RX63N Group, RX631 Group

True/False Determination of RTC Time Information on Cold Start

R01AN1713EJ0101
Rev. 1.01
Nov. 6, 2015

Abstract

This document describes the method to determine whether power is supplied from the VBATT while the voltage on the VCC pin is dropped when using the realtime clock (RTC) and the battery backup function together. The RTC clock settings and registers are used to determine whether the RTC continues operation or the RTC is initialized.

Products

- RX63N Group 177-pin and 176-pin packages with a ROM size between 768 KB and 2 MB
- RX63N Group 145-pin and 144-pin packages with a ROM size between 768 KB and 2 MB
- RX63N Group 100-pin package with a ROM size between 768 KB and 2 MB
- RX631 Group 177-pin and 176-pin packages with a ROM size between 256 KB and 2 MB
- RX631 Group 145-pin and 144-pin packages with a ROM size between 256 KB and 2 MB
- RX631 Group 100-pin package with a ROM size between 256 KB and 2 MB
- RX631 Group 64-pin package with a ROM size between 256 KB and 512 KB

When using this application note with other Renesas MCUs, careful evaluation is recommended after making modifications to comply with the alternate MCU.

Contents

1. Specifications

After a reset, the software determines whether to continue the RTC count operation or initialize the RTC, then performs the operation according to the determination.

The battery backup function enables the RTC and the sub-clock oscillator to continue operating with power supplied by the VBATT even when the voltage on the VCC pin is dropped. The RTC must be initialized only when the voltage on the VCC pin continues being dropped and power supply from the VBATT is stopped. But after both of them hung a VCC pin and a VBATT pin on 0V once when the voltage of VBATT was lower than a guarantee area after changing to VBATT, please do a power-on reset.

The VBATT circuit is assumed that it takes 15 minutes for charging the connected capacitor at a maximum, and the fully charged capacitor can supply power for at least 10 days.

In this application note, the following are verified for determination:

- If the VCC voltage is recovered and the reset is released within the period the VBATT can supply power.
- If the charging time for the VBATT is enough.
- If the values stored in the RTC registers are correct after a reset.

The RTC time information displayed on the Debug LCD of the RSK is updated every second while the VCC voltage is retained.

Also the RTC time information (backup) stored in the E2 DataFlash is updated in the following timings:

- After a reset and 1 second elapses.
- After 15 minutes, which is time required to charge the capacitor connected to the VBATT.
- When the hour counter (RHRCNT) for the RTC time information is updated.

Table 1.1 lists the Peripheral Functions and Their Applications and Figure 1.1 shows the Operation Overview.

Table 1.1 Peripheral Functions and Their Applications

Peripheral Function	Application
RTCa	Clock counter
VBATT	Power supply to the RTC and the sub-clock oscillator.
E2 DataFlash	Backup for the RTC time information

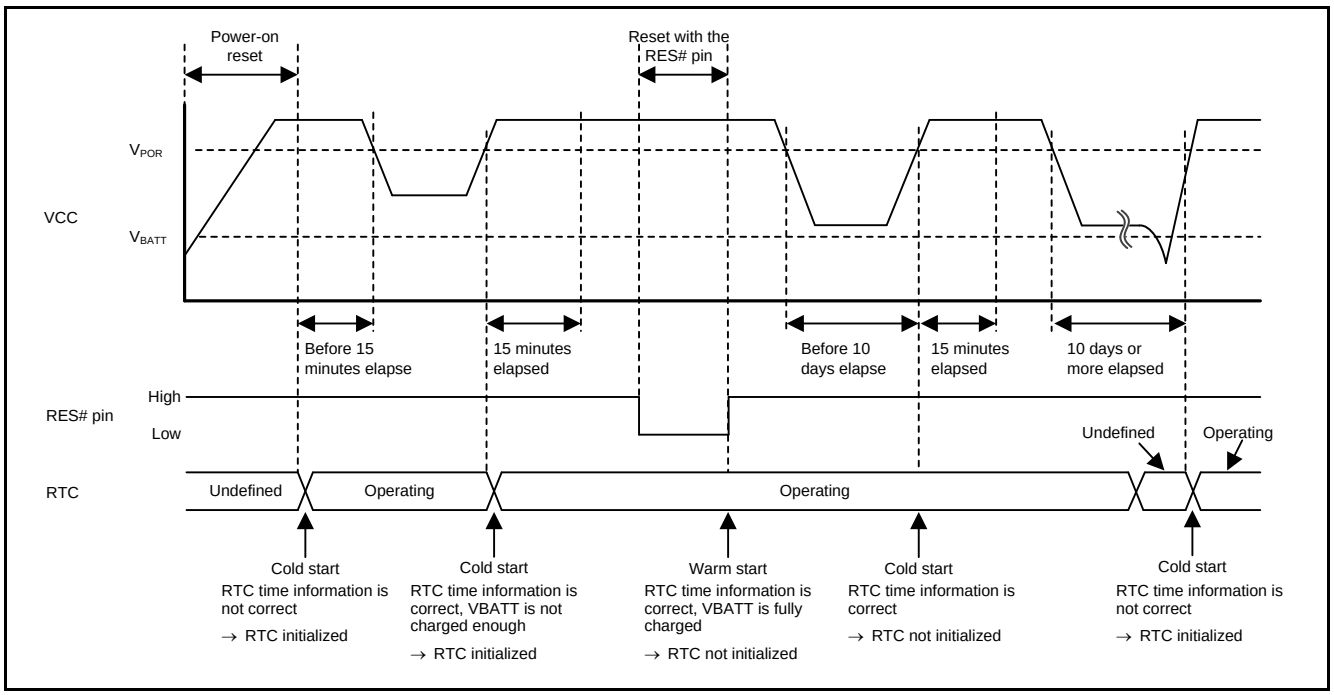


Figure 1.1 Operation Overview

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	R5F563NBDDFC (RX63N Group)
Operating frequencies	- Main clock: 12 MHz - Sub-clock: 32.768 kHz - PLL: 192 MHz (main clock divided by 1 and multiplied by 16) - LOCO: 125 kHz - System clock (ICLK): 96 MHz (PLL divided by 2) - Peripheral module clock B (PCLKB): 48 MHz (PLL divided by 4)
Operating voltage	3.3 V, VBATT is V_{BATT} or greater
Integrated development environment	Renesas Electronics Corporation High-performance Embedded Workshop Version 4.09.01
C compiler	Renesas Electronics Corporation C/C++ Compiler Package for RX Family V.1.02 Release 01
	Compile options -cpu=rx600 -output=obj="$(CONFIGDIR)\$(FILELEAF).obj" -debug -nologo (The default setting is used in the integrated development environment.)
iodefine.h version	Version 1.8
Endian	Little endian
Operating mode	Single-chip mode
Processor mode	Supervisor mode
Sample code version	Version 1.01
Board used	Renesas Starter Kit+ for RX63N (product part no.: R0K50563NC000BE)

3. Reference Application Notes

For additional information associated with this document, refer to the following application notes.

- RX63N Group, RX631 Group Initial Setting Rev. 1.10 (R01AN1245EJ)
- RX600 & RX200 Series Simple Flash API for RX Rev.2.40 (R01AN0544EU)
- RX Family Using the Simple Flash API for RX without the r_bsp Module Rev1.00 (R01AN1890EU)
- Renesas Starter Kit Sample Code for Hi-performance Embedded Workshop Rev.1.00 (R01AN1395EG)
- RX Family Coding Example of Wait Processing by Software Rev. 1.00 (R01AN1852EJ0100)

The initial setting functions, Debug LCD output functions, simple flash API functions, coding example of wait processing by software functions in the reference application notes are used in the sample code accompanying this application note. The revision numbers of the reference application notes are as of when these application notes were made. However the latest version is always recommended. Visit the Renesas Electronics Corporation website to check and download the latest version.

4. Hardware

4.1 Hardware Configuration

Figure 4.1 shows the Connection Example.

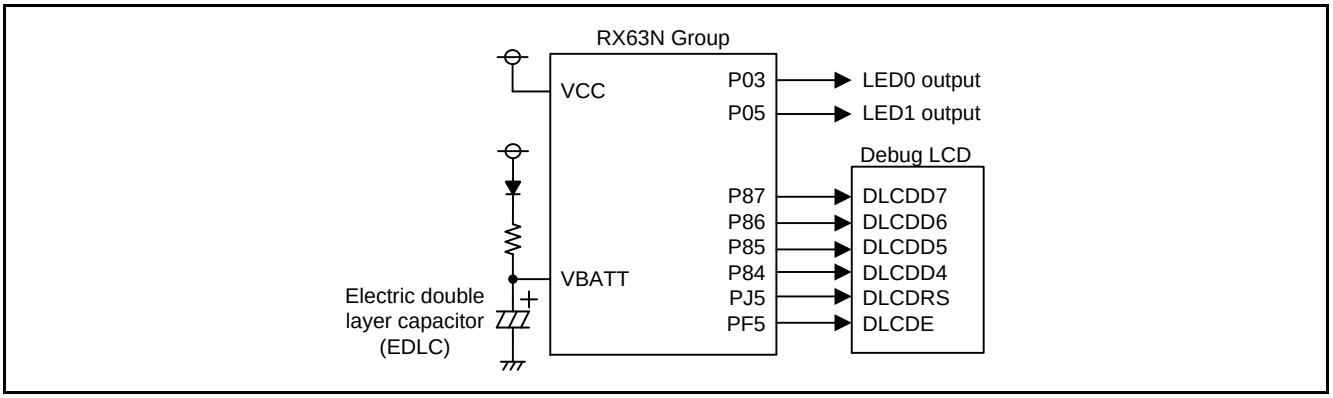


Figure 4.1 Connection Example

In this application note, the maximum charging time is 15 minutes and the shortest discharging time (time to retain power) is 10 days.

Note that the charging and discharging time vary depending on the circuit structure of the VBATT pins. Make sure to use the charge and discharge time carefully evaluated and confirmed in the user system.

4.2 Pins Used

Table 4.1 lists the Pins Used and Their Functions.

Table 4.1 Pins Used and Their Functions

Pin Name	I/O	Function
P03	Output	LED0 output to indicate completion of the RTC initialization
P05	Output	LED1 output to indicate E2 DataFlash backup error
P87	Output	“Debug LCD Data bit 7” output
P86	Output	“Debug LCD Data bit 6” output
P85	Output	“Debug LCD Data bit 5” output
P84	Output	“Debug LCD Data bit 4” output
PJ5	Output	“Debug LCD Enable” output
PF5	Output	“Debug LCD Register select” output

5. Software

5.1 Operation Overview

In this application note, the RTC time information is validated using the items below as criteria. According to the validation, the RTC is determined to continue operation or be initialized. Figure 5.1 shows the Validation Procedure.

- RTC count source: Sub-clock operating
- RTC control register: Same value as the initial value
- RTC automatic adjustment setting: Same value as the initial value when this setting is enabled
- RTC time information: Values that are correct as the time information and within 10 days from the backup time information in the E2 DataFlash.
- RTC alarm information: Values that are correct as the time information and within 10 days from the backup time information in the E2 DataFlash.
- Charge detection in the VBATT circuit: Cold start and 15 minutes elapsed from the previous RTC initialization.

RTC initial setting values are as follows:

- Hours mode: 12-hour mode
- Initial time setting: 00:00:00, Tuesday, January 1, 2013
- Alarm setting: Disabled
- RTCOUT output: Output disabled
- Automatic adjustment: Not used
- Count source: Sub-clock for standard clock loads
- Time capture: Not used
- Interrupts: The periodic interrupt (PRD) is generated every second.
  The alarm interrupt and the carry interrupt are disabled.

