RL78/G23

SMS Fire Detection Operation

Introduction

This application note describes how to use the SNOOZE mode sequencer to detect fires.

While the CPU operating clock is stopped, it is possible to control the sensor required for fire detection, measure the sensor output, and judge the measurement result, therefore this can achieve lower power consumption than before.

Target Device

RL78/G23

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.

Contents

1. Specifications

This application note shows how to detect fire by sensing smoke. The smoke sensor is configured a light emitting part, a light receiving part, and an amplifier. If smoke enters between the light emitting part and the light receiving part, the minute voltage of the light receiving part will change. Smoke is sensed by measuring the change in the amplified signal, and the outbreak of a fire is detected.

In the SNOOZE mode sequencer (SMS) processing, sensor control, measurements of sensor outputs and measurement result judgment are performed. When the measurement result exceeds the threshold value, the CPU starts.

The port function is used to control the sensor, and the A/D converter (ADC) is used to measure the sensor output value to judge the measurement result.

In SMS, port operation, A/D conversion, and processing to judge the conversion result are set in advance. The regular interval at which the sensor is started is generated using the 32-bit interval timer (TML32). After entering STOP mode, SMS is started by a TML32 interrupt (INTITL). By starting SMS, port operation, A/D conversion, and conversion result judgment processing are executed in sequence. If the measurement result exceeds the threshold value, an interrupt request signal (INTSMSE) is issued from SMS to start the CPU.

Figure 1-1 shows an example of the system configuration, and Figure 1-2 shows the flowchart of the entire system.

Figure 1-1 System Configuration

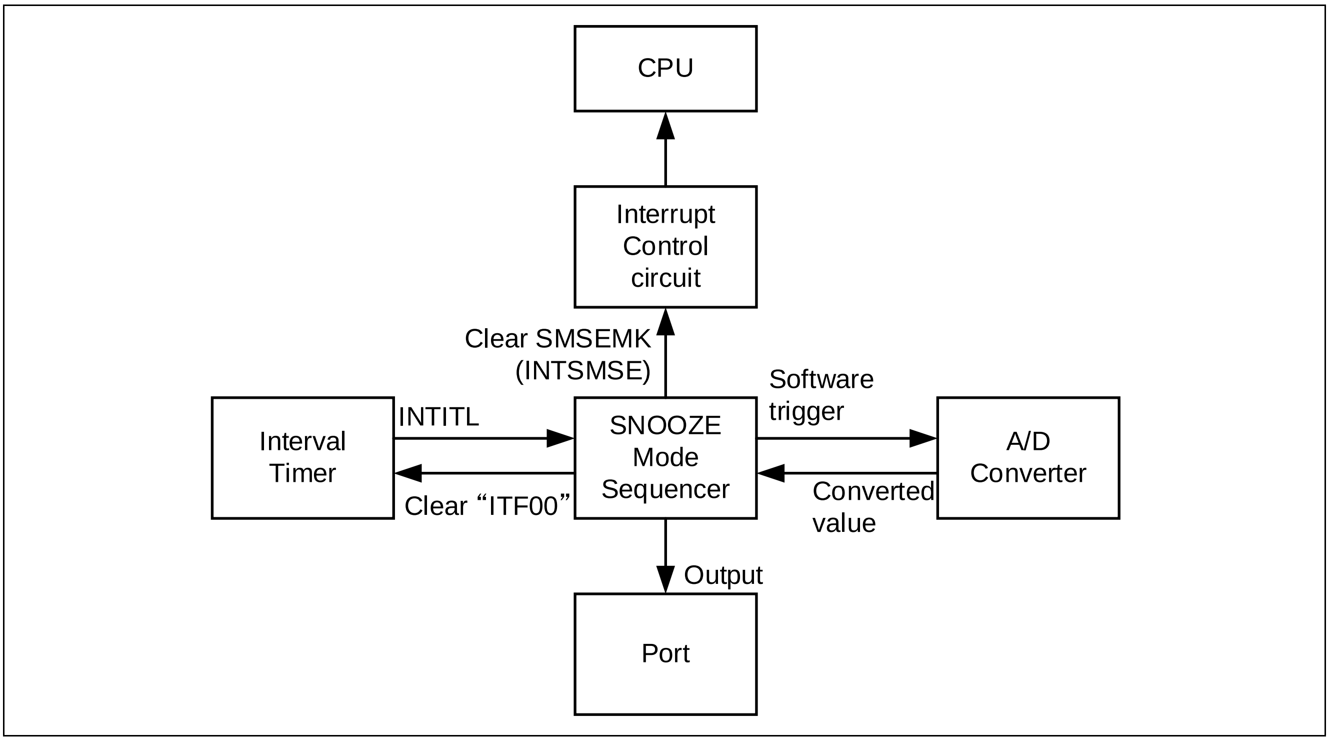
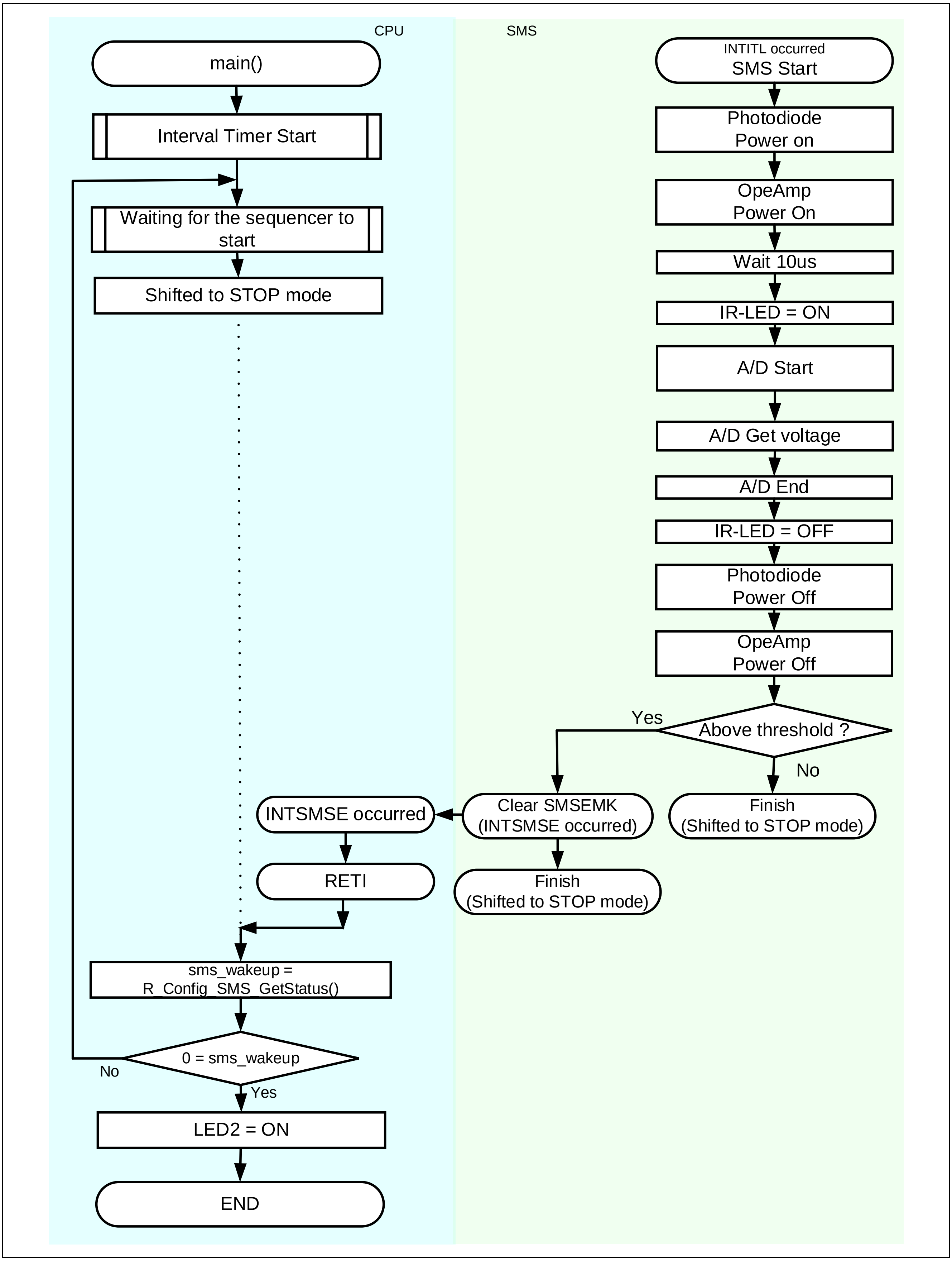


Figure 1-2 Entire Flowchart



2. Conditions for Operation Confirmation Test

The sample code with this application note runs properly under the condition below.

