# RX210, RX21A, and RX220 Groups

## Exiting Software Standby Mode Using the RTCb

R01AN1652EJ0101
Rev. 1.01
July 1, 2014

### Abstract

This document describes how to obtain the current time information while intermittently exiting software standby mode using the realtime clock (RTC) in the RX210, RX21A, and RX220 Groups.

### Products

- RX210, RX21A, and RX220 Groups

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

This document describes how to obtain the time information while intermittently exiting software standby mode using the RTC.

After a reset, if the reset processing is a cold start, the sub-clock oscillator and RTC are initialized. If the reset processing is a warm start, the RTC data is retained so initialization is not performed. Then the input level of the interrupt request pin is monitored. If the input level is low, the MCU enters software standby mode.

Once the MCU enters software standby mode, it exits software standby mode in concert with the periodic interrupt which occurs every 1/2 second. Then time information is obtained, and the MCU enters software standby mode according to the input level of the interrupt request pin.

- RTC count source: Sub-clock
- Intermittent period to exit software standby mode: 1/2 second

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 |
|---|---|
| RTCb | Used as clock function and to exit software standby mode. |
| IRQ1 | Used as an external input for entering software standby mode. |

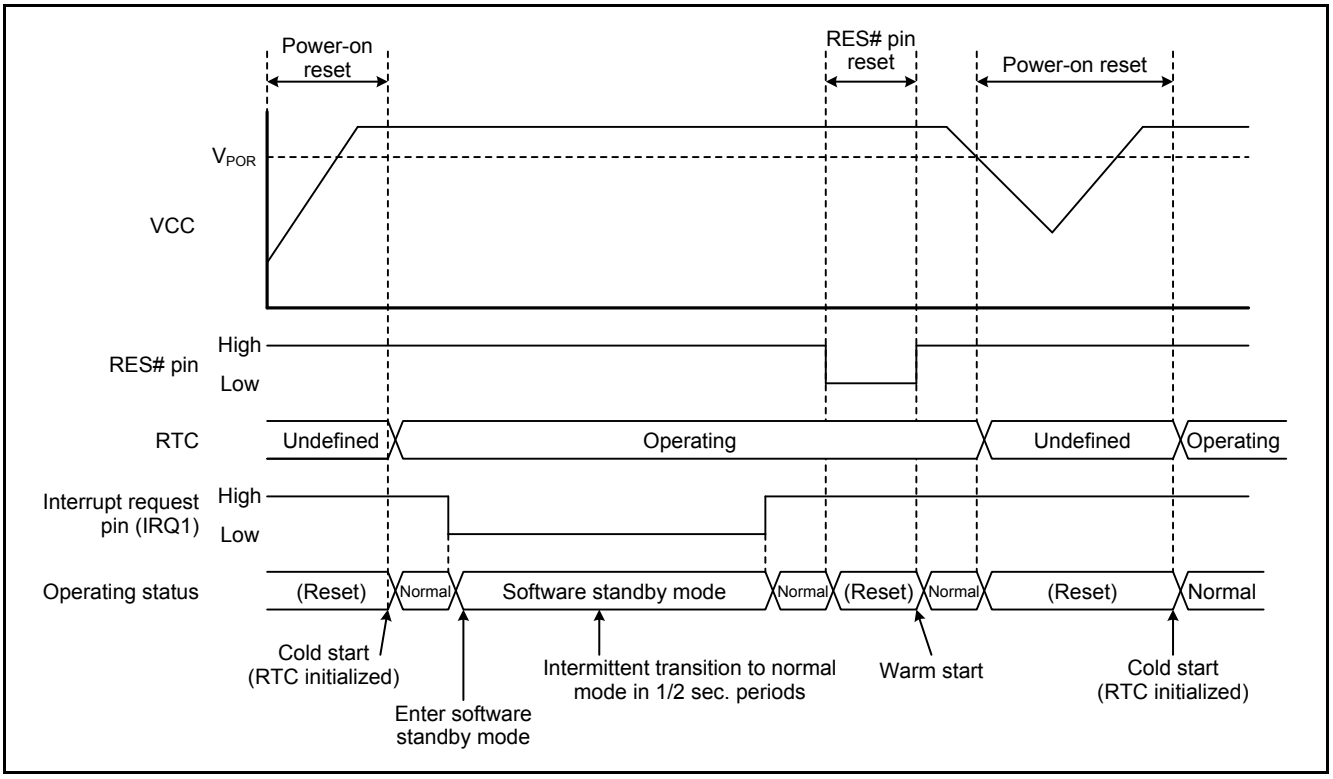



**Figure 1.1 Operation Overview**

## 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 | R5F5210BBDFP (RX210 Group) |
| Operating frequencies | - Main clock: 20 MHz<br>- Sub-clock: 32.768 kHz<br>- PLL: 100 MHz (main clock divided by 2 and multiplied by 10)<br>- LOCO: 125 kHz<br>- System clock (ICLK): 50 MHz (PLL divided by 2)<br>- Peripheral module clock B (PCLKB): 25 MHz (PLL divided by 4)<br>- RTC-dedicated sub-clock (RTCSCLK) |
| Operating voltage | 5.0 V |
| Integrated development environment | Renesas Electronics Corporation<br>High-performance Embedded Workshop Version 4.09.01 |
| C compiler | Renesas Electronics Corporation<br>C/C++ Compiler Package for RX Family V.1.02 Release 01 |
| | Compile options<br>-cpu=rx200 -output=obj="$(CONFIGDIR)\$(FILELEAF).obj" -debug -nologo<br>(The default setting is used in the integrated development environment.) |
| iodefine.h version | Version 1.4 |
| Endian | Little endian |
| Operating mode | Single-chip mode |
| Processor mode | Supervisor mode |
| Sample code version | Version 1.00 |
| Board used | Renesas Starter Kit for RX210 (product part no.: R0K505210C002BE) |

## 3. Reference Application Notes

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

- RX210 Group Initial Setting Rev. 2.10 (R01AN1002EJ)
- RX21A Group Initial Setting Rev. 1.10 (R01AN1486EJ)
- RX220 Group Initial Setting Rev. 1.10 (R01AN1494EJ)

The initial setting functions in the reference application notes are used in the sample code in this application note. The revision numbers of the reference application notes are current as of when this application note was 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.

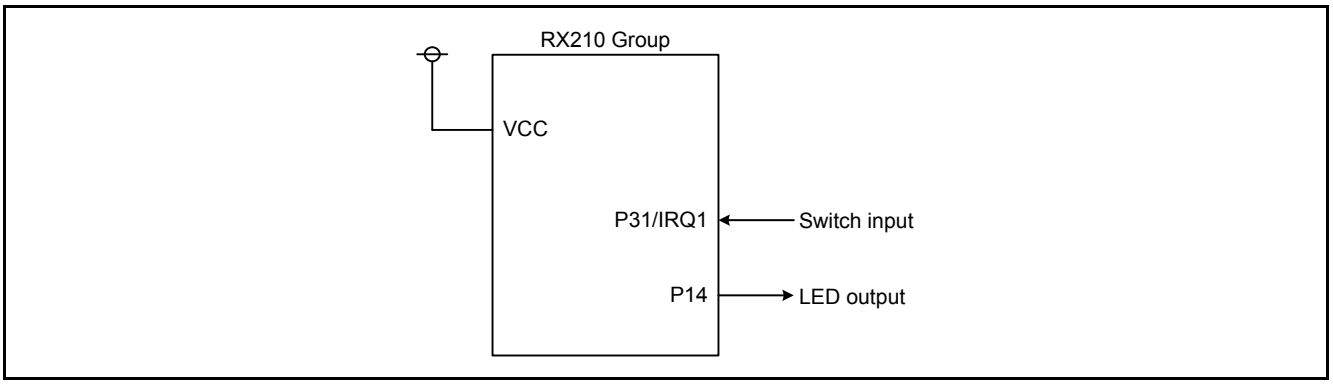



**Figure 4.1 Connection Example**

## 4.2 Pins Used

Table 4.1 lists the Pins Used and Their Functions.