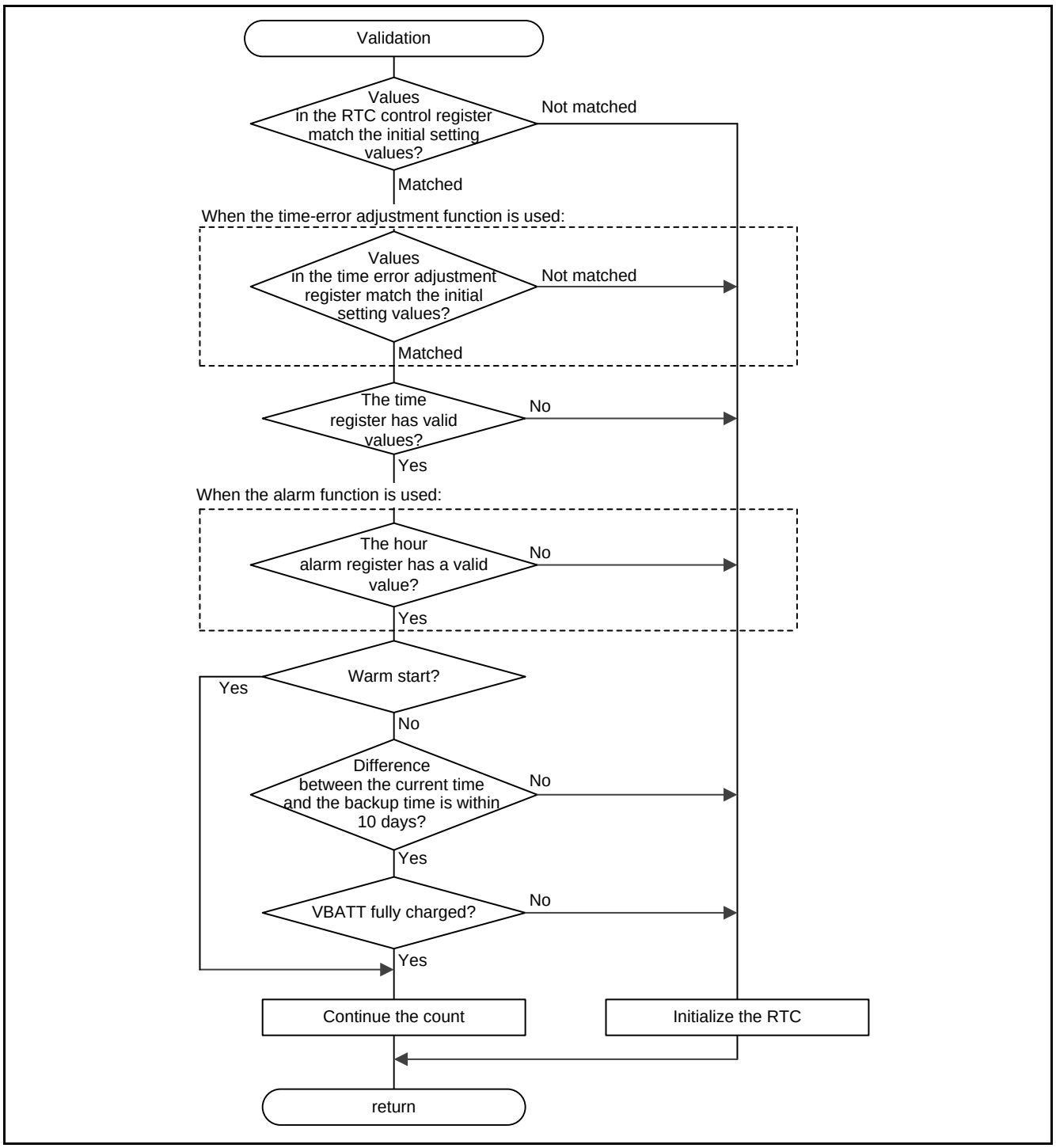


Figure 5.1 Validation Procedure

5.2 Battery Backup Function

With the connected circuit in this application note, an output voltage of V_{BATT} or greater on the VBATT pin can be retained for at least 10 days, and the capacitor can be fully charged within 15 minutes.

Note that the charging and discharging time vary depending on the connected circuit. Make sure to use the charging and discharging time carefully evaluated and confirmed in the user system.

5.3 Programming/Erasing the E2 DataFlash

In this application note, addresses 0010 0000h to 0010 003Fh (64 bytes) of the E2 DataFlash are used. 10 bytes of the RTC time information is overwritten and stored in addresses 0010 0000h to 0010 0009h and 0010 0020h to 0010 0029h alternately. Data is not retained for the other areas in the 64 bytes of the E2 DataFlash.

For details on the other Simple Flash API used for programming/erasing, refer to the RX600 & RX200 Series Simple Flash API for RX.

5.3.1 Erasing the E2 DataFlash

To erase the E2 DataFlash, the R_FlashEraseRange function of the Simple Flash API for RX is used. The erase error can be determined by the return value from the R_FlashEraseRang function or the register value in the FlashError function.

5.3.2 Programming the E2 DataFlash

To program the E2 DataFlash, the R_FlashWrite function of the Simple Flash API for RX is used. The program error can be determined by the return value from the R_FlashWrite function or the register value in the FlashError function.

5.3.3 Changes in the Simple Flash API

Programming and erasing of the E2 DataFlash is performed using the program in the flash memory (ROM). The main loop can be executed during programming or erasing.

In this application note, r_flash_api_rx_config.h is changed in the Simple Flash API.

Table 5.1 shows Changes in r_flash_api_rx_config.h.

Table 5.1 Changes in r_flash_api_rx_config.h

Changed Item	Description	Code After the Change
Settings in the Simple Flash API	No programming performed in the ROM	// #define FLASH_API_RX_CFG_ENABLE_ROM_PROGRAMMING
	Data to be programmed is stored in the RAM buffer.	#define FLASH_API_RX_CFG_FLASH_TO_FLASH
	E2 DataFlash operation is executed in background.	#define FLASH_API_RX_CFG_DATA_FLASH_BGO
	Flash API is not transferred to the RAM	// #define FLASH_API_RX_CFG_COPY_CODE_BY_API

Note: • When VCC power is cut off or reset during programming or erasing the E2 DataFlash, the RTC time information becomes incorrect. Thus the RTC is initialized after a reset.

5.4 File Composition

Table 5.2 and Table 5.3 list the files used in this application notes. Table 5.4 and Table 5.5 list the functions and setting values in the reference application notes. Table 5.6 lists files from the simple flash API module.

Files generated by the integrated development environment are not included in this table.

Table 5.2 Files Used in the Sample Code

File Name	Outline
main.c	Main processing, port setting
r_init_clock_an1713.c	Clock initialization
r_init_clock_an1713.h	Header file for r_init_clock_an1713.c
rtc_func.c	RTC register determination, RTC initialization, and periodic interrupt handling
rtc_func.h	Header file for rtc_func.c
flash_write.c	Processing for backing up the RTC time information to the E2 DataFlash

Table 5.3 Standard Include Files

File Name	Outline
stdbool.h	Defines macros regarding the boolean type and value.
stdint.h	Defines macros declaring the integer type with the specified width.
machine.h	Defines types of built-in function for the RX Family.

Table 5.4 Functions and Setting Values in the RX63N Group, RX631 Group Initial Setting Application Note

File Name	Function	Description
r_init_stop_module.c	R_INIT_StopModule()	—
r_init_stop_module.h	—	Module-stop state is canceled for DMAC/DTC, EXDMAC, RAM0, and RAM1.
r_init_non_existent_port.c	R_INIT_NonExistentPort()	—
r_init_non_existent_port.h	—	176 pin package is specified.
r_delay.c	R_DELAY_Us()	The waiting time is set.
r_delay.h	—	—

Table 5.5 Functions and Setting Values in the Sample Code Accompanying the Renesas Starter Kit for RX63N High-performance Embedded Workshop

File Name	Function	Description
lcd.c	- Init_LCD() - Display_LCD()	—
lcd.h	—	—
rskrx63ndef.h	—	—

Table 5.6 Simple Flash API Modules (RX600 & RX200 Series Simple Flash API for RX Application Note)

Module Name	Outline
r_flash_api_rx	RX Simple Flash API programs for RX600 and RX200 Series
u_bsp	Board support package for users (1)

Note:
1. The module is modified following the application note “Using the Simple Flash API for RX without the r_bsp Module”.

5.5 Option-Setting Memory

Table 5.7 lists the Option-Setting Memory Configured in the Sample Code. When necessary, set a value suited to the user system.

Table 5.7 Option-Setting Memory Configured in the Sample Code

Symbol	Address	Setting Value	Contents
OFS0	FFFF FF8Fh to FFFF FF8Ch	FFFF FFFFh	The IWDT is stopped after a reset. The WDT is stopped after a reset.
OFS1	FFFF FF8Bh to FFFF FF88h	FFFF FFFBh	The voltage monitor 0 reset is enabled after a reset. HOCO oscillation is disabled after a reset.
MDES	FFFF FF83h to FFFF FF80h	FFFF FFFFh	Little endian

5.6 Constants

Table 5.8 lists the Constants Used in the Sample Code.

Table 5.8 Constants Used in the Sample Code

Constant Name	Setting Value	Contents
RTC_INIT_RES	1	Reset with the RTC initialization required
RTC_RUN_RES	0	Reset with the RTC initialization not required
RTC_BACKUP_INIT	2	Backup of the RTC time information is requested. (Initialization source, backup request after 1 second)
RTC_BACKUP_ACK	1	Backup of the RTC time information is requested. (backup request after 15 minutes and every hour)
RTC_BACKUP_NACK	0	Backup of the RTC time information is not requested.
VBATT_FULL_CHARGE	1	VBATT circuit is fully charged.
VBATT_EMPTY	0	VBATT circuit is not charged enough.
MASK_DATA_SEC	0x01	Execution flag of second data verification, data for verifying the flag
MASK_DATA_MIN	0x02	Execution flag of minute data verification, data for verifying the flag
MASK_DATA_HR	0x04	Execution flag of hour data verification, data for verifying the flag
MASK_DATA_WK	0x08	Execution flag of day-of-week data verification, data for verifying the flag
MASK_DATA_DAY	0x10	Execution flag of day data verification, data for verifying the flag
MASK_DATA_MON	0x20	Execution flag of month data verification, data for verifying the flag
MASK_DATA_YR	0x40	Execution flag of year data verification, data for verifying the flag
ALL_DATA_CHECK	0x7F	Execution flag of all data verification
ADDRESS_BLOCK_DB0	0x100000	Start address of the DB0 block
ADDRESS_BLOCK_DB1	0x100020	Start address of the DB1 block
BUFF_SIZE	10	Sizes for storing the time information and programming
WRITE_BUSY	2	Programming mode (programming)
ERASE_BUSY	1	Programming mode (erasing)
WRITE_READY	0	Programming mode (preparing)
SELECT_SUB	PATTERN_D	Selection of the sub-clock setting pattern

5.7 Structure/Union List

Figure 5.2 shows the Structure/Union Used in the Sample Code.

```
/* **** Time data **** */
typedef struct
{
    uint8_t    second;          /* Second */
    uint8_t    minute;          /* Minute */
    uint8_t    hour;            /* Hour */
    uint8_t    dayweek;         /* Day of the week */
    uint8_t    day;             /* Day */
    uint8_t    month;           /* Month */
    uint16_t    year;           /* Year */
    uint8_t    charge_VBATT;    /* VBATT charge info */
    uint8_t    write_cnt;       /* Backup counter */
} time_bcd_t;
```

Figure 5.2 Structure/Union Used in the Sample Code

5.8 Variables

Table 5.9 lists the Global Variables, Table 5.10 lists the static Variables, and Table 5.11 lists the const Variable.

Table 5.9 Global Variables

Type	Variable Name	Contents	Function Used
uint8_t	init_RTC	RTC initialization request flag	- main - enable_RTC_an1713 - CGC_subclk_as_RTC_an1713
uint32_t	time_data	Buffer to obtain time information	- check_RTC - flash_check - check_RTC_flash - check_RTC_after10days - enable_RTC_an1713 - Excep_RTC_PRD - time_backup
uint8_t	flash_write	Backup request flag for the RTC time information	- main - check_RTC - enable_RTC_an1713 - Excep_RTC_PRD
int8_t	write_counter	Backup counter	- main - check_RTC - flash_check - check_RTC_flash - check_RTC_after10days - Excep_RTC_PRD
uint8_t	write_mode	Programming mode	- Excep_RTC_PRD - flash_check
uint8_t	lcd_buffer[9]	Buffer for Debug LCD	- main - uint32_ToBCDString
uint8_t	flash_is_ready	Flash processing end flag	- time_backup - check_RTC - Excep_RTC_PRD

Table 5.10 static Variables

Type	Variable Name	Contents	Function Used
static uint8_t	charge_min_count	Charge time (minutes) counter for the VBATT circuit	- check_RTC - Excep_RTC_PRD
static uint8_t	check_time_data	Time information check required/time information check not required	- check_RTC - Set_register_enable - flash_check - check_RTC_clock
static uint8_t	prog_buff	Time information for backup	- time_backup

Table 5.11 const Variable

Type	Variable Name	Contents	Function Used
const static uint32_t	backup_address[2] = {ADDRESS_BLOCK_DB0, ADDRESS_BLOCK_DB1};	- Start address of the DB0 block - Start address of the DB1 block	- check_RTC - flash_check - check_RTC_flash - check_RTC_after10days

5.9 Functions

Table 5.12 lists the Functions.

