement result of the input pulse interval) – 1

Note Ideal value = (1/15 kHz) / (1/8 MHz); measurement result of the input pulse interval = TDR01 + 1

③ Correction timing

The frequency of the low-speed on-chip oscillator clock changes with the temperature. Therefore, it is necessary to correct the interval time of the 8-bit interval timer 0 in accordance with the temperature change.

For reference, the temperature characteristics of the low-speed on-chip oscillator clock measured by us are shown in Figure5.1 and Figure5.2.

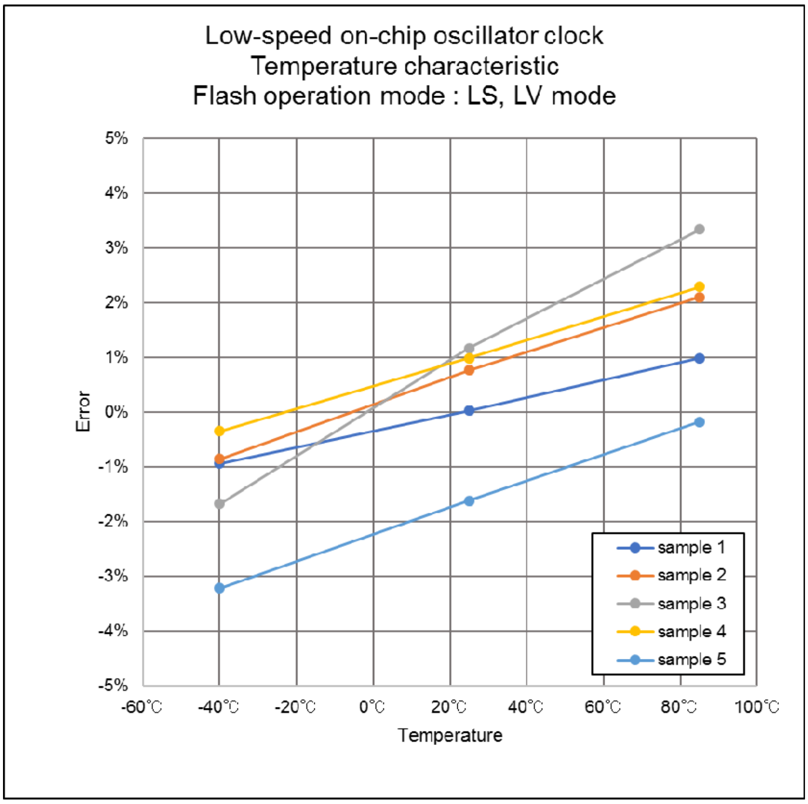

Figure5.1 Temperature characteristics of low-speed on-chip oscillator clock (LS,LV mode)

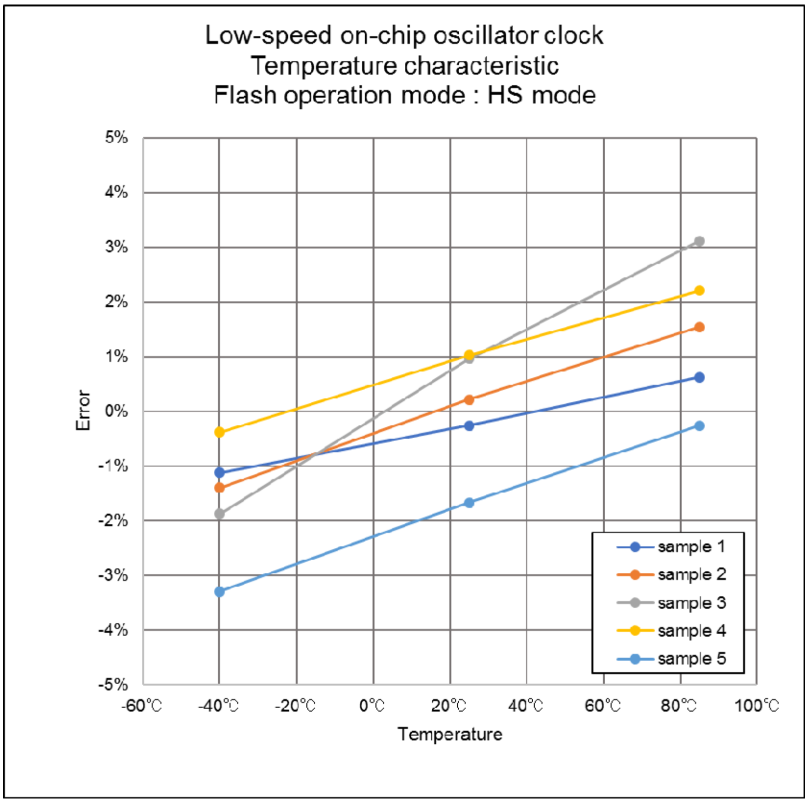

Figure5.2 Temperature characteristics of low-speed on-chip oscillator clock (HS mode)

5.3 Option Byte Settings

Table 5.2 lists the option byte settings.

Table 5.2 OptionByte Settings

Address	Setting Value	Contents
000C0H/010C0H	11101111B	Watchdog timer operation is stopped (count is stopped after reset)
000C1H/010C1H	01111111B	LVD operation (V_{LVD}): reset mode Detection voltage: 1.88V (1.84V to 1.91V)
000C2H/010C2H	10101010B	LS mode, High-speed on-chip oscillator clock: 8 MHz
000C3H/010C3H	10000100B	On-chip debugging enabled

5.4 Constants

Table 5.3 lists the constants that are used in this sample program.

Table 5.3 Constants for the Sample program

Constant Name	Setting Value	Contents
g_it8bit_data	19987500	Data for calculating the compare value of the 8-bit interval timer 0

5.5 Variables

Table 5.4 lists the variables.

Table 5.4 Variables

Type	Variable Name	Contents	Function Used
uint16_t	g_adcr_data	Stores the A/D conversion result	main, r_adc_interrupt
uint32_t	g_tau0_ch1_width	Stores the measurement result of the input pulse interval	r_main_timer_correction, r_tau0_channel1_interrupt

5.6 Functions

Table 5.5 lists the functions.

Table 5.5 Functions

Function Name	Outline
main	Main processing
R_MAIN_UserInit	Main initial setting
R_TAU0_Channel1_Start	Timer array unit channel1 operation start function
R_TAU0_Channel1_Stop	Timer array unit channel1 operation stop function
R_IT8Bit0_Channel0_Start	8-bit interval timer 0 count start function
R_IT8Bit0_Channel0_Stop	8-bit interval timer 0 count stop function
R_PCLBUZ0_Start	Buzzer output start function
R_PCLBUZ0_Stop	Buzzer output stop function
R_ADC_Start	A/D conversion operation start function
R_INTC0_Start	External interrupt enable function
r_main_timer_correction	Correction processing function
r_tau0_channel1_interrupt	Timer array unit channel1 capture complete interrupt function
r_adc_interrupt	End of A/D conversion interrupt processing
r_intc0_interrupt	External interrupt processing

5.7 Function Specifications

This part describes function specifications of the sample code.

[Function name]	main
Outline	Main processing
Header	r_cg_macrodriver.h, r_cg_cgc.h, r_cg_port.h, r_cg_tau.h, r_cg_it8bit.h, r_cg_pclbuz.h, r_cg_adc.h, r_cg_intp.h, r_cg_userdefine.h
Declaration	—
Description	Causes a transition to STOP mode after executing the main user initialization function. After approximately 20 seconds, a compare match interrupt request of the 8-bit interval timer 0 occurs to return from STOP mode. After returning to STOP mode, performs A/D conversion. Checks the A/D conversion result and if it is equal to or greater than the reference value (0x0E00), provides buzzer output. If it is smaller than the reference value (0x0E00), executes the correction processing function and causes a transition to STOP mode again.
Arguments	None
Remarks	None

[Function name]	R_MAIN_UserInit
Outline	Main initial setting
Header	r_cg_macrodriver.h, r_cg_cgc.h, r_cg_port.h, r_cg_tau.h, r_cg_it8bit.h, r_cg_pclbuz.h, r_cg_adc.h, r_cg_intp.h, r_cg_userdefine.h
Declaration	static void R_MAIN_UserInit(void);
Description	Waits for the oscillation of the low-speed on-chip oscillator clock to stabilize (210 us). After that, enables interrupts by using the EI instruction. Executes the correction processing function and starts the count operation of the 8-bit interval timer 0.
Arguments	None
Remarks	None

[Function name]	R_TAU0_Channel1_Start
Outline	Timer array unit channel1 operation start function
Header	r_cg_macrodriver.h, r_cg_tau.h
Declaration	void R_TAU0_Channel1_Start(void);
Description	This function is used for setting to enable count operation of channel 1 of the timer array unit.
Arguments	None
Remarks	None

[Function name]	R_TAU0_Channel1_Stop
Outline	Timer array unit channel1 operation stop function
Header	r_cg_macrodriver.h, r_cg_tau.h
Declaration	void R_TAU0_Channel1_Stop(void);
Description	This function is used for setting to disable count operation of channel 1 of the timer array unit.
Arguments	None
Remarks	None

[Function name]	R_IT8Bit0_Channel0_Start
Outline	8-bit interval timer 0 count start function
Header	r_cg_macrodriver.h, r_cg_it8bit.h
Declaration	void R_IT8Bit0_Channel0_Start(void);
Description	This function is used for setting to enable activation of the 8-bit interval timer 00.
Arguments	None
Remarks	None

[Function name]	R_IT8Bit0_Channel0_Stop
Outline	8-bit interval timer 0 count stop function
Header	r_cg_macrodriver.h, r_cg_it8bit.h
Declaration	void R_IT8Bit0_Channel0_Stop(void);
Description	This function is used for setting to disable activation of the 8-bit interval timer 00.
Arguments	None
Remarks	None

[Function name]	R_PCLBUZ0_Start
Outline	Buzzer output start function
Header	r_cg_macrodriver.h, r_cg_pclbuz.h
Declaration	void R_PCLBUZ0_Start(void);
Description	This function is used for setting to start of the buzzer output.
Arguments	None
Remarks	None

[Function name]	R_PCLBUZ0_Stop
Outline	Buzzer output stop function
Header	r_cg_macrodriver.h, r_cg_pclbuz.h
Declaration	void R_PCLBUZ0_Stop(void);
Description	This function is used for setting to stop of the buzzer output.
Arguments	None
Remarks	None

[Function name]	R_ADC_Start
Outline	A/D conversion operation start function
Header	r_cg_macrodriver.h, r_cg_adc.h
Declaration	void R_ADC_Start(void);
Description	This function is used for setting to start the A/D conversion operation of the A/D converter.
Arguments	None
Remarks	None

[Function name]	R_INTC0_Start
Outline	External interrupt enable function
Header	r_cg_macrodriver.h, r_cg_intp.h
Declaration	void R_INTC0_Start(void);
Description	This function is used for setting to enable operation of the external interrupt.
Arguments	None
Remarks	None

[Function name]	r_main_timer_correction
Outline	Correction processing function
Header	r_cg_macrodriver.h, r_cg_tau.h, r_cg_it8bit.h
Declaration	void r_main_timer_correction(void);
Description	Starts the count operation of the timer array unit channel 1, and acquires the measurement result of the input pulse interval upon generation of the second capture end interrupt request. Based on the acquired measurement result of the input pulse interval, calculates the compare value of the 8-bit interval timer 0, and sets it to the 8-bit interval timer compare register 0. Then stops the count operation of the timer array unit channel 1.
Arguments	None
Remarks	None

[Function name]	r_tau0_channel1_interrupt
Outline	Timer array unit channel1 capture complete interrupt function
Header	r_cg_macrodriver.h, r_cg_tau.h
Declaration	#pragma interrupt r_tau0_channel1_interrupt(vect=INTTM01)
Description	Stores the measurement result of the input pulse interval by the timer array unit channel 1 in the global variable.
Arguments	None
Remarks	None

[Function name]	r_adc_interrupt
Outline	End of A/D conversion inrterrupt processing
Header	r_cg_macrodriver.h, r_cg_adc.h
Declaration	static void __near r_adc_interrupt(void)
Description	Stores the A/D conversion result in the global variable.
Arguments	None
Remarks	None

[Function name]	r_intc0_interrupt
Outline	External interrupt processing
Header	r_cg_macrodriver.h, r_cg_intp.h
Declaration	#pragma interrupt r_intc0_interrupt(vect=INTP0)
Description	Performs chattering prevention processing. Then disables the external interrupts and buzzer output, and starts the count operation of the 8-bit interval timer 0.
Arguments	None
Remarks	None

5.8 Flowcharts

Figure 5.3 shows an overall flow of the sample code.