The number of pins in the sample code is set for the 100-pin package. When using products with less than 100 pins, select pins appropriate to the package used.

**Table 4.1 Pins Used and Their Functions**

| Pin Name | I/O | Function |
|---|---|---|
| P31/IRQ1 | Input | Switch input for mode transition |
| P14 | Output | LED output |

## 5. Software

In normal mode, the RTC time information is updated and the level of the interrupt request pin is monitored. The time information is stored in the time data storage area (global variable). The level of the interrupt request pin is monitored using the IR flag of the IRQ1 interrupt.

When the IR flag of the IRQ1 interrupt is 1, the RTC periodic interrupt (PRD interrupt) is enabled, and the MCU enters software standby mode. The MCU exits software standby mode by a PRD interrupt request, and the PRD interrupt is disabled.

After the MCU exits software standby mode, the RTC time information is updated, the level of the interrupt request pin is monitored, and the above processing is repeated.

Settings for the peripheral functions are listed below.

RTC

Count source: Sub-clock

Initial time setting: 00:00:00, Tuesday, January 1, 2013

Time mode: 24-hour mode

RTCOUT output: Disabled

Error adjustment: Not used

Time capture: Not used

Interrupts used: PRD interrupt (generated in 1/2 sec. periods), Carry interrupt (CUP)

IRQ1 input pin

Detection method: Low level is detected

Digital filter: Enabled (sampling clock: PCLKB/8)

Interrupt used: External pin interrupt (IRQ1 interrupt)

## 5.1 Operation Overview

(1) Initial setting

After a reset, if the RSTSR1.CWSF bit is 0 (cold start), the sub-clock oscillator and RTC are initialized, and the RSTSR1.CWSF bit is set to 1 (warm start). Then the LED is turned on, the RTC time information is read, and the IR flag of the IRQ1 interrupt is monitored.

(2) Entering software standby mode

When the IR flag of the IRQ1 interrupt becomes 1, the PRD interrupt request is enabled. The LED is turned off, the WAIT instruction is executed, and the MCU enters software standby mode.

(3) Exiting software standby mode

The MCU exits software standby mode by the PRD interrupt request generated every 1/2 second. Then the LED is turned on in the PRD interrupt handling. The PRD interrupt request is disabled, a wait for 1/128 second is performed, and the RTC time information is read. If the IR flag of the IRQ1 interrupt is 1, the processing in step (3) is performed again; if the IR flag is 0, the RTC time information is read. Also the IR flag of the IRQ1 is monitored.

(4) Warm start

After a reset, if the RSTSR1.CWSF bit is 1, the operation is continued without initializing the sub-clock oscillator and RTC.

Figure 5.1 shows the Timing Diagram.

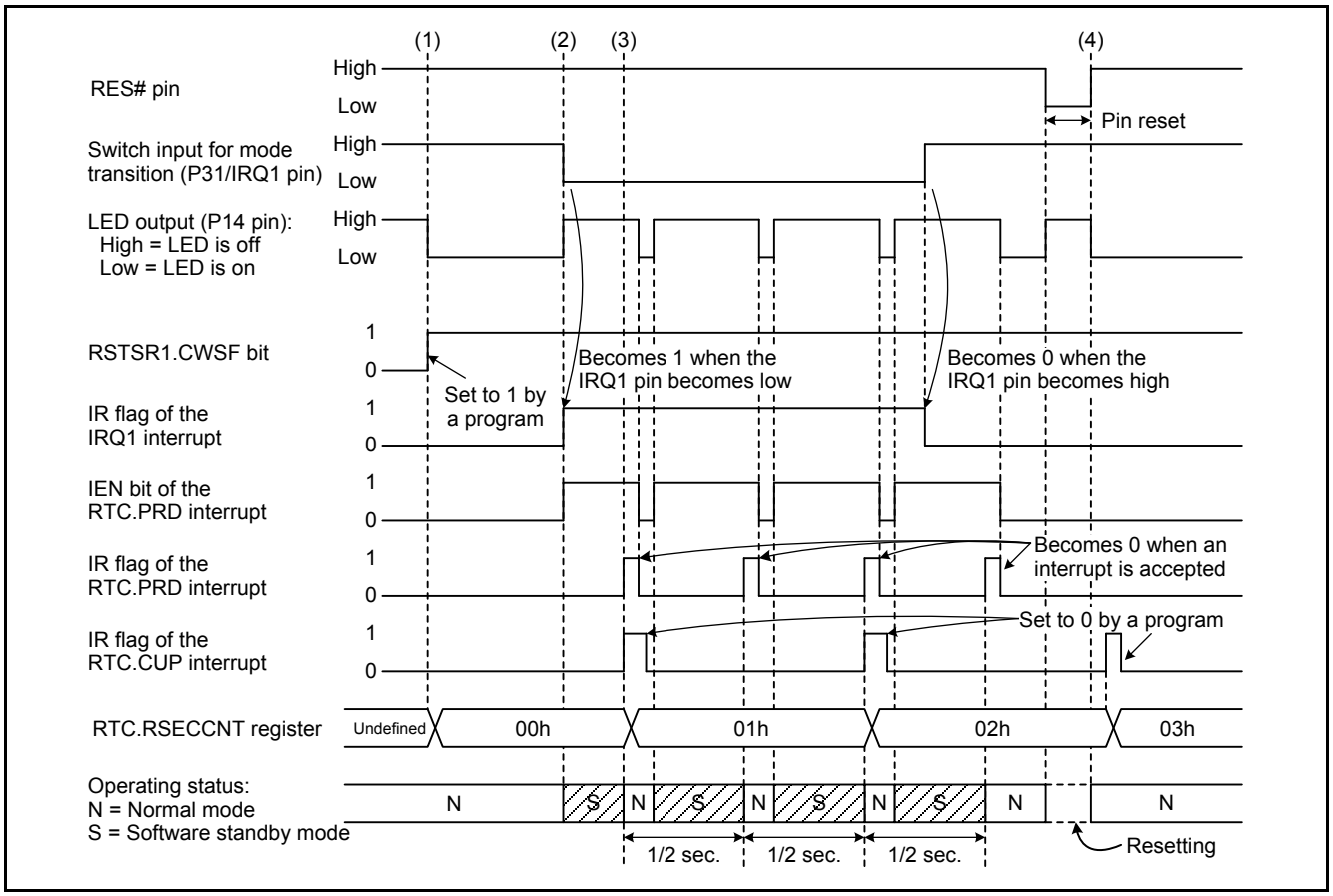



**Figure 5.1 Timing Diagram**

## 5.2 File Composition

Table 5.1 lists the Files Used in the Sample Code. Files generated by the integrated development environment are not included in this table.

**Table 5.1 Files Used in the Sample Code**

| File Name | Outline | Remarks |
|---|---|---|
| main.c | Main processing | |
| r_init_stop_module.c | Stop processing for active peripheral functions after a reset | |
| r_init_stop_module.h | Header file for r_init_stop_module.c | |
| r_init_non_existent_port.c | Nonexistent port initialization | |
| r_init_non_existent_port.h | Header file for r_init_non_existent_port.c | |
| r_init_clock.c | Clock initialization | |
| r_init_clock.h | Header file for r_init_clock.c | |

## 5.3 Option-Setting Memory

Table 5.2 lists the Option-Setting Memory Configured in the Sample Code. When necessary, set a value suited to the user system.