Table 5.12 Functions

Function Name	Outline	File the Function is in
main	Main processing	main.c
port_init	Port output setting	main.c
check_RTC	Determination of RTC initialization requirement	rtc_func.c
Set_register_enable	Setting the validity of the alarm time information	rtc_func.c
flash_check	Determination of the backup time information for use	rtc_func.c
check_RTC_clock	Verification of the RTC time information value	rtc_func.c
check_RTC_flash	Comparison of the RTC time information with the backup time information	rtc_func.c
check_RTC_after10days	Verifying the range of the RTC time information	rtc_func.c
rtc_time_read	Reading the RTC time information	rtc_func.c
R_INIT_Clock_an1713	Clock initialization	r_init_clock_an1713.c
CGC_subclk_as_RTC_ an1713	Sub-clock setting (sub-clock is used for the RTC count source and not used for the system clock)	r_init_clock_an1713.c
enable_RTC_an1713	RTC initialization	rtc_func.c
Excep_RTC_PRD	RTC periodic interrupt handling	rtc_func.c
time_backup	Backup processing for the RTC time information	flash_write.c

5.10 Function Specifications

The following tables list the sample code function specifications.

main	
Outline	Main processing
Header	None
Declaration	void main(void)
Description	After initialization, if a backup request of the RTC time information is generated, the backup processing function is called.
Arguments	None
Return Value	None
Remarks	In the main loop processing, do not rewrite registers that are write protected while operating in E2 DataFlash P/E mode. When additional processing is added to the main loop, and if operating in E2 DataFlash P/E mode, any processing including the additional processing is executed in E2 DataFlash P/E mode.

port_init	
Outline	Port initialization
Header	None
Declaration	static void port_init(void)
Description	Initializes the ports.
Arguments	None
Return Value	None

check_RTC	
Outline	Determination of RTC initialization requirement
Header	rtc_func.h
Declaration	uint8_t check_RTC(void)
Description	Compares the RTC register value with the RTC initial setting value specified by the enable_RTC_an1713 function, performs processing and verification regarding the backup time information, determines cold/warm start, verifies the charge status of the VBATT circuit at a backup timing, and then determines whether to continue the RTC operation or initialize the RTC.
Arguments	None
Return Value	RTC_INIT_RES: RTC time information is invalid. RTC_RUN_RES: RTC time information is valid.

flash_check	
Outline	Determination of the backup time information for use
Header	rtc_func.h
Declaration	static uint8_t flash_check(void)
Description	Verifies the backup counters DB0 and DB1, and determines data to be used for the subsequent comparison processing. If the backup counter is invalid, reads the backup time information to determine the data to be used for the subsequent comparison processing.
Arguments	None
Return Value	RTC_INIT_RES: RTC time information is invalid. RTC_RUN_RES: RTC time information is valid.

Set_register_enable	
Outline	Setting the validity of the alarm time information
Header	rtc_func.h
Declaration	static void Set_register_enable(void)
Description	Only when the ENB bit of the alarm time information is set to 1, the ENB bit of the RTC alarm register is read to verify the register value.
Arguments	None
Return Value	None

check_RTC_clock	
Outline	Verification of the RTC time information value
Header	rtc_func.h
Declaration	static unsigned char check_RTC_clock(void)
Description	Verifies that the register value of the RTC time information is correct.
Arguments	None
Return Value	RTC_INIT_RES: RTC time information is invalid. RTC_RUN_RES: RTC time information is valid.

check_RTC_flash	
Outline	Comparison of the RTC time information with the backup time information
Header	rtc_func.h
Declaration	static unsigned char check_RTC_flash(void)
Description	Compares the RTC time information with the backup time information and verifies the RTC time information is newer than the backup time information.
Arguments	None
Return Value	RTC_INIT_RES: RTC time information is invalid. RTC_RUN_RES: RTC time information is valid.

check_RTC_after10days	
Outline	Verifying the range of the RTC time information
Header	rtc_func.h
Declaration	static unsigned char check_RTC_after10days(void)
Description	Compares the RTC time information with the backup time information and verifies the RTC time information is within normal range (10 days after the backup time information).
Arguments	None
Return Value	RTC_INIT_RES: RTC time information is invalid. RTC_RUN_RES: RTC time information is valid.

rtc_time_read	
Outline	Reading the RTC time information
Header	None
Declaration	static void rtc_time_read(void)
Description	Reads the RTC time information and stores it in the RAM area for the RTC time information.
Arguments	None
Return Value	None

R_INIT_Clock_an1713	
Outline	Clock initialization
Header	r_init_clock.h
Declaration	void R_INIT_Clock_an1713(void)
Description	Initializes the clock.
Arguments	None
Return Value	None
Remarks	The sample code selects processing which uses PLL as the system clock and the sub-clock as the RTC count source. Refer to the RX63N Group, RX631 Group Initial Setting Rev. 1.10 application note for details on this function.

CGC_subclk_as_RTC_an1713	
Outline	Sub-clock setting
Header	r_init_clock.h
Declaration	void CGC_subclk_as_RTC_an1713(void)
Description	Configures the setting when the sub-clock is used as the RTC count source and not used as the system clock.
Arguments	None
Return Value	None
Remarks	Refer to pattern D in the RX63N Group, RX631 Group Initial Setting Rev. 1.10 application note for details on this function.

enable_RTC_an1713	
Outline	RTC Initialization
Header	None
Declaration	static void enable_RTC_an1713(void)
Description	Performs the initialization when the RTC is used (setting for clock provision and RTC software reset).
Arguments	None
Return Value	None

Excep_RTC_PRD	
Outline	RTC periodic interrupt handling
Header	rtc_func.h
Declaration	static void Excep_RTC_PRD(void)
Description	Reads the RTC time information in this interrupt handling and displays it on the Debug LCD. After a reset and time required to charge the VBATT circuit elapsed, updates the VBATT charge flag and the backup counter and sets the backup request. Also updates the backup counter every hour and sets the backup request.
Arguments	None
Return Value	None

time_backup	
Outline	Backup processing for the RTC time information
Header	None
Declaration	void time_backup(uint32_t time_data, uint8_t write_block)
Description	Stores the RTC time information in the E2 DataFlash. The RTC time information is stored in 10-byte area from the start address of block 0 or block 1 in the E2 DataFlash. The block to store data is determined with bit 0 of the backup counter.
Arguments	uint32_t time_data: RTC time information data to be backed up uint8_t write_block Data to specify the area used for backup
Return Value	None
Remarks	The program placed in the ROM area is executed even when the CPU is programming/erasing E2 DataFlash.

5.11 Flowcharts

5.11.1 Main Processing

Figure 5.3 shows the Main Processing.

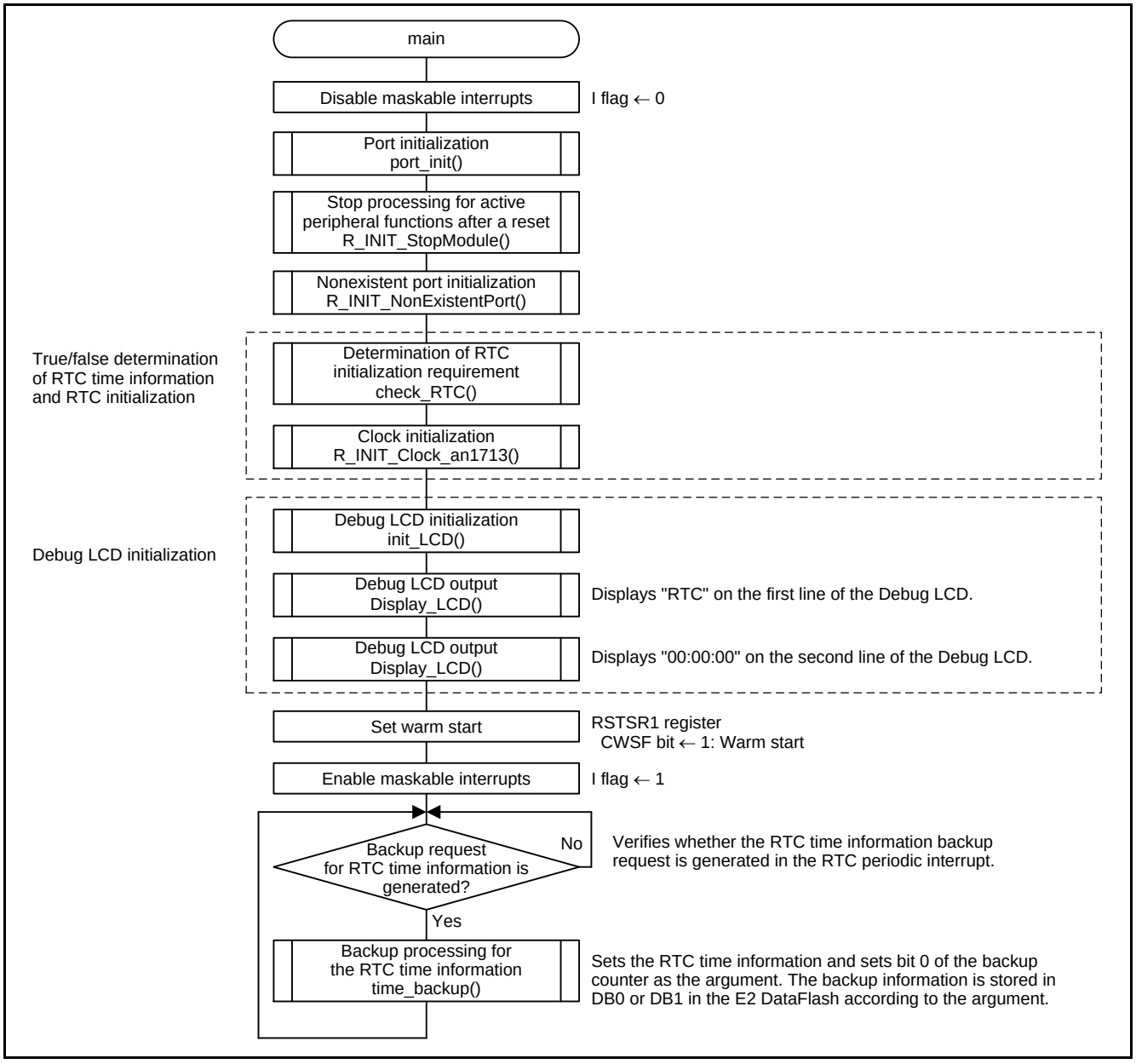


Figure 5.3 Main Processing

5.11.2 Port Initialization

Figure 5.4 shows the Port Initialization.