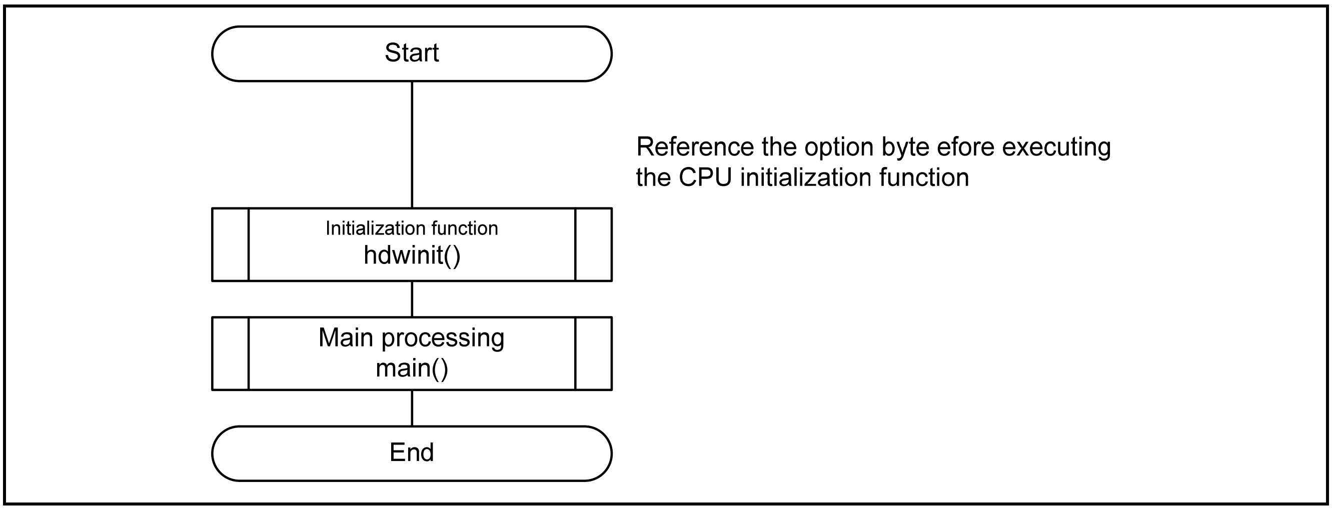

Figure 5.3 Overall Flow

5.8.1 Initialization Function

Figure 5.4 shows the flowchart for the initialization function.

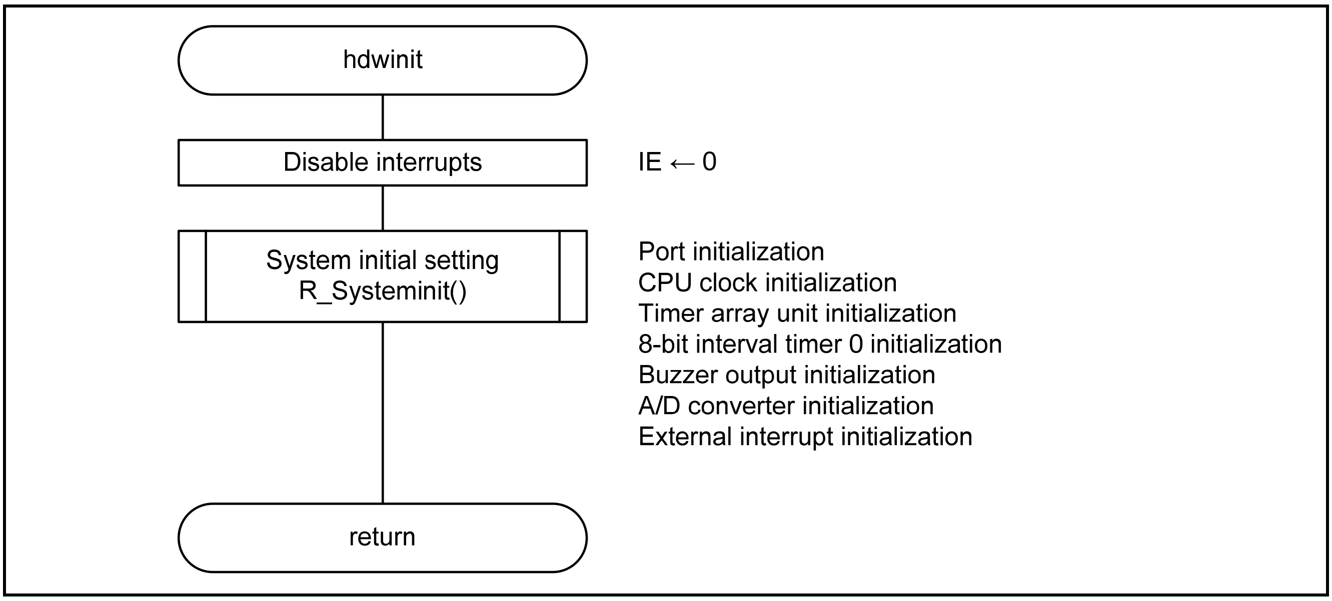

Figure 5.4 Initialization Function

5.8.2 System Initial Setting

Figure 5.5 shows the flowchart for the system initial setting.

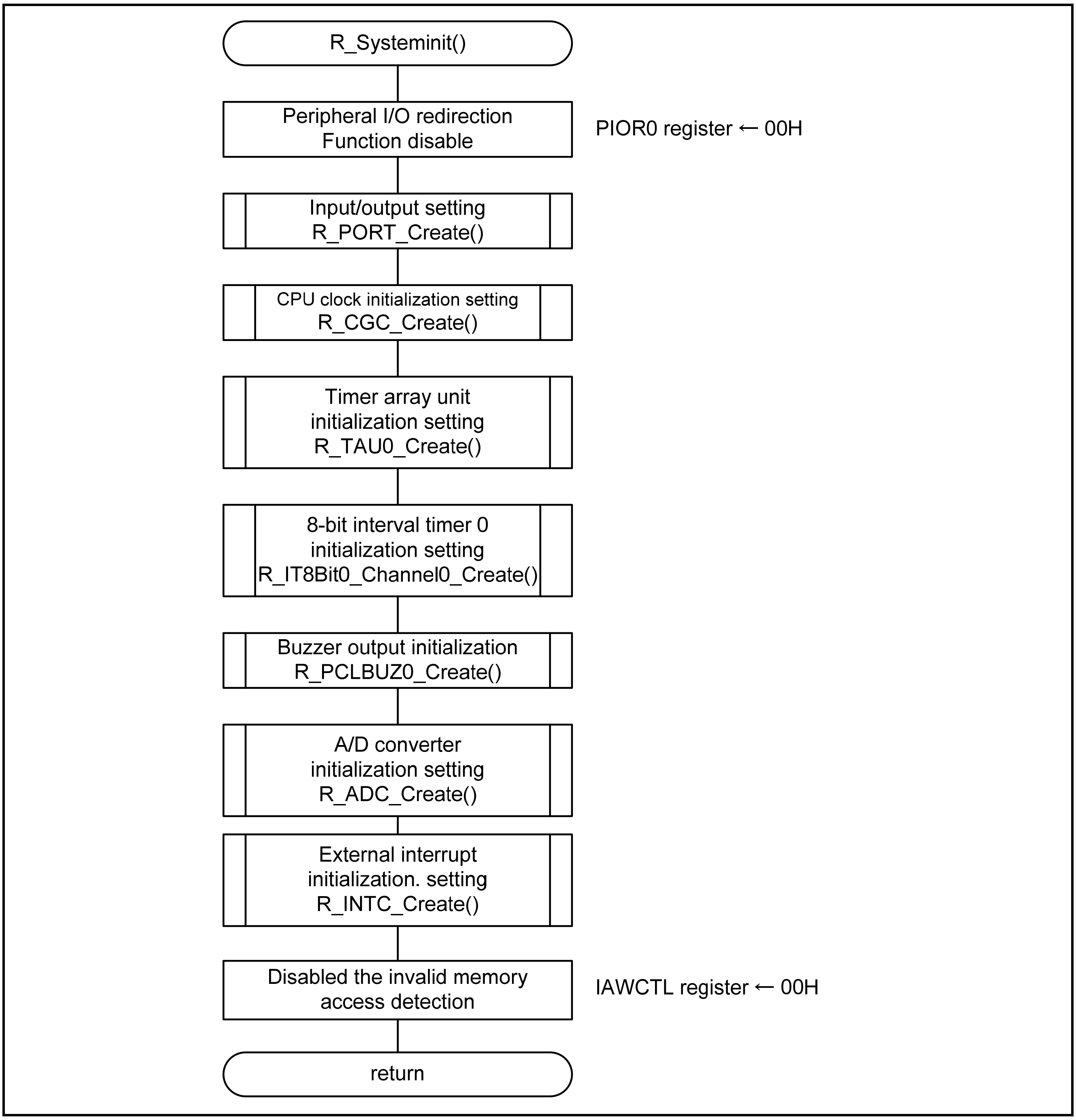

Figure 5.5 System Initial Setting

5.8.3 Ports Initial Setting

Figure 5.6 shows the flowchart for the ports initial setting.

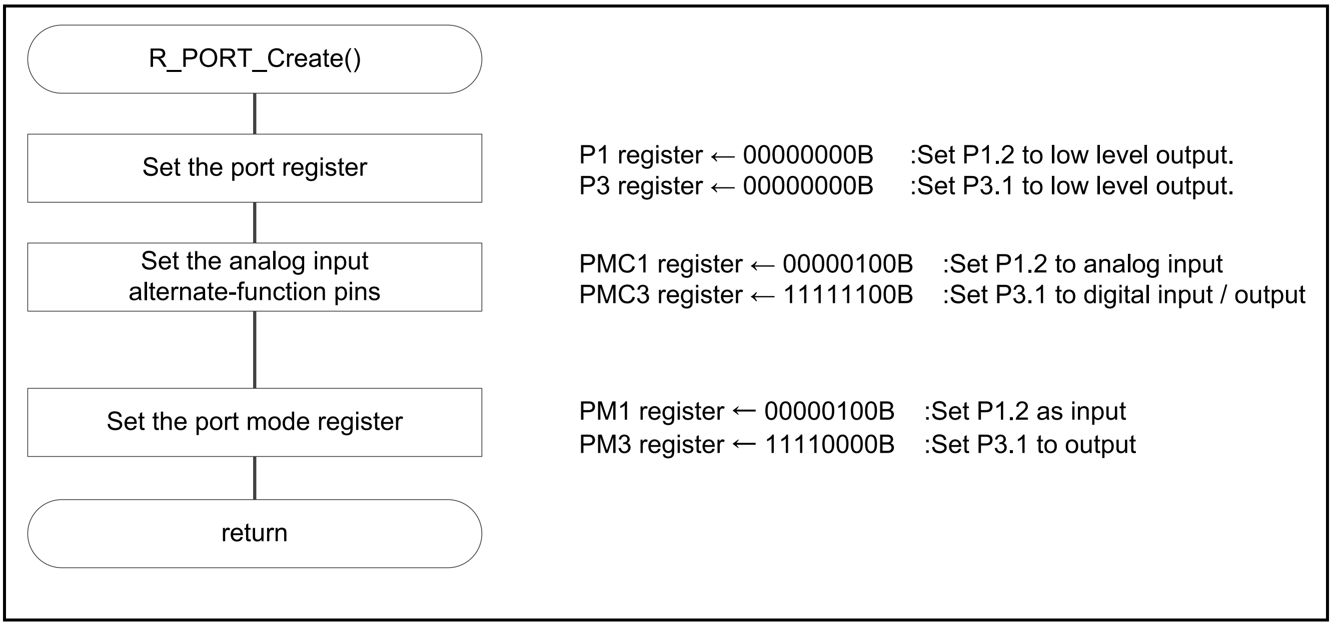

Figure 5.6 Port Initial Setting

Note: Refer to the initialization flowchart in the RL78/G13 Initialization CC-RL (R01AN2575E) Application Note for details on how to set unused ports.

Caution: When designing circuits, always make sure unused ports are properly processed and all electrical characteristics are met. Also make sure each unused input-only port is connected to VDD or VSS through a resister.

5.8.4 CPU Initial Setting

Figure 5.7 shows the flowchart for the CPU initial setting.