**Table 5.2 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.<br>The WDT is stopped after a reset. |
| OFS1 | FFFF FF8Bh to FFFF FF88h | FFFF FFFFh | The voltage monitor 0 reset is disabled after a reset. HOCO oscillation is disabled after a reset. |
| MDES | FFFF FF83h to FFFF FF80h | FFFF FFFFh | Little endian |

## 5.4 Constants

Table 5.3 lists the Constants Used in the Sample Code.

**Table 5.3 Constants Used in the Sample Code**

| Constant Name | Setting Value | Contents |
|---|---|---|
| SW_STANDBY | IR(ICU,IRQ1) | IRQ1 interrupt status flag: Switch input for mode transition |
| SW_ON | 1 | Switch on |
| SW_OFF | 0 | Switch off |
| LED_RUN | PORTC.PODR.BIT.B4 | P14 output data store bit: LED |
| LED_ON | 0 | LED turned on |
| LED_OFF | 1 | LED turned off |
| LOOP_COUNT | 50,000,000L/128 | Loop counter: Wait 1/128 second or longer (ICLK = 50 MHz) |

## 5.5 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 */
} time_bcd_t;
```

**Figure 5.2 Structure/Union Used in the Sample Code**

## 5.6 Variable

Table 5.4 lists the Global Variable.

**Table 5.4 Global Variable**

| Type | Variable Name | Contents | Function Used |
|---|---|---|---|
| time_bcd_t | time | Time data storage area | rtc_time_read |

## 5.7 Functions

Table 5.5 lists the Functions.

**Table 5.5 Functions**

| Function Name | Outline |
|---|---|
| main | Main processing |
| port_init | Port initialization |
| R_INIT_StopModule | Stop processing for active peripheral functions after a reset |
| R_INIT_NonExistentPort | Nonexistent port initialization |
| R_INIT_Clock_an1652 | Clock initialization |
| peripheral_init | Peripheral function initialization |
| irq_init | IRQ initialization |
| rtc_init | RTC initialization |
| rtc_time_read | Reading RTC time information |
| run_to_standby | Preparation to enter software standby mode |
| standby_to_run | Exiting software standby mode |
| Excep_RTC_PRD | PRD interrupt handling |