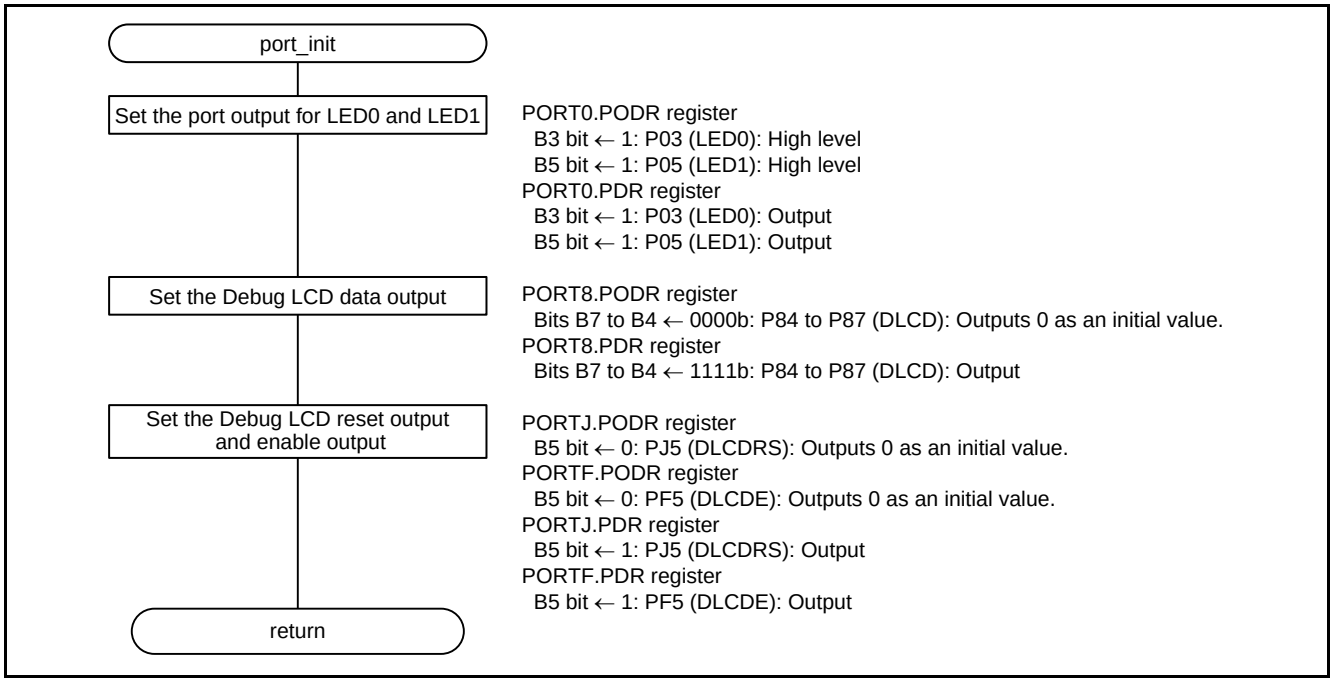


Figure 5.4 Port Initialization

5.11.3 Determination of RTC Initialization Requirement

Figure 5.5 and Figure 5.6 show the Determination of RTC Initialization Requirement.

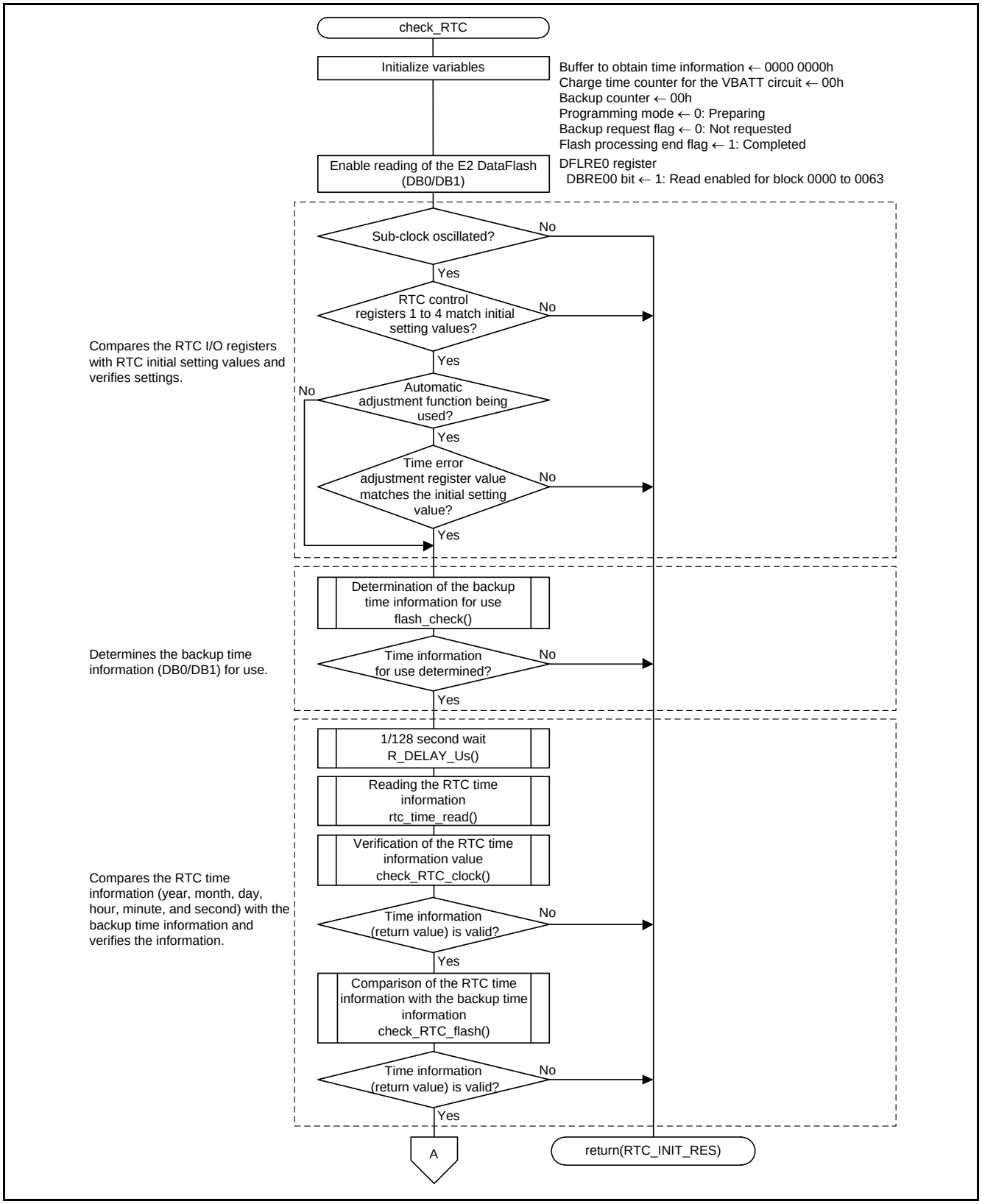


Figure 5.5 Determination of RTC Initialization Requirement (1/2)

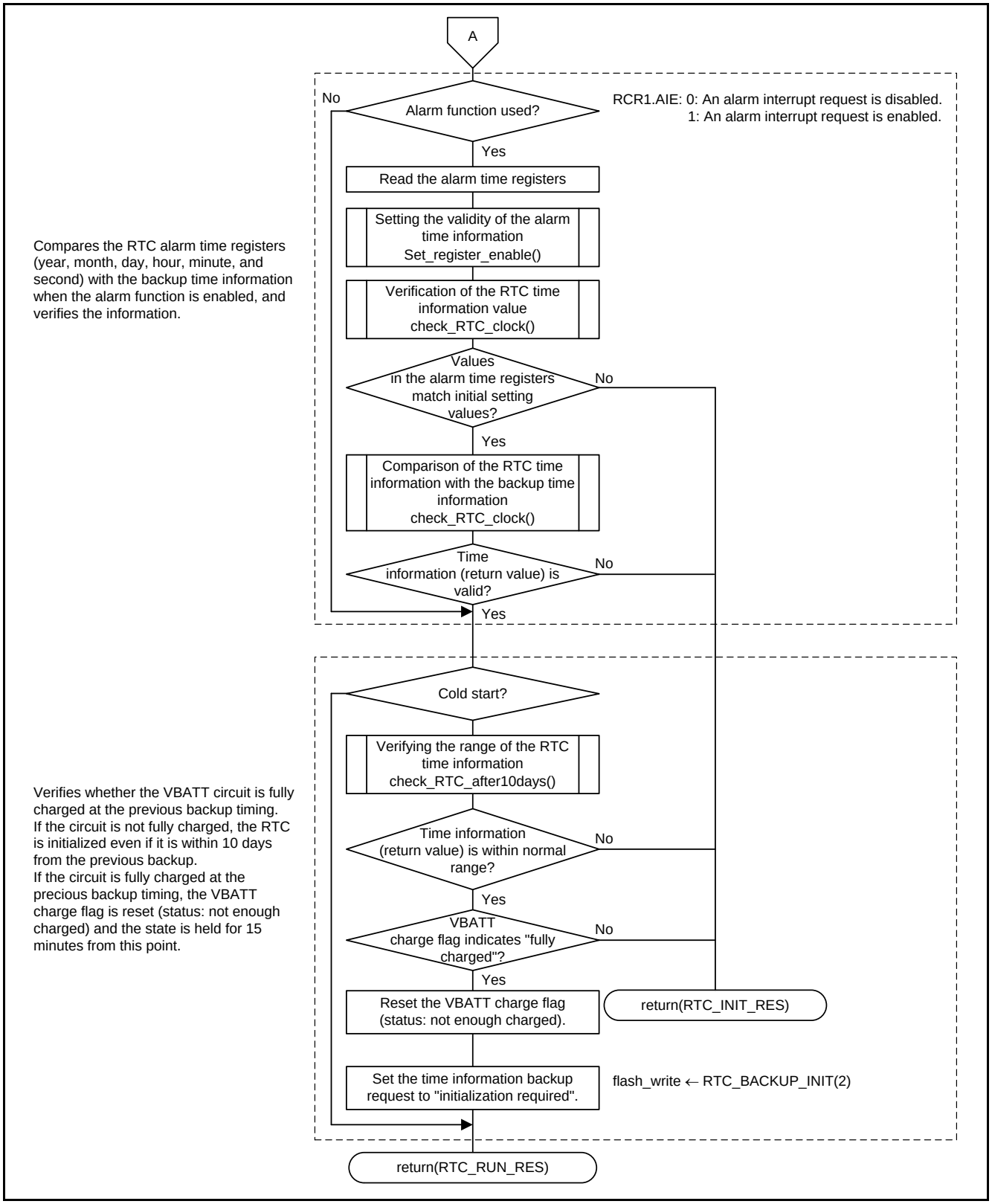


Figure 5.6 Determination of RTC Initialization Requirement (2/2)

5.11.4 Setting the Validity of the Alarm Time Information

Figure 5.7 shows the Setting the Validity of the Alarm Time Information.

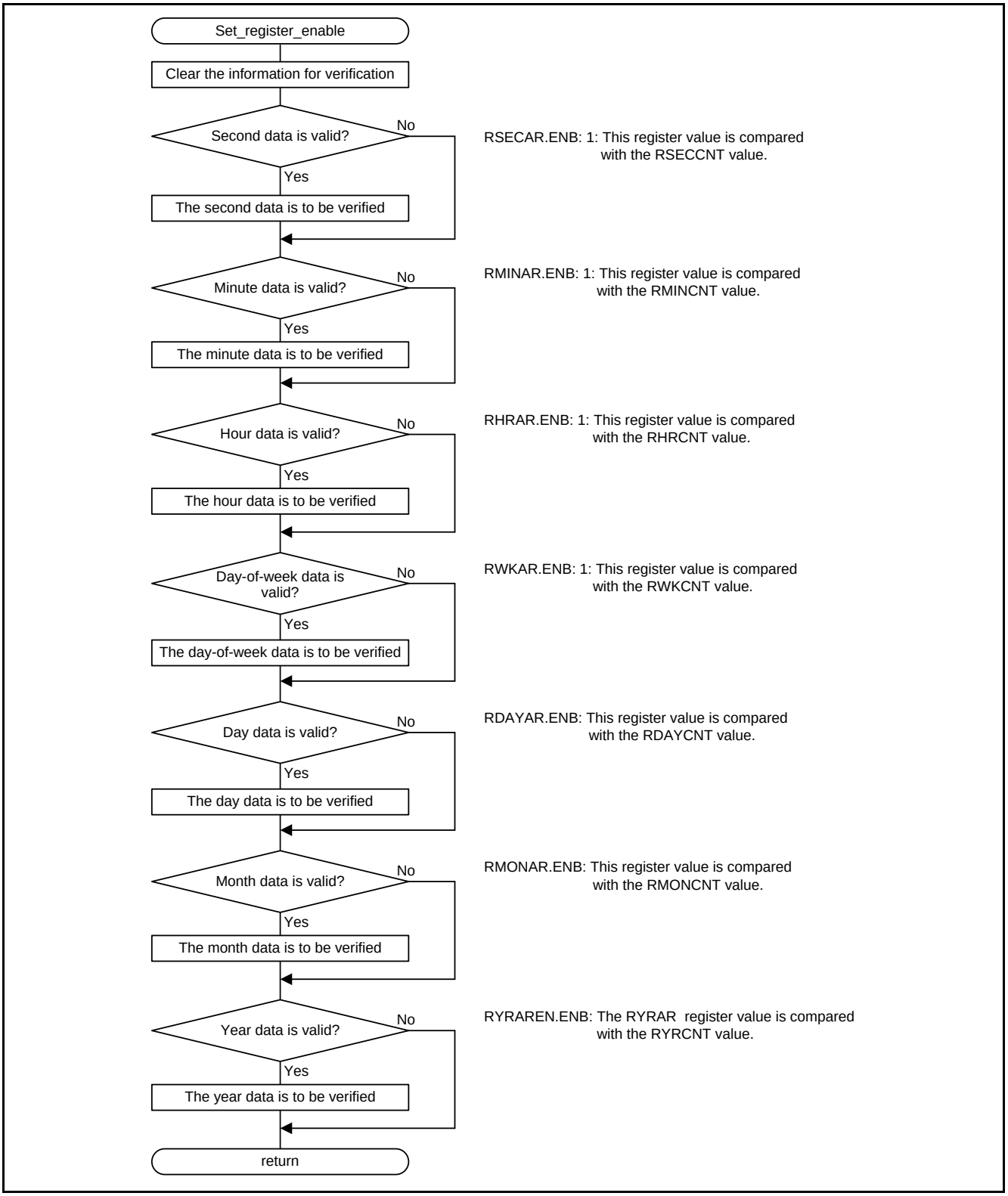


Figure 5.7 Setting the Validity of the Alarm Time Information

5.11.5 Determination of the Backup Time Information for Use

Figure 5.8 shows the Determination of the Backup Time Information for Use.

