RX210 and RX220 Groups

Monitoring the Voltage Using the DOC and S12AD

R01AN1003EJ0101
Rev. 1.01
July 1, 2014

Abstract

This document describes how to use the RX210 and RX220 Groups' data operation circuit (DOC) and the 12-bit A/D converter (S12AD) to exit sleep mode when the voltage input to an analog input pin is at least the reference voltage.

Products

RX210 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 MCU operates in normal operating mode (program execution state) after a reset.

Voltage input to an analog input pin is A/D converted in 100 ms intervals. The A/D conversion result and reference value are compared (subtracted) by the DOC to see if the conversion result is the same value as the reference value or higher.

The MCU enters sleep mode by pushing a switch. As with normal operating mode, A/D conversion is performed in 100 ms intervals. The MCU exits sleep mode when the A/D conversion result is the same value as the reference value or higher.

- Sampling interval for the analog input pin: 100 ms

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
MTU2a channel 0 (MTU0)	Activation source (100 ms interval) for the S12AD
S12AD	Measures voltage on the analog input pin
DOC	Determines the reference value for the measured voltage
Data transfer controller (DTC)	Specifies the reference value for the measured voltage and transfers the S12AD conversion result

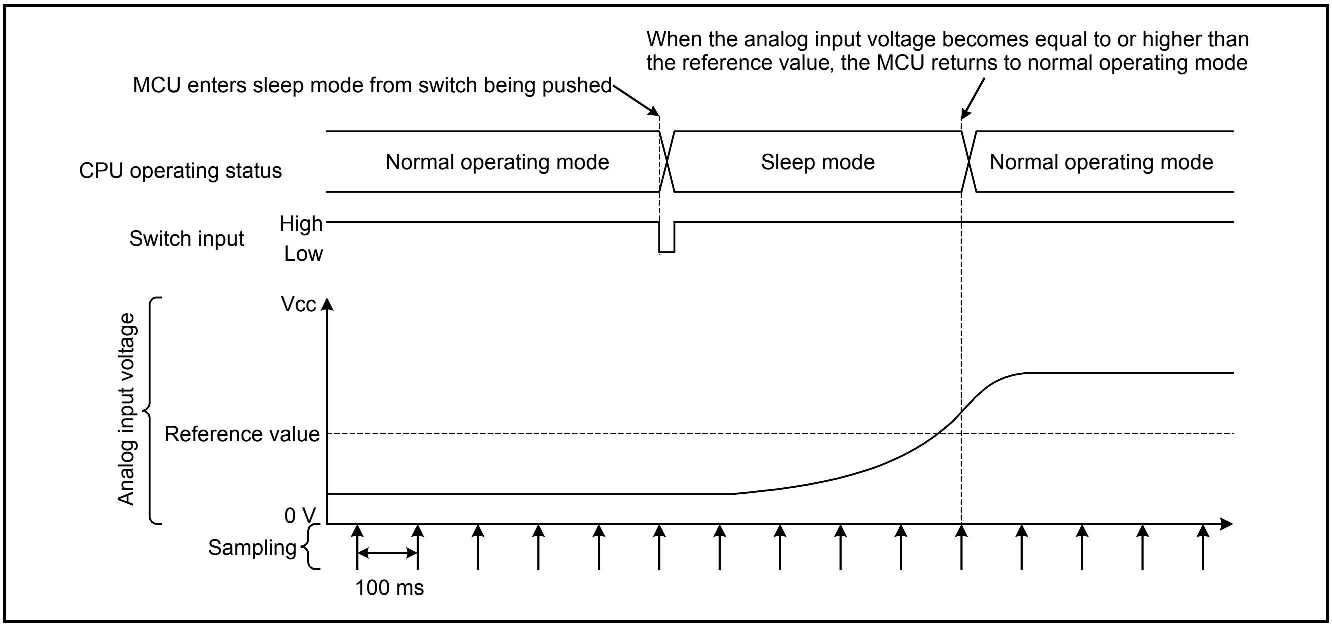


Figure 1.1 Operation Overview

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	R5F52108ADFP (RX210 Group)
Operating frequencies	• HOCO clock: 32 MHz • System clock (ICLK): 16 MHz (HOCO clock divided by 2) • Peripheral module clock B (PCLKB): 1 MHz (HOCO clock divided by 32) • Peripheral module clock D (PCLKD): 1 MHz (HOCO clock divided by 32)
Operating voltage	5.0 V
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=rx200 –output=obj="$(CONFIGDIR)\$(FILELEAF).obj" -debug -nologo The default setting is used in the integrated development environment.
iodefine.h version	Version 1.2A
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 number: R0K505210C000BE)

3. Reference Application Notes

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

- RX210 Group Initial Setting Rev. 2.00 (R01AN1002EJ)
- 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 Pins Used

Table 4.1 lists the Pins Used and Their Functions. This table assumes the 100-pin product is used. When using products with less than 100 pins, select the pins appropriate to the product used.

Table 4.1 Pins Used and Their Functions

Pin Name	I/O	Function
P14	Output	Output to show LED operation (LED is on in normal operating mode and off in sleep mode)
P31/IRQ1	Input	Input for SW1 (the input switch to enter sleep mode)
P40/AN000	Input	Analog input pin

5. Software

The MCU operates in normal operating mode after a reset.

MTU0 is used to trigger A/D conversion, and input to the analog input pin is A/D converted in 100 ms intervals. After A/D conversion is performed, the DOC data subtraction mode is used and the reference value is compared to the A/D converted result.

If the result of data subtraction is 0000h or higher (DOCR.DOPCF flag is 0), the input level to the analog input pin is determined to be the reference value or lower. If the result of the data subtraction is less than 0000h (DOCR.DOPCF flag is 1), the input level to the analog input pin is determined to be higher than the reference value.

When SW1 (falling edge of IRQ1) is input, enable the DOC interrupt. Next, execute the WAIT instruction and enter sleep mode.

Even in sleep mode, A/D conversion is performed on input to the analog input pin in 100 ms intervals just as in normal operating mode. When the input level to the analog input pin is higher than the reference value (when the DOCR.DOPCF flag becomes 1), a DOC interrupt request is generated, and the MCU exits sleep mode.

After exiting sleep mode, the MCU operates in normal operating mode.

Settings for the peripheral functions are listed below.

MTU0
TGIA0 interrupt period: 100 ms
Count clock: PCLKB divided by 4
TGIA0 interrupt: Enabled (used as a DTC activation source)

S12AD
Input channel: AN000
A/D conversion clock: 1 MHz (ADCLK = PCLKD)
Operating mode: Single scan mode
A/D conversion start condition: MTU0.TGRA compare match
A/D scan end interrupt: Enabled (used as a DTC activation source)
Sample-and-hold function: Used
A/D conversion time: Approximately 90 µs

DOC
Operating mode: Data subtraction mode
Data operation circuit interrupt: Disabled in normal operating mode and enabled in sleep mode
Interrupt priority level: Level 1

DTC

Activation source: MTU0.TGIA0 interrupt request or A/D scan end interrupt request
Address mode: Full-address mode

- DTC transfer setting when the activation source is the TGIA0 interrupt request
  Transfer mode: Repeat transfer
  Transfer source addressing mode: SAR register is set to address-fixed mode
  Transfer source address: RAM (storage destination for data operation reference value)
  Transfer destination addressing mode: DAR register is set to address-fixed mode
  Transfer destination address: DOC.DODSR register
  Data transfer length: 16 bits
  Chain transfer: Disabled

- DTC transfer setting when the activation source is the A/D scan end interrupt request
  Transfer mode: Repeat transfer
  Transfer source addressing mode: SAR register is set to address-fixed mode
  Transfer source address: S12AD.ADDR0 register
  Transfer destination addressing mode: DAR register is set to address-fixed mode
  Transfer destination address: DOC.DODIR register
  Data transfer length: 16 bits
  Chain transfer: Disabled

External pin interrupt: IRQ1

IRQ1 detection setting: Falling edge
Digital filter: Enabled
Digital filter sampling: PCLKB

5.1 Operation Overview

(1) After a reset, the MCU operates in normal operating mode.