## 5.8 Function Specifications

The following tables list the sample code function specifications.

| main | |
|---|---|
| **Outline** | Main processing |
| **Header** | None |
| **Declaration** | void main(void) |
| **Description** | After the initial setting, turns on the LED and reads the RTC time information. If the switch input for mode transition indicates on, enters software standby mode. |
| **Arguments** | None |
| **Return Value** | None |

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

| R_INIT_StopModule | |
|---|---|
| **Outline** | Stop processing for active peripheral functions after a reset |
| **Header** | r_init_stop_module.h |
| **Declaration** | void R_INIT_StopModule(void) |
| **Description** | Configures the setting to enter the module stop state. |
| **Arguments** | None |
| **Return Value** | None |
| **Remarks** | Transition to the module stop state is not performed in the sample code. For details on this function, refer to the Initial Setting application note for the product used. |

| R_INIT_NonExistentPort | |
|---|---|
| **Outline** | Nonexistent port initialization |
| **Header** | r_init_non_existent_port.h |
| **Declaration** | void R_INIT_NonExistentPort(void) |
| **Description** | Initializes port direction registers for ports that do not exist in products with less than 144 pins. |
| **Arguments** | None |
| **Return Value** | None |
| **Remarks** | The number of pins in the sample code is set for the 100-pin package (PIN_SIZE=100). After this function is called, when writing in byte units to the PDR registers or PODR registers which have nonexistent ports, set the corresponding bits for nonexistent ports as follows: set the I/O select bits in the PDR registers to 1 and set the output data store bits in the PODR registers to 0.<br>For details on this function, refer to the Initial Setting application note for the product used. |

| R_INIT_Clock_an1652 | |