Table 2-1 Operation Confirmation Conditions

Items	Contents
MCU	RL78/G23 (R7F100GLG)
Operating frequencies	• High-speed on-chip oscillator clock: 32 MHz • CPU/peripheral hardware clock: 32 MHz
Operating voltage	• 3.3V • LVD0 operations (V_{LVD0}) : Reset mode Rising edge TYP.1.875V Falling edge TYP.1.835V
Integrated development environment (CS+)	CS+ for CC V8.05.00 from Renesas Electronics Corp.
C compiler (CS+)	CC-RL V1.10 from Renesas Electronics Corp.
Integrated development environment (e^2 studio)	e^2 studio 2021-04 (21.4.0) from Renesas Electronics Corp.
C compiler (e^2 studio)	CC-RL V1.10 from Renesas Electronics Corp.
Integrated development environment (IAR)	IAR Embedded Workbench for Renesas RL78 v4.21.1 from IAR Systems
C compiler (IAR)	
Smart Configurator	V.1.0.1
Board support package (r_bsp)	V.1.10
Emulator	CS＋, e^2 studio: COM port IAR: E2 Emulator Lite
Board	RL78/G23 Fast Prototyping Board (RTK7RLG230CLG000BJ)

3. Related application note

The following application note is related to this application note.

Please refer to them as well.

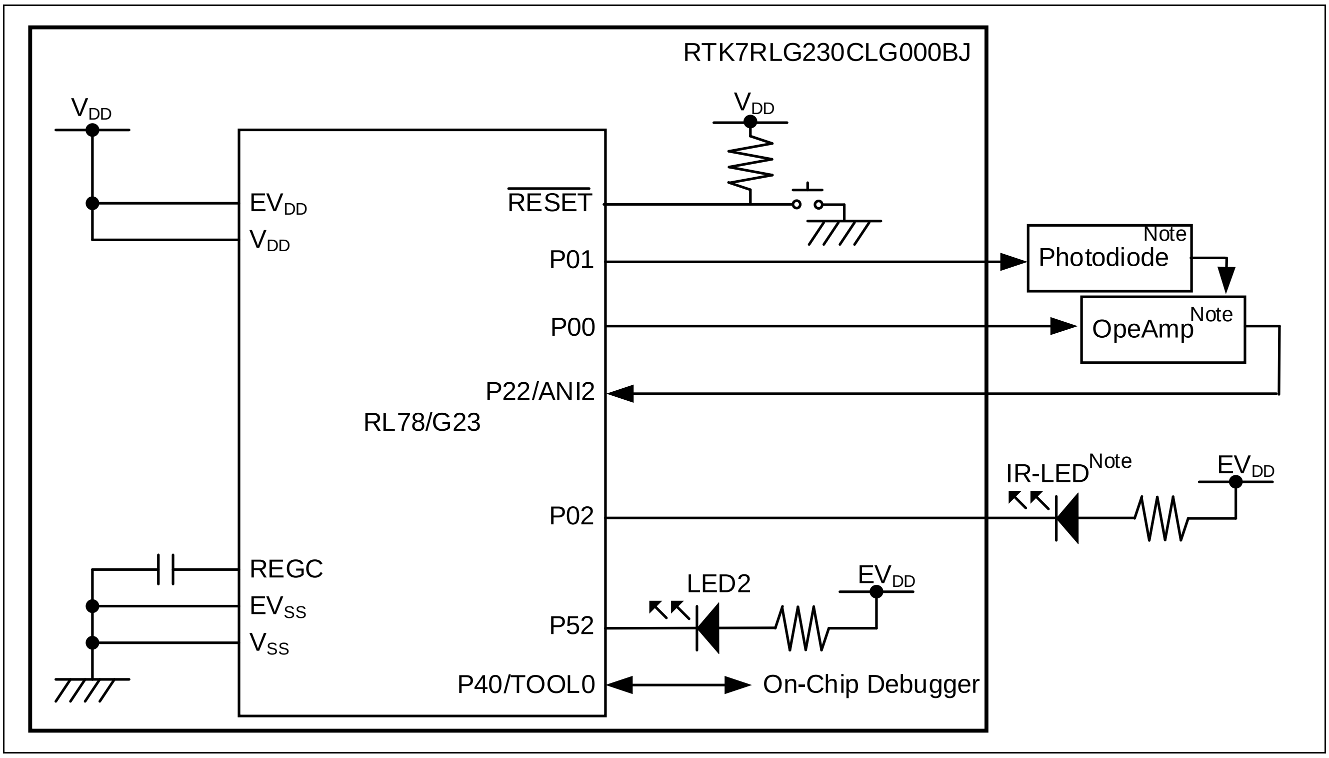
4. Hardware

4.1 Example of Hardware Configuration

Figure 4-1 shows an example of the hardware configuration in this application.

This application note shows an example of using a light emitting part: infrared LED, a light receiving part: photodiode, and an amplifier: operational amplifier as sensors for fire detection.

Figure 4-1 Hardware Configuration



Note. Must be configured outside the board (RTK7RLG230CLG000BJ)

Caution 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.)

Caution 2. Connect the EV_{SS} pin to V_{SS} and the EV_{DD} pin to V_{DD}.

Caution 3. V_{DD} must be held at not lower than the reset release voltage (V_{LVD0}) that is specified as LVD.

4.2 Used Pins

Table 4-1 shows list of used pins and assigned functions.

Table 4-1 List of Pins and Functions

Pin Name	Input/Output	Function
P00	Output	ON/OFF control of operational amplifier Note
P01	Output	ON/OFF control of photodiodeNote
P02	Output	Infrared LEDNote (Low Active)
P52	Output	LED2 lights (Low Active)
P22/ANI2	Input	Input for A/D conversion

Note. Must be configured outside the RTK7RLG230CLG000BJ.

Caution. In this application note, only the used pins are processed. When actually designing your circuit, make sure the design includes sufficient pin processing and meets electrical characteristic requirements.

5. Software

5.1 Overview of the sample program

In this sample code, the CPU shifts from STOP mode to SNOOZE mode by a 32-bit interval timer (TML32) interrupt request (INTITL). Then controls the photodiode, operational amplifier, and infrared LED that are configured externally, and the A/D conversion and judgment of the operational amplifier output are performed by SMS.

The A/D conversion is measured with setting the supply voltage (V_{DD}) as the reference voltage and the ANI2 as the analog input channel. SMS compares the A/D conversion result with the threshold value, and if it exceeds the threshold value, it determines that a fire has occurred and shifts from SNOOZE mode to normal operation to start the CPU.

Caution. When using the RL78/G23 and LED with the same power supply as in the hardware configuration shown in this application note, the LED may not meet the forward voltage standard of the LED and the LED may not light.

The outline of the processing performed by this sample code is shown below.

(1) Starts counting TML32
(2) Shifts to STOP mode
(3) Shifts to SNOOZE mode with TML32 compare match
(4) Starts the photodiode and operational amplifier
(5) Waits until the voltage to be converted to A/D stabilizes
(6) Turns on the infrared LED
(7) Performs A/D conversion
(8) Stops photodiodes and operational amplifiers
(9) Turns off the infrared LED
(10) Branches to (2) if A/D conversion result does not exceed threshold, else branches (11)
(11) Starts the CPU
(12) Shifts to normal operation from SNOOZE mode
(13) Turns on LED2

(4) to (11) are processed by SMS.

5.2 Folder Configuration

Table 5-1 shows folder configuration of source file and header files using by sample code except the files generated by integrated development environment and the files in the bsp environment.

Table 5-1 Folder configuration

Folder/File configuration	Outline	Created by Smart configurator
¥r01an5617_sms_fire_detection<DIR>	Root folder of this sample code	
¥src<DIR>	Folder for program source	
main.c	Sample code source file	
¥smc_gen<DIR>[Note 2]	Folder created by Smart Configurator	√
¥Config_ADC<DIR>	Folder for ADC program	√
Config_ADC.c	Source file for ADC	√
Config_ADC.h	Header file for ADC	√
Config_ADC_user.c	Interrupt source file for ADC	√[Note 1]
¥Config_ITL000_ITL001<DIR>	Folder for TML32 program	√
Config_ITL000_ITL001.c	Source file for TML32	√
Config_ITL000_ITL001.h	Header file for TML32	√
Config_ITL000_ITL001_user.c	Interrupt source file for TML32	√[Note 1]
¥Config_PORT<DIR>	Folder for PORT program	√
Config_PORT.c	Source file for PORT	√
Config_PORT.h	Header file for PORT	√
Config_PORT_user.c	Interrupt source file for PORT	√[Note 1]
¥Config_SMS<DIR>	Folder for SMS program	√
Config_SMS.c	Source file for SMS	√
Config_SMS.h	Header file for SMS	√
Config_SMS_ASM.smsasm	ASM source file for SMS	√
Config_SMS_user.c	Interrupt source file for SMS	√
¥general<DIR>	Folder for initialize or common program	√
¥r_bsp<DIR>	Folder for BSP program	√
¥r_config<DIR>	Folder for BSP_CFG program	√

Note. <DIR> means directory.

Note 1. Not used in this sample code.

Note 2. The sample code of the IAR version has a different configuration. Check the sample code of the IAR version for details. In addition, stores r01an5617_sms_fire_detection.ipcf. For details, refer to "RL78 Smart Configurator User's Guide: IAREW (R20AN0581)".

5.3 Option Byte Settings

Table 5-2 shows the option byte settings.