(2) 12-bit A/D conversion starts when an MTU0.TGIA0 interrupt request is generated. Also, DTC transfer is used to write the reference value (0800h) to the DODSR register.

(3) 12-bit A/D conversion is completed, an A/D scan end interrupt request is generated, and DTC transfer is used to write the A/D conversion result to the DODIR register.

(4) If the A/D conversion result is lower than the reference value (0800h), the operation result from the DOC will not underflow and the DOCR.DOPCF flag will not change.

(5) Perform the 12-bit A/D conversion again. If the conversion result is higher than the reference value (0800h), the operation result from the DOC will underflow and the DOCR.DOPCF flag becomes 1.

(6) When a falling edge is detected on the IRQ1 pin, the DOC interrupt is enabled.

(7) Execute the WAIT instruction and the MCU enters sleep mode.

(8) As with operations in normal operating mode, 12-bit A/D conversion starts when the TGIA0 interrupt request is generated. At the same time, DTC transfer is used to write the reference value to the DODSR register.

(9) If the A/D conversion result is lower than the reference value, the operation result from the DOC will not underflow and the MCU will remain in sleep mode.

(10) If the A/D conversion result is the reference value or higher, the operation result from the DOC underflows, a DOC interrupt request is generated, and the MCU exits sleep mode.

(11) After the MCU exits sleep mode, the DOC interrupt is disabled.

Figure 5.1 shows the Operation Timing Diagram.

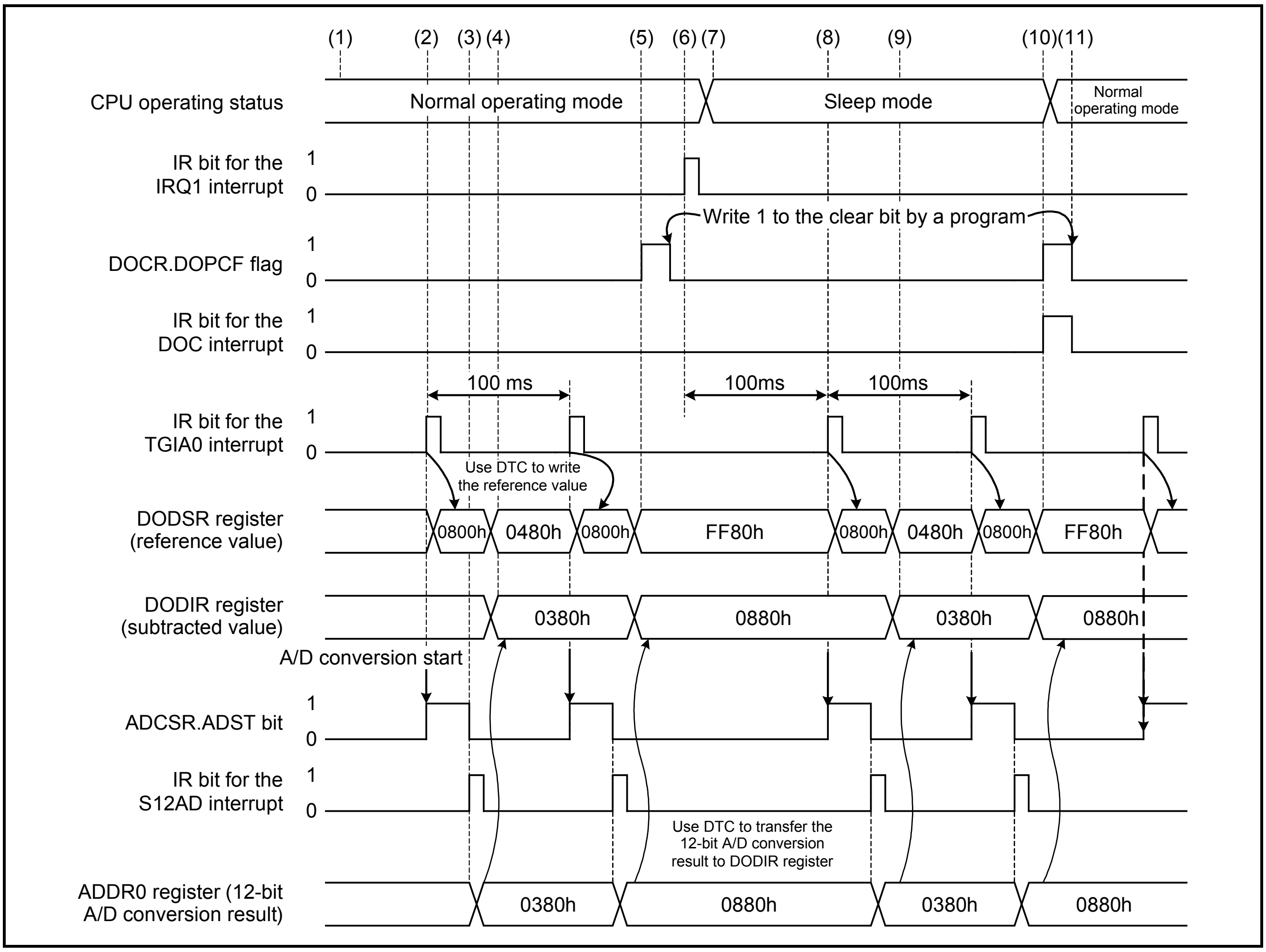


Figure 5.1 Operation Timing Diagram

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. 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_SLEEP	IR(ICU,IRQ1)	IR flag for SW1
SW_ON	1	Switch input on
SW_OFF	0	Switch input off
LED_RUN	PORT1.PODR.BIT.B4	Output data store bit for the LED output pin
LED_RUN_PDR	PORT1.PDR.BIT.B4	I/O select bit for the LED output pin
LED_ON	0	LED on
LED_OFF	1	LED off
WAKE_LEVEL	0800h	Voltage level to exit sleep mode (VREFH0 ÷ 2)
ADDR_DTC_VECT_TOP	FFFF F000h	Start address of the DTC vector table
ADDR_DTC_VECT_S12AD	FFFF F198h	DTC vector address: S12AD
ADDR_DTC_VECT_TGIA0	FFFF F1C8h	DTC vector address: MTU0.TGIA0
WAIT_ADCLK_2	32	Wait two cycles of the ADCLK: (ICLK ÷ ADCLK) × 2 = 32

