RX63T Group

Initialization Example

R01AN1252EJ0101
Rev.1.01
Jul 29, 2016

Introduction

This application note describes the settings required after a reset, including the clock settings and stopping the peripheral functions that operate after a reset.

Target Device

- RX63T Group 144-pin versions with ROM capacities from 256 to 512 KB
- RX63T Group 120-pin versions with ROM capacities from 256 to 512 KB
- RX63T Group 112-pin versions with ROM capacities from 256 to 512 KB
- RX63T Group 100-pin versions with ROM capacities from 256 to 512 KB
- RX63T Group 64-pin versions with ROM capacities from 32 to 64 KB
- RX63T Group 48-pin versions with ROM capacities from 32 to 64 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

This application note shows how to stop the peripheral functions that operate after a reset, set up the ports that do not exist, and set the clocks. This application note assumes that the processing required when power is first applied has been performed.

1.1 Stopping the Peripheral Functions that Operate after a Reset

There are some peripheral functions that operate after power is applied and others for which the module stop function is disabled. This application note provides the following processing related to this item.

Processing to stop the functional safety of DMAC, DTC, and RAM0.

Note, however, that the sample code does not perform this processing. If this processing is required, the corresponding settings in the sample code must be changed so that the required processing is performed.

1.2 Nonexistent Port Settings

In the versions of these microcontrollers that have fewer than 144 pins, the ports that are not connected to package pins must be set to output mode. The sample code performs initial settings that corresponding to packages with 64 pins. These settings must be modified to correspond to the version of the microcontroller actually used.

1.3 Clock Settings

1.3.1 Overview

The clock settings are performed in the following order.

1. Main clock settings
2. PLL clock settings
3. System clock switching

The sample code sets up the clocks so that the PLL is used as the system clock.

1.3.2 Clock Specifications Assumed in the Sample Code

Table 1.1 lists the clock specifications assumed by the sample code. The oscillator stabilization time and other settings are calculated for the specifications of the oscillator element listed in the table.

Table 1.2 lists the peripheral functions and their application used in this processing.

Table 1.1 Sample Code Operating Conditions

	Oscillator Frequency	Oscillator Stabilization Time	Remarks
Main clock oscillator	16 MHz	4.2 ms*1	The 144-pin versions use 12 MHz.
PLL clock	192 MHz (Main clock divided by 1 and multiplied by 12)	500 µs (maximum)*2	The 144-pin versions use 144 MHz.

Notes: 1. The oscillator stabilization time required in an actual system will depend on the printed circuit board wiring pattern, the oscillator circuit component constants, and other conditions. We recommend that users contact the manufacturer of the oscillator element used and request an evaluation of the actual system.
2. See the Electrical Characteristics section in the User's Manual: Hardware.

Table 1.2 Peripheral Function and Its Applications

Peripheral Function	Application
Compare match timer channel 0 (CMT0)	Clock oscillator stabilization time period measurement*

Note: * If an OS is used, select a timer channel that is not used by the OS.

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 (64-Pin Version)

Item	Contents
MCU used	R5F563T6EDFM (RX63T Group) (64-pin version)
Operating frequency	Main clock: 16 MHz PLL: 192 MHz (main clock divided by 1 and multiplied by 12) System clock (ICLK): 96 MHz (PLL divided by 2) Timer module clock (PCLKA): 96 MHz (PLL divided by 2) Peripheral module clock (PCLKB): 48 MHz (PLL divided by 4) S12AD clock (PCLKD): 48 MHz (PLL divided by 4) Flash interface clock (FCLK): 48 MHz (PLL divided by 4)
Operating voltage	3.3 V
Integrated development environment	Renesas Electronics Corporation High-performance Embedded Workshop Version 4.09.01.007
C compiler	Renesas Electronics Corporation C/C++ Compiler Package for RX Family V.1.02 Release 01
	Compiler option (The integrated development environment default settings are used.)
iodefine.h version	2.1b
Endian order	Little endian
Operating mode	Single-chip mode
Processor mode	Supervisor mode
Sample code version	1.00
Board used	Renesas Starter Kit for RX63T (Product number: R0K50563TS000BE)

Table 2.2 Operation Confirmation Conditions (144-Pin Version)

Item	Contents
MCU used	R5F563TEADFB (RX63T Group) (144-pin version)
Operating frequency	Main clock: 12 MHz PLL: 144 MHz (main clock divided by 1 and multiplied by 12) System clock (ICLK): 72 MHz (PLL divided by 2) Timer module clock (PCLKA): 72 MHz (PLL divided by 2) Peripheral module clock (PCLKB): 36 MHz (PLL divided by 4) AD clock (PCLKC): 36 MHz (PLL divided by 4) S12AD clock (PCLKD): 36 MHz (PLL divided by 4) External bus clock (BCLK): 36 MHz (PLL divided by 4) Flash interface clock (FCLK): 36 MHz (PLL divided by 4)
Operating voltage	5.0 V
Integrated development environment	Renesas Electronics Corporation High-performance Embedded Workshop Version 4.09.01.007
C compiler	Renesas Electronics Corporation C/C++ Compiler Package for RX Family V.1.02 Release 01
	Compiler option (The integrated development environment default settings are used.)
iodefine.h version	2.1b
Endian order	Little endian
Operating mode	Single-chip mode
Processor mode	Supervisor mode
Sample code version	1.00
Board used	Renesas Starter Kit for RX63T-H (Product number: R0K5563THS000BE)

3. Software

This software stops the peripheral functions that are operating after a reset and sets up the ports that don't exist, and then sets up the clocks.

3.1 Stopping the Peripheral Functions Operating after a Reset

This software stops the peripheral functions that are operating after a reset.