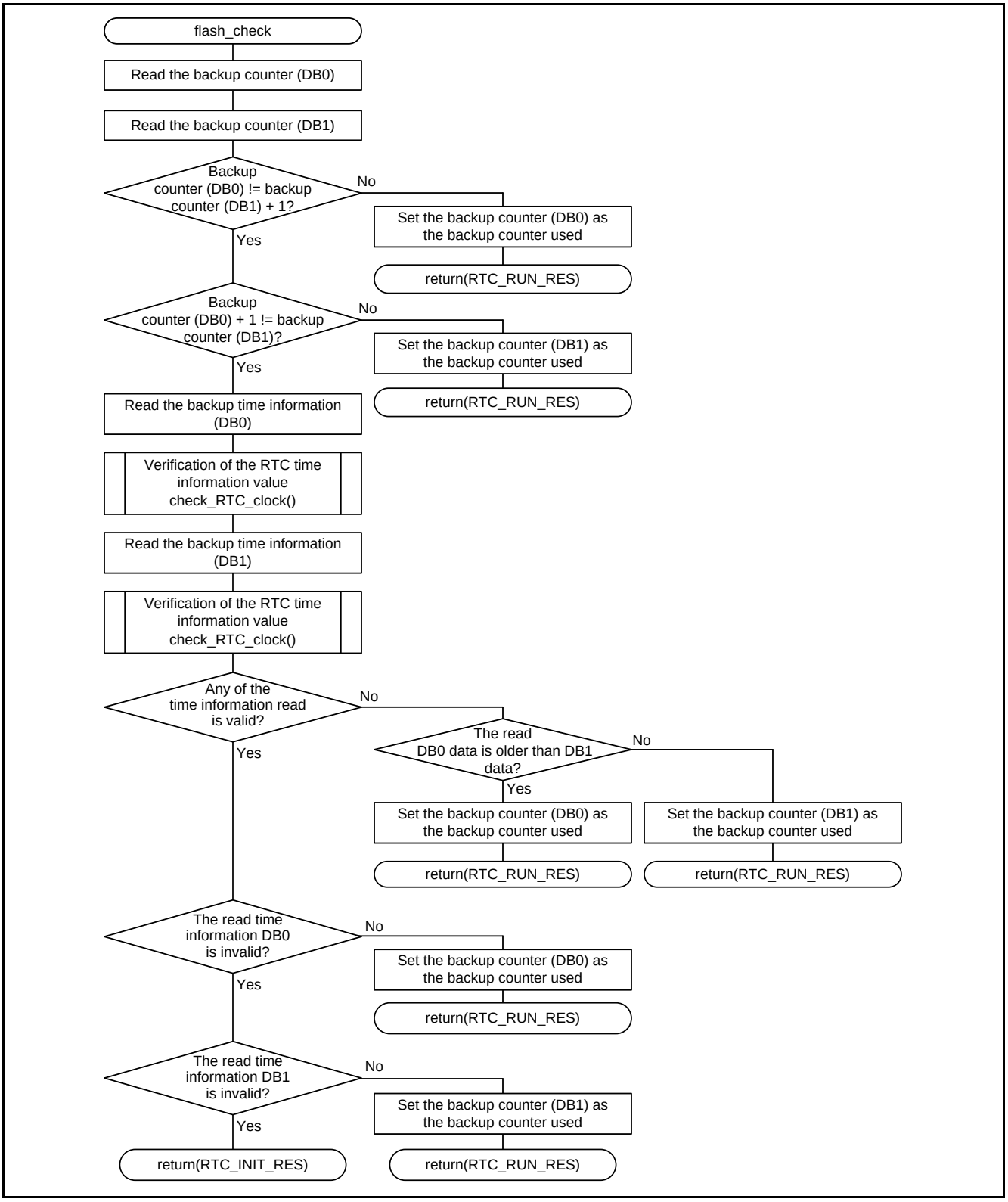


Figure 5.8 Determination of the Backup Time Information for Use

5.11.6 Verification of the RTC Time Information Value

Figure 5.9 shows the Verification of the RTC Time Information Value.

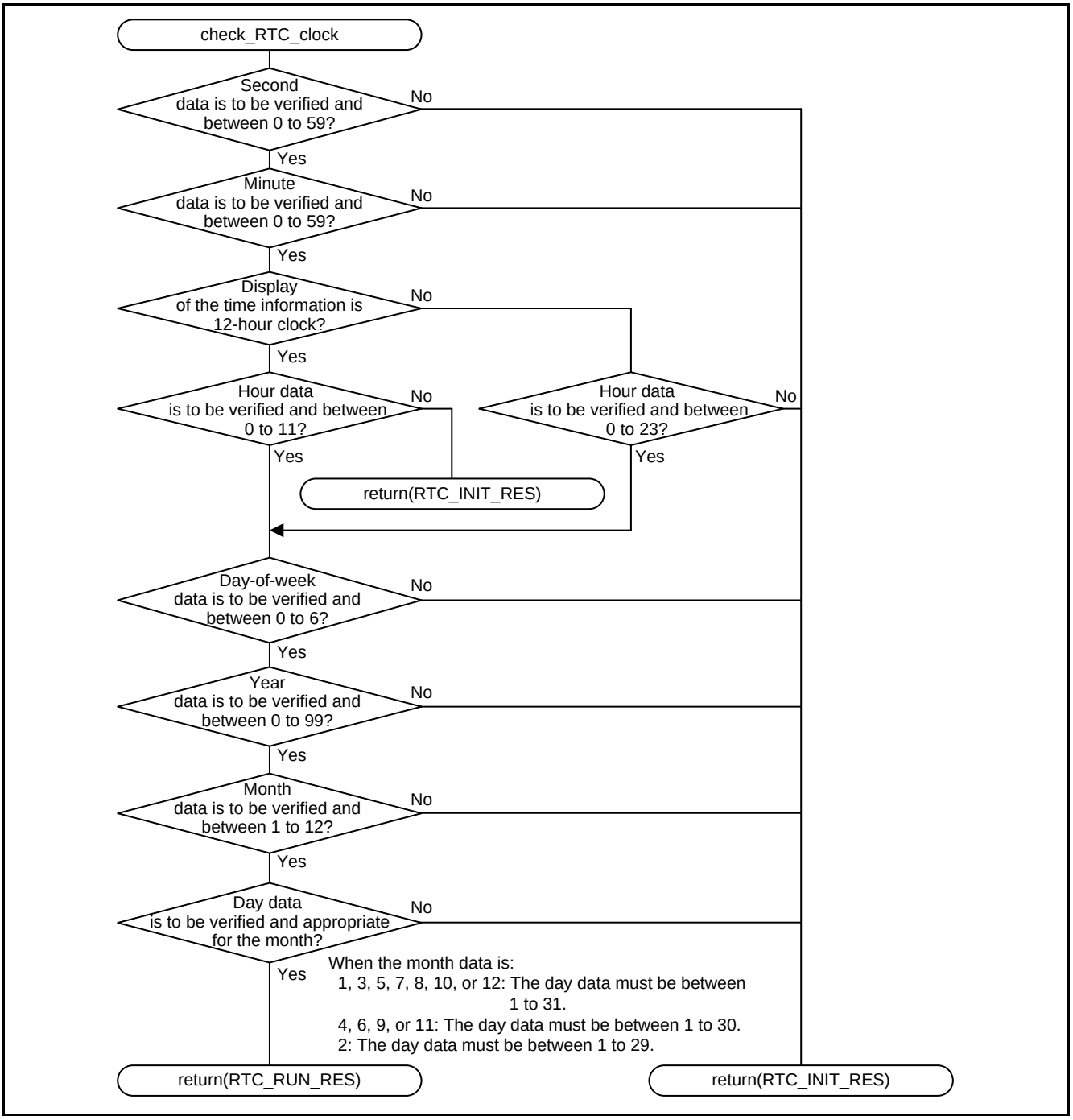
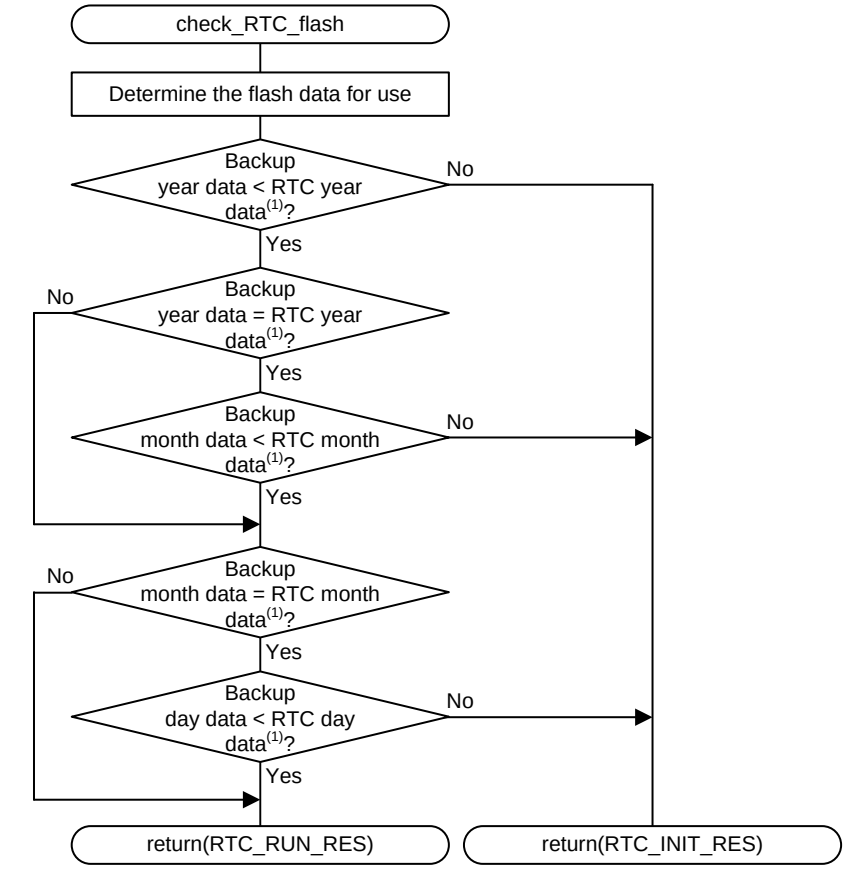


Figure 5.9 Verification of the RTC Time Information Value

5.11.7 Comparison of the RTC Time Information with the Backup Time Information

Figure 5.10 shows the Comparison of the RTC Time Information with the Backup Time Information.