5.5 Structure/Union List

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

The bit field members are allocated from the upper bits of the #pragma bit field order function. Refer to the RX Family C/C++ Compiler User's Manual for details.

```
/* **** DTC transfer information **** */
#pragma bit_order    left          /* Bit field order: The bit field members are allocated from upper bits */
#pragma unpack                     /* Boundary alignment value for structure members: Alignment by member type */
struct st_dtc_full
{
    union
    {
        uint32_t  LONG;
        struct
        {
            uint32_t  MRA_MD      :2;
            uint32_t  MRA_SZ      :2;
            uint32_t  MRA_SM      :2;
            uint32_t              :2;
            uint32_t  MRB_CHNE    :1;
            uint32_t  MRB_CHNS    :1;
            uint32_t  MRB_DISEL   :1;
            uint32_t  MRB_DTS     :1;
            uint32_t  MRB_DM      :2;
            uint32_t              :2;
            uint32_t              :16;
        } BIT;
    } MR;
    void * SAR;
    void * DAR;
    struct
    {
        uint32_t  CRA:16;
        uint32_t  CRB:16;
    } CR;
};
#pragma packoption                 /* End of specification for the boundary alignment value for structure members */
#pragma bit_order                  /* End of specification for the bit field order */
```

Figure 5.2 Structure/Union Used in the Sample Code

5.6 Variables

Table 5.4 lists the Global Variables and Table 5.5 lists the const Variables.

Table 5.4 Global Variables

Type	Variable Name	Contents	Function Used
struct st_dtc_full	s12ad_dtc_tbl	DTC transfer information for S12AD	dtc_init
struct st_dtc_full	tgia0_dtc_tbl	DTC transfer information for MTU0.TGIA0	dtc_init
uint16_t	doc_init_data	DOC initial value	dtc_init, doc_init

Table 5.5 const Variables

Type	Variable Name	Contents	Function Used
const void *	s12ad_dtc_vector	DTC vector for S12AD	None
const void *	tgia0_dtc_vector	DTC vector for MTU0.TGIA0	None

5.7 Functions

Table 5.6 lists the Functions.

Table 5.6 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	Clock initialization
peripheral_init	Peripheral function initialization
lpc_init	LPC initialization
mpc_init	MPC initialization
mtu_init	MTU initialization
s12ad_init	S12AD initialization
doc_init	DOC initialization
dtc_init	DTC initialization
irq_init	IRQ initialization
run_to_sleep	Settings prior to entering sleep mode
sleep_to_run	Settings after exiting sleep mode
Excep_DOC_DOPCF	DOC.DOPCF interrupt handling

5.8 Function Specifications

The tables below list the sample code function specifications.

main

Outline	Main processing
Header	None
Declaration	void main(void)
Description	After the initial setting, the MTU0 count starts. When an IRQ1 falling edge is detected, the MCU enters sleep 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	Performs settings 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 100 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. For details on this function, refer to the Initial Setting application note for the product used.

R_INIT_Clock

Outline	Clock initialization
Header	r_init_clock.h
Declaration	void R_INIT_Clock(void)
Description	Initializes the clock.
Arguments	None
Return Value	None
Remarks	The sample code selects processing which uses HOCO as the system clock and uses middle-speed operating mode 1A as the operating power control mode without using the main clock, PLL, or sub-clock. 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	void peripheral_init(void)
Description	Performs settings to initialize the peripheral functions used.
Arguments	None
Return Value	None

lpc_init

Outline	LPC initialization
Header	None
Declaration	void lpc_init(void)
Description	Initializes the low power consumption functions.
Arguments	None
Return Value	None

mpc_init

Outline	MPC initialization
Header	None
Declaration	void mpc_init(void)
Description	Allocates input pins to the peripheral functions used.
Arguments	None
Return Value	None

mtu_init

Outline	MTU initialization
Header	None
Declaration	void mtu_init(void)
Description	Performs settings to initialize the MTU.
Arguments	None
Return Value	None

s12ad_init

Outline	S12AD initialization
Header	None
Declaration	void s12ad_init(void)
Description	Performs settings to initialize the S12AD.
Arguments	None
Return Value	None

doc_init

Outline	DOC initialization
Header	None
Declaration	void doc_init(void)
Description	Performs settings to initialize the DOC.
Arguments	None
Return Value	None

dtc_init

Outline	DTC initialization
Header	None
Declaration	void dtc_init(void)
Description	Performs settings to initialize the DTC.
Arguments	None
Return Value	None

irq_init	
Outline	IRQ initialization
Header	None
Declaration	void irq_init(void)
Description	Performs settings to initialize the IRQ.
Arguments	None
Return Value	None

run_to_sleep	
Outline	Settings prior to entering sleep mode
Header	None
Declaration	void run_to_sleep(void)
Description	Performs processing prior to entering sleep mode.
Arguments	None
Return Value	None

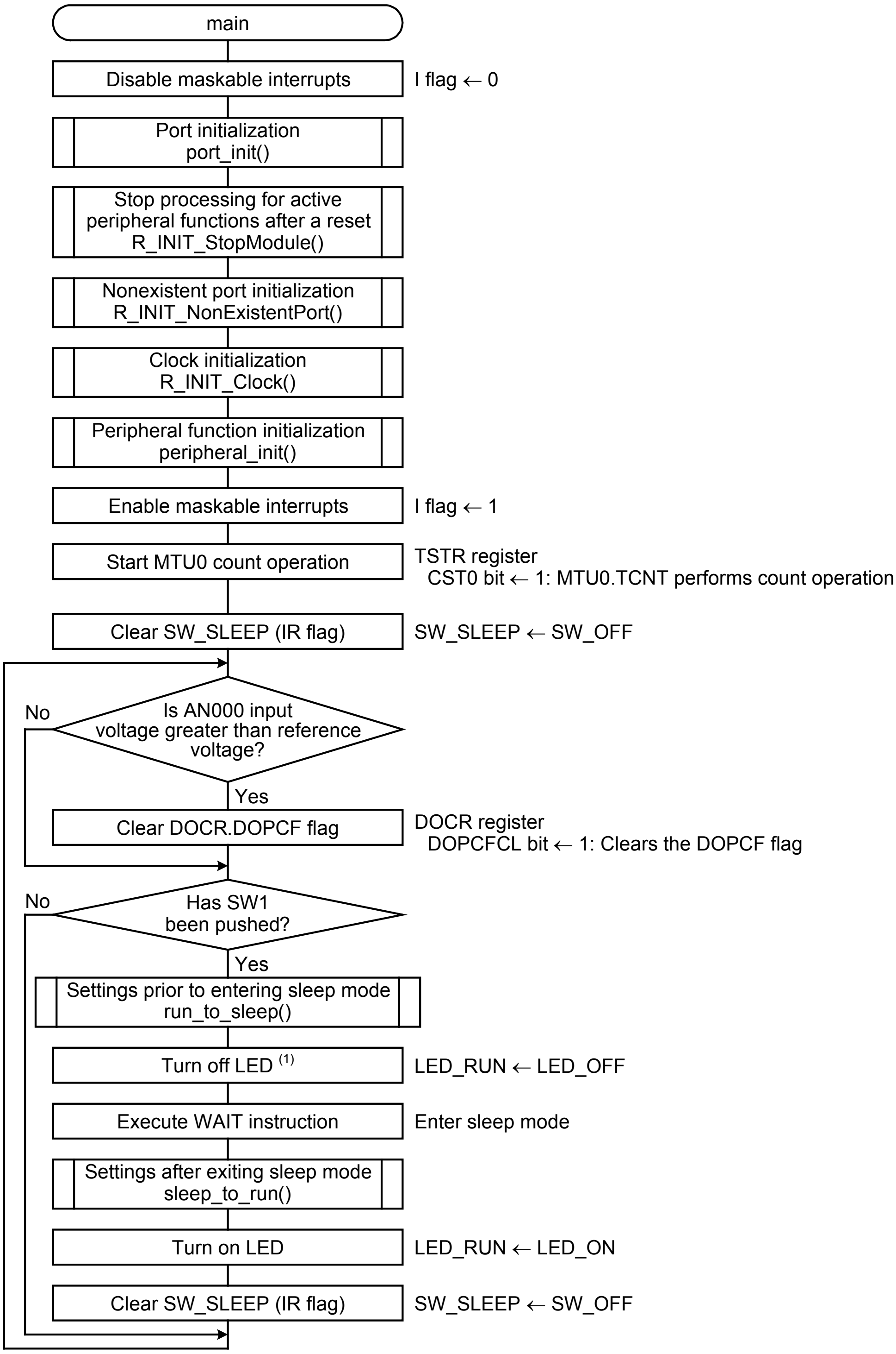
sleep_to_run	
Outline	Settings after exiting sleep mode
Header	None
Declaration	void sleep_to_run(void)
Description	Performs processing prior to exiting sleep mode.
Arguments	None
Return Value	None