|---|---|
| **Outline** | Clock initialization |
| **Header** | r_init_clock.h |
| **Declaration** | void R_INIT_Clock_an1652(void) |
| **Description** | Initializes clocks. Determines whether the reset processing is a cold start or a warm start. If the reset processing is a cold start, configures the sub-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.<br>For details on this function, refer to the Initial Setting application note for the product used. |

| peripheral_init | |
|---|---|
| **Outline** | Peripheral function initialization |
| **Header** | None |
| **Declaration** | static void peripheral_init(void) |
| **Description** | Initializes peripheral functions used. |
| **Arguments** | None |
| **Return Value** | None |

| irq_init | |
|---|---|
| **Outline** | IRQ initialization |
| **Header** | None |
| **Declaration** | void irq_init(void) |
| **Description** | Initializes the IRQ. |
| **Arguments** | None |
| **Return Value** | None |

| rtc_init | |
|---|---|
| **Outline** | RTC initialization |
| **Header** | None |
| **Declaration** | void rtc_init(void) |
| **Description** | Initializes the RTC. |
| **Arguments** | None |
| **Return Value** | None |

| rtc_time_read | |
|---|---|
| **Outline** | Reading RTC time information |
| **Header** | None |
| **Declaration** | void rtc_time_read(void) |
| **Description** | Reads the RTC time information and stores it in the time data storage area. |
| **Arguments** | None |
| **Return Value** | None |

| run_to_standby | |
|---|---|
| **Outline** | Preparation to enter software standby mode |
| **Header** | None |
| **Declaration** | void run_to_standby(void) |
| **Description** | Performs processing required before entering software standby mode. |
| **Arguments** | None |
| **Return Value** | None |

| standby_to_run | |
|---|---|
| **Outline** | Exiting software standby mode |
| **Header** | None |
| **Declaration** | void standby_to_run(void) |