Table 5-2 Option Byte Settings

Address	Setting Value	Contents
000C0H/040C0H	1110 1111B (EFH)	Operation of Watchdog timer is stopped (counting is stopped after reset)
000C1H/040C1H	1111 1110B (FEH)	LVD0 operating mode: reset mode Detection voltage: Rising edge 1.875V Falling edge 1.835V
000C2H/040C2H	1110 1000B (E8H)	Flash operating mode: HS mode High-speed on-chip oscillator clock: 32MHz
000C3H/040C3H	1000 0101B (85H)	On-chip debugging is enabled

5.4 Constants

Table 5-3 shows the constants that are used in this sample code.

Table 5-3 Constants used in the sample code

Constant Name	Setting Value	Contents	File
LED2	P5_bit.no2	P52	main.c
LED_ON	0	Setting value for turning on the LED	
THRESHOLD	0E8BH	A/D conversion result threshold	

5.5 Variables

Global variables are not used in this sample code.

5.6 Functions

Table 5-4 shows the functions used in the sample code. However, the unchanged functions generated by the Smart Configurator are excluded.

Table 5-4 Functions

Function name	Outline	Source file
main	Main process	main.c

5.7 Function Specifications

This part describes function specifications of the sample code.

[Function name] main

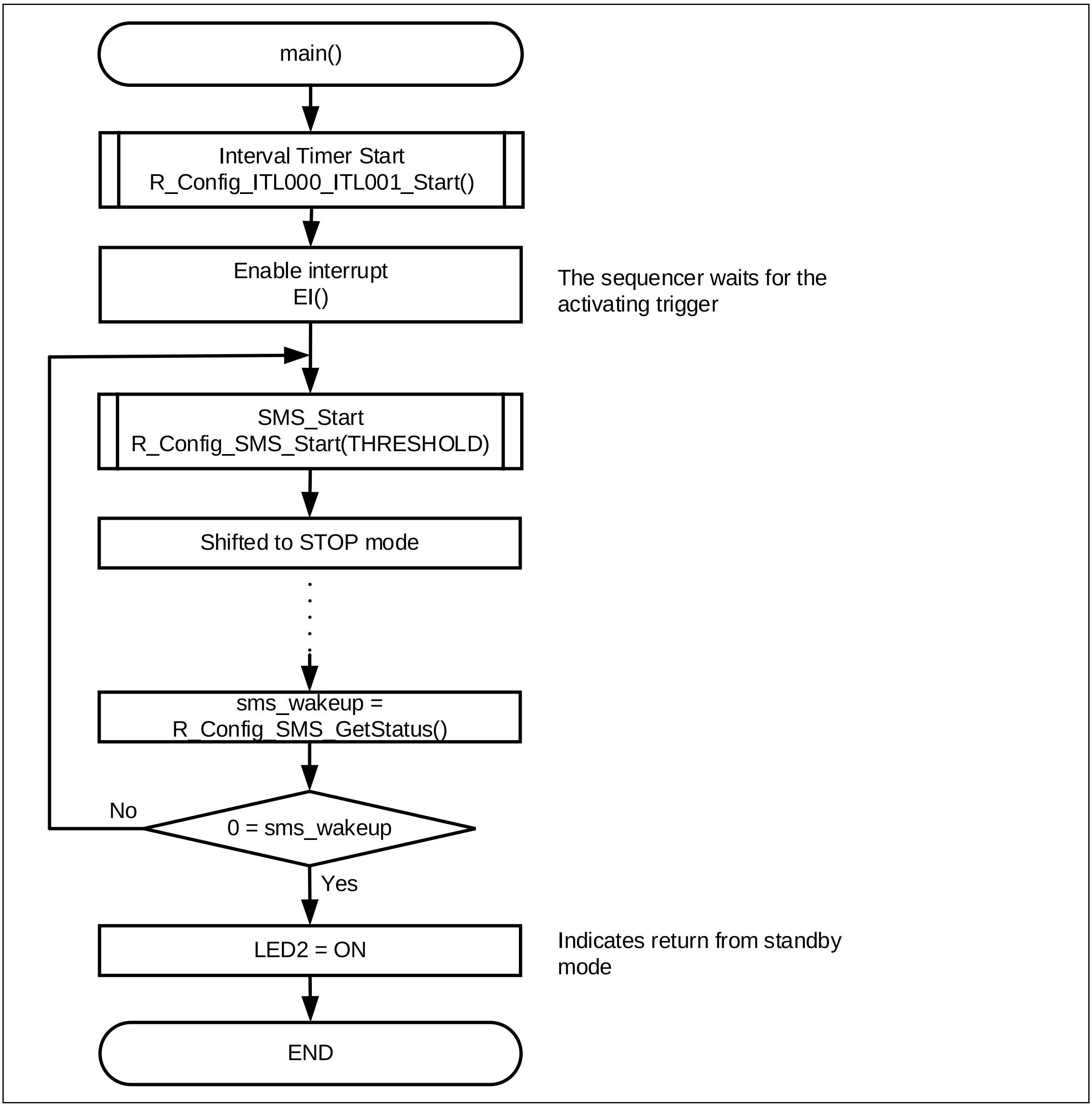
Outline	Main process
Header	r_smc_entry.h
Declaration	void main (void);
Description	This function sets the threshold for starting the CPU from standby mode, then starts TML32 operation and shifts to STOP mode. LED2 lights up when returning from SNOOZE mode.
Arguments	None
Return value	None
Remarks	None

5.8 Flow Charts

5.8.1 Main Process

Figure 5-1 shows flowchart of main process.

Figure 5-1 Main process



5.9 SNOOZE Mode Sequencer settings

When the event set in the start trigger occurs, SMS executes the processing commands stored in the sequencer instruction register (SMSI0-31) in order. When executing a processing command, the Sequencer general-purpose register (SMSG0-15) is used to store the source address, destination address, calculated data, and so on.

SMSI0-31 and SMSG0-15 are set by writing the SMS program (.SMSASM file) in assembly language. The SMS program can also be created by combining processing blocks using the SNOOZE mode sequencer component of the Smart Configurator. The created SMS program is converted to a C language file by the SMS assembler and incorporated into the program.

The specifications of SMS processing executed by the sample code are shown below.

Outline	SMS process
Description	SMS starts with TML32 interrupt and uses the port to starts the photodiode and operational amplifier and wait for stabilization. After turning on the infrared LED, it performs ADC configuration and A/D conversion. It compares the A/D conversion result with the threshold and generates INTSMSE if the result exceeds the threshold.
Arguments Note1	val_adcr_th
Return value	None
Remarks	None

Note1. Argument to be specified in the R_Config_SMS_Start function setting. For details, refer to 6.2.1 and 6.2.12.

Figure 5-2 shows flowchart of SMS process.
Table 5-5 to Table 5-6 show the register settings that control the SNOOZE mode sequencer.