Excep_DOC_DOPCF	
Outline	DOC.DOPCF interrupt handling
Header	None
Declaration	void Excep_DOC_DOPCF(void)
Description	Clears the DOCR.DOPCF flag.
Arguments	None
Return Value	None

5.9 Flowcharts

5.9.1 Main Processing

Figure 5.3 shows the Main Processing.



Note:
1. Read the value of the register that was written to immediately before executing the WAIT instruction and confirm that the written value can be read.

Figure 5.3 Main Processing

5.9.2 Port Initialization

Figure 5.4 shows the Port Initialization.

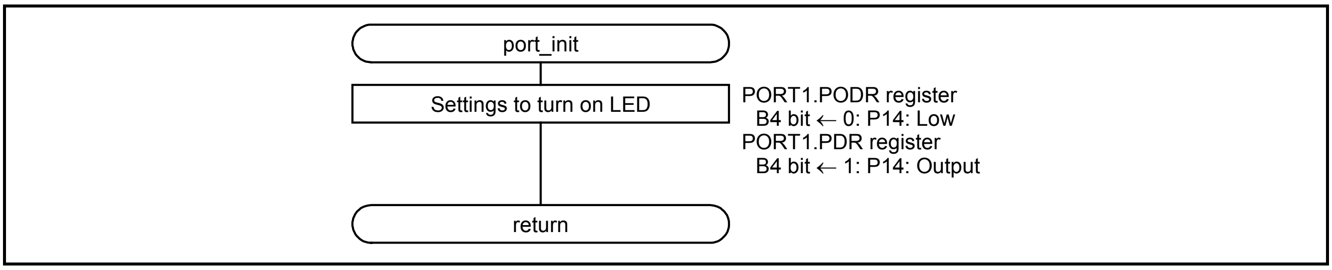


Figure 5.4 Port Initialization

5.9.3 Peripheral Function Initialization

Figure 5.5 shows the Peripheral Function Initialization.

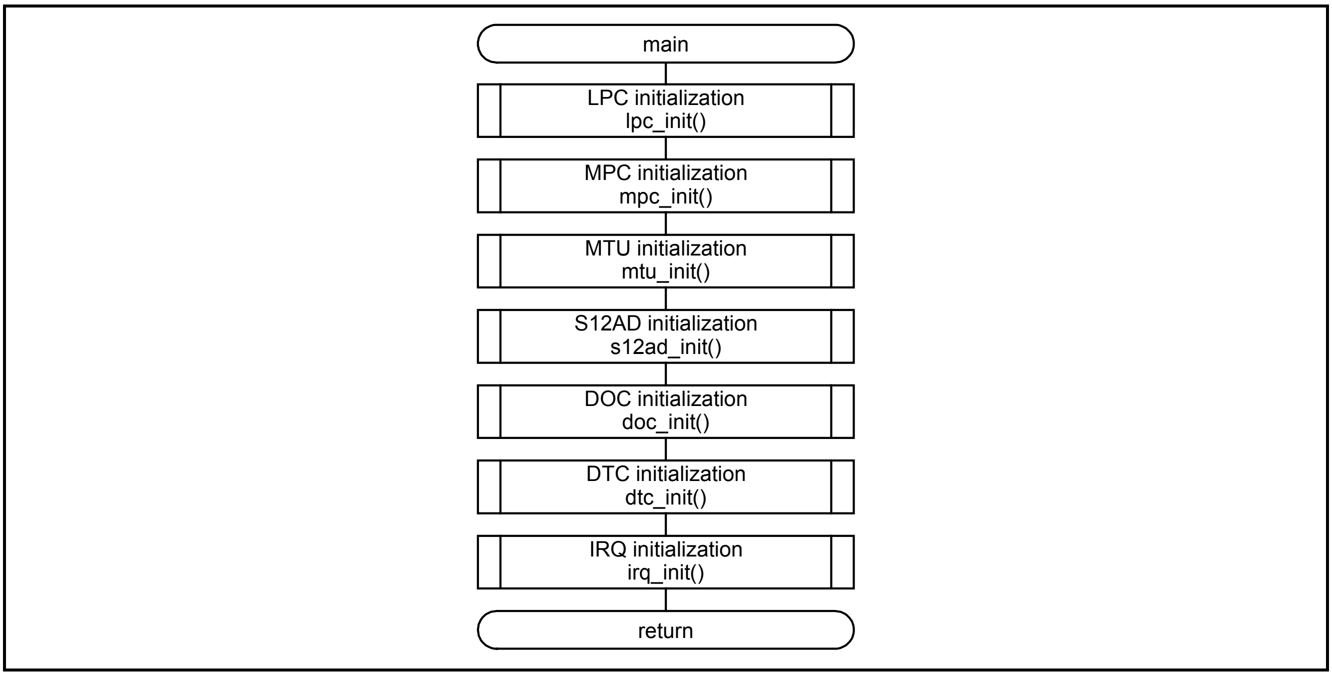


Figure 5.5 Peripheral Function Initialization

5.9.4 LPC Initialization

Figure 5.6 shows the LPC Initialization.