| **Description** | Performs processing after exiting software standby mode. |
| **Arguments** | None |
| **Return Value** | None |

| Excep_RTC_PRD | |
|---|---|
| **Outline** | RTC period interrupt handling |
| **Header** | None |
| **Declaration** | void Excep_RTC_PRD(void) |
| **Description** | Turns on the LED. |
| **Arguments** | None |
| **Return Value** | None |

## 5.9 Flowcharts

### 5.9.1 Main Processing

Figure 5.3 shows the Main Processing.

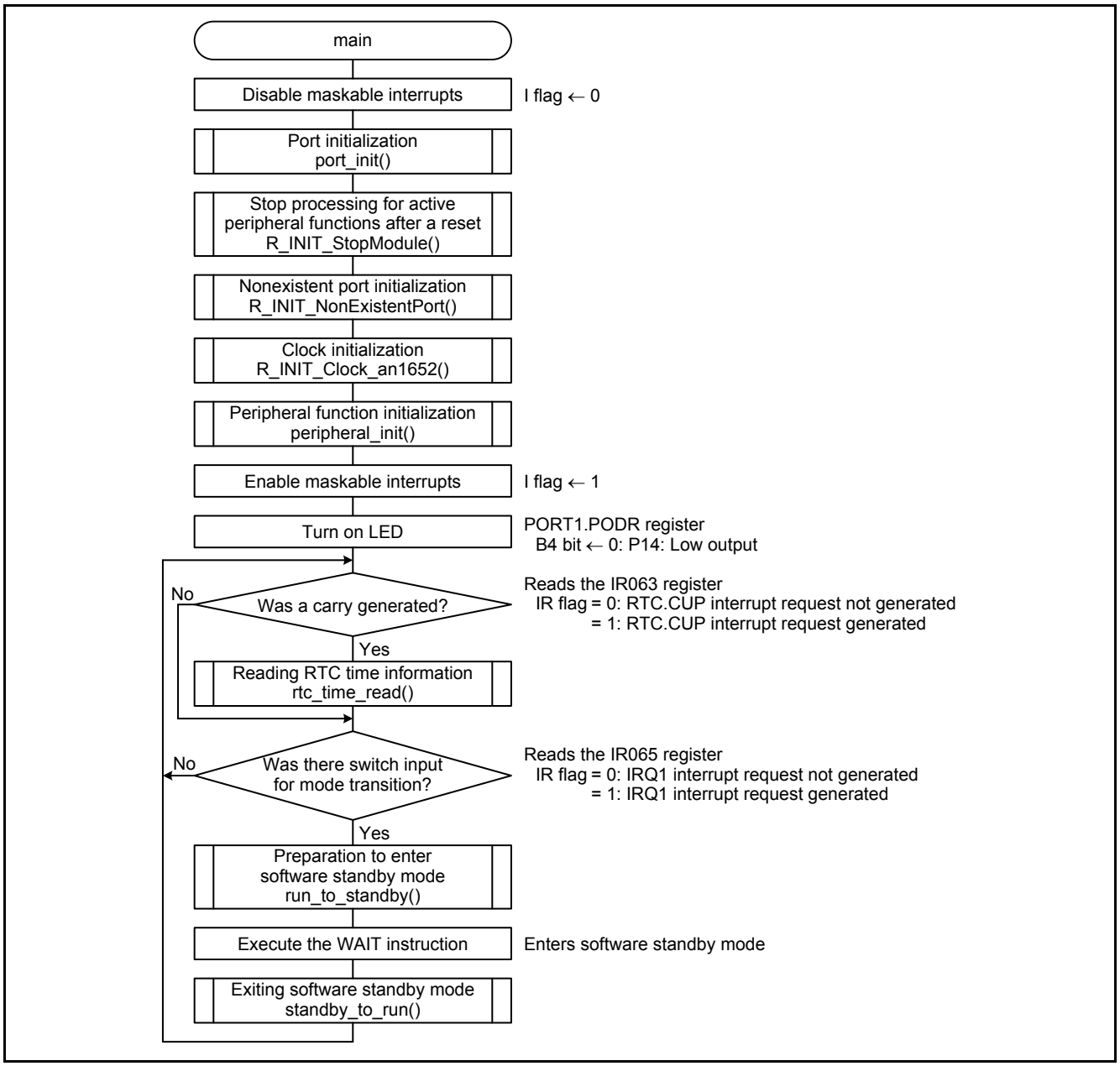



**Figure 5.3 Main Processing**

### 5.9.2 Port Initialization

Figure 5.4 shows Port Initialization.

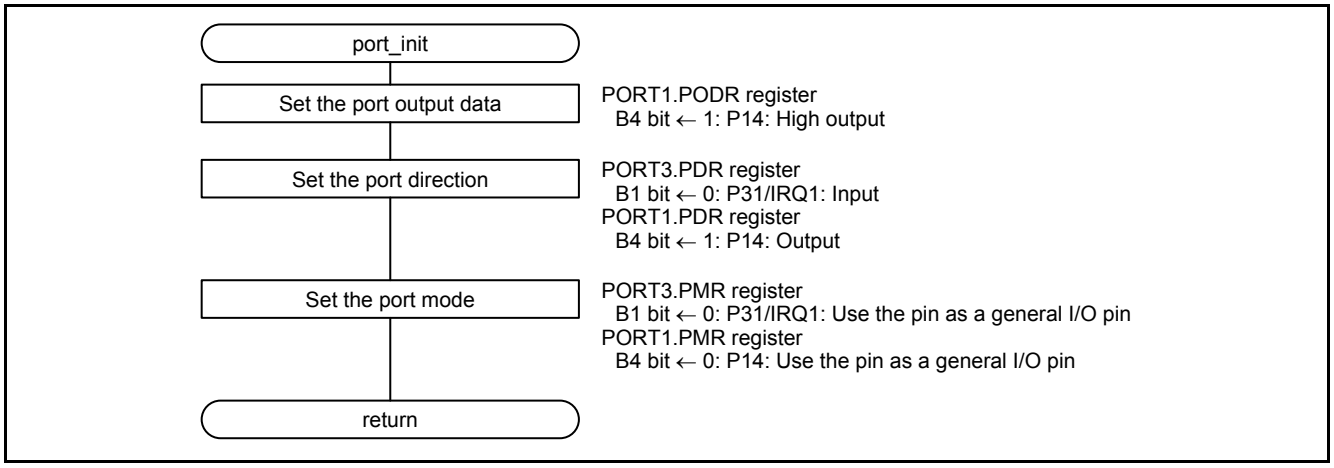
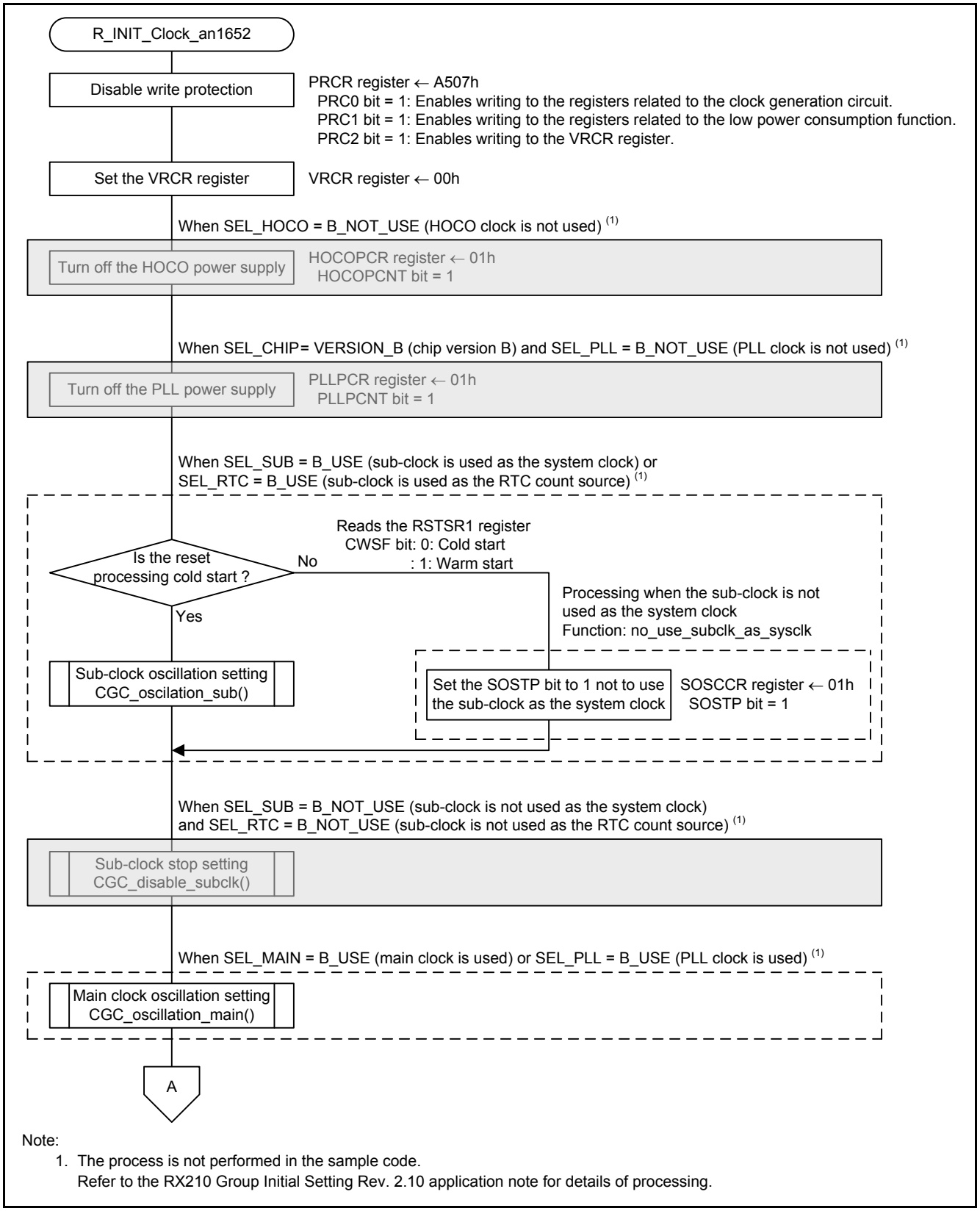



**Figure 5.4 Port Initialization**

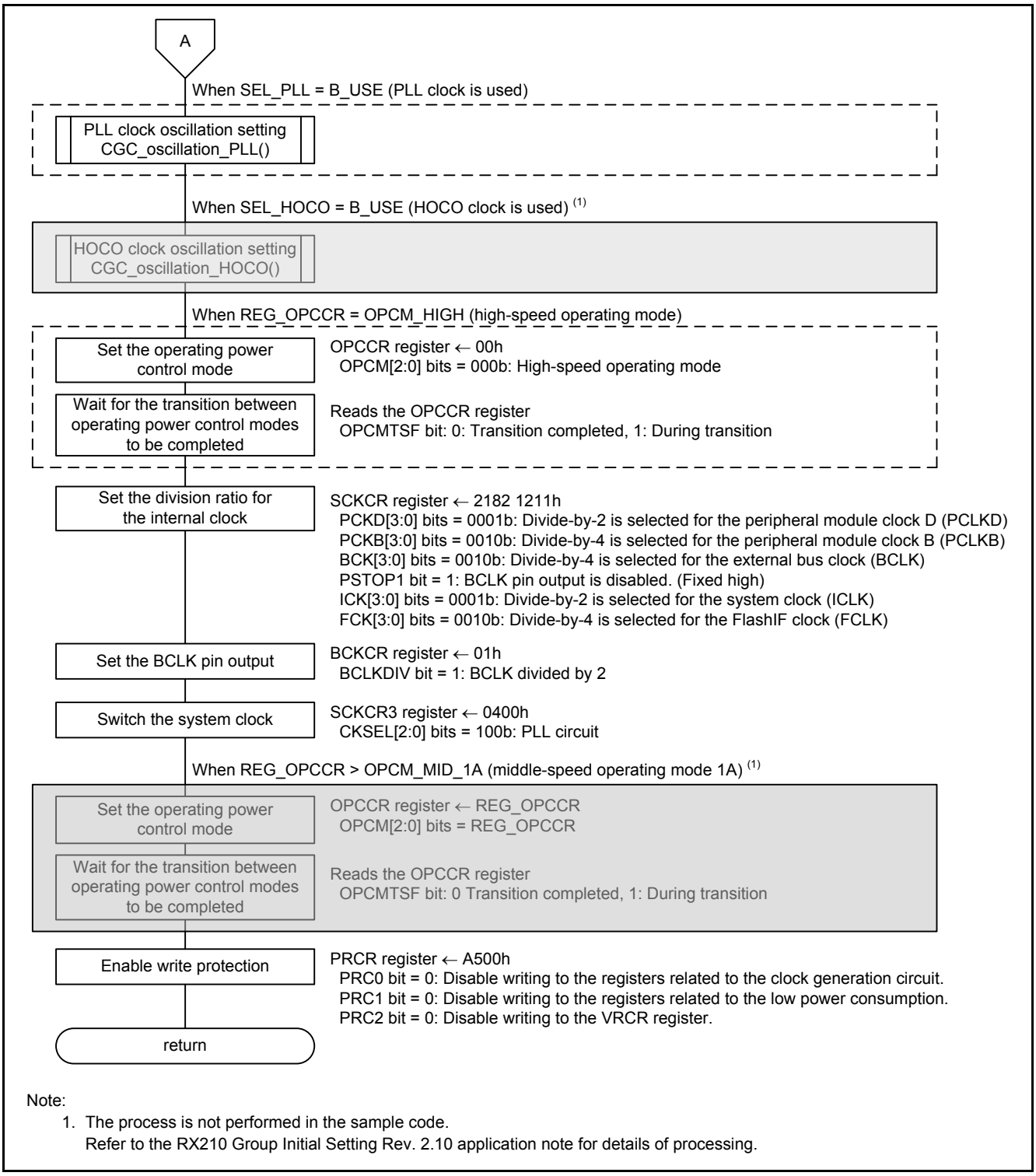
### 5.9.3 Clock Initialization

Figure 5.5 and Figure 5.6 show the Clock Initialization.




Note:
1. The process is not performed in the sample code.
   Refer to the RX210 Group Initial Setting Rev. 2.10 application note for details of processing.

**Figure 5.5 Clock Initialization (1/2)**

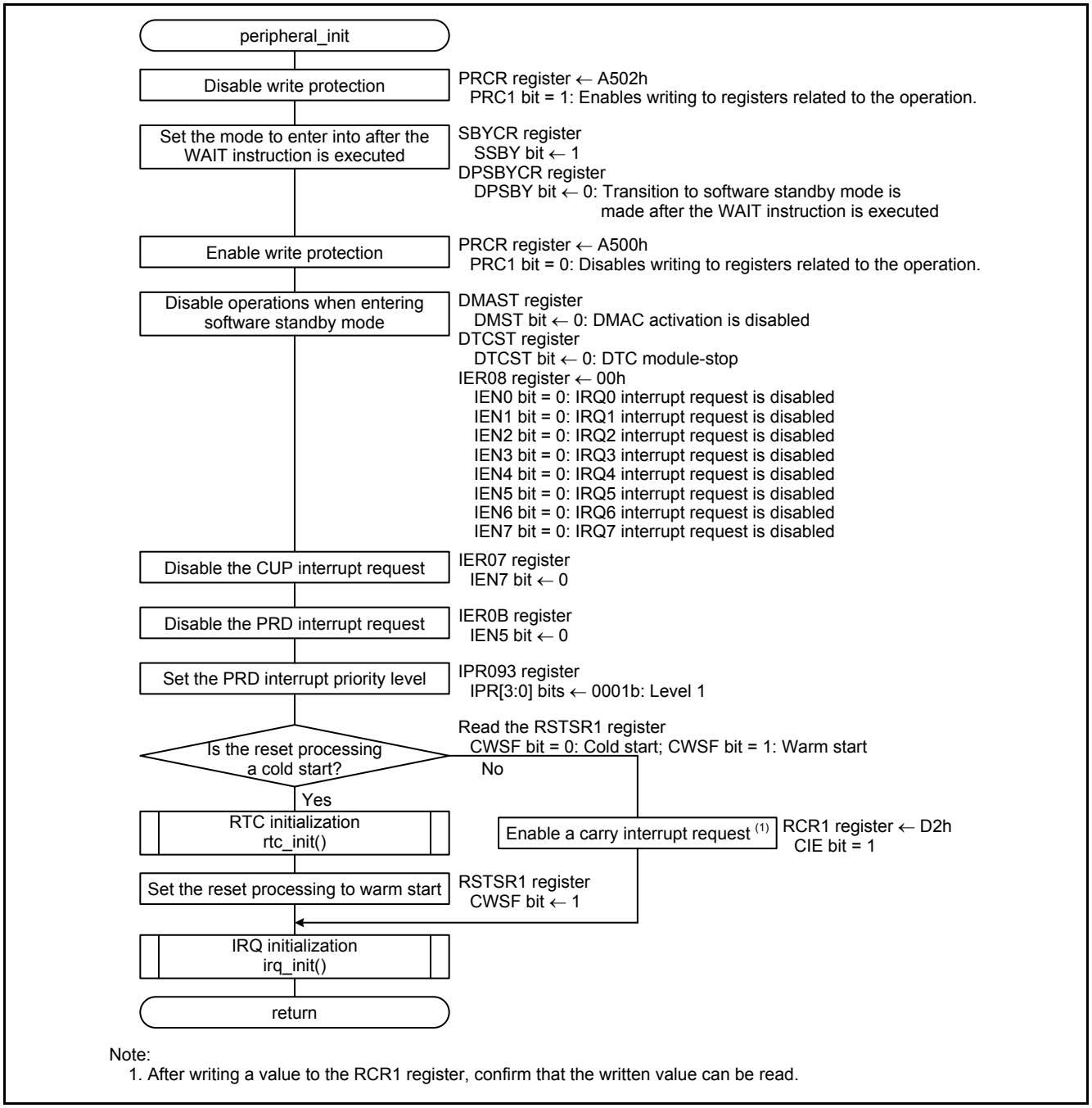



Note:
1. The process is not performed in the sample code.
   Refer to the RX210 Group Initial Setting Rev. 2.10 application note for details of processing.

**Figure 5.6 Clock Initialization (2/2)**

### 5.9.4 Peripheral Function Initialization

Figure 5.7 shows Peripheral Function Initialization.




Note:
1. After writing a value to the RCR1 register, confirm that the written value can be read.

**Figure 5.7 Peripheral Function Initialization**

### 5.9.5 IRQ Initialization

Figure 5.8 shows IRQ Initialization.

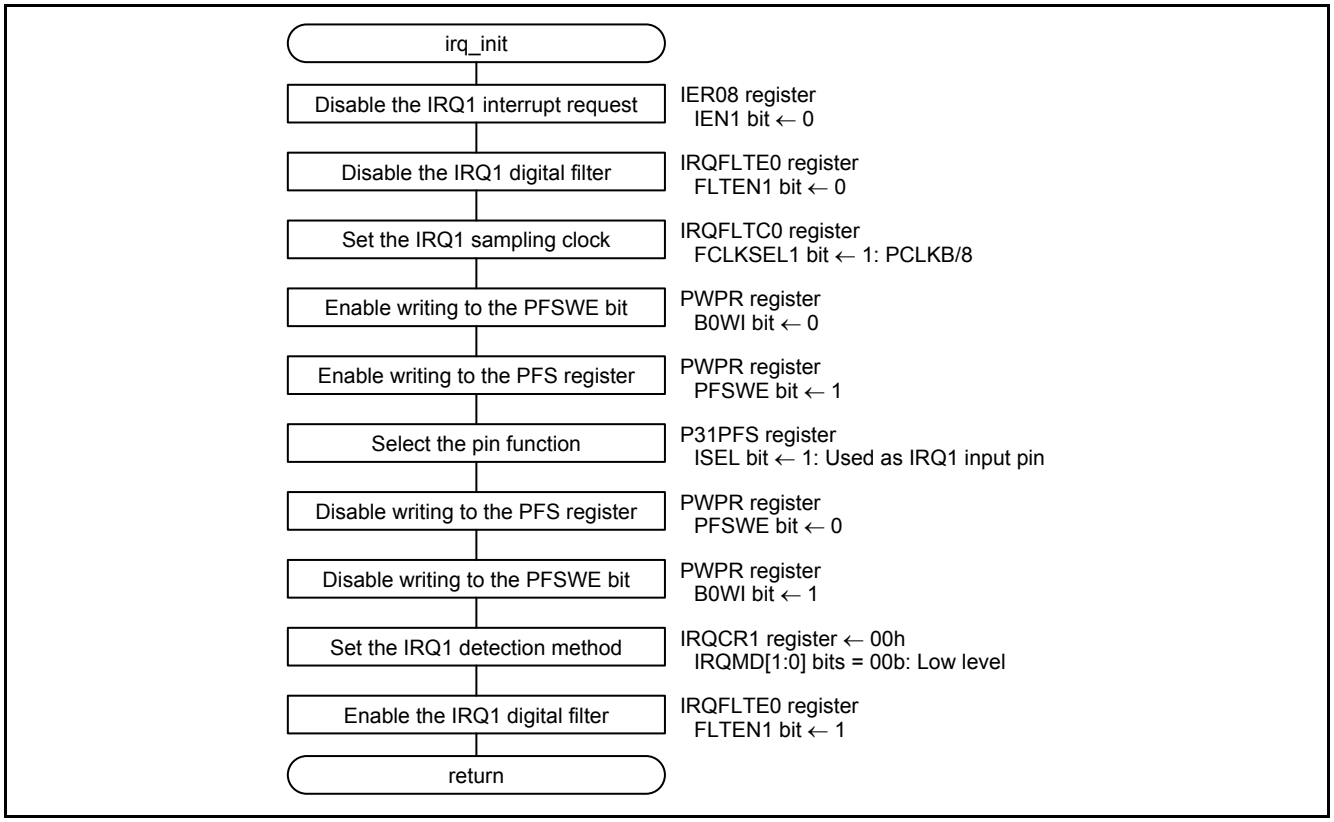



**Figure 5.8 IRQ Initialization**

### 5.9.6 RTC Initialization

Figure 5.9 shows RTC Initialization.

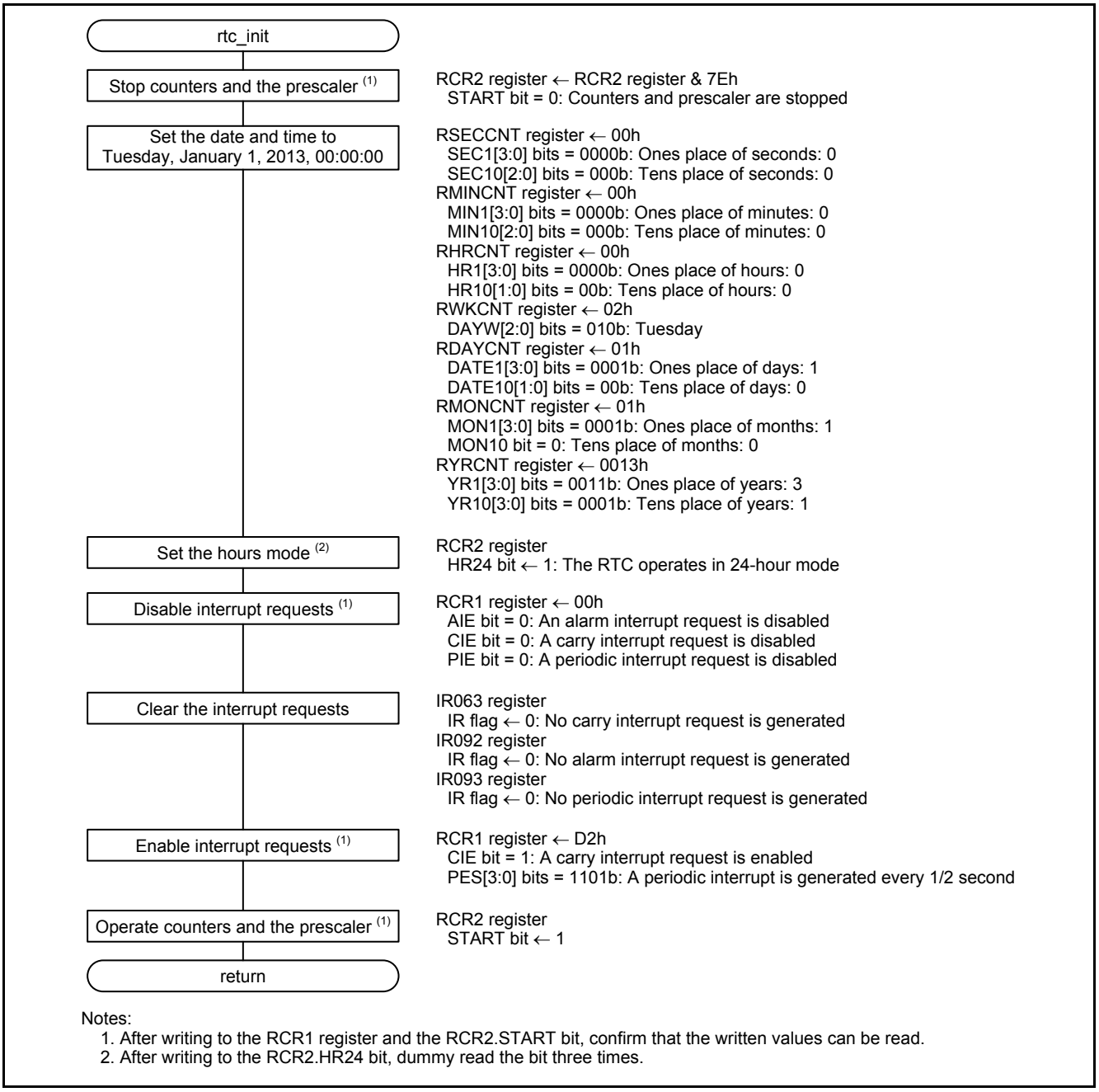



**Figure 5.9 RTC Initialization**

### 5.9.7 Reading RTC Time Information

Figure 5.10 shows Reading RTC Time Information.