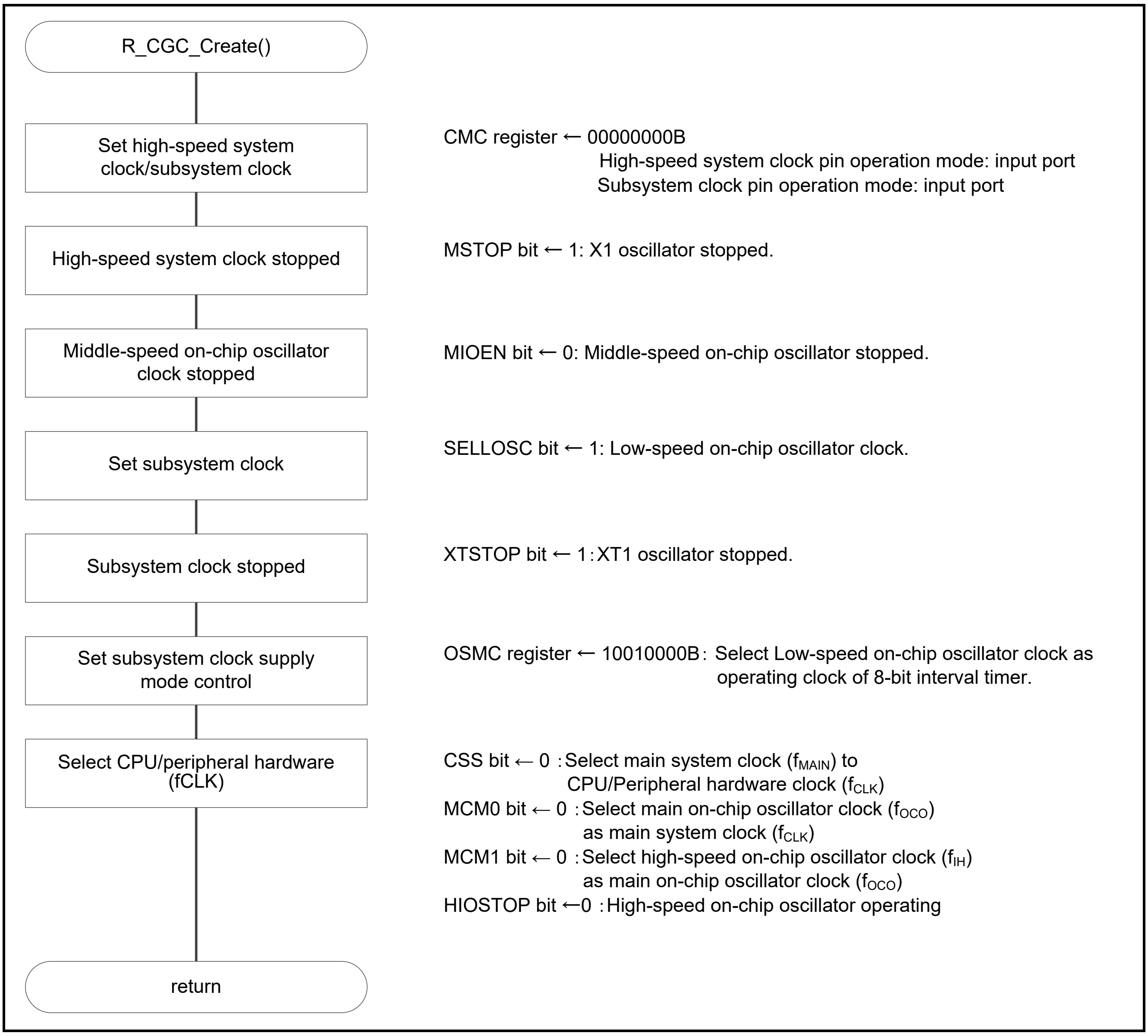

Figure 5.7 CPU Initial Setting

Clock operation mode setting

- Clock operation mode control register (CMC)
  Set the high-speed system clock pin operation mode to input mode.
  Set the subsystem clock pin operation mode to input mode.

Symbol: CMC

7	6	5	4	3	2	1	0
EXCLK	OSCSEL	EXCLKS	OSCSELS	0	AMPHS1	AMPHS0	AMPH
0	**0**	**0**	**0**	0	**0**	**0**	**0**

Bits 7-6

EXCLK	OSCSEL	High-seed oscillation clock pin operation mode	X1/P121 Port	X2/EXCLK/P122 Port
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

Bits 5-4

EXCLKS	OSCSELS	Subsystem clock pin operation mode	XT1/P123 Pin	XT2/EXCLKS/P124 Pin
0	**0**	**Input port mode**	**Input port**	
0	1	XT1 oscillation mode	Crystal resonator connection	
1	0	Input port mode	Input port	
1	1	External clock input mode	Input port	External clock input

Bits 2-1

AMPHS1	AMPHS0	XT1 oscillator oscillation mode selection
0	**0**	**Low-power consumption oscillation (default)**
0	1	Normal oscillation
1	0	Ultra-low power consumption oscillation
1	1	Setting prohibited

Bit 0

AMPH	Control of X1 clock oscillation frequency
0	**1MHz ≤ f_X ≤ 10MHz**
1	10MHz < f_X ≤ 20MHz

Note: Refer to the RL78/I1D User's Manual (Hardware version) for details on how to set registers.

Operation control of clocks

- Clock operation status control register (CSC)

High-speed system clock operation control: X1 oscillator stopped
Subsystem clock operation control: XT1 oscillator stopped
Middle-speed on-chip oscillator clock operation control: Middle-speed on-chip oscillator stopped
High-speed on-chip oscillator clock operation control: High-speed on-chip oscillator operating

Symbol: CSC

7	6	5	4	3	2	1	0
MSTOP	XTSTOP	0	0	0	0	MIOEN	HIOSTOP
1	**1**	x	x	x	x	**0**	**0**

Bit 7

MSTOP	High-speed system clock operation control		
	X1 oscillation mode	External clock input mode	Input port mode
0	X1 oscillator operating	External clock from EXCLK pin is valid	**Input port**
1	**X1 oscillator stopped**	External clock from EXCLK pin is invalid	

Bit 6

XTSTOP	Subsystem clock operation control		
	XT1 oscillation mode	External clock input mode	Input port mode
0	XT1 oscillator operating	External clock from EXCLKS pin is valid	**Input port**
1	**XT1 oscillator stopped**	External clock from EXCLKS pin is invalid	

Bit 1

MIOEN	Middle-speed on-chip oscillator clock operation control
0	**Middle-speed on-chip oscillator stopped**
1	Middle-speed on-chip oscillator operating

Bit 0

HIOSTOP	High-speed on-chip oscillator clock operation control
0	**High-speed on-chip oscillator operating**
1	High-speed on-chip oscillator stopped

Note: Refer to the RL78/I1D User’s Manual (Hardware version) for details on how to set registers.

Subsystem clock setting

- Subsystem clock select register (CKSEL)

Select low-speed on-chip oscillator clock as subsystem clock.

Symbol: CKSEL

7	6	5	4	3	2	1	0
0	0	0	0	0	0	0	SELLOSC
0	0	0	0	0	0	0	**1**

Bit 0

SELLOSC	Selection of sub clock/low-speed on-chip oscillator clock
0	Sub clock
1	**Low-speed on-chip oscillator clock**

Subsystem clock supply mode control

- Subsystem clock supply mode control register (OSMC)

Setting in STOP mode or in HALT mode while subsystem clock is selected as CPU clock

: Stops supply of subsystem clock to peripheral functions other than the real-time clock 2, 12-bit interval timer, 8-bit interval timer, and clock output/buzzer output controller.

Selection of count clock for real-time clock and 12-bit interval timer: low-speed on-chip oscillator clock

Symbol: OSMC

7	6	5	4	3	2	1	0
RTCLPC	0	0	WUTMM CK0	0	0	0	0
1	0	0	**1**	0	0	0	0

Bit 7

RTCLPC	Setting in STOP mode or in HALT mode while subsystem clock is selected as CPU clock
0	Enables supply of subsystem clock to peripheral functions
1	**Stops supply of subsystem clock to peripheral functions other than the real-time clock 2, 12-bit interval timer, 8-bit interval timer, and clock output/buzzer output controller.**

Bit 4

WUTMMCK0	Selection of operation clock for real-time clock 2, frequency measurement circuit, 12-bit interval timer, 8-bit interval timer, and clock output/buzzer output controller
0	Subsystem clock (f_{SUB})
1	**Low-speed internal oscillator clock**

Note: Refer to the RL78/I1D User's Manual (Hardware version) for details on how to set registers.

System clock control setting

- System clock control register (CKC)

Select the high-speed on-chip oscillator clock as a CPU/peripheral hardware clock.

Symbol: CKC

7	6	5	4	3	2	1	0
CLS	CSS	MCS	MCM0	0	0	MCS1	MCM1
0	**0**	**0**	**0**	0	0	**0**	**0**

Bit 6

CSS	Selection of CPU/peripheral hardware clock (f_{CLK})
0	**Main system clock (f_{MAIN})**
1	Subsystem clock (f_{SUB})

Bit 4

MCM0	Main system clock (fMAIN) operation control
0	**Selects the main on-chip oscillator clock (f_{OCO}) as the main system clock (f_{MAIN})**
1	Selects the high-speed system clock (f_{MX}) as the main system clock (f_{MAIN})

Bit 0

MCM1	Main on-chip oscillator clock (f_{oco}) operation control
0	**High-speed on-chip oscillator clock (f_{IH})**
1	Middle-speed on-chip oscillator clock (f_{IM})

Note: Refer to the RL78/I1D User’s Manual (Hardware version) for details on how to set registers.

5.8.5 Timer Array Unit Initial Setting

Figure 5.8 and Figure 5.9 shows the flowchart for the timer array unit initial setting.

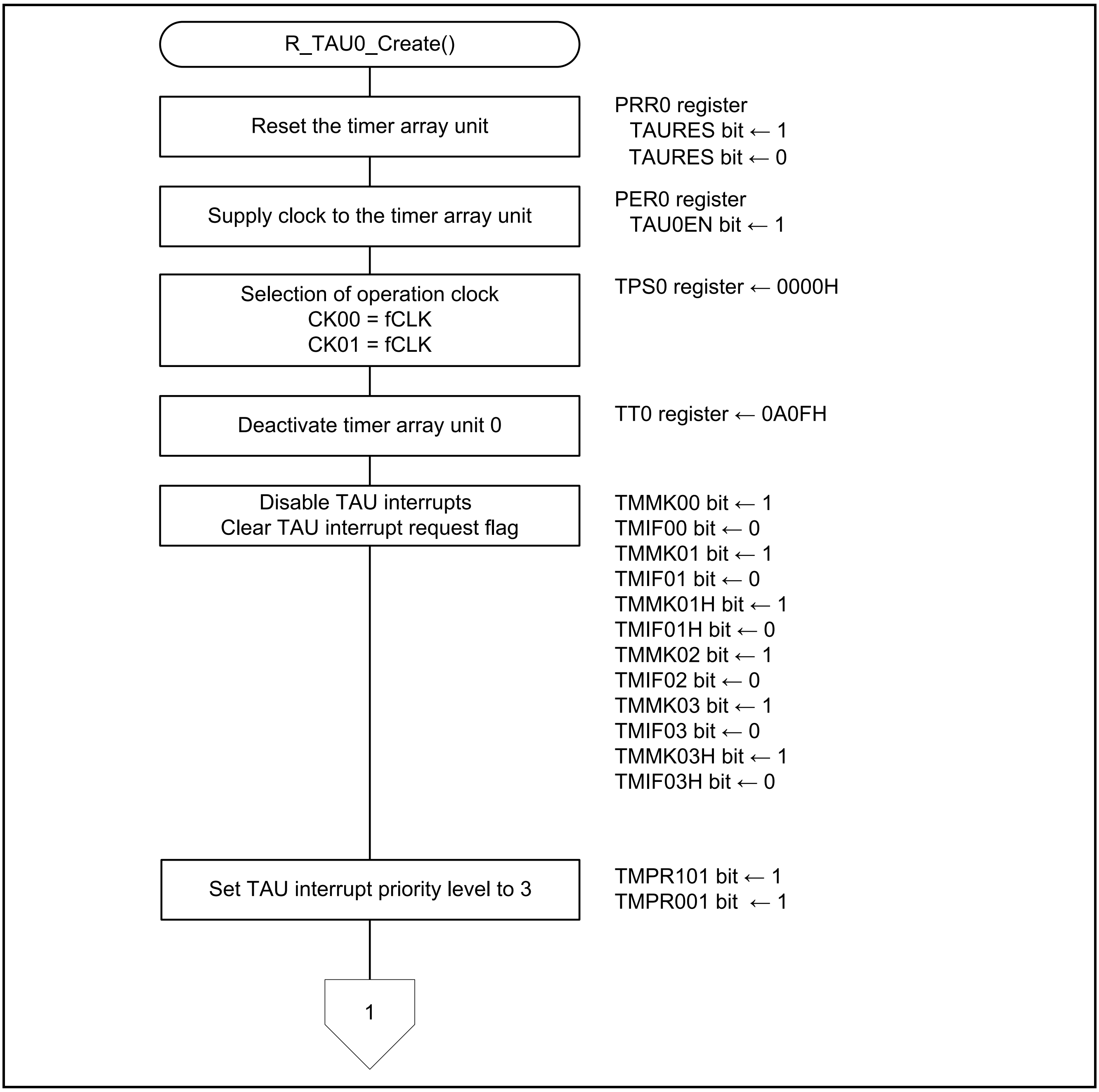

Figure 5.8 Timer Array Unit Initial Setting (1/2)