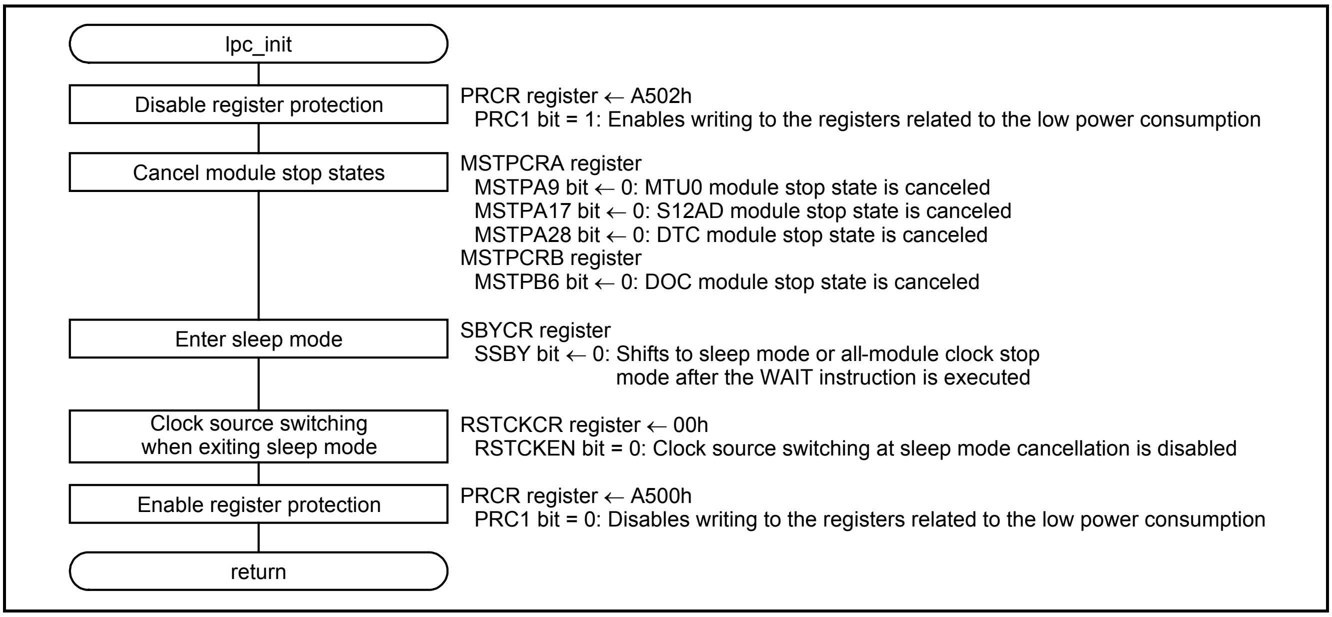


Figure 5.6 LPC Initialization

5.9.5 MPC Initialization

Figure 5.7 shows the MPC Initialization.

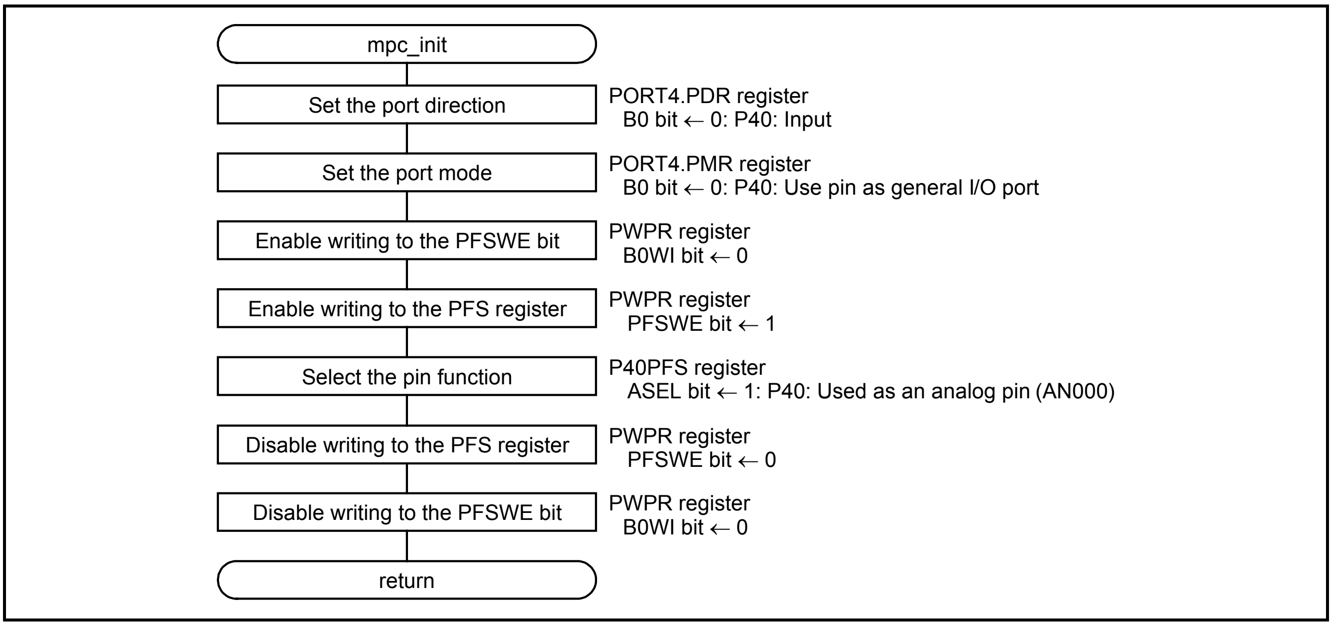


Figure 5.7 MPC Initialization

5.9.6 MTU Initialization

Figure 5.8 shows the MTU Initialization.