Figure 5-2 SMS process

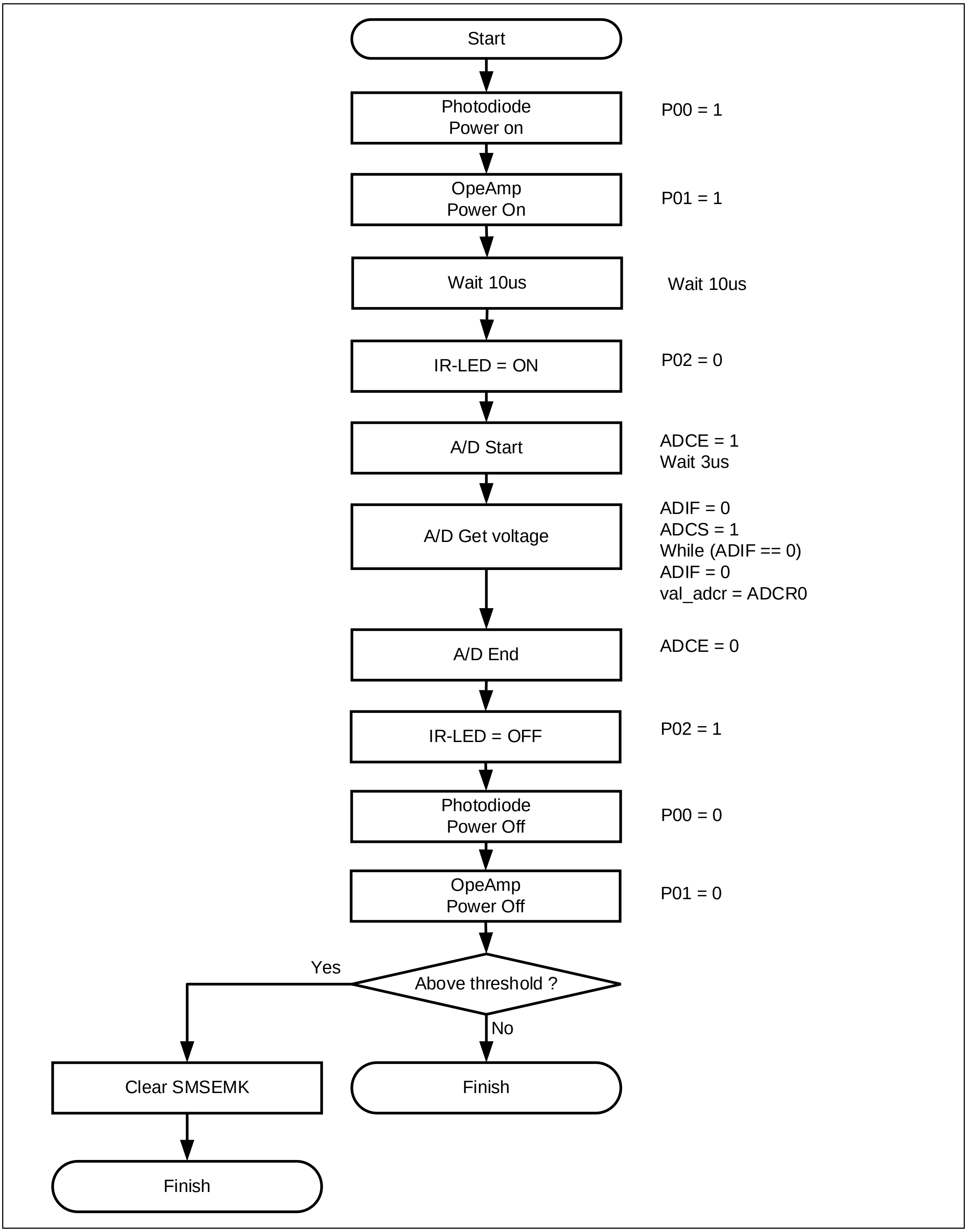


Table 5-5 Sequencer general-purpose registers 0-15

Register Symbol	Setting	Remark
SMSG0	0000H	fixed value: 0000H
SMSG1	0000H	Variable for storing ADCR register: val_adcr
SMSG2	0000H	Threshold of AD conversion result: val_adcr_th
SMSG3	0xFFE5	MK0H address
SMSG4	&ITLS0	ITLS0 address
SMSG5	&P0	P0 address
SMSG6	&ADCR0	ADCR0 address
SMSG7	&ADM0	ADM0 address
SMSG8	&IF1H	IF1H address
SMSG9	0000H	Unused
SMSG10	0000H	Unused
SMSG11	0000H	Unused
SMSG12	0000H	Unused
SMSG13	0000H	Unused
SMSG14	0000H	Unused
SMSG15	FFFFH	fixed value: FFFFH

Table 5-6 Sequencer instruction registers 0-3

Register Symbol	Setting	Remark
SMSI0	0400H	MOV [SMSG4+0], SMSG0
SMSI1	4500H	SET1 [SMSG5+0].0
SMSI2	4510H	SET1 [SMSG5+0].1
SMSI3	9502H	WAIT 80, 2
SMSI4	5520H	CLR1 [SMSG5+0].2
SMSI5	4600H	SET1 [SMSG6+0].0
SMSI6	9600H	WAIT 96, 0
SMSI7	5700H	CLR1 [SMSG7+0].0
SMSI8	4670H	SET1 [SMSG6+0].7
SMSI9	B700H	WHILE0 [SMSG7+0].0
SMSI10	5700H	CLR1 [SMSG7+0].0
SMSI11	3310H	MOVW SMSG1, [SMSG3+0]
SMSI12	5600H	CLR1 [SMSG6+0].0
SMSI13	4520H	SET1 [SMSG5+0].2
SMSI14	5500H	CLR1 [SMSG5+0].0
SMSI15	5510H	CLR1 [SMSG5+0].1
SMSI16	7122H	CMPW SMSG1, SMSG2
SMSI17	8021H	BNC $2
SMSI18	F000H	FINISH
SMSI19	5340H	CLR1 [SMSG3+0].4
SMSI20	F000H	FINISH
SMSI21-31	0000H	Unused

6. Application example

In addition to the sample code, this application note stores the following Smart Configurator configuration files.

r01an5617_sms_fire_detection.scfg

r01an5617_sms_fire_detection.sms

The following is a description of the file and setting examples and precautions for use.

6.1 r01an5617_sms_fire_detection.scfg

This is the Smart Configurator configuration file used in the sample code. It contains all the features configured in the Smart Configurator. The sample code settings are as follows.

Table 6-1 Parameters of Smart Configurator

Tag name	Components	Contents
Clocks	-	Operation mod: High-speed main mode 2.4 (V) ~ 5.5 (V) EV_{DD} setting: 1.8V≦EV_{DD0}＜5.5V High-speed on-chip oscillator: 32MHz f_{IHP}: 32MHz f_{CLK}: 32000kHz (High-speed on-chip oscillator) f_{SXP}: 32.768kHz (Low-speed on-chip oscillator)
System	-	On-chip debug operation setting: COM port Note 1 Pseudo-RRM/DMM function setting: Used Start/Stop function setting: Unused Trace function setting: Used Security ID setting: Use security ID Security ID：0x00000000000000000000 Security ID authentication failure setting: Do not erase flash memory data
Components	r_bsp	Start up select : Enable (use BSP startup) Control of invalid memory access detection : Disable RAM guard space (GRAM0-1) : Disabled Guard of control registers of port function (GPORT) : Disabled Guard of registers of interrupt function (GINT) : Disabled Guard of control registers of clock control function, voltage detector, and RAM parity error detection function (GCSC) : Disabled Data flash access control (DFLEN) : Disables Initialization of peripheral functions by Code Generator/Smart Configurator : Enable API functions disable : Enable Parameter check enable : Enable Setting for starting the high-speed on-chip oscillator at the times of release from STOP mode and of transitions to SNOOZE mode : High-speed Enable user warm start callback (PRE) : Unused Enable user warm start callback (POST) : Unused Watchdog Timer refresh enable : Unused
	Config_LVD0	Operation mode setting: Reset mode Voltage detection setting: Reset generation level (V_{LVD0}): 1.835 (V)

Table 6.2 Parameters of Smart Configurator

Tag name	Components	Contents
Components	Config_ITL000 _ITL001	Components: Interval Timer Operation mode: 16 bit count mode Resource: ITL000_ITL001 Operation clock: f_{SXP} Clock source: f_{ITL0}/128 Interval value: 10 s Interrupt setting: unused
	Config_ADC	Components: A/D Converter Comparator operation setting: Stop Resolution setting: 12 bits VREF (+) setting: V_{DD} VREF (-) setting: V_{SS} Trigger mode setting: Software trigger no-wait mode Operation mode setting: One-shot select mode A/D channel selection: ANI2 Conversion time mode: Normal 1 Conversion time: 66/f_{CLK} Conversion result upper/lower bound value setting: Generates an interrupt request (INTAD) when ADLL≦ADCRn≦ADUL Upper bound (ADUL) value: 255 Lower bound (ADLL) value: 0 Interrupt setting: unused
	Config_SMS	Components: SNOOZE Mode Sequencer Start trigger: Interval detection interrupt (INTITL)
	Config_PORT	Components: Port Port selection: PORT5 P52: Out (Output 1) Port selection: PORT0 P00: Out (Output 0) P01: Out (Output 0) P01: Out (Output 1)

Note 1. When using IAR, use the following settings.
On-chip debug operation setting: Use emulator
Emulator setting: E2 Emulator Lite

6.1.1 Clocks

Set the clock used in the sample code.

In this sample code, 32000KHz is set for f_{CLK} and the conversion time mode is set to "Standard 1 (2.4 V ≤ V_{DD} ≤ 5.5 V)" with Config_ADC, so the operation mode is "High-speed main mode 2.4 (V) ~ 5.5 (V)”. Note that changing the settings.

6.1.2 System

Set the on-chip debug of the sample code.

"Control of on-chip debug operation" and "Security ID authentication failure setting" affect "On-chip debugging is enabled" in "Table 5-2 Option Byte Settings". Note that changing the settings.

6.1.3 r_bsp

Set the startup of the sample code.

6.1.4 Config_LVD0

Set the power management of the sample code.

Affects "Setting of LVD0" in "Table 5-2 Option Byte Settings". Note that changing the settings

6.1.5 Config_IT000_ITL001

Initialize the interval timer for the sample code.

The interval timer interrupt (INTITL) is used to start the SMS in the sample code. Therefore, "Interrupt setting" is set to "Not used". "Interrupt Settings" can also be changed to "Use".

Since INTITL is masked by the R_Config_SMS_Start function, the CPU will not start even if INTITL is generated during STOP or SNOOZE mode. After returning from STOP mode and SNOOZE mode, INTITL is in a masked state, so unmask INTITL if necessary.

6.1.6 Config_ADC

Initialize the ADC for the sample code.

In the sample code, "VREF (+) setting" is set to V_{DD} and "A/D channel selection" is set to ANI0. It is also possible to change "A/D channel selection" to another ANI pin. And "the internal reference voltage" or "the temperature sensor output voltage" can be selected too. However, the A/D converter reference voltage current and temperature sensor operating current will flow during STOP mode in this case.