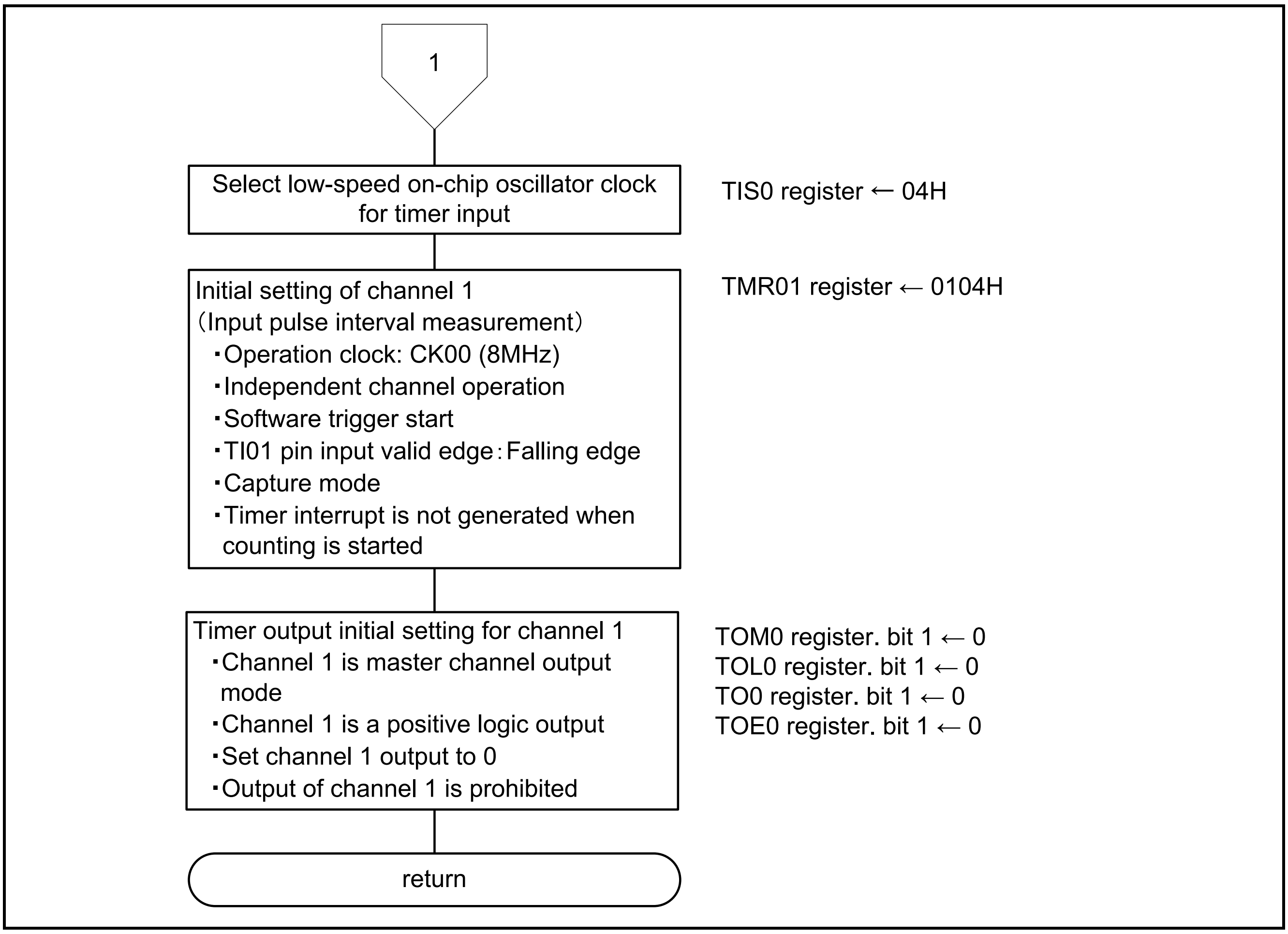

Figure 5.9 Timer Array Unit Initial Setting (2/2)

Reset control of timer array unit

- Peripheral reset control register 0（PRR0）

Reset timer array unit

Symbol: PRR0

7	6	5	4	3	2	1	0
0	0	ADCRES	0	0	SAU0RES	0	TAU0RES
0	0	x	0	0	x	0	**0**

Bit 0

TAU0RES	Reset control of timer array unit 0
0	Reset control of timer array unit 0
1	Reset state of timer array unit 0.

Clock supply to timer array unit started

- Peripheral enable register 0 (PER0)

Clock supply to timer array unit

Symbol: PER0

7	6	5	4	3	2	1	0
RTCWEN	0	ADCEN	0	0	SAU0EN	0	TAU0EN
x	0	x	0	0	x	0	**1**

Bit 0

TAU0EN	Control of timer array 0 unit input clock
0	Stops supply of input clock.
1	**Supplies input clock.**

Note: Refer to the RL78/I1D User’s Manual (Hardware version) for details on how to set registers.

Operation clock setting

- Timer clock select register 0 (TPS0)

  Selection of operation clock (CK00)

Symbol: TPS0

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	0	PRS 031	PRS 030	0	0	PRS 021	PRS 020	PRS 013	PRS 012	PRS 011	PRS 010	PRS 003	PRS 002	PRS 001	PRS 000
0	0	**0**	**0**	0	0	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**1**	**0**

PRS 003	PRS 002	PRS 001	PRS 000		Operation Clock (CK00) Selection				
					f_{CLK} = 2MHz	f_{CLK} = 5MHz	f_{CLK} = 10MHz	f_{CLK} = 20MHz	f_{CLK} = 24MHz
0	**0**	**0**	**0**	**f_{CLK}**	**2MHz**	**5MHz**	**10MHz**	**20MHz**	**24MHz**
0	0	0	1	$f_{CLK}/2$	1MHz	2.5MHz	5MHz	10MHz	12MHz
0	0	1	0	$f_{CLK}/2^2$	500kHz	1.25MHz	2.5MHz	5MHz	6MHz
0	0	1	1	$f_{CLK}/2^3$	250kHz	625kHz	1.25MHz	2.5MHz	3MHz
0	1	0	0	$f_{CLK}/2^4$	125kHz	312.5kHz	625kHz	1.25MHz	1.5MHz
0	1	0	1	$f_{CLK}/2^5$	62.5kHz	156.2kHz	312.5KHz	625KHz	750kHz
0	1	1	0	$f_{CLK}/2^6$	31.25kHz	78.1kHz	156.2kHz	312.5kHz	375kHz
0	1	1	1	$f_{CLK}/2^7$	15.62kHz	39.1kHz	78.1kHz	156.2kHz	187.5kHz
1	0	0	0	$f_{CLK}/2^8$	7.81kHz	19.5kHz	39.1kHz	78.1kHz	93.8kHz
1	0	0	1	$f_{CLK}/2^9$	3.91kHz	9.76kHz	19.5kHz	39.1kHz	46.9kHz
1	0	1	0	$f_{CLK}/2^{10}$	1.95kHz	4.88kHz	9.76kHz	19.5kHz	23.4kHz
1	0	1	1	$f_{CLK}/2^{11}$	976Hz	2.44kHz	4.88kHz	9.76kHz	11.7kHz
1	1	0	0	$f_{CLK}/2^{12}$	488Hz	1.22kHz	2.44kHz	4.88kHz	5.86kHz
1	1	0	1	$f_{CLK}/2^{13}$	244Hz	610Hz	1.22kHz	2.44kHz	2.93kHz
1	1	1	0	$f_{CLK}/2^{14}$	122Hz	305Hz	610Hz	1.22kHz	1.46kHz
1	1	1	1	$f_{CLK}/2^{15}$	61Hz	153Hz	305Hz	610Hz	732Hz

Channel stop control

- Timer channel stop register 0 (TT0)

  Stop the counting operation of each channel

Symbol: TT0

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	0	0	0	TT H03	0	TT H01	0	0	0	0	0	TT 03	TT 02	TT 01	TT 00
0	0	0	0	1	0	1	0	0	0	0	0	1	1	**1**	1

Bit 1

TT00	Operation stop trigger of channel 0
0	TE01 bit is cleared to 0 and the count operation is stopped.
1	**Operation is stopped (stop trigger is generated).**

Note: Refer to the RL78/I1D User’s Manual (Hardware version) for details on how to set registers.

Selection of timer input used with channel 1 of timer array unit

- Timer input select register 0 (TIS0)

Select low-speed on-chip oscillator clock (f_{IL}) for channel 1 timer input

Symbol: TIS0

7	6	5	4	3	2	1	0
0	0	0	TIS04	0	TIS02	TIS01	TIS00
0	0	0	x	0	**1**	**0**	**0**

Bit 2-0

TIS02	TIS01	TIS00	Selection of timer input used with channel 1
0	0	0	Input signal of timer input pin (TI01)
0	0	1	Event input signal from ELC
0	1	0	Input signal of timer input pin (TI01)
0	1	1	Middle-speed on-chip oscillator clock (f_{IM})
1	**0**	**0**	**Low-speed on-chip oscillator clock (f_{IL})**
1	0	1	Subsystem clock (f_{SUB})
Other than above			Setting prohibited

Note: Refer to the RL78/I1D User’s Manual (Hardware version) for details on how to set registers.

Timer array unit 0 channel 1 initialization

- Timer mode register 01 (TMR01)

Selection of operation mode、Software trigger start,
Selection of operation clock

Symbol: TMR01

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CKS 011	CKS 010	0	CCS 01	SPLIT 01	STS 012	STS 011	STS 010	CIS 011	CIS 010	0	0	MD 013	MD 012	MD 011	MD 010
0	**0**	0	**0**	**0**	**0**	**0**	**1**	**0**	**0**	0	0	**0**	**1**	**0**	**0**

Bits 15-14

CKS 011	CKS 010	Selection of operation clock (f_{MCK}) for channel n
0	**0**	**Operation clock CKm0 set in timer clock selection register m (TPSm)**
0	1	Operation clock CKm2 set in timer clock selection register m (TPSm)
1	0	Operation clock CKm1 set in timer clock selection register m (TPSm)
1	1	Operation clock CKm3 set in timer clock selection register m (TPSm)

Bit 12

CCS01	Selection of channel n operation clock (f_{TCLK})
0	**Operation clock (f_{MCK}) set in bits CKSmn0 and CKSmn1**
1	Valid edge of input signal from TImn pin

Bit 11

SPLIT01	Selection of 8-bit/16-bit timer operation for channels 1 and 3
0	**Operates as 16-bit timer**
1	Operates as 8-bit timer

Bits 10-8

STS 012	STS 011	STS 010	Setting start or capture trigger of channel 1
0	0	0	Only software trigger start is valid (other trigger sources are unselected)
0	**0**	**1**	**Valid edge of TImn pin input is used as both the start trigger and capture trigger**
0	1	0	Both edges of TImn pin input are used as the start trigger and capture trigger
1	0	0	Interrupt signal of master channel is used (when using slave channel with simultaneous channel operation function)
Other than above			Setting prohibited

Note: Refer to the RL78/I1D User's Manual (Hardware version) for details on how to set registers.

Symbol: TMR01

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CKS 011	CKS 010	0	CCS 01	SPLIT 01	STS 012	STS 011	STS 010	CIS 011	CIS 010	0	0	MD 013	MD 012	MD 011	MD 010
0	**0**	0	**0**	**0**	**0**	**0**	**1**	**0**	**0**	0	0	**0**	**1**	**0**	**0**

Bits 7-6

CIS 011	CIS 010	Selection of TI01 pin input valid edge
0	**0**	**Falling edge**
0	1	Rising edge
1	0	Both edges (when low-level width is measured)
1	1	Both edges (when high-level width is measured)

Bits 3-0

MD 013	MD 012	MD 011	MD 010	Operation mode of channel 1
0	0	0	0	Interval timer mode (Timer interrupt is not generated when counting is started).
			1	Interval timer mode (Timer interrupt is generated when counting is started）.
0	**1**	**0**	**0**	**Capture mode** **(Timer interrupt is not generated when counting is started).**
			1	Capture mode (Timer interrupt is generated when counting is started).
0	1	1	0	Event counter mode (Timer interrupt is not generated when counting is started).
1	0	0	0	One-count mode (Start trigger is invalid during counting operation).
			1	One-count mode (Start trigger is valid during counting operation).
1	1	0	0	Capture & one-count mode (Timer interrupt is not generated when counting is started Start trigger is invalid during counting operation).
Other than above				Setting prohibited

Note: Refer to the RL78/I1D User's Manual (Hardware version) for details on how to set registers.

Interrupt request flag setting

- Interrupt request flag register (IF1L)

Clear timer interrupt request flag

Symbol: IF1L

7	6	5	4	3	2	1	0
0	0	TMIF03	TMIF02	TMIF01	TMIF03H	TMIF01H	FMIF
0	0	x	x	**0**	x	x	x

Bit 3

TMIF01	Interrupt request flag
0	**No interrupt request signal is generated**
1	Interrupt request is generated, interrupt request status

Setting interrupt mask flag

- Interrupt mask flag registers (MK1L)

Interrupt servicing enabled

Symbol: MK1L

7	6	5	4	3	2	1	0
0	0	TMMK03	TMMK02	TMMK01	TMMK03H	TMMK01H	FMMK
0	0	x	x	**0**	x	x	x

Bit 3

TMMF01	Interrupt servicing control
0	**Interrupt servicing enabled**
1	Interrupt servicing disabled

Note: Refer to the RL78/I1D User’s Manual (Hardware version) for details on how to set registers.

5.8.6 8-bit Interval Timer 0 Initial Setting

Figure 5.10 shows the flowchart for the 8-bit interval timer 0 initial setting.