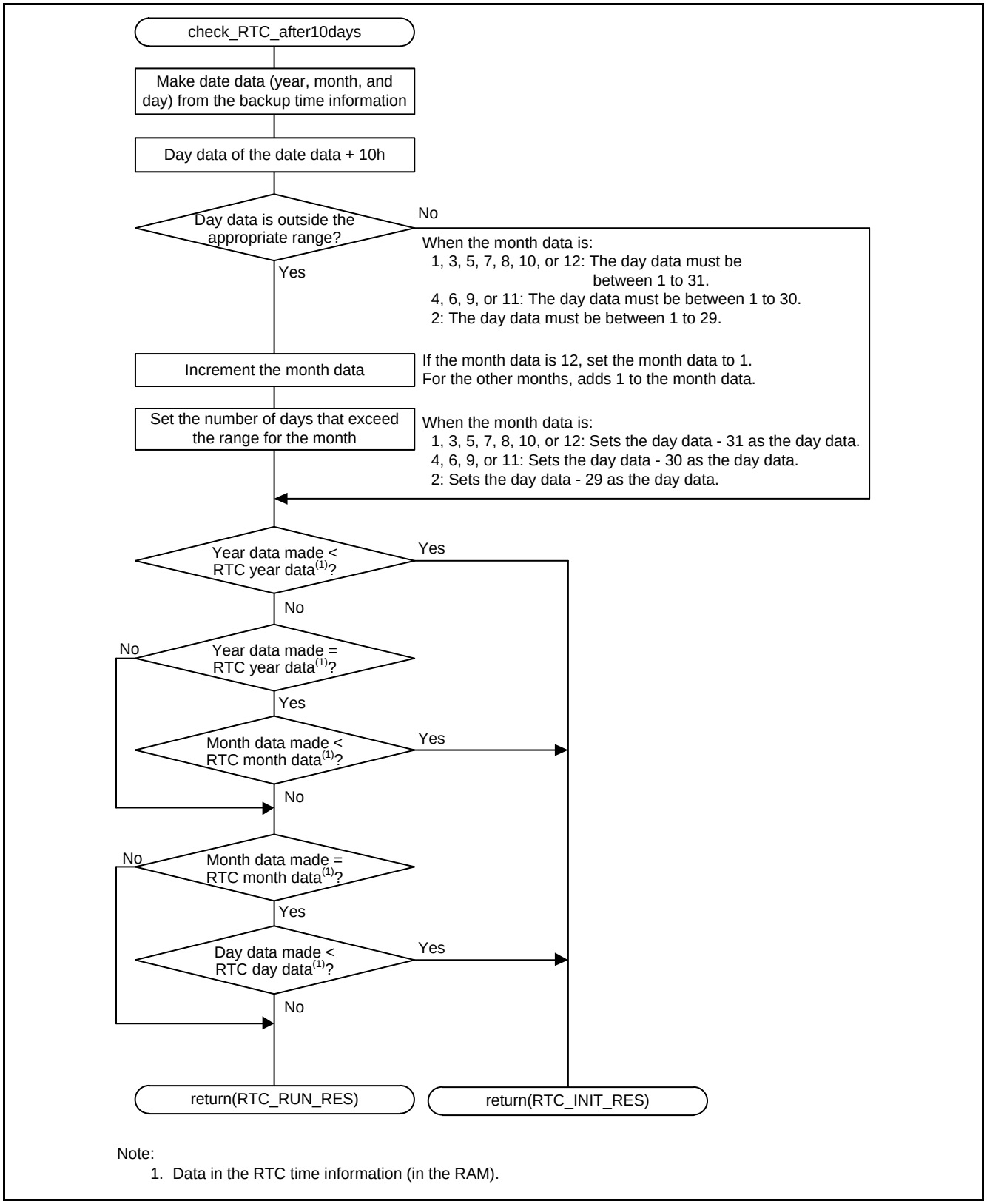


Note:
1. Data in the RTC time information (in the RAM).

Figure 5.10 Comparison of the RTC Time Information with the Backup Time Information

5.11.8 Verifying the Range of the RTC Time Information

Figure 5.11 shows Verifying the Range of the RTC Time Information.



Note:
1. Data in the RTC time information (in the RAM).

Figure 5.11 Verifying the Range of the RTC Time Information

5.11.9 Reading the RTC Time Information

Figure 5.12 shows the Reading the RTC Time Information.

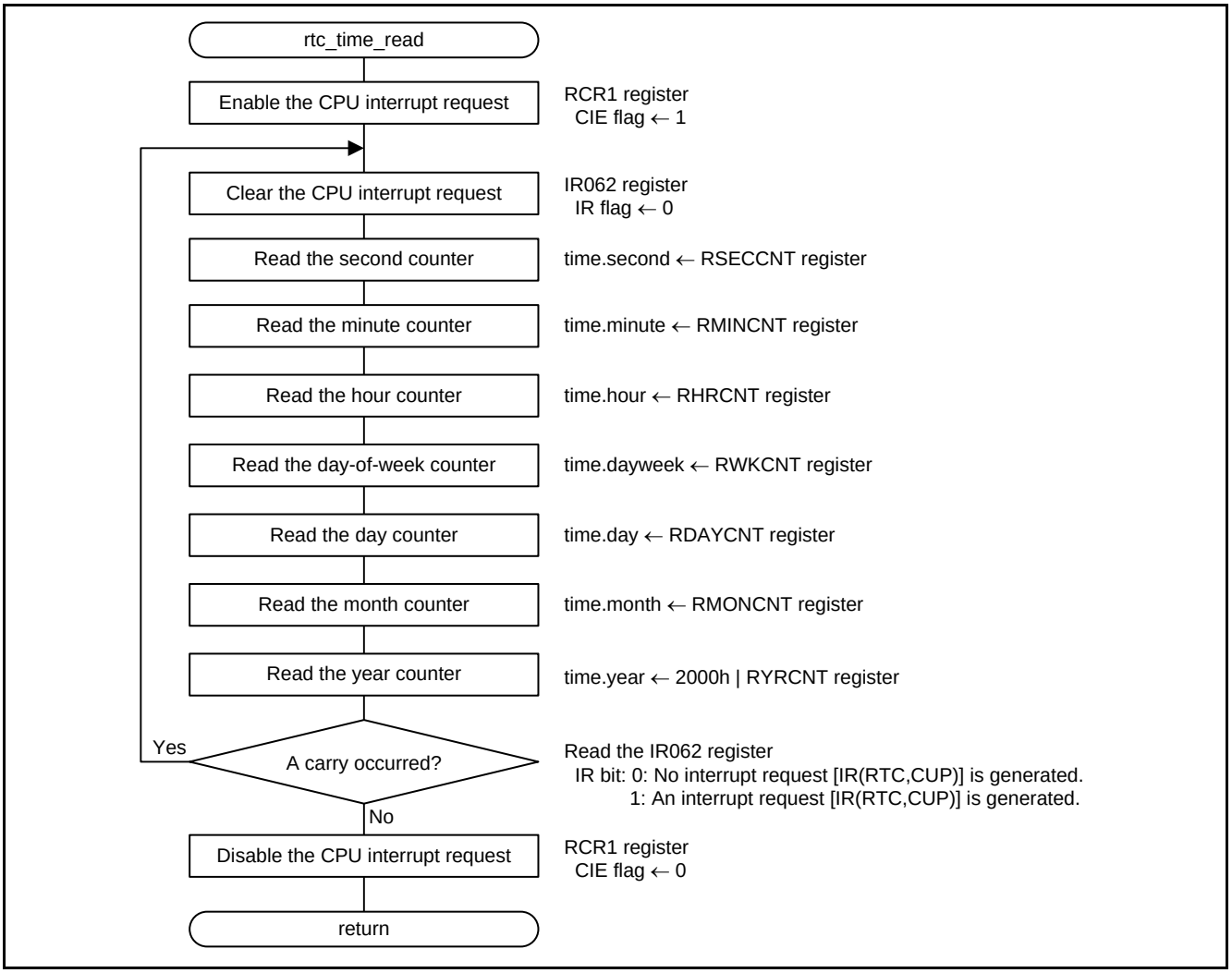


Figure 5.12 Reading the RTC Time Information

5.11.10 Clock Initialization

Figure 5.13 shows the Clock Initialization.

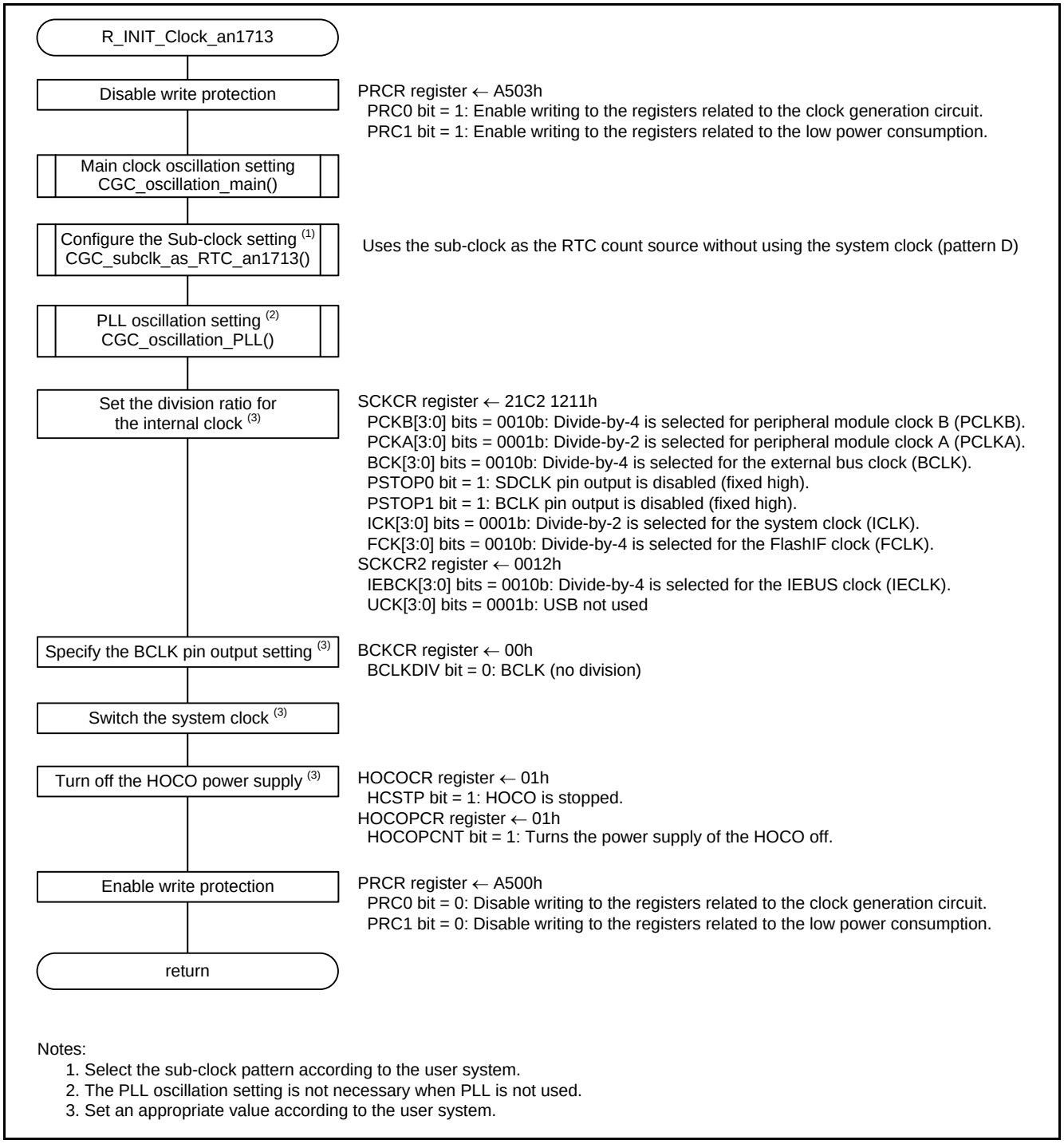
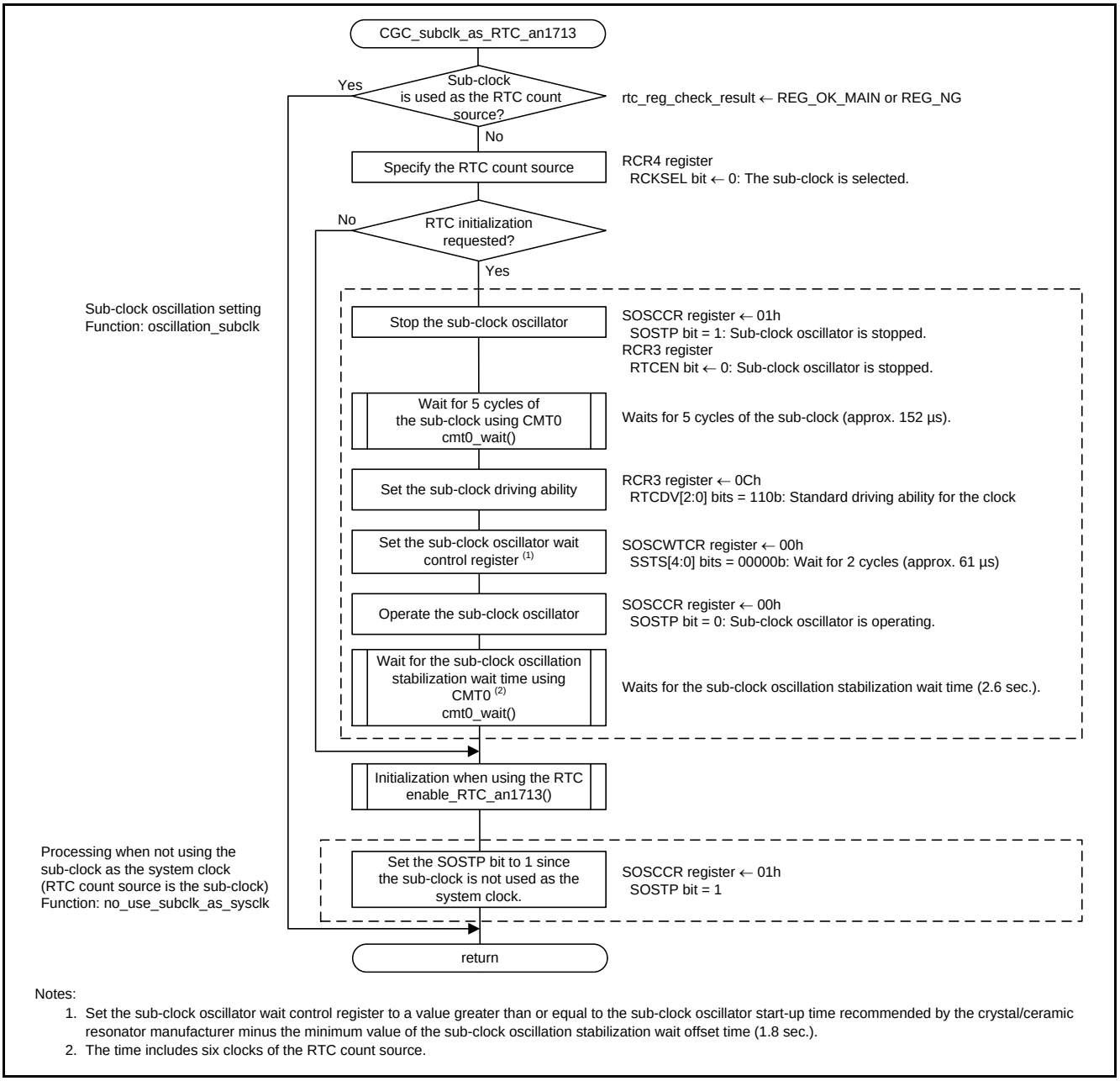