In the sample code, A/D conversion is not performed when the device is not in SOOZE mode, so "Interrupt Settings" is set to "Not Used". "Interrupt Settings" can also be changed to "Use". Since INTAD is masked by the R_Config_SMS_Start function, the CPU will not start even if INTAD is generated during STOP or SNOOZE mode. After returning from STOP mode and SNOOZE mode, INTAD is in a masked state, so unmask INTAD if necessary.

6.1.7 Config_SMS

Set the sample code SMS.

For details, refer to "6.2 r01an5617_sms_fire_detection.sms".

6.1.8 Config_PORT

Set the port of the sample code.

In the sample code, P52 is used to control LED2, P00 to control the operational amplifier, P01 to control the photodiode, and P02 to control the infrared LED. P00, P01, and P02 can be changed to other pins, but the settings in Config_SMS must be changed accordingly. For details, refer to “6.3 Example of changing the output pins with this sample code”.

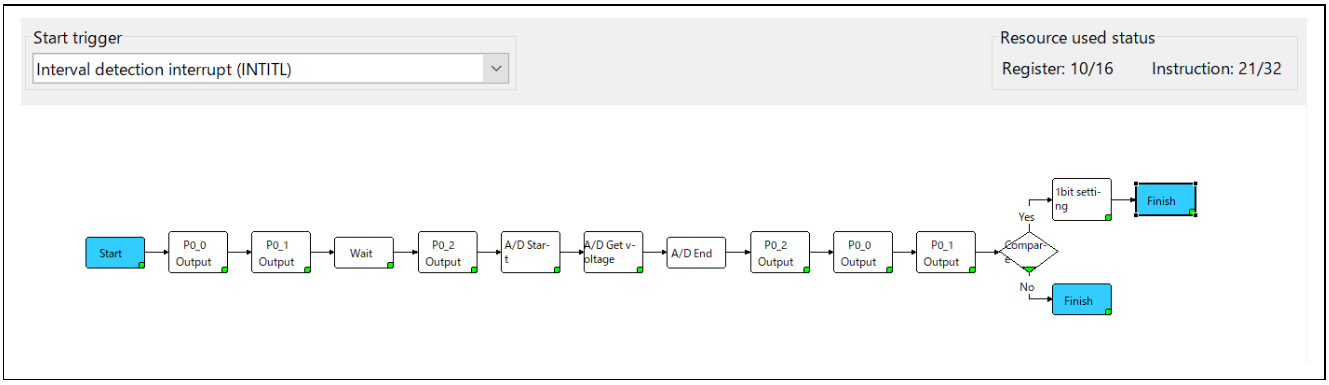
6.2 r01an5617_sms_fire_detection.sms

This is the data for Config_SMS alone. In the sample code, the interrupt of the interval timer is used to start SMS, and A/D is used in the operation of SMS. Note that it is necessary to set the interval timer and A/D separately.

The r01an5617_sms_fire_detection.sms can also be imported into the Smart Configurator of another project. After setting up the SMS component in another project, go to [Import SMS Sequence] -> [Browse] and select "r01an5617_sms_fire_detection.sms" to import it.

When imported into the smart configurator, the flow chart will be as shown in Figure 6-1. This flow chart is the same as "Figure 5-2 SMS process".

Figure 6-1 Config_SMS flow chart



Start trigger

Interval detection interrupt (INTITL)

Resource used status

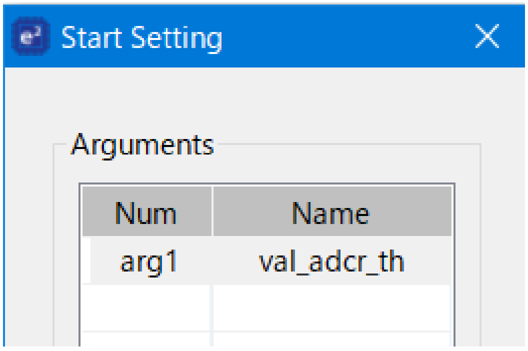
Register: 10/16 Instruction: 21/32

A description of each block is shown below.

6.2.1 Start

When the SMS starts, the value of THESHOLD passed as an argument in the SMS start function (R_Config_SMS_Start function) is set to val_adcr_th (A/D conversion result threshold).

Figure 6-2 Start Setting



Start Setting

Arguments

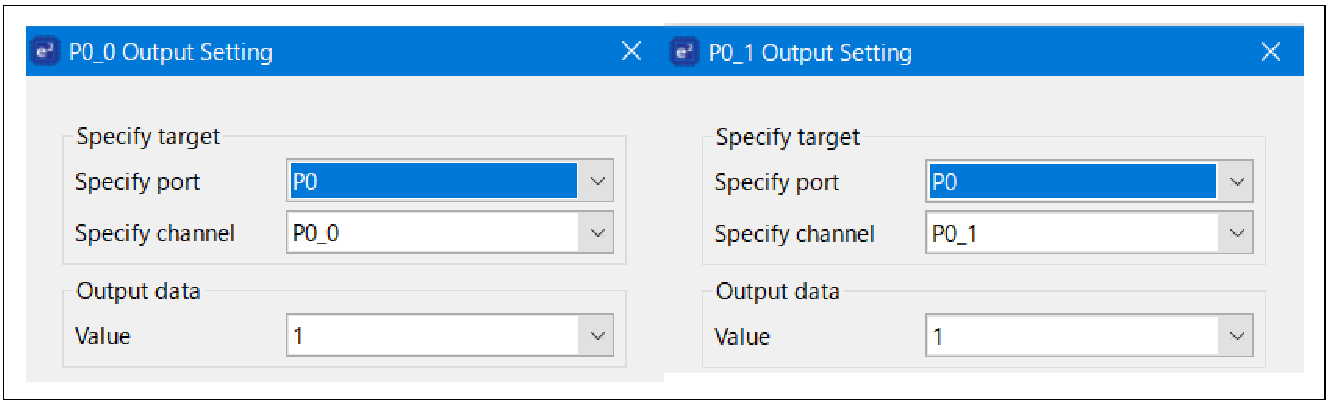
Num	Name
arg1	val_adcr_th

6.2.2 P0_0 Output、P0_1 Output

Outputs the value of the output data to the target specified port. In the sample code, "1" is output to P0_0 and "1" is output to P0_1. To set the port (e.g., change to output mode), use the Config_Port component.

In the sample code, P0_0 controls the ON of the operational amplifier and P0_1 controls the ON of the photodiode, so make sure to do this before starting the A/D operation.

Figure 6-3 P0_0, P0_1 Output Setting



P0_0 Output Setting

Specify target
Specify port: P0
Specify channel: P0_0

Output data
Value: 1

P0_1 Output Setting

Specify target
Specify port: P0
Specify channel: P0_1

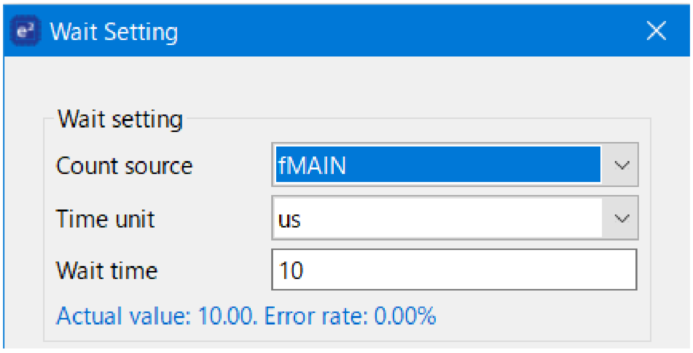
Output data
Value: 1

6.2.3 Wait

Wait for the SMS processing for the set wait time. In the sample code, the waiting time for the photodiode to start up is set, and the processing waits for 10us in the count source (f_{MAIN}).

When changing the waiting time, if you set a value that cannot be set, the value will be in red. Note that changing the waiting time.

Figure 6-4 Wait Setting



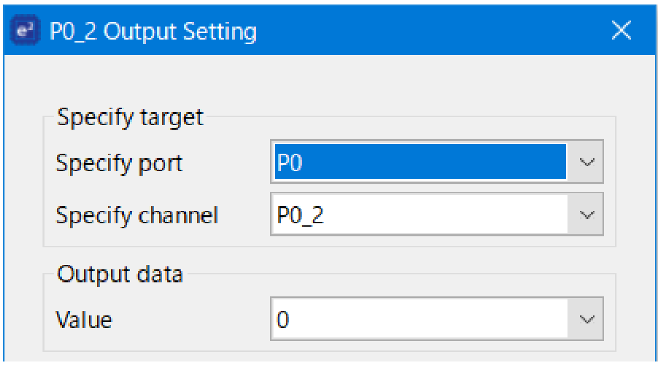
Wait Setting

Wait setting
Count source: fMAIN
Time unit: us
Wait time: 10
Actual value: 10.00. Error rate: 0.00%

6.2.4 P0_2 Output

Outputs "0" to P0_2.