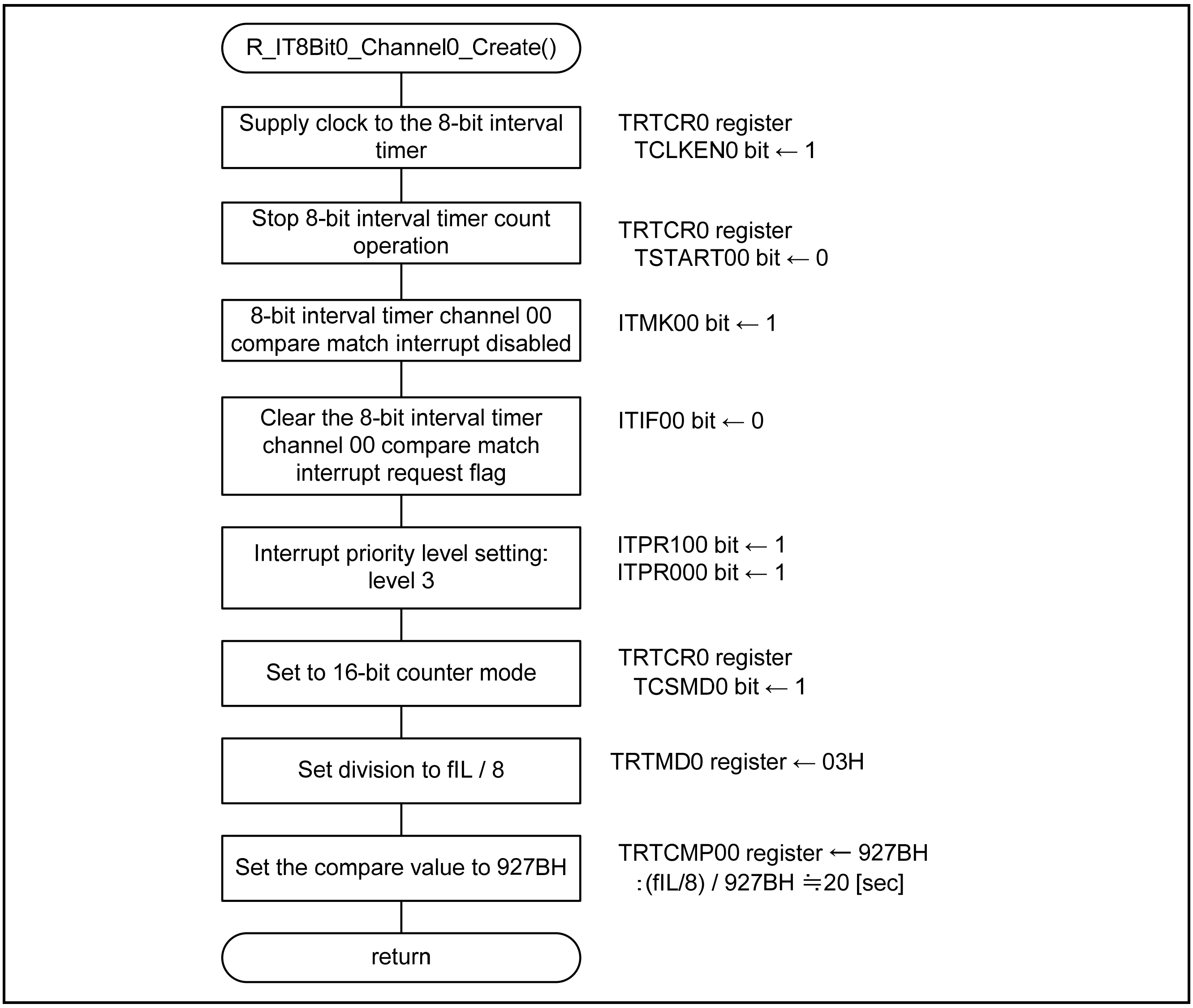

Figure 5.10 The 8-bit Interval Timer 0 Initial Setting

8-bit interval timer 0 control setting

- 8-bit interval timer control register 0 (TRTCR0)
  Start clock supply as the 16-bit counter.

Symbol: TRTCR0

7	6	5	4	3	2	1	0
TCSMD0	0	0	TCLKEN0	0	TSTART01	0	TSTART00
1	0	0	**1**	0	x	0	**0**

Bit 7

TCSMD0	Mode selection
0	Operates as 8-bit counter
1	**Operates as 16-bit counter (channel 0 and channel 1 are connected)**

Bit 4

TCLKEN0	8-bit interval timer clock enable
0	Clock is stopped
1	**Clock is supplied**

Bit 0

TSTART00	8-bit interval timer 0 count start
0	**Count stops**
1	Count starts

Note: Refer to the RL78/I1D User's Manual (Hardware version) for details on how to set registers.

8-bit interval timer 0 interrupt setting

- Interrupt mask flag register (MK2L)
  Disable interrupt servicing.
- Interrupt request flag register (IF2L)
  Clear the interrupt request flag.

Symbol: MK2L

7	6	5	4	3	2	1	0
FLMK	0	0	0	ITMK11	ITMK10	ITMK01	ITMK00
x	0	0	0	x	x	x	**1**

Bit 0

ITMK00	Interrupt servicing control
0	Interrupt servicing enabled
1	**Interrupt servicing disabled**

Symbol: IF2L

7	6	5	4	3	2	1	0
FLIF	0	0	0	ITIF11	ITIF10	ITIF01	ITIF00
x	0	0	0	x	x	x	**0**

Bit 0

ITIF00	Interrupt request flag
0	**No interrupt request signal is generated**
1	Interrupt request is generated, interrupt request status

Note: Refer to the RL78/I1D User’s Manual (Hardware version) for details on how to set registers.

8-bit interval timer 0 count source setting

- 8-bit interval timer division register 0 (TRTMD0)
  Set the division of 8-bit interval timer 0.

Symbol: TRTMD0

7	6	5	4	3	2	1	0
0	TCK01			0	TCK00		
0	x	x	x	0	**101**		

Bits 2-0

TCK00			8-bit interval timer 0 division selection
Bit 2	Bit 1	Bit 0	
0	0	0	fSXR or fIL
0	0	1	fSXR/2 or fIL/2
0	1	0	fSXR/4 or fIL/4
0	**1**	**1**	**fSXR/8 or fIL/8**
1	0	0	fSXR/16 or fIL/16
1	0	1	fSXR/32 or fIL/32
1	1	0	fSXR/64 or fIL/64
1	1	1	fSXR/128 or fIL/128

8-bit interval timer 0 count value setting

- 8-bit interval timer compare register 00 (TRTCMP00)
  Set a count value.

Symbol：TRTCMP0

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
1	**0**	**0**	**1**	**0**	**0**	**1**	**0**	**0**	**1**	**1**	**1**	**1**	**0**	**1**	**1**

Bits 15 - 0

Function
16-bit counter

Note: Refer to the RL78/I1D User's Manual (Hardware version) for details on how to set registers.

5.8.7 Buzzer Output Initial Setting

Figure 5.11 shows the flowchart for the buzzer output initial setting.

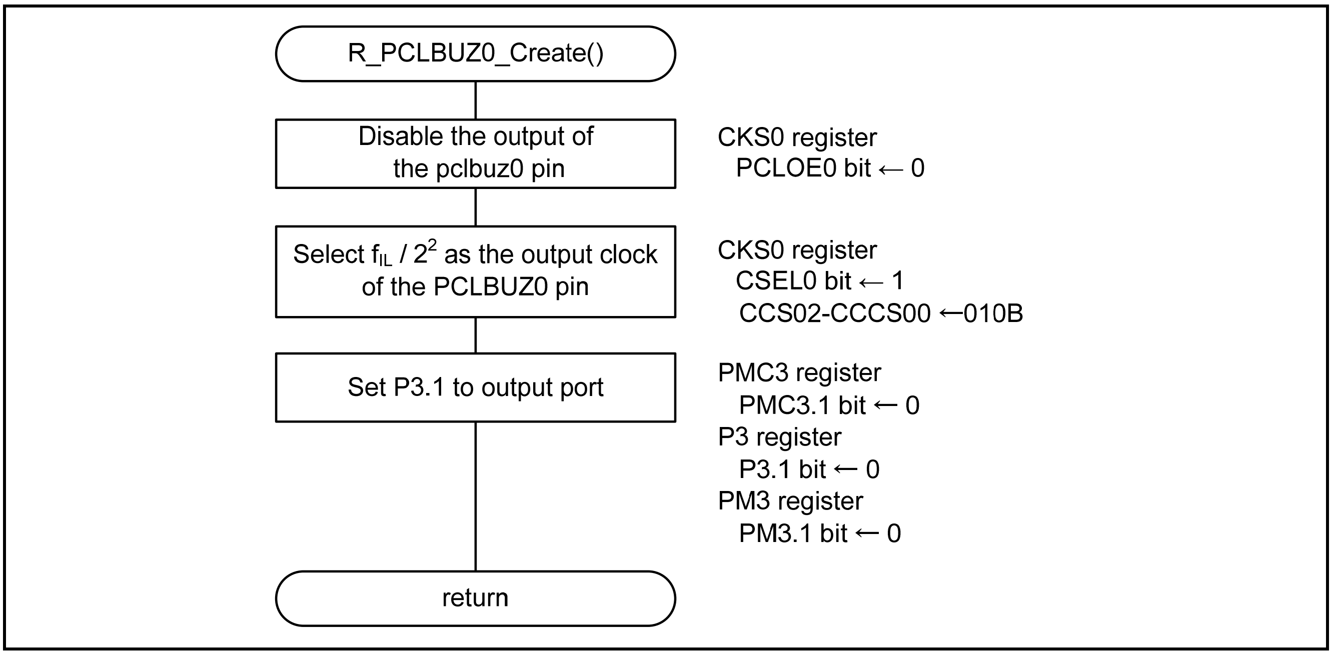

Figure 5.11 Buzzer Output Initial Setting

5.8.8 A/D Converter Initial Setting

Figure 5.12 shows the flowchart for the A/D converter initial setting.

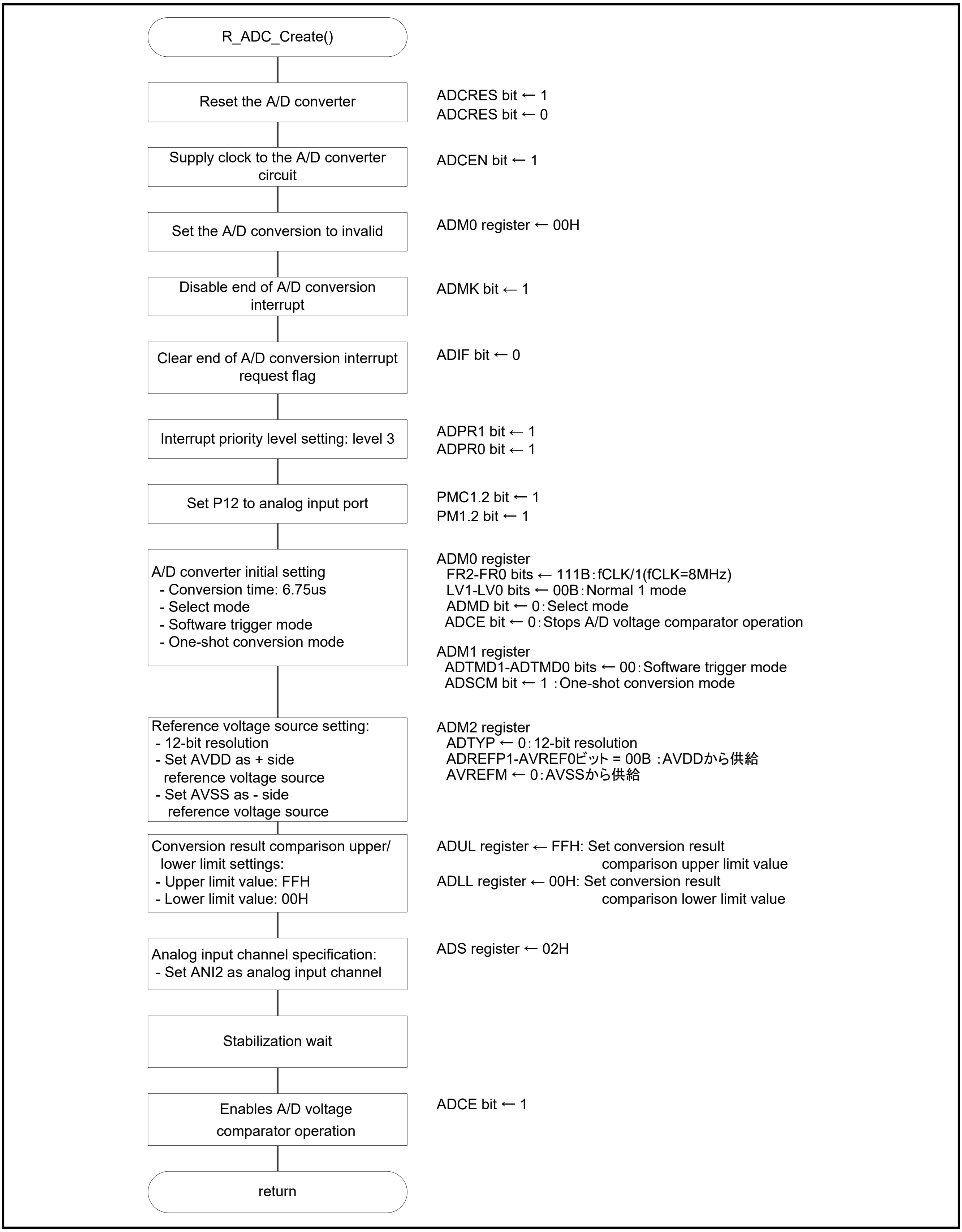

Figure 5.12 A/D Converter Initial Setting

5.8.9 External Interrupt Initial Setting

Figure 5.13 shows the flowchart of the external interrupt initial setting.