Figure 5.13 Clock Initialization

5.11.11 Sub-Clock Oscillation Setting

Figure 5.14 shows the Sub-Clock Oscillation Setting.



Notes:
1. Set the sub-clock oscillator wait control register to a value greater than or equal to the sub-clock oscillator start-up time recommended by the crystal/ceramic resonator manufacturer minus the minimum value of the sub-clock oscillation stabilization wait offset time (1.8 sec.).
2. The time includes six clocks of the RTC count source.

Figure 5.14 Sub-Clock Oscillation Setting

5.11.12 RTC Initialization

Figure 5.15 shows the RTC Initialization.

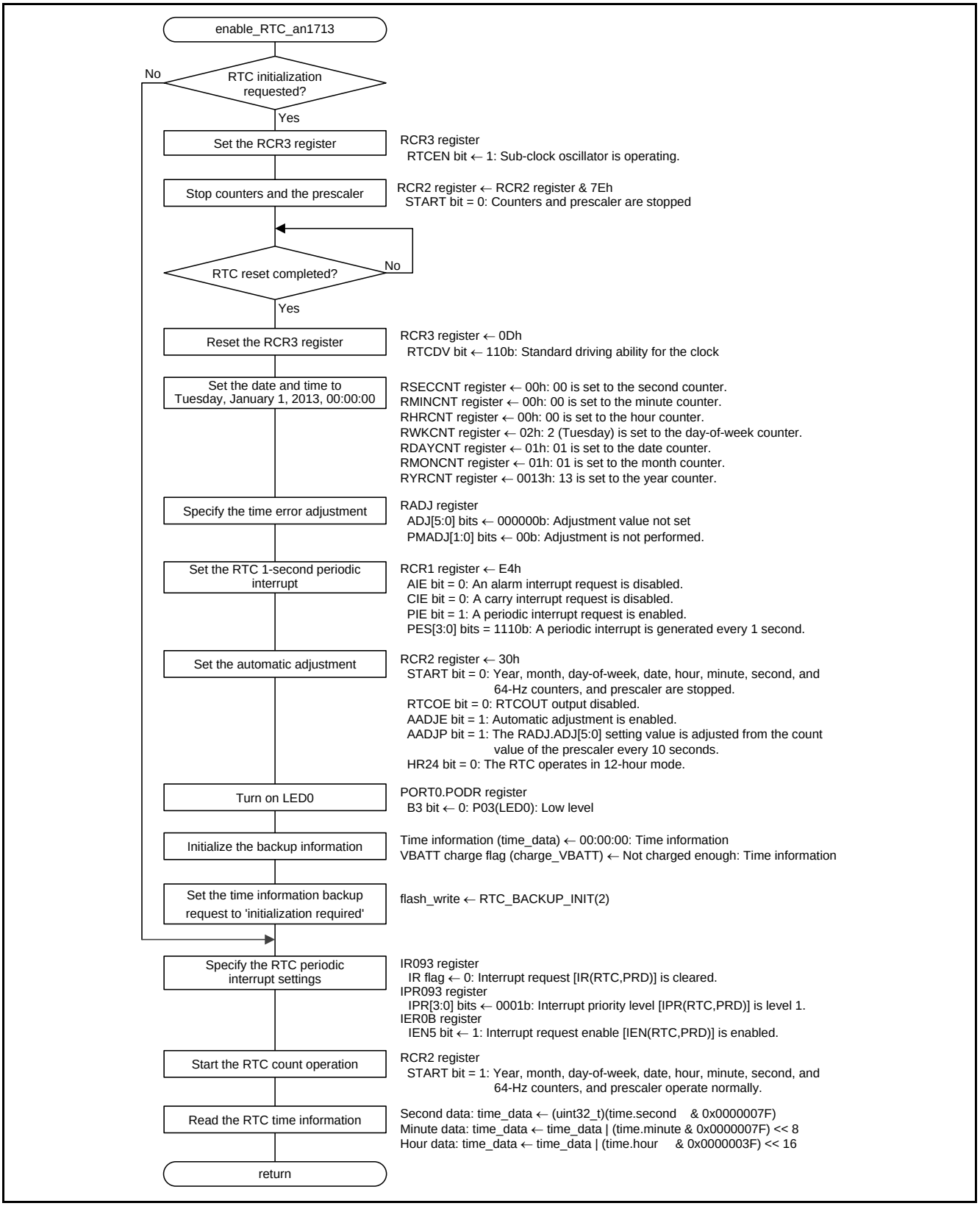


Figure 5.15 RTC Initialization

5.11.13 RTC Periodic Interrupt Handling

Figure 5.16 shows the RTC Periodic Interrupt Handling.

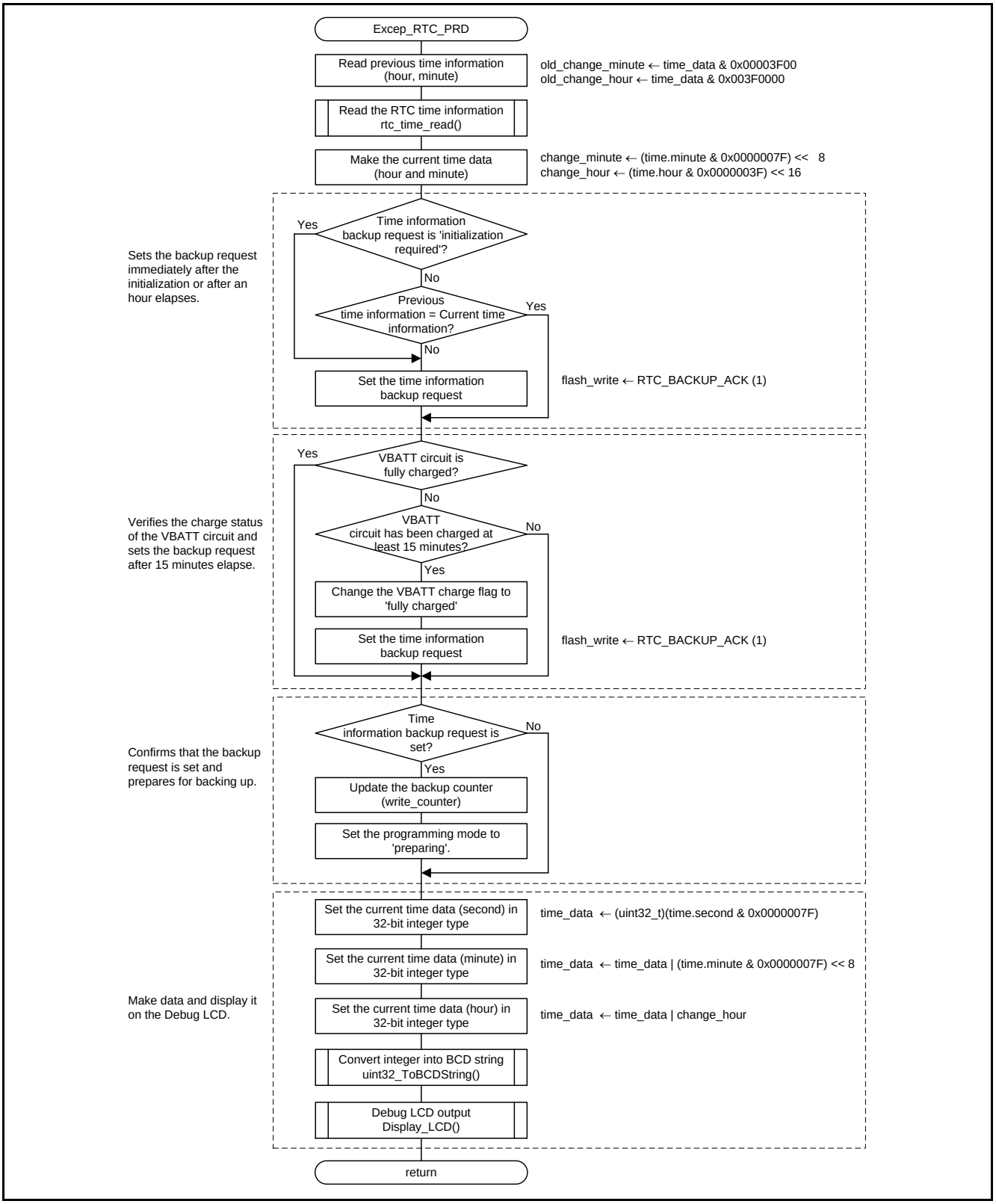


Figure 5.16 RTC Periodic Interrupt Handling

5.11.14 Backup Processing for the RTC Time Information

Figure 5.17 and Figure 5.18 show the Backup Processing for the RTC Time Information.