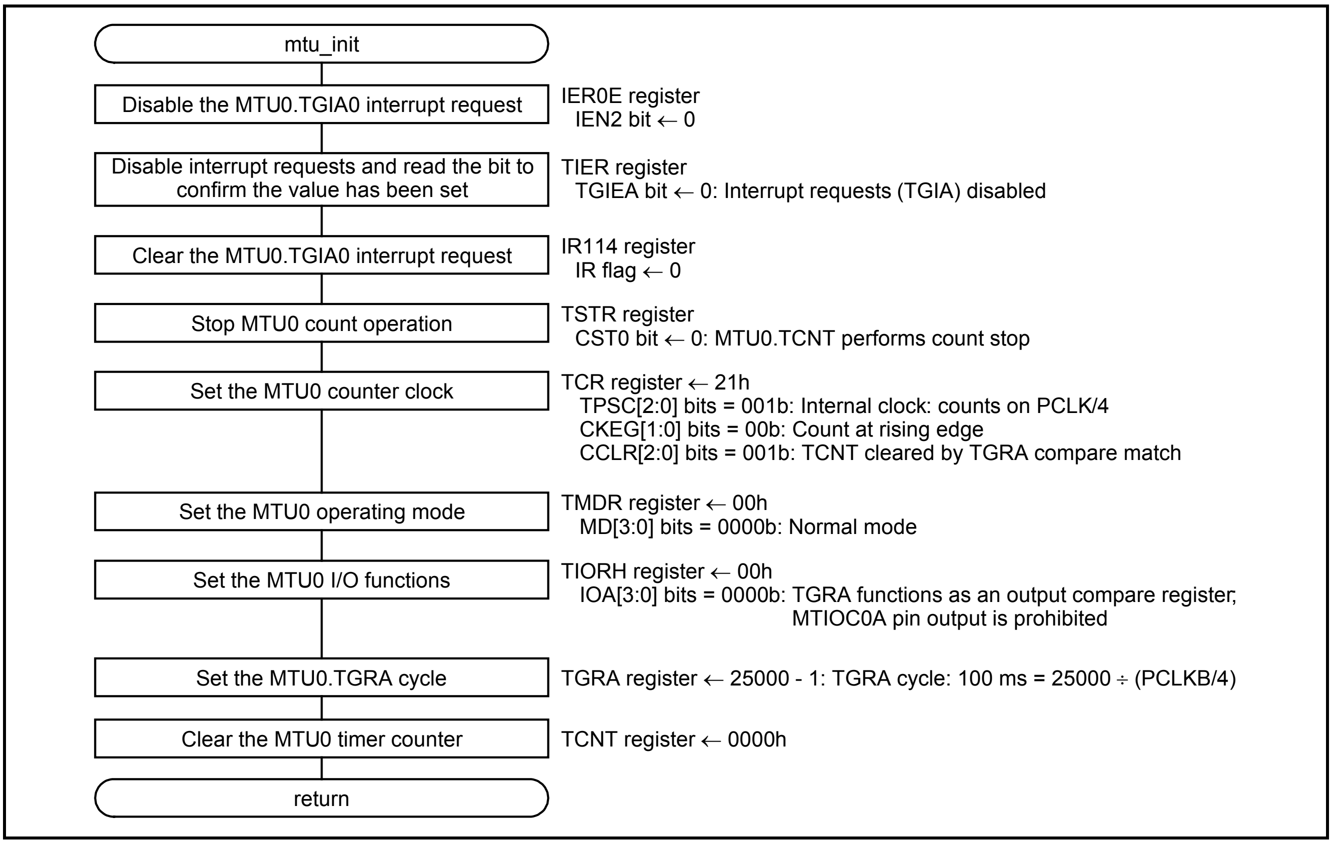


Figure 5.8 MTU Initialization

5.9.7 S12AD Initialization

Figure 5.9 shows the S12AD Initialization.

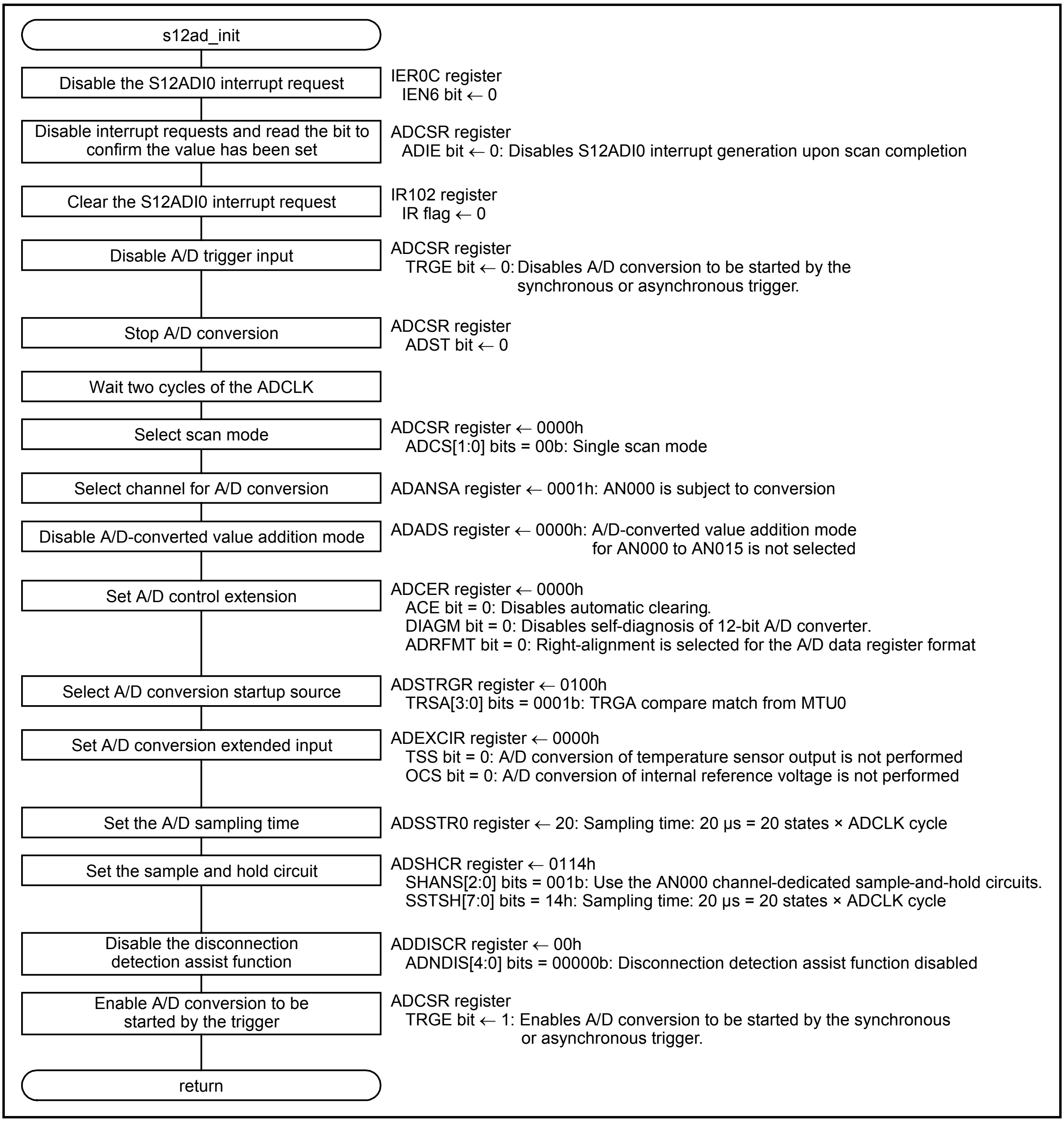


Figure 5.9 S12AD Initialization

5.9.8 DOC Initialization

Figure 5.10 shows the DOC Initialization.

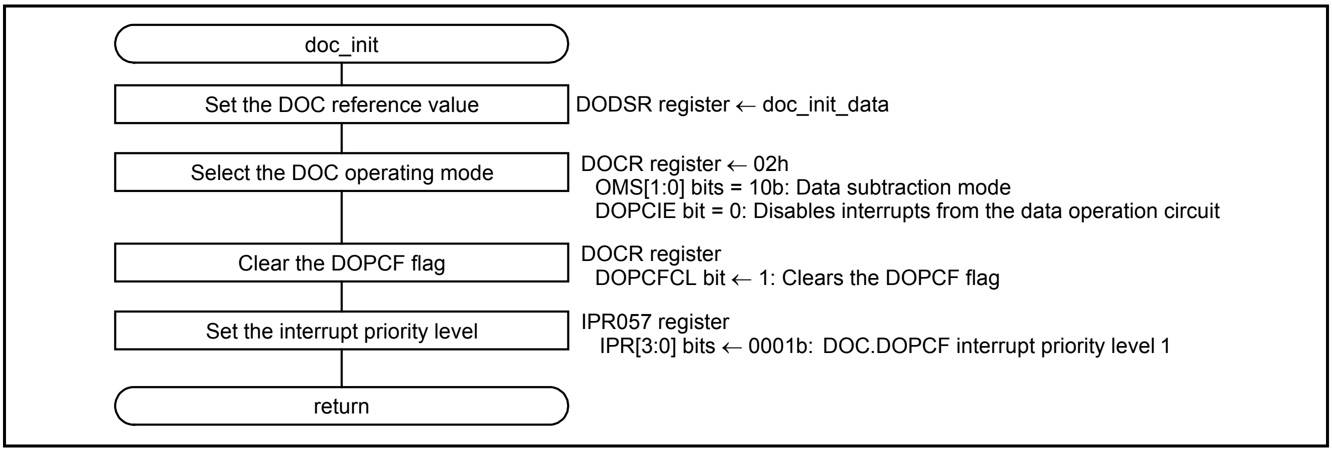


Figure 5.10 DOC Initialization

5.9.9 DTC Initialization

Figure 5.11 shows the DTC Initialization.