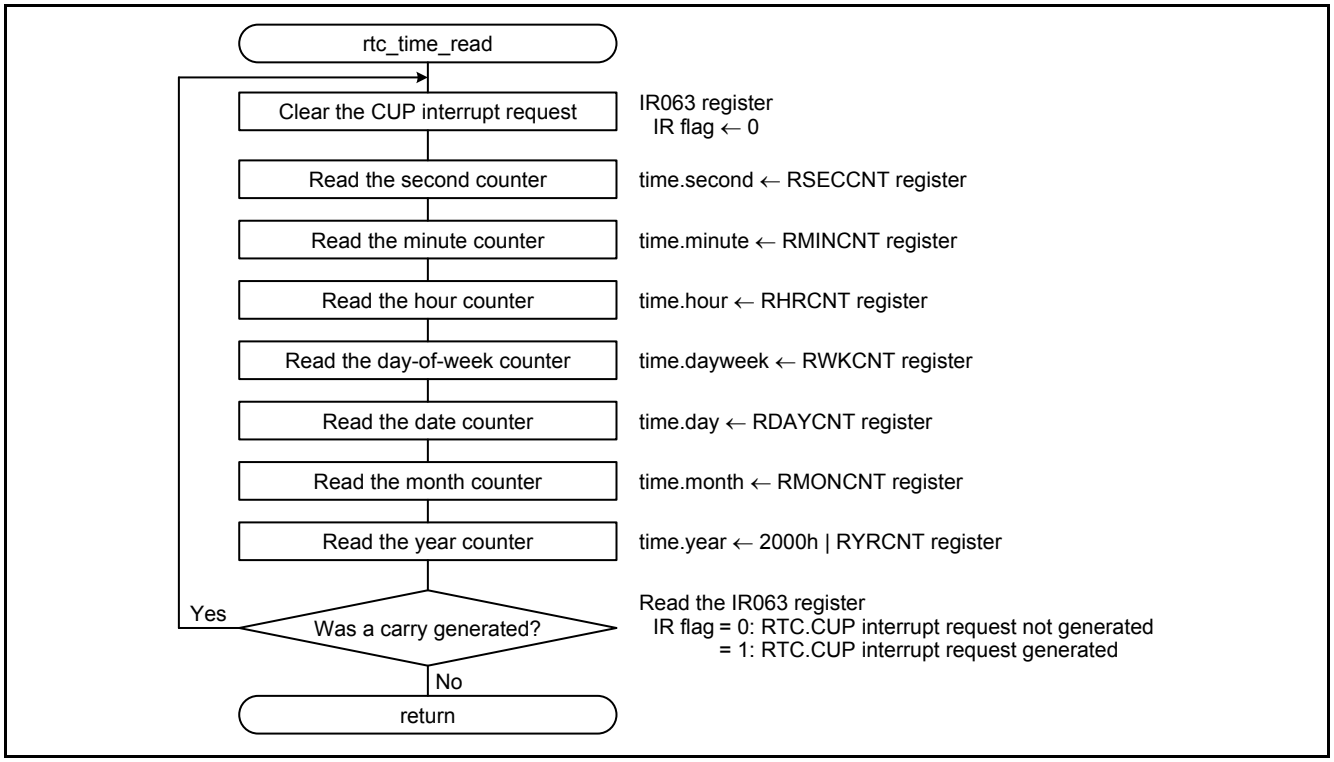



**Figure 5.10 Reading RTC Time Information**

### 5.9.8 Preparation to Enter Software Standby Mode

Figure 5.11 shows Preparation to Enter Software Standby Mode.

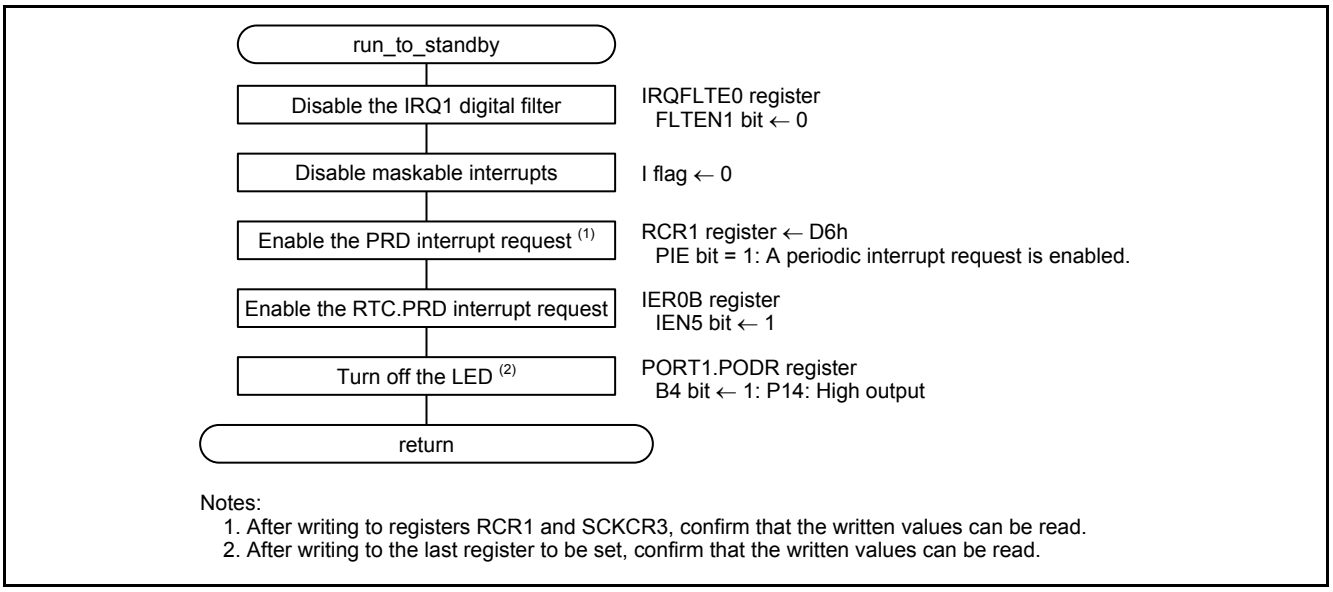



**Figure 5.11 Preparation to Enter Software Standby Mode**

### 5.9.9 Exiting Software Standby Mode

Figure 5.12 shows Exiting Software Standby Mode.

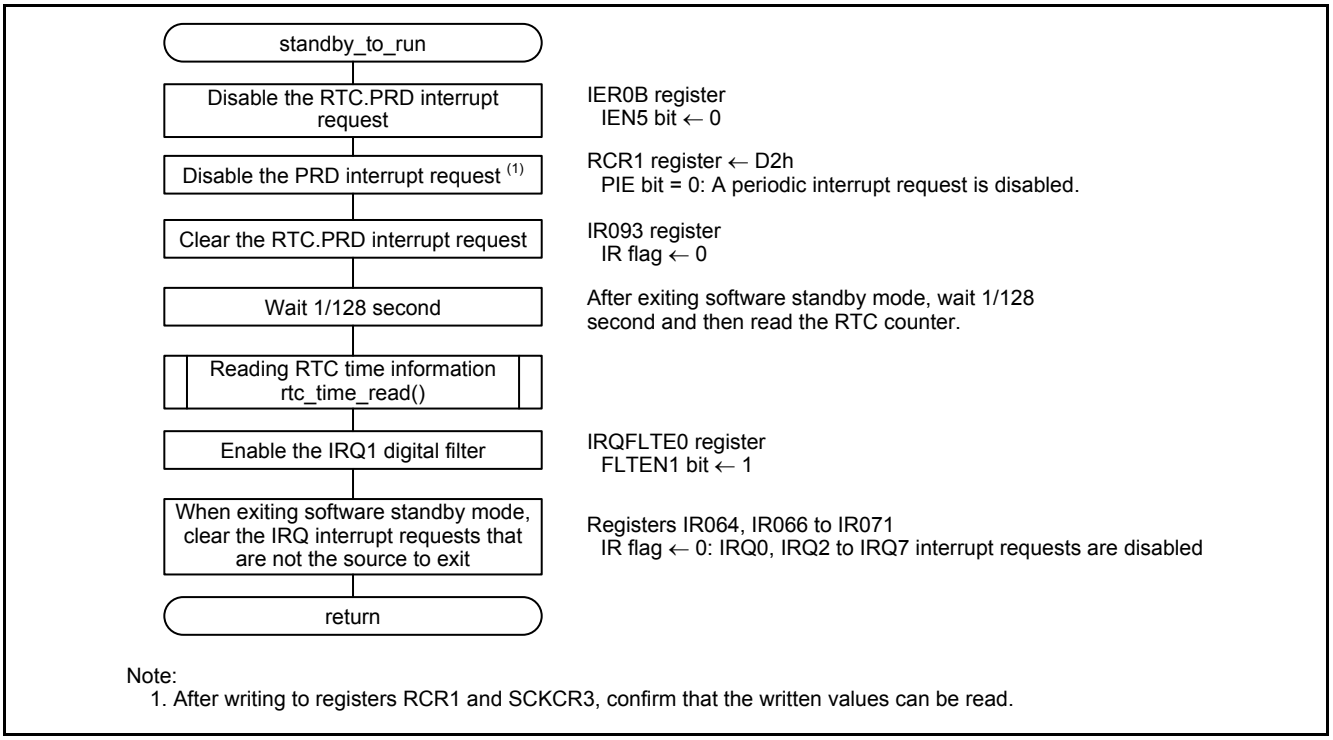



**Figure 5.12 Exiting Software Standby Mode**

### 5.9.10 PRD Interrupt Handling

Figure 5.13 shows PRD Interrupt Handling.

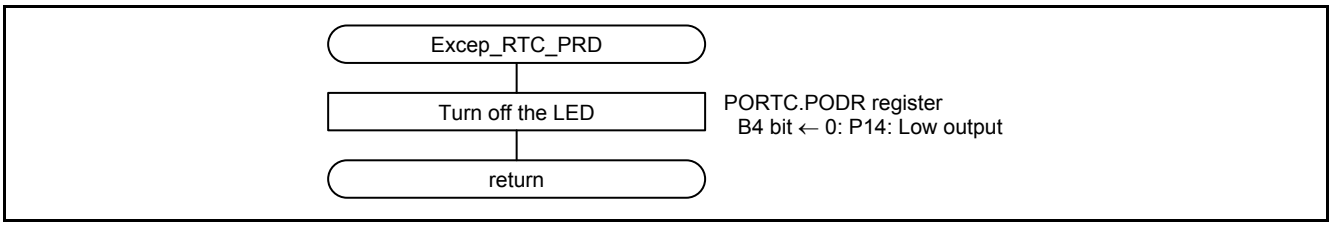



**Figure 5.13 PRD Interrupt Handling**

## 6. Applying This Application Note to the RX21A or RX220 Group

The sample code accompanying this application note has been confirmed to operate with the RX210 Group. To make the sample code operate with the RX21A or RX220 Group, use this application note in conjunction with the Initial Setting application note for each group.

For details on using this application note with the RX21A and RX220 Groups, refer to “5. Applying the RX210 Group Application Note to the RX21A Group” in the RX21A Group Initial Setting application note, and “4. Applying the RX210 Group Application Note to the RX220 Group” in the RX220 Group Initial Setting application note.

Note: • The r_init_clock.h file will be overwritten when applying the Initial Setting for the product used. Make the settings in the overwritten file be same as the original settings in the r_init_clock.h file accompanying this application note.

## 7. Sample Code

Sample code can be downloaded from the Renesas Electronics website.

## 8. Reference Documents

User's Manual: Hardware

RX210 Group User's Manual: Hardware Rev.1.50 (R01UH0037EJ)
RX21A Group User's Manual: Hardware Rev.1.00 (R01UH0251EJ)
RX220 Group User's Manual: Hardware Rev.1.10 (R01UH0292EJ)
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.