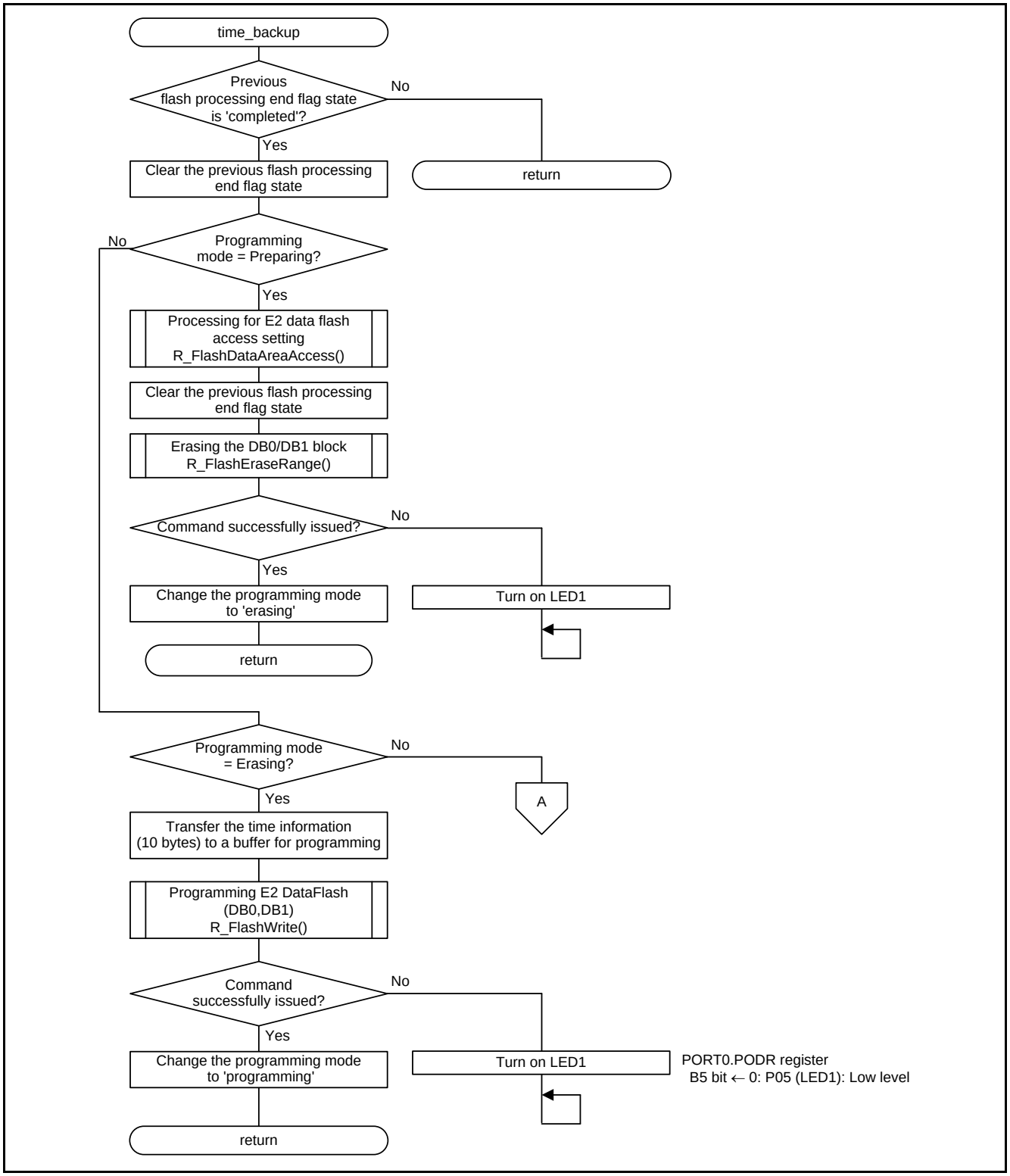


Figure 5.17 Backup Processing for the RTC Time Information (1/2)

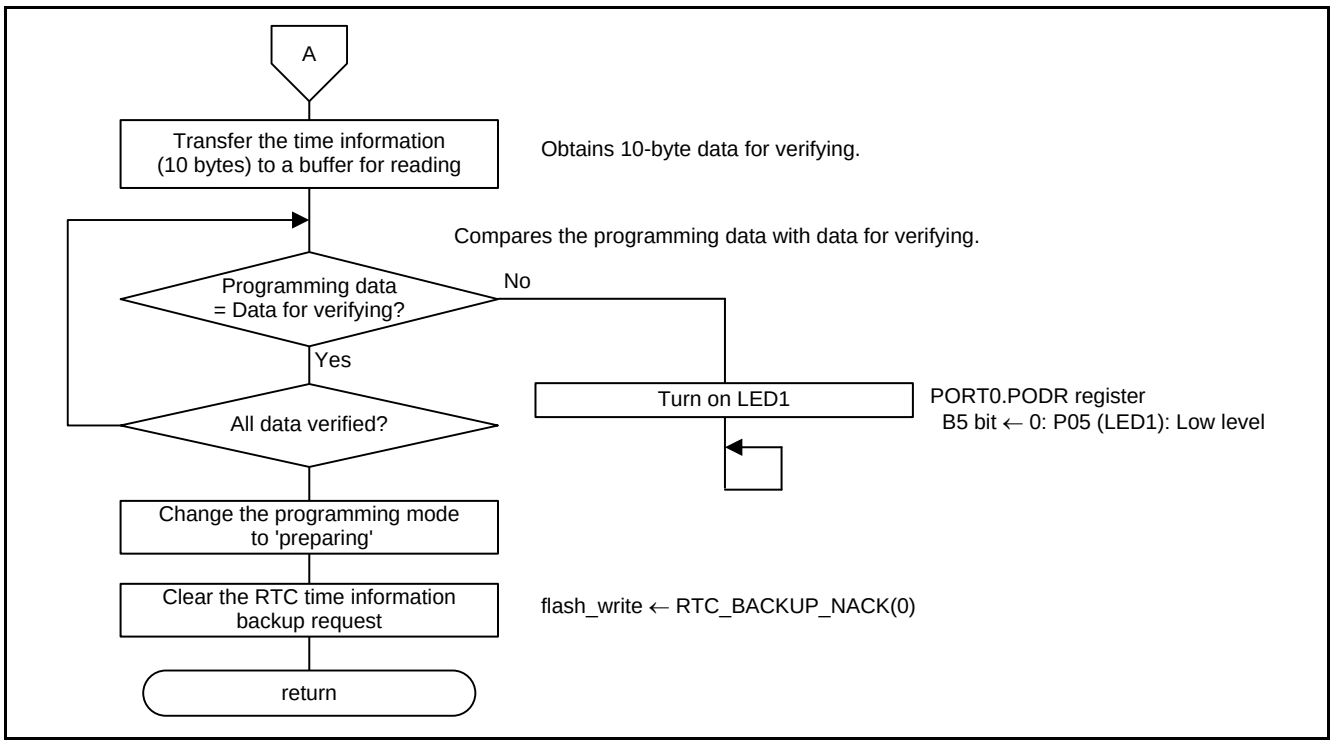


Figure 5.18 Backup Processing for the RTC Time Information (2/2)

6. Sample Code

Sample code can be downloaded from the Renesas Electronics website.

7. Reference Documents

User's Manual: Hardware

RX63N Group, RX631 Group User's Manual: Hardware Rev.1.80 (R01UH0041EJ)
The latest version can be downloaded from the Renesas Electronics website.

Technical Update/Technical News

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

User's Manual: Development Tools

RX Family C/C++ Compiler Package V.1.01 User's Manual Rev.1.00 (R20UT0570EJ)
The latest version 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	RX63N Group, RX631 Group Application Note True/False Determination of RTC Time Information on Cold Start

Rev.	Date	Description	
		Page	Summary
1.00	Apr. 1, 2014	—	First edition issued
1.01	Nov. 6, 2015	3	The specification was corrected.
		27	A figure was corrected.
		Program	The set value of OFS0 register and OFS1 register was corrected.
			The waiting time to read time counter after a reset was added.

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.

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 of these circuits, software, and information in the design of your equipment. Renesas Electronics assumes no responsibility for any losses incurred by you or third parties arising from the use of these circuits, software, or information.
2. Renesas Electronics has used reasonable care in preparing the information included in this document, but Renesas Electronics does not warrant that such information is error free. Renesas Electronics assumes no liability whatsoever for any damages incurred by you resulting from errors in or omissions from the information included herein.
3. Renesas Electronics does not assume any liability for infringement of 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. 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 should not alter, modify, copy, or otherwise misappropriate any Renesas Electronics product, whether in whole or in part. Renesas Electronics assumes no responsibility for any losses incurred by you or third parties arising from such alteration, modification, copy or otherwise misappropriation of Renesas Electronics product.
5. Renesas Electronics products are classified according to the following two quality grades: "Standard" and "High Quality". The recommended 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; and industrial robots etc.
   "High Quality": Transportation equipment (automobiles, trains, ships, etc.); traffic control systems; anti-disaster systems; anti-crime systems; and safety equipment etc.
   Renesas Electronics products are neither intended nor 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 damages (nuclear reactor control systems, military equipment etc.). You must check the quality grade of each Renesas Electronics product before using it in a particular application. You may not use any Renesas Electronics product for any application for which it is not intended. Renesas Electronics shall not be in any way liable for any damages or losses incurred by you or third parties arising from the use of any Renesas Electronics product for which the product is not intended by Renesas Electronics.
6. You should use the Renesas Electronics products described in this document within the range specified by Renesas Electronics, especially with respect to the maximum rating, operating supply voltage range, movement power voltage range, heat radiation characteristics, installation and other product characteristics. Renesas Electronics shall have no liability for malfunctions or damages arising out of the use of Renesas Electronics products beyond such specified ranges.
7. Although Renesas Electronics endeavors to improve the quality and reliability of its products, semiconductor products have specific characteristics such as the occurrence of failure at a certain rate and malfunctions under certain use conditions. Further, Renesas Electronics products are not subject to radiation resistance design. Please be sure to implement safety measures to guard them against the possibility of physical injury, and injury or damage caused by fire in the event of the failure of a Renesas Electronics product, 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, please evaluate the safety of the final products or systems manufactured by you.
8. Please contact a Renesas Electronics sales office for details as to environmental matters such as the environmental compatibility of each Renesas Electronics product. Please use Renesas Electronics products in compliance with all applicable laws and regulations that regulate the inclusion or use of controlled substances, including without limitation, the EU RoHS Directive. Renesas Electronics assumes no liability for damages or losses occurring as a result of your noncompliance with applicable laws and regulations.
9. Renesas Electronics products and technology may 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 should not use Renesas Electronics products or technology described in this document for any purpose relating to military applications or use by the military, including but not limited to the development of weapons of mass destruction. When exporting the Renesas Electronics products or technology described in this document, you should comply with the applicable export control laws and regulations and follow the procedures required by such laws and regulations.
10. It is the responsibility of the buyer or distributor of Renesas Electronics products, who distributes, disposes of, or otherwise places the product with a third party, to notify such third party in advance of the contents and conditions set forth in this document, Renesas Electronics assumes no responsibility for any losses incurred by you or third parties as a result of unauthorized use of Renesas Electronics products.
11. This document may not be reproduced or duplicated in any form, in whole or in part, without prior written consent of Renesas Electronics.
12. Please contact a Renesas Electronics sales office if you have any questions regarding the information contained in this document or Renesas Electronics products, or if you have any other inquiries.

(Note 1) "Renesas Electronics" as used in this document means Renesas Electronics Corporation and also includes its majority-owned subsidiaries.

(Note 2) "Renesas Electronics product(s)" means any product developed or manufactured by or for Renesas Electronics.

RENESAS

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

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

Renesas Electronics America Inc.
2801 Scott Boulevard Santa Clara, CA 95050-2549, U.S.A.
Tel: +1-408-588-6000, Fax: +1-408-588-6130

Renesas Electronics Canada Limited
9251 Yonge Street, Suite 8309 Richmond Hill, Ontario Canada L4C 9T3
Tel: +1-905-237-2004

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, HALII 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 5.0