Figure 6-5 P0_2 Output Setting



P0_2 Output Setting

Specify target

Specify port: P0

Specify channel: P0_2

Output data

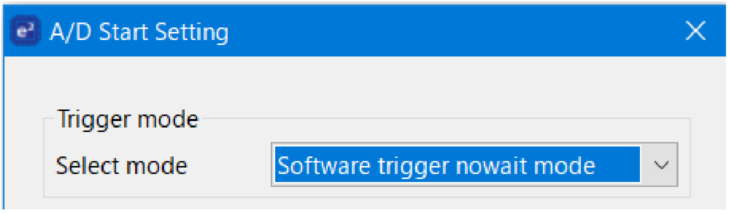
Value: 0

6.2.5 A/D Start

Set the trigger mode for A/D. The waiting time is automatically added according to the mode.

In the sample code, the A/D operation is started without changing the A/D conversion target, but it is also possible to change the A/D conversion target before the A/D operation starts. Before starting the A/D operation, add "A/D Change channel" and change the A/D conversion target.

Figure 6-6 A/D Start Setting



A/D Start Setting

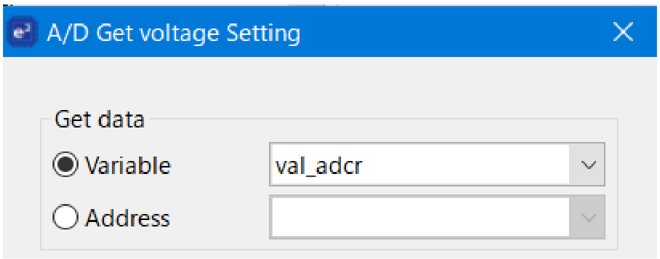
Trigger mode

Select mode: Software trigger nowait mode

6.2.6 A/D Get voltage

Converts A/D and stores the value of the A/D conversion result (ADCR0) in the variable val_adcr.

Figure 6-7 A/D Get voltage Setting



A/D Get voltage Setting

Get data

● Variable: val_adcr

○ Address:

6.2.7 A/D End

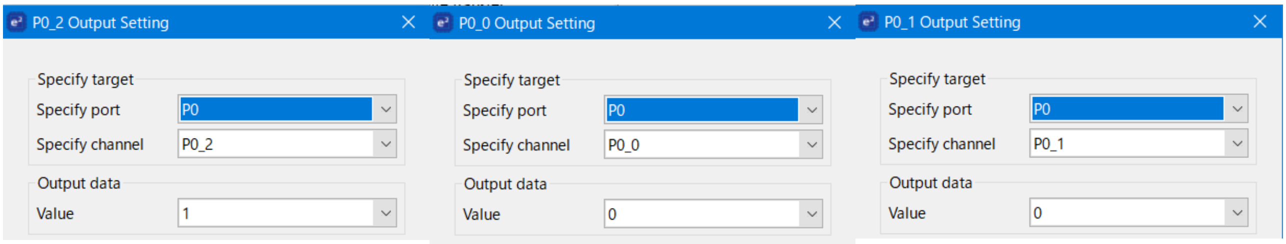
End the A/D conversion.

6.2.8 Port Output

Output "1" to P0_2, "0" to P0_0, and "0" to P0_1.

In the sample code, P0_0 controls the OFF of the operational amplifier and P0_1 controls the OFF of the photodiode, so be sure to do this after the start of A/D operation.

Figure 6-8 Port Output Setting



P0_2 Output Setting

Specify target

Specify port: P0

Specify channel: P0_2

Output data

Value: 1

P0_0 Output Setting

Specify target

Specify port: P0

Specify channel: P0_0

Output data

Value: 0

P0_1 Output Setting

Specify target

Specify port: P0

Specify channel: P0_1

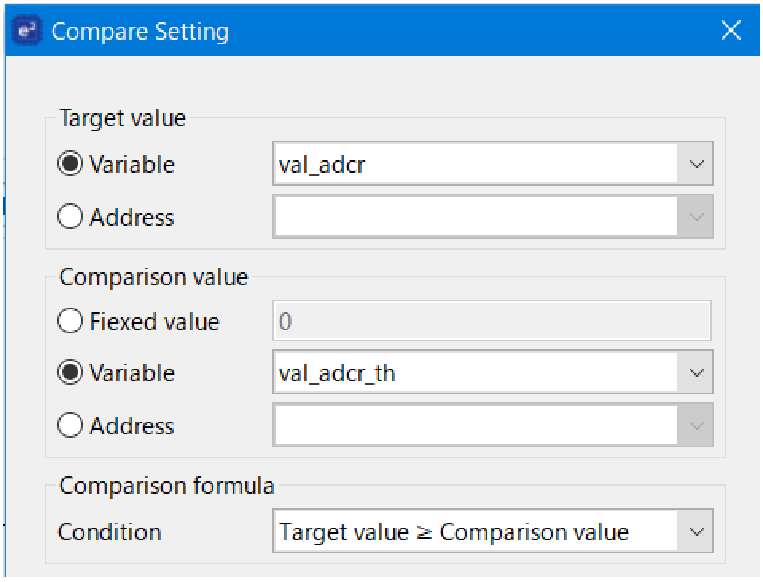
Output data

Value: 0

6.2.9 Compare

This function compares whether the A/D conversion result stored in the variable val_adcr is greater than the threshold value stored in the variable val_adcr_th. If val_adcr is greater than the threshold value, it judges that the power supply voltage is less than the arbitrary value and returns to the CPU operation mode from the SNOOZE mode. va_adcr is less than the threshold value, the supply voltage is judged to be greater than or equal to the arbitrary value, and the mode shifts to STOP mode.

Figure 6-9 Compare Setting



Compare Setting

Target value

● Variable: val_adcr

○ Address:

Comparison value

○ Fiexed value: 0

● Variable: val_adcr_th

○ Address:

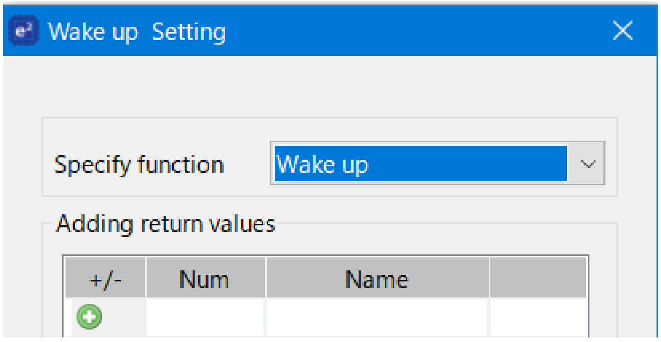
Comparison formula

Condition: Target value ≥ Comparison value

6.2.10 Wake up

Returns to the CPU operation mode. In the sample code, the return value is not used.

Figure 6-10 Wake up Setting



Wake up Setting

Specify function: Wake up

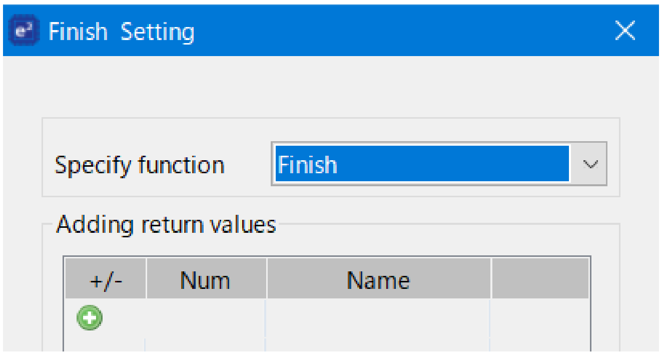
Adding return values

+/-	Num	Name	
⊕			

6.2.11 Finish

It shifts to STOP mode. In the sample code, the return value is not used.

Figure 6-11 Finish Setting



Finish Setting

Specify function: Finish

Adding return values

+/-	Num	Name	
⊕			

6.2.12 Variable Setting

The settings of the variables used in SMS are shown below.