dtc_init	
Stop the DTC module	DTCST register ← 00h DTCST bit = 0: DTC module stop
Disable the read skip	DTCCR register RRS bit ← 0: Transfer data read is not skipped
Select the address mode	DTCADMOD register ← 00h SHORT bit = 0: Full-address mode
Set the DTC transfer information for MTU0	tgia0_dtc_tbl.MR.LONG ← 0000 0000h: Initialized by setting 0 tgia0_dtc_tbl.MR.BIT.MRA_MD ← 01b: Repeat transfer mode tgia0_dtc_tbl.MR.BIT.MRA_SZ ← 01b: Word transfer tgia0_dtc_tbl.MR.BIT.MRA_SM ← 00b: Address in the SAR register is fixed tgia0_dtc_tbl.MR.BIT.MRB_CHNE ← 0: Chain transfer is disabled tgia0_dtc_tbl.MR.BIT.MRB_DISEL ← 0: An interrupt request to the CPU is generated when specified data transfer is completed tgia0_dtc_tbl.MR.BIT.MRB_DTS ← 0: Transfer destination side is repeat area tgia0_dtc_tbl.MR.BIT.MRB_DM ← 00b: Address in the DAR register is fixed tgia0_dtc_tbl.SAR ← Address for doc_init_data tgia0_dtc_tbl.DAR ← Address for the DODSR register tgia0_dtc_tbl.CR.CRA ← 0101h: DTC transfer count is 1. tgia0_dtc_tbl.CR.CRB ← 0000h: Repeat transfer is not used
Set the DTC transfer information for S12AD	s12ad_dtc_tbl.MR.LONG ← 0000 0000h: Initialized by setting 0 s12ad_dtc_tbl.MR.BIT.MRA_MD ← 01b: Repeat transfer mode s12ad_dtc_tbl.MR.BIT.MRA_SZ ← 01b: Word transfer s12ad_dtc_tbl.MR.BIT.MRA_SM ← 00b: Address in the SAR register is fixed s12ad_dtc_tbl.MR.BIT.MRB_CHNE ← 0: Chain transfer is disabled s12ad_dtc_tbl.MR.BIT.MRB_DISEL ← 0: An interrupt request to the CPU is generated when specified data transfer is completed s12ad_dtc_tbl.MR.BIT.MRB_DTS ← 0: Transfer destination side is repeat area s12ad_dtc_tbl.MR.BIT.MRB_DM ← 00b: Address in the DAR register is fixed s12ad_dtc_tbl.SAR ← Address for the ADDR0 register s12ad_dtc_tbl.DAR ← Address for the DODIR register s12ad_dtc_tbl.CR.CRA ← 0101h: DTC transfer count is 1. s12ad_dtc_tbl.CR.CRB ← 0000h: Repeat transfer is not used
Set the DTC vector base address	DTCVBR register ← DTC vector start address
Select the address mode	DTCCR register RRS bit ← 1: Transfer data read is skipped when vector numbers match.
Enable DTC activation by TGIA0 interrupt	DTCER114 register DTCE bit ← 1: DTC activation is enabled by TGIA0 interrupt IER0E register IEN2 bit ← 1: TGIA0 interrupt request is enabled
Enable DTC activation by S12AD interrupt	DTCER102 register DTCE bit ← 1: DTC activation is enabled by S12AD interrupt IER0C register IEN6 bit ← 1: S12ADI0 interrupt request is enabled
Enable the MTU0 interrupt request	TIER register TGIEA bit ← 1: Interrupt requests (TGIA0) enabled
Enable the S12ADI0 interrupt request	ADCSR register ADIE bit ← 1: Enables S12ADI0 interrupt generation upon scan completion
Start the DTC module	DTCST register ← 01h DTCST bit = 1: DTC module start
return	

Figure 5.11 DTC Initialization

5.9.10 IRQ Initialization

Figure 5.12 shows the IRQ Initialization.

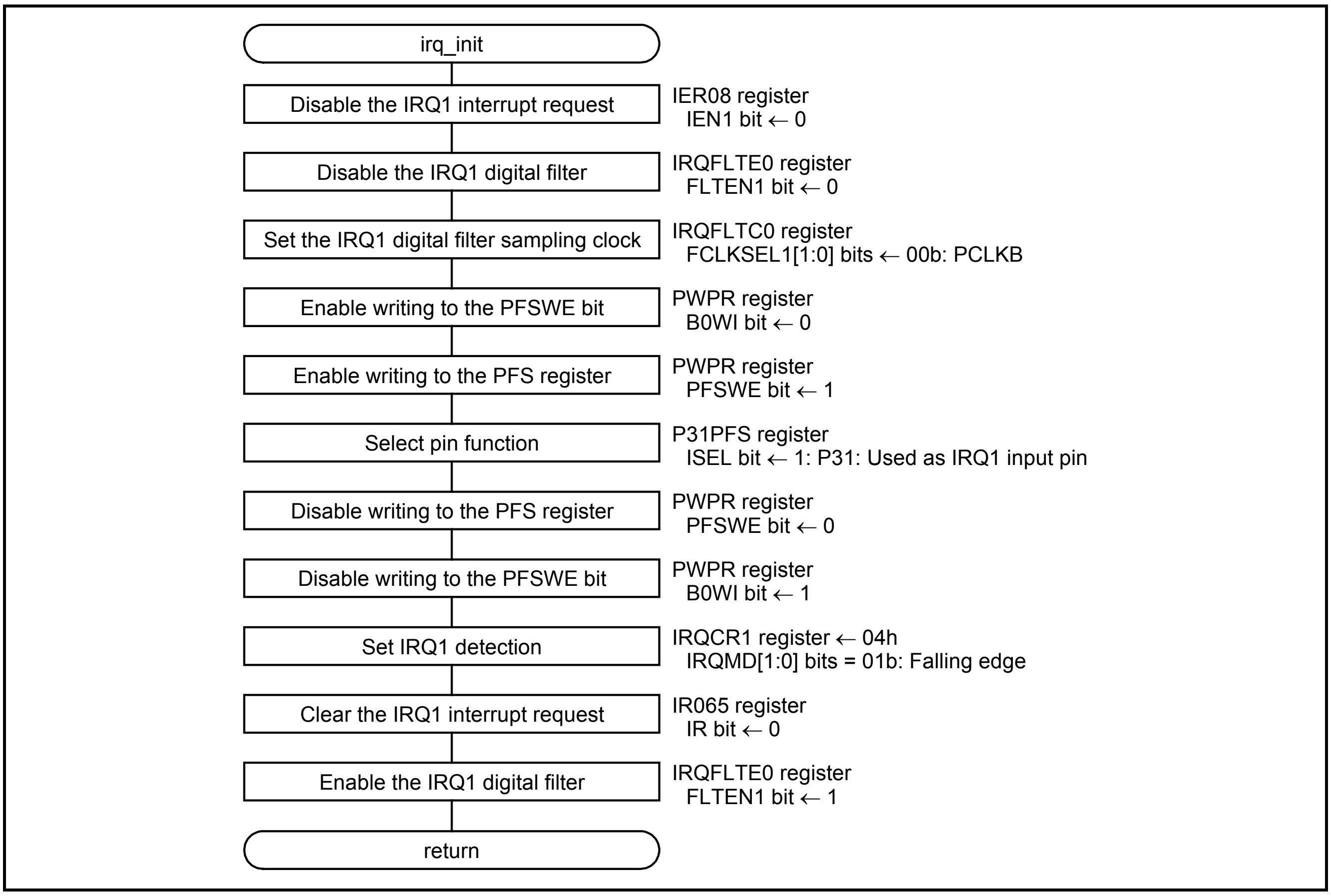


Figure 5.12 IRQ Initialization

5.9.11 Settings Prior to Entering Sleep Mode

Figure 5.13 shows the Settings Prior to Entering Sleep Mode.