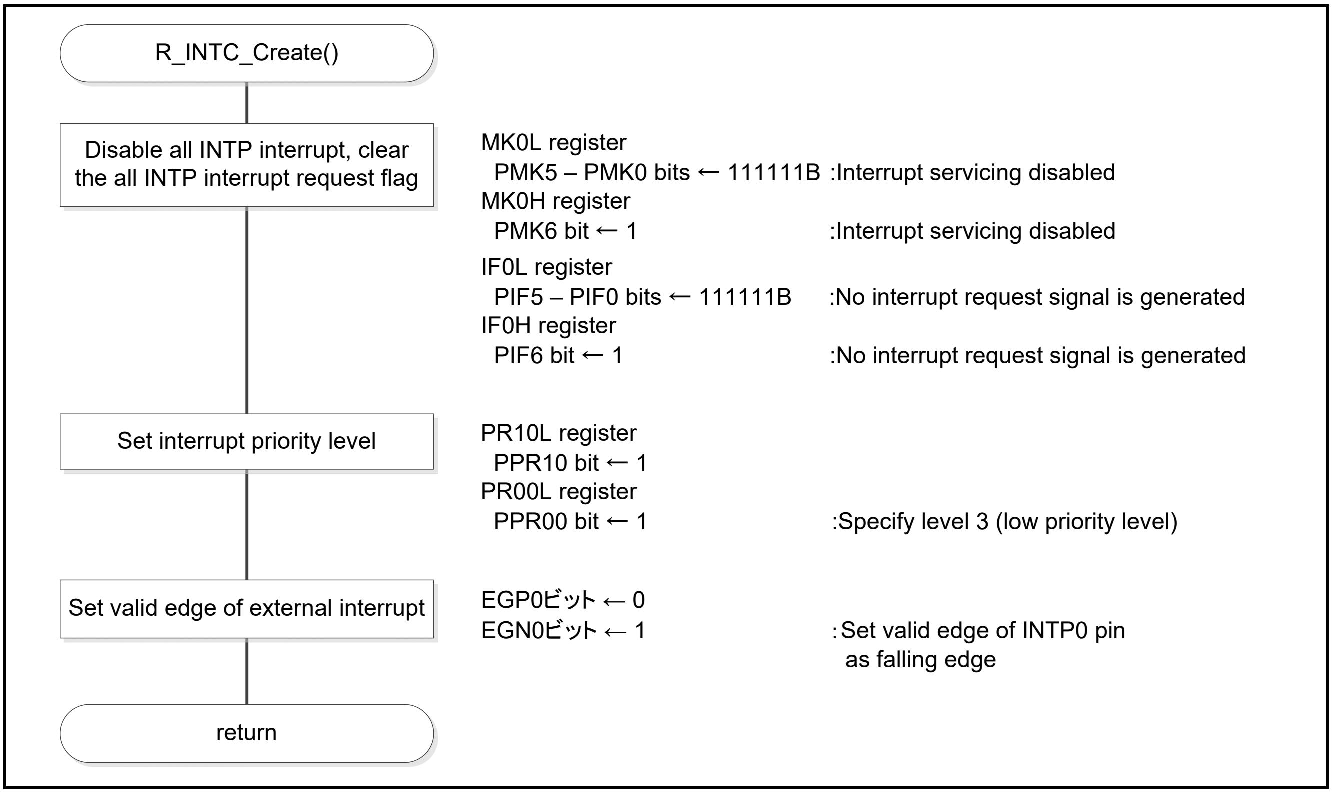

Figure 5.13 External Interrupt Initial Setting

5.8.10 Main Processing

Figure 5.14 shows the flowchart of the main processing.

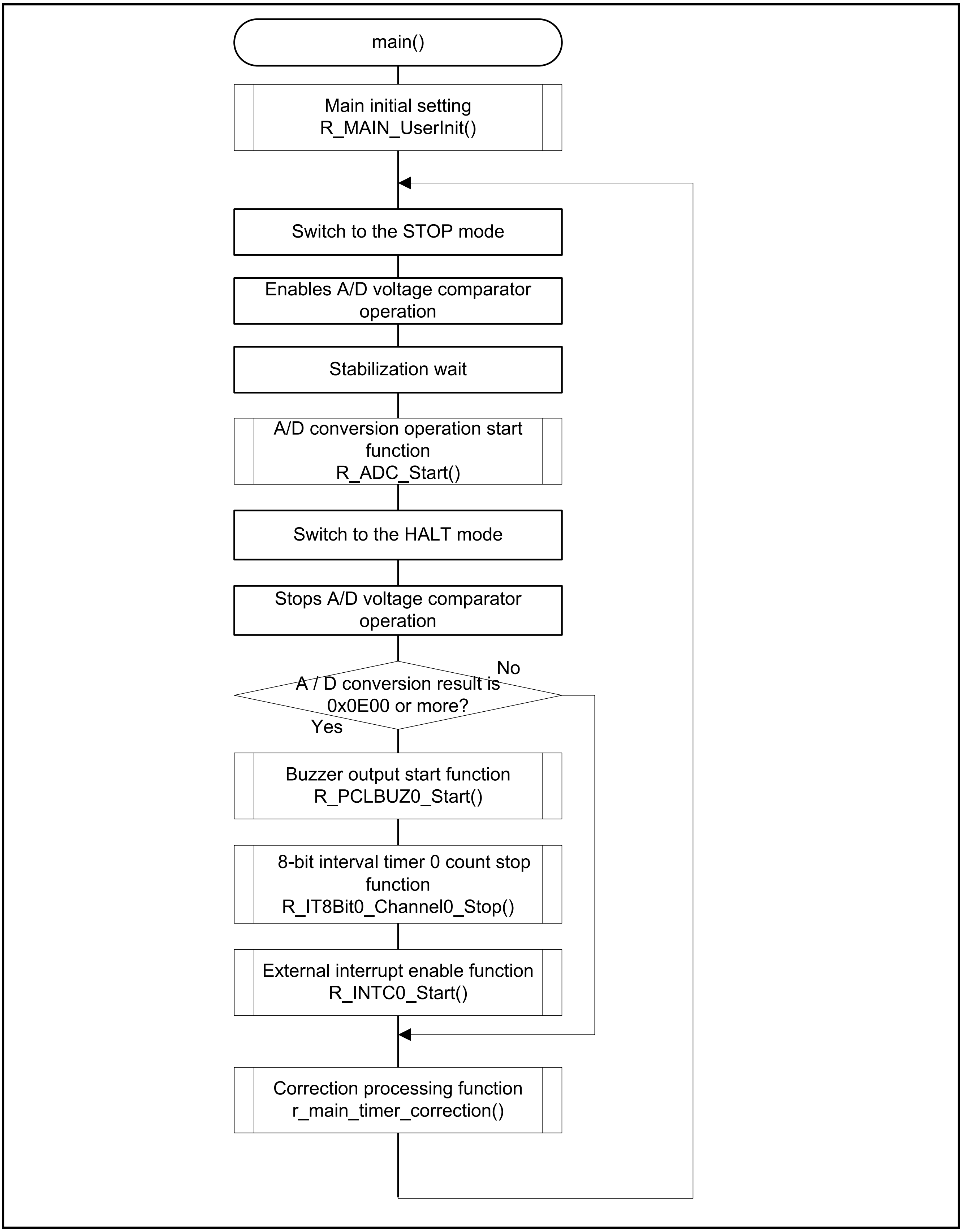

Figure 5.14 Main Processing

5.8.11 Main Initial Setting

Figure 5.15 shows the flowchart of the main initial setting.

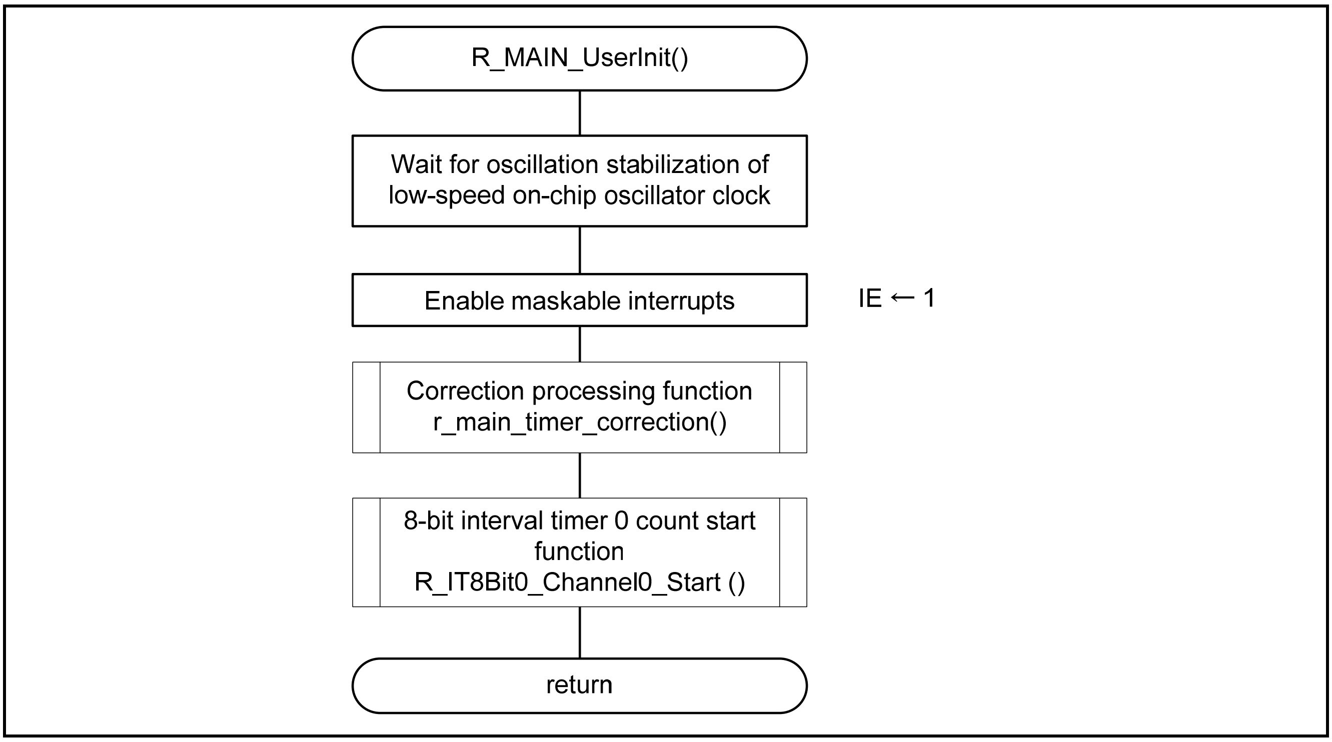

Figure 5.15 Main Initial Setting

5.8.12 A/D Conversion Operation Start Function

Figure 5.16 shows the flowchart of the A/D conversion operation start function.

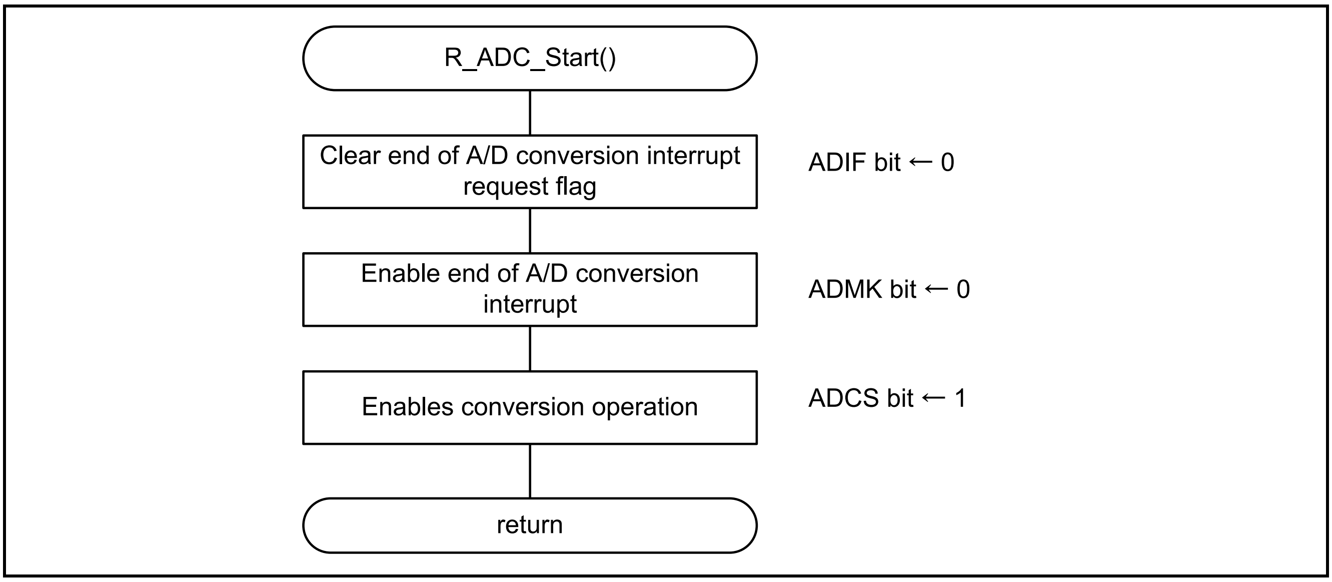

Figure 5.16 A/D Conversion Operation Start Function

5.8.13 Buzzer Output Start Function

Figure 5.17 shows the flowchart of the buzzer output start function.