Table 6.3 Variables used in SMS

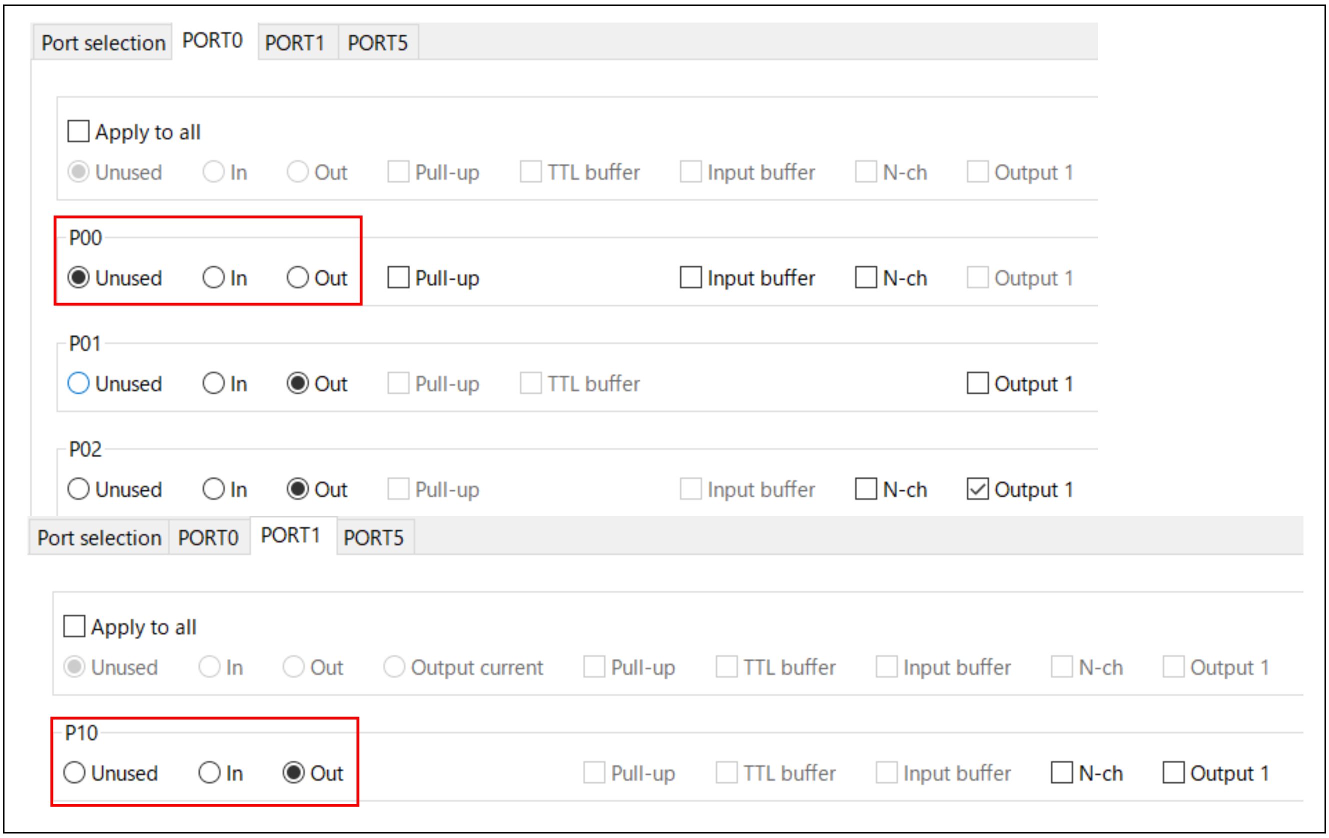
Data name	Initialization mode	Initial value	Description
val_adcr	No initialization	-	Stores the A/D conversion result.
val_adcr_th	Pass argument via SMS start function	-	Stores the address of the threshold. The value of THESHOLD is set as an argument in the R_Config_SMS_Start function.

6.3 Example of changing the output pins with this sample code

The following is an example of changing the output pin P0_0 to P1_0.

(1) Change the settings of P00 and P10 in Config_PORT as shown in Figure 6-12.

Figure 6-12 Config_PORT Setting



Port selection | PORT0 | PORT1 | PORT5

□ Apply to all
● Unused ○ In ○ Out □ Pull-up □ TTL buffer □ Input buffer □ N-ch □ Output 1

P00
● Unused ○ In ○ Out □ Pull-up □ Input buffer □ N-ch □ Output 1

P01
○ Unused ○ In ● Out □ Pull-up □ TTL buffer □ Output 1

P02
○ Unused ○ In ● Out □ Pull-up □ Input buffer □ N-ch ☑ Output 1

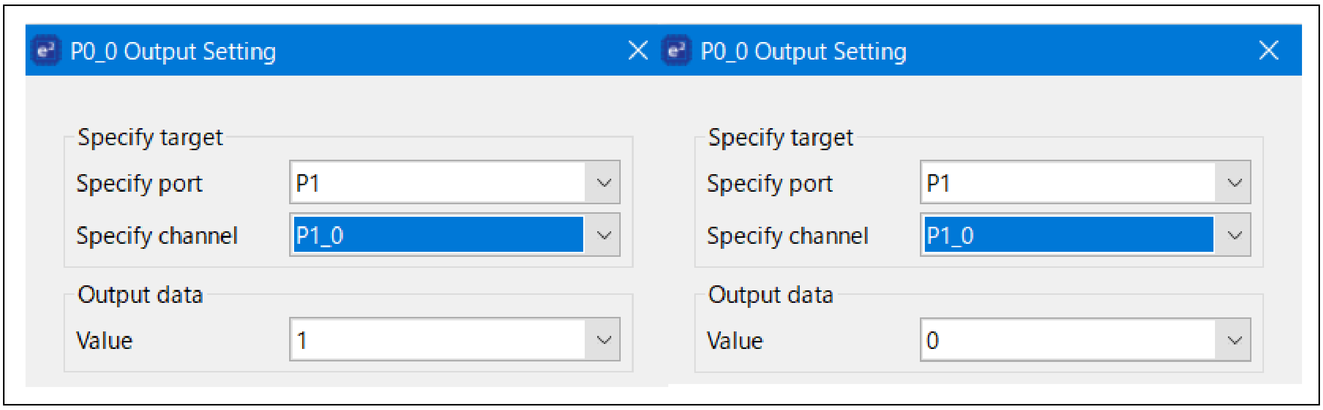
Port selection | PORT0 | PORT1 | PORT5

□ Apply to all
● Unused ○ In ○ Out ○ Output current □ Pull-up □ TTL buffer □ Input buffer □ N-ch □ Output 1

P10
○ Unused ○ In ● Out □ Pull-up □ TTL buffer □ Input buffer □ N-ch □ Output 1

(2) In the Config_SMS flow, change “P0_0 Output” to as shown in Figure 6-13.

Figure 6-13 P0_0 Setting



P0_0 Output Setting

Specify target
Specify port: P1
Specify channel: P1_0

Output data
Value: 1

P0_0 Output Setting

Specify target
Specify port: P1
Specify channel: P1_0

Output data
Value: 0

7. Sample Code

Sample code can be downloaded from the Renesas Electronics website.

Furthermore, if you regenerate the code with Smart Configurator, please modify the R_Config_SMS_Start function as indicated in the red frame below.

```
void R_Config_SMS_Start(uint16_t val_adcr_th)
{
    /* Set the sms data from arguments */
    SMSG2 = val_adcr_th;
    /* Initialize SMS data */
    SMSG3 = 65509U;
    /* Disable related interrupts */
    ITLMK = 1U;
    ADMK = 1U;
    /* Start sms */
    //SMSEIF = 0U; /* clear INTSMSE interrupt flag */
    //SMSEMK = 0U; /* enable INTSMSE interrupt */
    SMSEMK = 1U; /* disable INTSMSE interrupt */
    SMSEIF = 1U; /* set INTSMSE interrupt flag */
    g_sms_wakeup_flag = 0U;
    ITLS0 = _00_INTITL_CLEAR;
    SMSSTART = 1U;
}
```

8. Reference

RL78/G23 User’s Manual: Hardware (R01UH0896E)

RL78 Family User’s Manual: Software (R01US0015E)

SMS assembler User’s Manual (R20UT4792J)

RL78 Smart Configurator User's Guide: CS+ (R20AN0580E)

RL78 Smart Configurator User's Guide: e² studio (R20AN0579E)

RL78 Smart Configurator User's Guide: IAREW (R20AN0581E)

(The latest version can be downloaded from the Renesas Electronics website.)

Technical Update / Technical News

(The latest version can be downloaded from the Renesas Electronics website.)

Revision History

Rev.	Date	Description	
		Page	Summary
1.00	Apr.13.21	-	First edition
1.10	Jun.1.21	6	Updated tool version Table 2-1 Operation Confirmation Conditions Integrated development environment (CS+) : E8.05.00f -> V8.05.00 C compiler (CS+) : V1.09.00 -> V1.10 Integrated development environment (e^2 studio) : 2021-01 (21.01.0) -> 2021-04 (21.4.0) C compiler (e^2 studio) : V1.09.00 -> V1.10 Integrated development environment (IAR) : V4.20.1 -> V4.21.1 Smart Configurator : V.1.0.0 -> V.1.0.1 Board support package (r_bsp) : V.1.0.0 -> V.1.10
		6, 8 18, 19	Changed due to COM port support Table 2-1 Operation Confirmation Conditions Emulator: E2 Emulator Lite -> CS＋, e^2 studio: COM port IAR: E2 Emulator Lite Figure 4-1 Hardware Configuration Added P11/TOOLRxD and P12/TOOLTxD Table 6-1 Smart Configurator Settings Note 1 added
		11, 13	Changed due to IAR version sample code update Table 5-1 Folder configuration Added the note about reference documents in folder configuration 5.7 Function Specifications [Function name] main, Header e^2 studio, CS+ : r_smc_entry.h IAR : ior7f100g.h, ior7f100g_ext.h, r_cg_macrodriver.h, Config_SMS.h, Config_ITL000_ITL001.h -> r_smc_entry.h
		18, 19	Changed clock abbreviation Table 6-1 Parameters of Smart Configurator Clocks : f_{SXL} -> f_{SXP} Table 6-2 Parameters of Smart Configurator Operation clock : f_{SXL} -> f_{SXP}
		27	Added of RL78 Smart Configurator User's Guide 8. Reference RL78 Smart Configurator User's Guide: CS+ (R20AN0580E) RL78 Smart Configurator User's Guide: e^2 studio (R20AN0579E) RL78 Smart Configurator User's Guide: IAREW (R20AN0581E)
1.20	Jan.9.24	-	Changed the flowchart for SMS processing