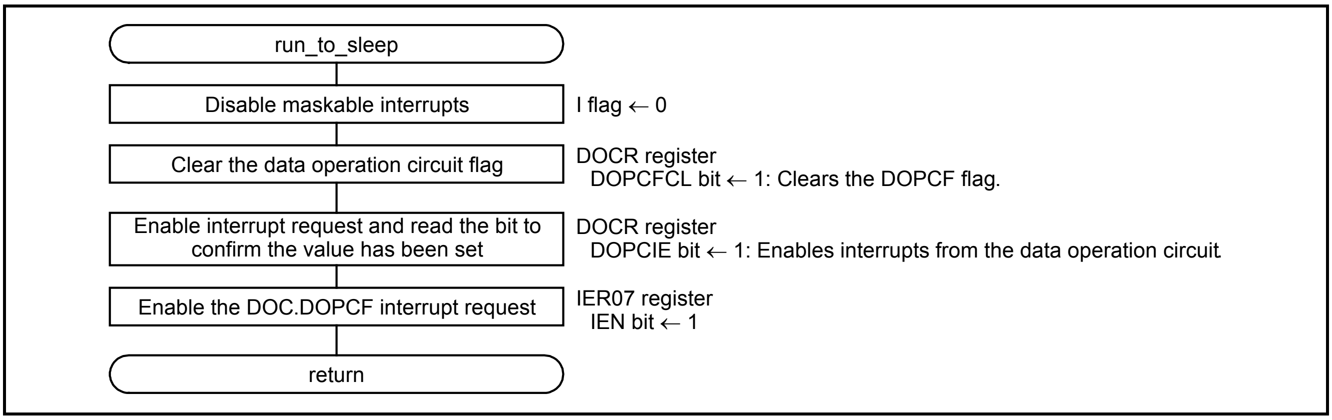


Figure 5.13 Settings Prior to Entering Sleep Mode

5.9.12 Settings After Exiting Sleep Mode

Figure 5.14 shows the Settings After Exiting Sleep Mode.

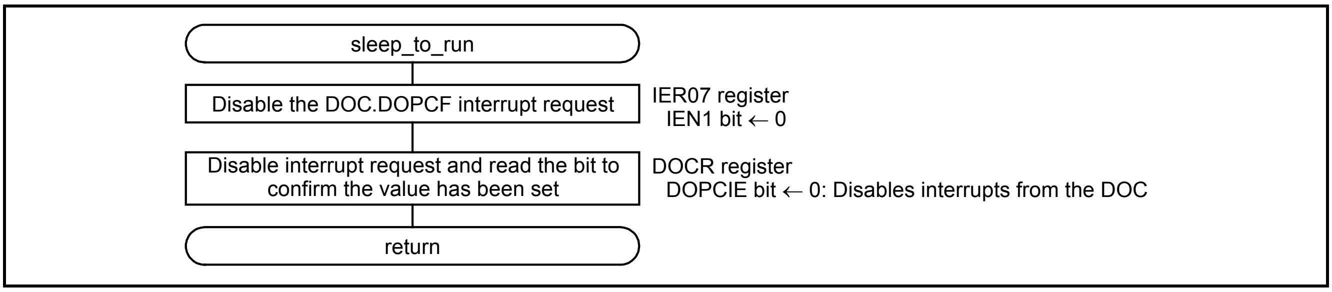


Figure 5.14 Settings After Exiting Sleep Mode

5.9.13 DOC.DOPCF Interrupt Handling

Figure 5.15 shows the DOC.DOPCF Interrupt Handling.

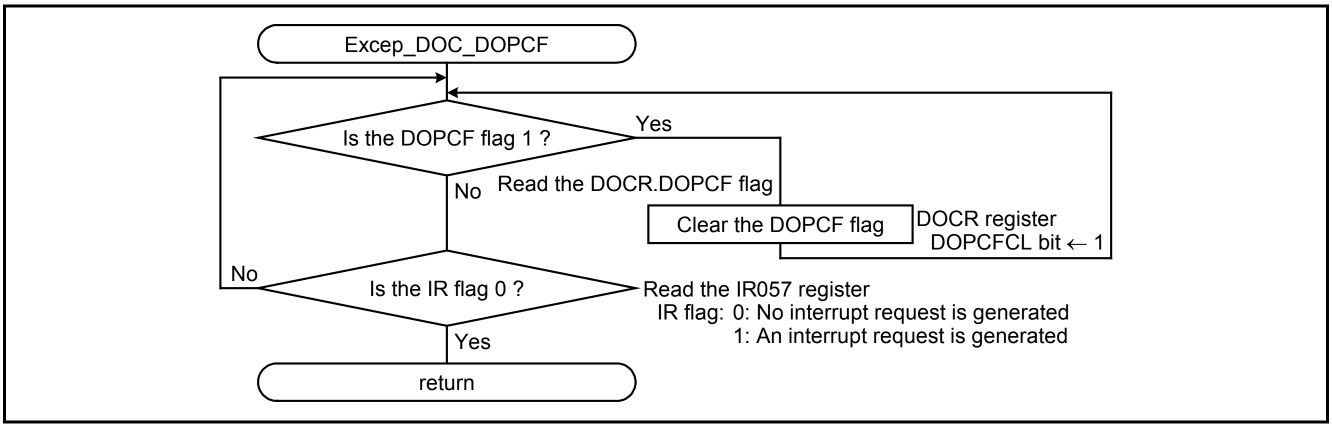


Figure 5.15 DOC.DOPCF Interrupt Handling

6. Applying This Application Note to the 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 RX220 Group, use this application note in conjunction with the RX220 Group Initial Setting application note.

For details on using this application note with the RX220 Group, refer to “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 RX220 Group Initial Setting. 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)
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 and RX220 Groups Application Note Monitoring the Voltage Using the DOC and S12AD

Rev.	**Date**	**Description**	
		Page	**Summary**
1.00	Aug. 1, 2013	—	First edition issued
1.01	July 1, 2014	1	Products: Added the RX220 Group.
		4	3. Reference Application Notes: Added the RX220 Group Initial Setting application note.
		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 RX220 Group: Added.
		24	8. Reference Documents: Added the RX220 Group User's Manual: Hardware.

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.

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