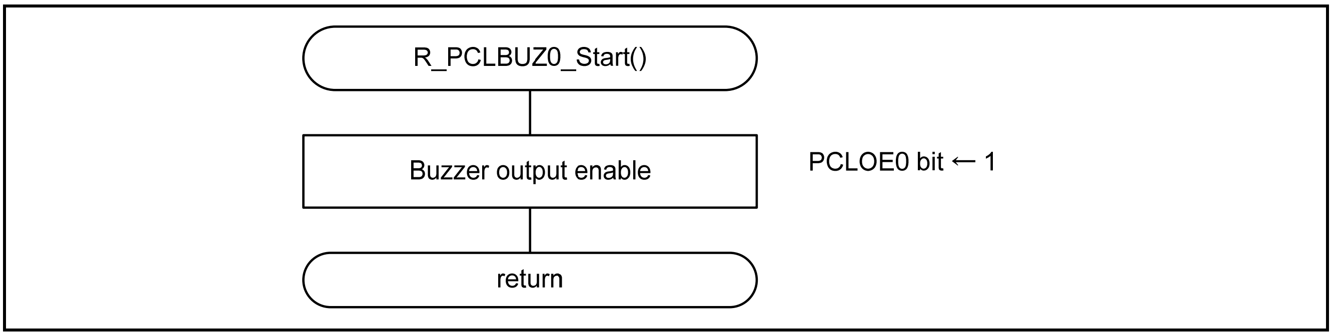

Figure 5.17 Buzzer Output Start Function

5.8.14 8-bit Interval Timer 0 Count Stop Function

Figure 5.18 shows the flowchart of the 8-bit interval timer 0 count stop function.

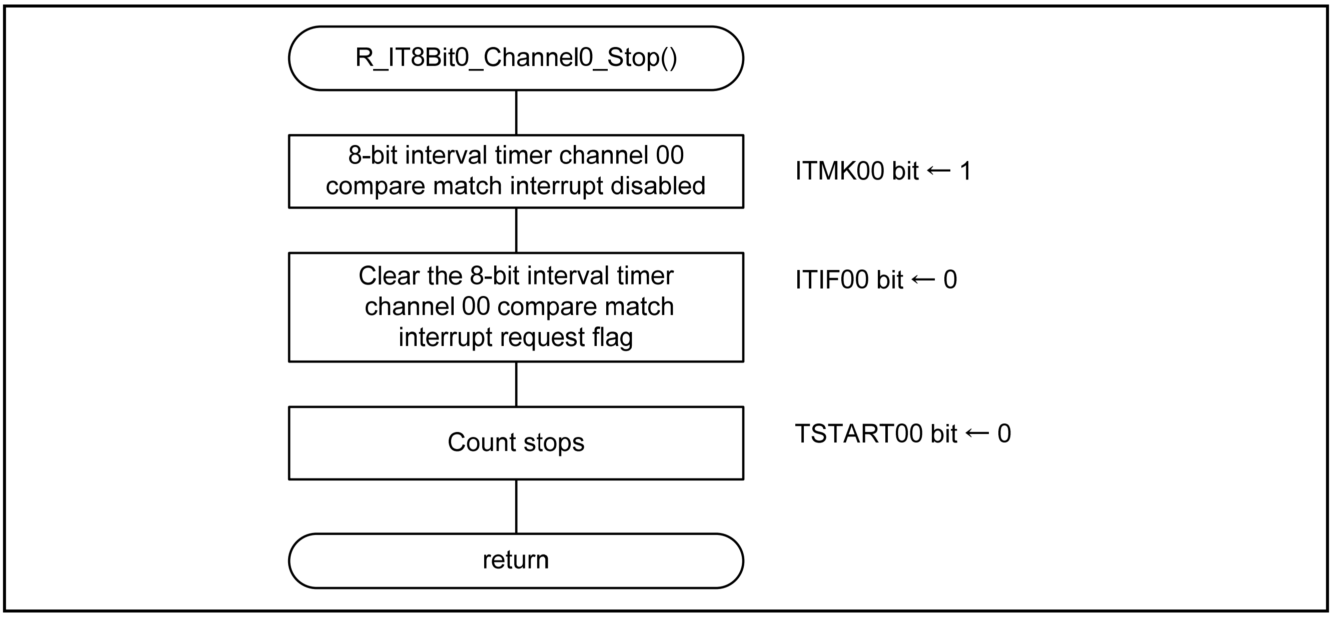

Figure 5.18 8-bit Interval Timer 0 Count Stop Function

5.8.15 External Interrupt Enable Function

Figure 5.19 shows the flowchart of the external interrupt enable function.

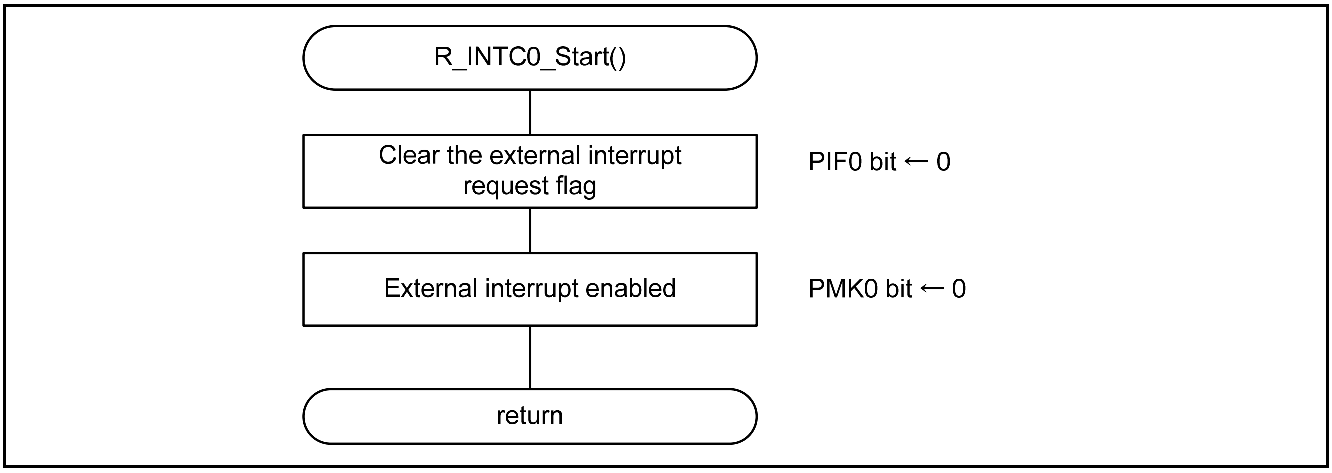

Figure 5.19 External Interrupt Enable Function

5.8.16 Correction Processing Function

Figure 5.20 shows the flowchart of the correction processing function.

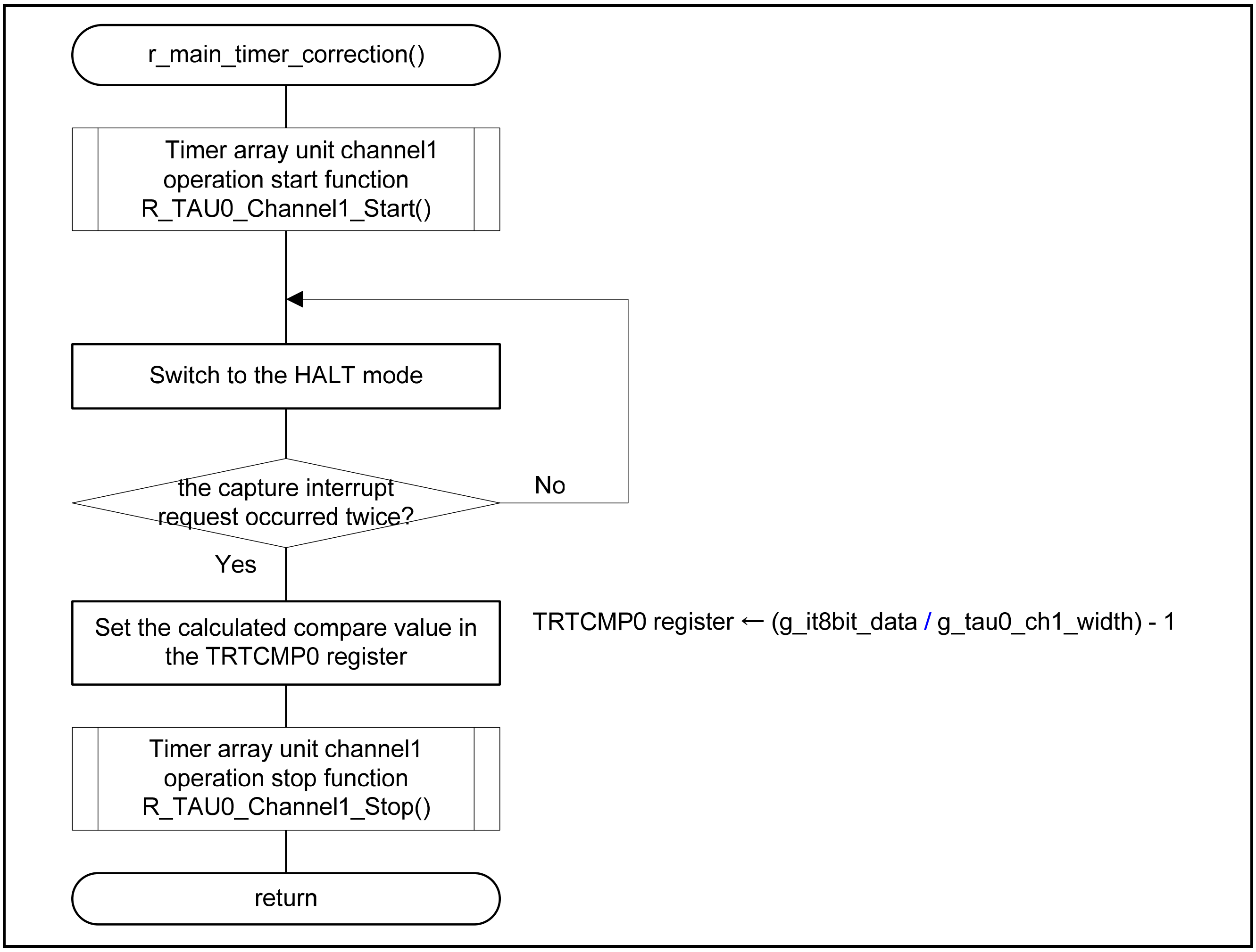

Figure 5.20 Correction Processing Function

5.8.17 8-bit Interval Timer 0 Count Start Function

Figure 5.21 shows the flowchart of the 8-bit interval timer 0 count start function.

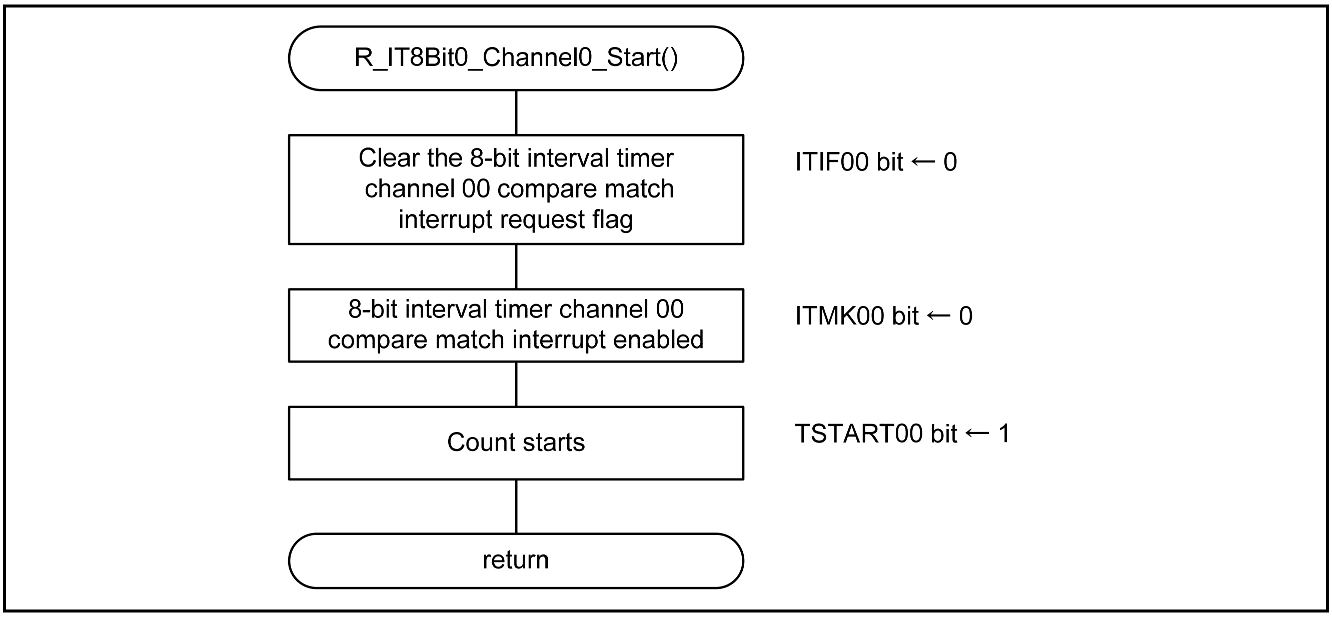

Figure 5.21 8-bit Interval Timer 0 Count Start Function

5.8.18 Timer Array Unit Channel 1 Capture Complete Interrupt Function

Figure 5.22 shows the flowchart of the timer array unit channel 1 capture complete interrupt function.