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 | RX210, RX21A, and RX220 Groups Application Note<br>Exiting Software Standby Mode Using the RTCb |
|---|---|

| Rev. | Date | Description | |
|---|---|---|---|
| | | Page | Summary |
| 1.00 | Dec. 16, 2013 | — | First edition issued |
| 1.01 | July 1, 2014 | 1 | Products: Added the RX21A and RX220 Groups. |
| | | 4 | 3. Reference Application Notes: Added the Initial Setting application notes for the RX21A and RX220 Groups. |
| | | 10, 11 | Modified the description of reference application note in the following functions: R_INIT_StopModule, R_INIT_NonExistentPort, and R_INIT_Clock. |
| | | 23 | 6. Applying This Application Note to the RX21A or RX220 Group: Added. |
| | | 24 | 8. Reference Documents: Added the User's Manual: Hardware for the RX21A and RX220 Groups. |
| | | | |

## General Precautions in the Handling of MPU/MCU Products

The following usage notes are applicable to all MPU/MCU 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.

   — 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.

   — 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.

   — 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.

   — 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.

   — The characteristics of an MPU or MCU 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**
1101 Nicholson Road, Newmarket, Ontario L3Y 9C3, Canada
Tel: +1-905-898-5441, Fax: +1-905-898-3220

**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 1601-1613, 16/F., Tower 2, Grand Century Place, 193 Prince Edward Road West, Mongkok, Kowloon, Hong Kong
Tel: +852-2265-6688, Fax: +852 2886-9022/9044

**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 906, 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 Korea Co., Ltd.**
12F., 234 Teheran-ro, Gangnam-Ku, Seoul, 135-920, Korea
Tel: +82-2-558-3737, Fax: +82-2-558-5141





Colophon 4.0