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. Precaution against Electrostatic Discharge (ESD)

   A strong electrical field, when exposed to a CMOS device, can cause destruction of the gate oxide and ultimately degrade the device operation. Steps must be taken to stop the generation of static electricity as much as possible, and quickly dissipate it when it occurs. Environmental control must be adequate. When it is dry, a humidifier should be used. This is recommended to avoid using insulators that can 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 must be grounded. The operator must also be grounded using a wrist strap. Semiconductor devices must not be touched with bare hands. Similar precautions must be taken for printed circuit boards with mounted semiconductor devices.
2. Processing at power-on

   The state of the product is undefined at the time when power is supplied. The states of internal circuits in the LSI are indeterminate and the states of register settings and pins are undefined at the time 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 time 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 time when power is supplied until the power reaches the level at which resetting is specified.
3. Input of signal during power-off state

   Do not input signals or an I/O pull-up power supply while the device is powered off. 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. Follow the guideline for input signal during power-off state as described in your product documentation.
4. Handling of unused pins

   Handle unused pins in accordance with the directions given under handling of unused pins in the manual. 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 the 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.
5. Clock signals

   After applying a reset, only release the reset line after the operating clock signal becomes stable. When switching the clock signal during program execution, wait until the target clock signal is stabilized. 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. Additionally, 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.
6. 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 V_{IL} (Max.) and V_{IH} (Min.) due to noise, for example, 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 V_{IL} (Max.) and V_{IH} (Min.).
7. Prohibition of access to reserved addresses

   Access to reserved addresses is prohibited. The reserved addresses are provided for possible future expansion of functions. Do not access these addresses as the correct operation of the LSI is not guaranteed.
8. Differences between products

   Before changing from one product to another, for example to a product with a different part number, confirm that the change will not lead to problems. The characteristics of a microprocessing unit or microcontroller unit products in the same group but having a different part number might differ in terms of 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.

Notice

1. Descriptions of circuits, software and other related information in this document are provided only to illustrate the operation of semiconductor products and application examples. You are fully responsible for the incorporation or any other use of the circuits, software, and information in the design of your product or system. Renesas Electronics disclaims any and all liability for any losses and damages incurred by you or third parties arising from the use of these circuits, software, or information.
2. Renesas Electronics hereby expressly disclaims any warranties against and liability for infringement or any other claims involving patents, copyrights, or other intellectual property rights of third parties, by or arising from the use of Renesas Electronics products or technical information described in this document, including but not limited to, the product data, drawings, charts, programs, algorithms, and application examples.
3. No license, express, implied or otherwise, is granted hereby under any patents, copyrights or other intellectual property rights of Renesas Electronics or others.
4. You shall be responsible for determining what licenses are required from any third parties, and obtaining such licenses for the lawful import, export, manufacture, sales, utilization, distribution or other disposal of any products incorporating Renesas Electronics products, if required.
5. You shall not alter, modify, copy, or reverse engineer any Renesas Electronics product, whether in whole or in part. Renesas Electronics disclaims any and all liability for any losses or damages incurred by you or third parties arising from such alteration, modification, copying or reverse engineering.
6. Renesas Electronics products are classified according to the following two quality grades: “Standard” and “High Quality”. The intended applications for each Renesas Electronics product depends on the product’s quality grade, as indicated below.

   "Standard": Computers; office equipment; communications equipment; test and measurement equipment; audio and visual equipment; home electronic appliances; machine tools; personal electronic equipment; industrial robots; etc.

   "High Quality": Transportation equipment (automobiles, trains, ships, etc.); traffic control (traffic lights); large-scale communication equipment; key financial terminal systems; safety control equipment; etc.

   Unless expressly designated as a high reliability product or a product for harsh environments in a Renesas Electronics data sheet or other Renesas Electronics document, Renesas Electronics products are not intended or authorized for use in products or systems that may pose a direct threat to human life or bodily injury (artificial life support devices or systems; surgical implantations; etc.), or may cause serious property damage (space system; undersea repeaters; nuclear power control systems; aircraft control systems; key plant systems; military equipment; etc.). Renesas Electronics disclaims any and all liability for any damages or losses incurred by you or any third parties arising from the use of any Renesas Electronics product that is inconsistent with any Renesas Electronics data sheet, user’s manual or other Renesas Electronics document.
7. No semiconductor product is absolutely secure. Notwithstanding any security measures or features that may be implemented in Renesas Electronics hardware or software products, Renesas Electronics shall have absolutely no liability arising out of any vulnerability or security breach, including but not limited to any unauthorized access to or use of a Renesas Electronics product or a system that uses a Renesas Electronics product. RENESAS ELECTRONICS DOES NOT WARRANT OR GUARANTEE THAT RENESAS ELECTRONICS PRODUCTS, OR ANY SYSTEMS CREATED USING RENESAS ELECTRONICS PRODUCTS WILL BE INVULNERABLE OR FREE FROM CORRUPTION, ATTACK, VIRUSES, INTERFERENCE, HACKING, DATA LOSS OR THEFT, OR OTHER SECURITY INTRUSION (“Vulnerability Issues”). RENESAS ELECTRONICS DISCLAIMS ANY AND ALL RESPONSIBILITY OR LIABILITY ARISING FROM OR RELATED TO ANY VULNERABILITY ISSUES. FURTHERMORE, TO THE EXTENT PERMITTED BY APPLICABLE LAW, RENESAS ELECTRONICS DISCLAIMS ANY AND ALL WARRANTIES, EXPRESS OR IMPLIED, WITH RESPECT TO THIS DOCUMENT AND ANY RELATED OR ACCOMPANYING SOFTWARE OR HARDWARE, INCLUDING BUT NOT LIMITED TO THE IMPLIED WARRANTIES OF MERCHANTABILITY, OR FITNESS FOR A PARTICULAR PURPOSE.
8. When using Renesas Electronics products, refer to the latest product information (data sheets, user’s manuals, application notes, “General Notes for Handling and Using Semiconductor Devices” in the reliability handbook, etc.), and ensure that usage conditions are within the ranges specified by Renesas Electronics with respect to maximum ratings, operating power supply voltage range, heat dissipation characteristics, installation, etc. Renesas Electronics disclaims any and all liability for any malfunctions, failure or accident arising out of the use of Renesas Electronics products outside of such specified ranges.
9. Although Renesas Electronics endeavors to improve the quality and reliability of Renesas Electronics products, semiconductor products have specific characteristics, such as the occurrence of failure at a certain rate and malfunctions under certain use conditions. Unless designated as a high reliability product or a product for harsh environments in a Renesas Electronics data sheet or other Renesas Electronics document, Renesas Electronics products are not subject to radiation resistance design. You are responsible for implementing safety measures to guard against the possibility of bodily injury, injury or damage caused by fire, and/or danger to the public in the event of a failure or malfunction of Renesas Electronics products, such as safety design for hardware and software, including but not limited to redundancy, fire control and malfunction prevention, appropriate treatment for aging degradation or any other appropriate measures. Because the evaluation of microcomputer software alone is very difficult and impractical, you are responsible for evaluating the safety of the final products or systems manufactured by you.
10. Please contact a Renesas Electronics sales office for details as to environmental matters such as the environmental compatibility of each Renesas Electronics product. You are responsible for carefully and sufficiently investigating applicable laws and regulations that regulate the inclusion or use of controlled substances, including without limitation, the EU RoHS Directive, and using Renesas Electronics products in compliance with all these applicable laws and regulations. Renesas Electronics disclaims any and all liability for damages or losses occurring as a result of your noncompliance with applicable laws and regulations.
11. Renesas Electronics products and technologies shall not be used for or incorporated into any products or systems whose manufacture, use, or sale is prohibited under any applicable domestic or foreign laws or regulations. You shall comply with any applicable export control laws and regulations promulgated and administered by the governments of any countries asserting jurisdiction over the parties or transactions.
12. It is the responsibility of the buyer or distributor of Renesas Electronics products, or any other party who distributes, disposes of, or otherwise sells or transfers the product to a third party, to notify such third party in advance of the contents and conditions set forth in this document.
13. This document shall not be reprinted, reproduced or duplicated in any form, in whole or in part, without prior written consent of Renesas Electronics.
14. Please contact a Renesas Electronics sales office if you have any questions regarding the information contained in this document or Renesas Electronics products.

(Note1) “Renesas Electronics” as used in this document means Renesas Electronics Corporation and also includes its directly or indirectly controlled subsidiaries.

(Note2) “Renesas Electronics product(s)” means any product developed or manufactured by or for Renesas Electronics.

(Rev.5.0-1 October 2020)

Corporate Headquarters

TOYOSU FORESIA, 3-2-24 Toyosu,
Koto-ku, Tokyo 135-0061, Japan
www.renesas.com

Trademarks

Renesas and the Renesas logo are trademarks of Renesas Electronics Corporation. All trademarks and registered trademarks are the property of their respective owners.

Contact information

For further information on a product, technology, the most up-to-date version of a document, or your nearest sales office, please visit: www.renesas.com/contact/.