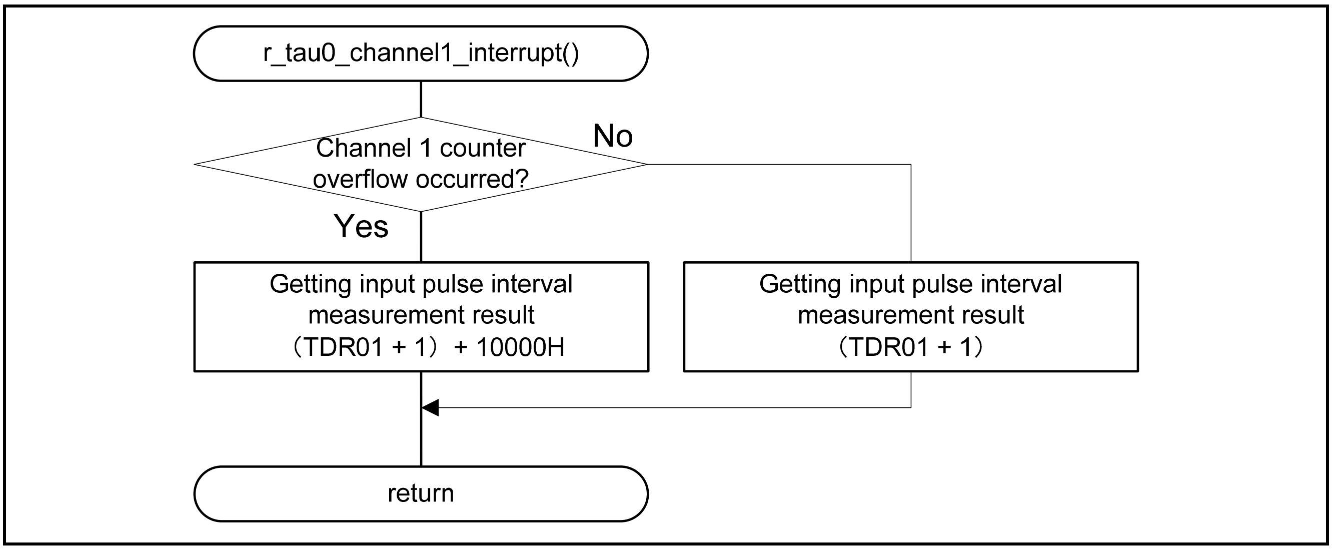

Figure 5.22 Timer Array Unit Channel 1 Capture Complete Interrupt Function

5.8.19 End of A/D conversion interrupt processing

Figure 5.23 shows the flowchart of the end of A/D conversion interrupt processing.

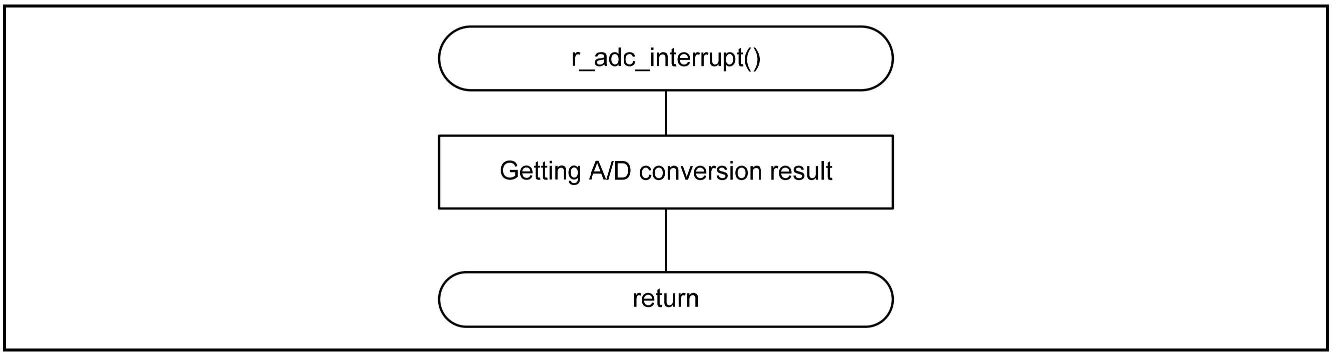

Figure 5.23 End of A/D conversion interrupt processing

5.8.20 External Interrupt Processing

Figure 5.24 shows the flowchart of the external interrupt processing.

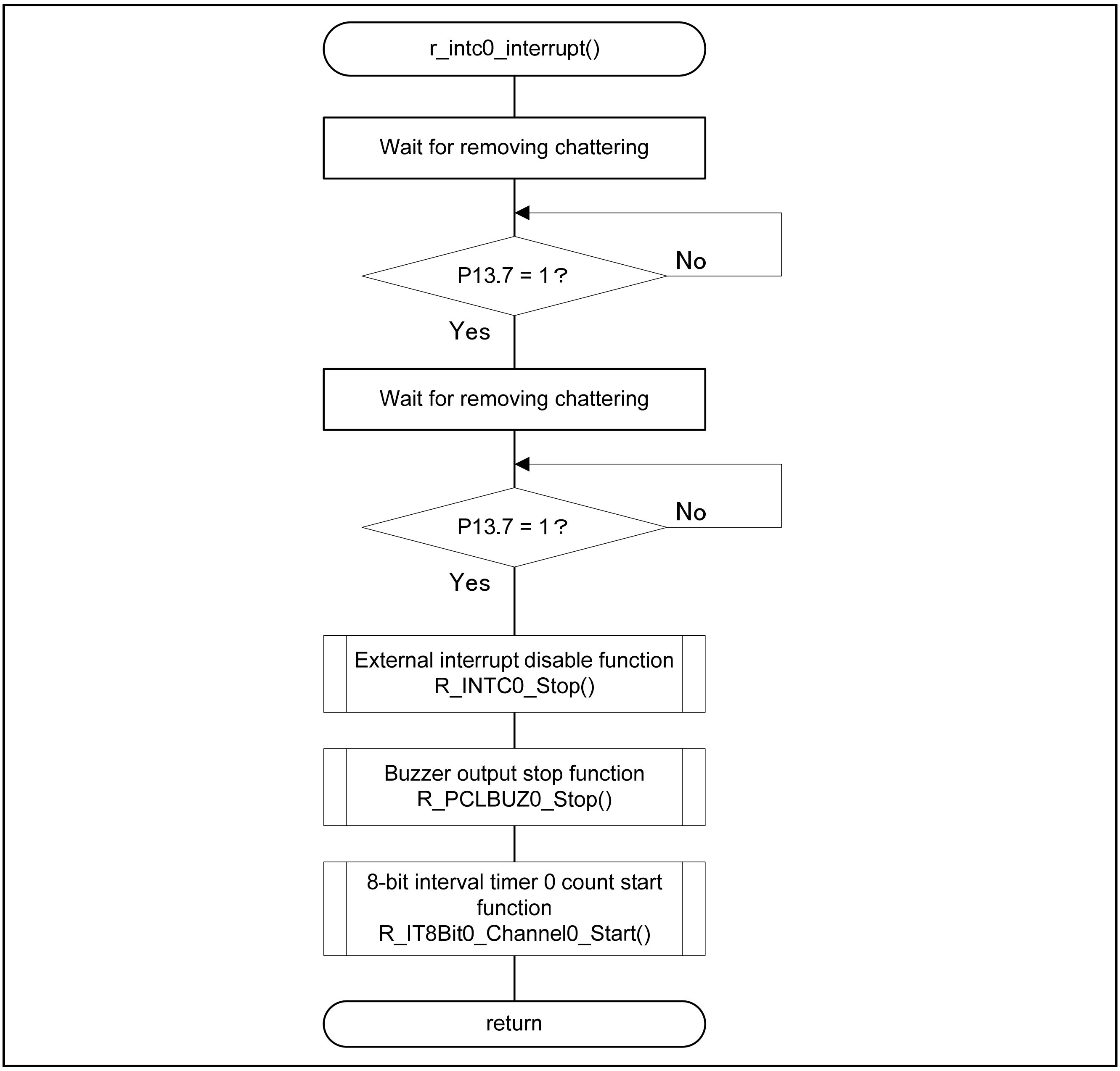

Figure 5.24 External Interrupt Processing

6. Sample Code

Sample code can be downloaded from the Renesas Electronics website.

7. Reference Documents

RL78/I1D User's Manual: Hardware (R01UH0474E)

RL78 Family User's Manual: Software (R01US0015E)

The latest versions can be downloaded from the Renesas Electronics website.

Technical Update/Technical News

The latest information can be downloaded from the Renesas Electronics website.

Website and Support

Renesas Electronics website
http://www.renesas.com

Inquiries
http://www.renesas.com/contact/

Revision History

Rev.	Date	Description	
		Page	Summary
1.00	Nov. 9, 2017	—	First edition issued

General Precautions in the Handling of Microprocessing Unit and Microcontroller Unit Products

The following usage notes are applicable to all Microprocessing unit and Microcontroller unit products from Renesas. For detailed usage notes on the products covered by this document, refer to the relevant sections of the document as well as any technical updates that have been issued for the products.

1. Handling of Unused Pins
   Handle unused pins in accordance with the directions given under Handling of Unused Pins in the manual.
   ¾ The input pins of CMOS products are generally in the high-impedance state. In operation with an unused pin in the open-circuit state, extra electromagnetic noise is induced in the vicinity of LSI, an associated shoot-through current flows internally, and malfunctions occur due to the false recognition of the pin state as an input signal become possible. Unused pins should be handled as described under Handling of Unused Pins in the manual.
2. Processing at Power-on
   The state of the product is undefined at the moment when power is supplied.
   ¾ The states of internal circuits in the LSI are indeterminate and the states of register settings and pins are undefined at the moment when power is supplied.
   In a finished product where the reset signal is applied to the external reset pin, the states of pins are not guaranteed from the moment when power is supplied until the reset process is completed. In a similar way, the states of pins in a product that is reset by an on-chip power-on reset function are not guaranteed from the moment when power is supplied until the power reaches the level at which resetting has been specified.
3. Prohibition of Access to Reserved Addresses
   Access to reserved addresses is prohibited.
   ¾ The reserved addresses are provided for the possible future expansion of functions. Do not access these addresses; the correct operation of LSI is not guaranteed if they are accessed.
4. Clock Signals
   After applying a reset, only release the reset line after the operating clock signal has become stable. When switching the clock signal during program execution, wait until the target clock signal has stabilized.
   ¾ When the clock signal is generated with an external resonator (or from an external oscillator) during a reset, ensure that the reset line is only released after full stabilization of the clock signal. Moreover, when switching to a clock signal produced with an external resonator (or by an external oscillator) while program execution is in progress, wait until the target clock signal is stable.
5. Differences between Products
   Before changing from one product to another, i.e. to a product with a different part number, confirm that the change will not lead to problems.
   ¾ The characteristics of Microprocessing unit or Microcontroller unit products in the same group but having a different part number may differ in terms of the internal memory capacity, layout pattern, and other factors, which can affect the ranges of electrical characteristics, such as characteristic values, operating margins, immunity to noise, and amount of radiated noise. When changing to a product with a different part number, implement a system-evaluation test for the given product.

SALES OFFICES Renesas Electronics Corporation http://www.renesas.com

Refer to "http://www.renesas.com/" for the latest and detailed information.

California Eastern Laboratories, Inc.
4590 Patrick Henry Drive, Santa Clara, California 95054-1817, U.S.A.
Tel: +1-408-919-2500, Fax: +1-408-988-0279

Renesas Electronics Europe Limited
Dukes Meadow, Millboard Road, Bourne End, Buckinghamshire, SL8 5FH, U.K
Tel: +44-1628-585-100, Fax: +44-1628-585-900

Renesas Electronics Europe GmbH
Arcadiastrasse 10, 40472 Düsseldorf, Germany
Tel: +49-211-6503-0, Fax: +49-211-6503-1327

Renesas Electronics (China) Co., Ltd.
Room 1709, Quantum Plaza, No.27 ZhiChunLu Haidian District, Beijing 100191, P.R.China
Tel: +86-10-8235-1155, Fax: +86-10-8235-7679

Renesas Electronics (Shanghai) Co., Ltd.
Unit 301, Tower A, Central Towers, 555 Langao Road, Putuo District, Shanghai, P. R. China 200333
Tel: +86-21-2226-0888, Fax: +86-21-2226-0999

Renesas Electronics Hong Kong Limited
Unit 1601-1611, 16/F., Tower 2, Grand Century Place, 193 Prince Edward Road West, Mongkok, Kowloon, Hong Kong
Tel: +852-2265-6688, Fax: +852 2886-9022

Renesas Electronics Taiwan Co., Ltd.
13F, No. 363, Fu Shing North Road, Taipei 10543, Taiwan
Tel: +886-2-8175-9600, Fax: +886 2-8175-9670

Renesas Electronics Singapore Pte. Ltd.
80 Bendemeer Road, Unit #06-02 Hyflux Innovation Centre, Singapore 339949
Tel: +65-6213-0200, Fax: +65-6213-0300

Renesas Electronics Malaysia Sdn.Bhd.
Unit 1207, Block B, Menara Amcorp, Amcorp Trade Centre, No. 18, Jln Persiaran Barat, 46050 Petaling Jaya, Selangor Darul Ehsan, Malaysia
Tel: +60-3-7955-9390, Fax: +60-3-7955-9510

Renesas Electronics India Pvt. Ltd.
No.777C, 100 Feet Road, HAL II Stage, Indiranagar, Bangalore, India
Tel: +91-80-67208700, Fax: +91-80-67208777

Renesas Electronics Korea Co., Ltd.
12F., 234 Teheran-ro, Gangnam-Gu, Seoul, 135-080, Korea
Tel: +82-2-558-3737, Fax: +82-2-558-5141

Colophon 6.0