After the reset is cleared, the module stop state for just those peripheral functions listed below is cleared. To cause the module to transition to the module stop state, set the module stop bit (transition to the module stop state) to 1. Power consumption can be reduced by performing this module stop operation.

In the sample code, the constants of the form MSTP_STATE_<corresponding module name> are all set to 0 (MODULE_STOP_DISABLE), and the corresponding modules do not transition to the module stop state. When it is desirable, according to system being used, to cause certain modules to transition to the stop state, set the values of the corresponding constants in the file r_init_stop_module.h to the value 1 (MODULE_STOP_ENABLE).

Table 3.1 lists the peripheral modules whose module state is cleared after a reset.

Table 3.1 Peripheral Modules whose Module State Is Cleared after a Reset

Corresponding Module	Module Stop Setting Bit	Value after Reset	Setting when This Module Is Not Used
DMAC/DTC	MSTPCRA.MSTPA28 bit	0 (Module stop state cleared)	1 (Module stop state)
RAM0	MSTPCRC.MSTPC0 bit		

3.2 Nonexistent Port Settings

3.2.1 Processing Overview

When a version of this product with less than 144 pins is used, the corresponding bit in the PDR register for the port that does not exist is set to 1 (output). After this function is called, if the application writes data in byte units to a PDR register or the PODR register that includes ports that do not exist, for the ports that do not exist, the application should write 1 for the direction control bit and 0 for the port output data storage bit.

Tables 3.2 and 3.3 list the ports that do not exist.

Table 3.2 Nonexistent Ports (1)

Port Symbol	120-Pin Products	Number of Pins	112-Pin Products	Number of Pins	100-Pin Products	Number of Pins
PORT0	P02 to P05	4	P02, P03	2	P02 to P05	4
PORT1	P14	1	P13, P14	2	P12 to P14	3
PORT2	—	—	P25, P26	2	P25, P26	2
PORT3	P34, P35	2	P34, P35	2	P34, P35	2
PORT4*	—	—	—	—	—	—
PORT5*	P56, P57	2	P56, P57	2	P56, P57	2
PORT6*	—	—	—	—	—	—
PORT7	—	—	—	—	—	—
PORT8	—	—	—	—	—	—
PORT9	—	—	—	—	—	—
PORTA	PA6	1	PA6	1	PA6	1
PORTB	—	—	—	—	—	—
PORTC*	PC0 to PC5	6	PC0 to PC5	6	PC0 to PC5	6
PORTD	—	—	—	—	—	—
PORTE	—	—	—	—	—	—
PORTF	PF4	1	PF0, PF1	2	PF0 to PF4	5
PORTG	—	—	PG6	1	PG0 to PG6	7

Note: * PORT4 to PORT6 and PORTC are input-only pins.

Table 3.3 Nonexistent Ports (2)

Port Symbol	64-Pin Products	Number of Pins	48-Pin Products	Number of Pins
PORT0	P02 to P05	4	P00 to P05	6
PORT1	P12 to P14	3	P10 to P14	5
PORT2	P20, P21, P25, P26	4	P20, P21, P25, P26	4
PORT3	P34, P35	2	P31 to P35	5
PORT4*1	—	—	P45, P46	2
PORT7	—	—	—	—
PORT8*2	P80 to P82	3	P80 to P82	3
PORT9	P90, P95, P96	3	P90 to P96	7
PORTA	PA0, PA1, PA6	3	PA0, PA1, PA4 to PA6	5
PORTB	—	—	PB7	1
PORTD	PD0 to PD2	3	PD0 to PD2	3

Notes: 1. PORT4 to PORT6 and PORTC are input-only pins.
2. In the 48-pin and 64-pin versions, there is no PDR register for PORT5, PORT6, PORT8, PORTC, PORTF, and PORTG.

3.2.2 Selecting the Pin Count

The sample code is set up for 64-pin versions (PIN_SIZE = 64). The sample code can support pin counts of 48, 64, 100, 112, 120, and 144 pins. To use the sample code with a version with a pin count other than 64, set the PIN_SIZE constant in the r_init_non_existent_port.h header file to the pin count of the product used.

3.3 Clock Setup

3.3.1 Clock Setup Procedure

Table 3.4 lists the clock setup procedure steps and the processing performed and the settings performed by the sample code.

This sample code performs all the settings in steps 1 through 4. It starts both the main clock and the PLL. After that is switches the system clock to the PLL.

Table 3.4 Clock Setup Procedure

Step	Processing	Operations Performed	Settings Performed by the Sample Code
1	Main clock oscillator setup	After setting the wait time until the main clock output is supplied to the internal clock with the MOSCWTCR register, the sample code starts the main clock. After that, it waits in software for the main clock oscillator stabilization time*1 to elapse.	Sets the main clock oscillator to the operating state.
2	PLL oscillator setup*2	After setting PLL input divisor and frequency multiplier values and the wait period until the PLL output is supplied to the internal clock with the PLLWTCR register, the sample code starts the PLL clock. After that, it waits in software for the PLL oscillator stabilization time*1 to elapse.	Starts PLL operation.
3	Clock divisor setting*3	This step changes the clock divisor.	• ICLK, PCLKA: Divided by 2 • PCLKB, PCLKD, FCLK: Divided by 4
4	System clock source switching	The system clock source is switched to a clock appropriate for the actual application.	Switches to PLL operation

Notes: 1. See section 3.3.2, Notes on the Clock Oscillator Stabilization Times, for further information on the clock oscillator stabilization times.
2. The PLL setup step is not required if the PLL is not used.
3. If the main clock is selected as the system clock, do not set a divisor of 1 or 2.

3.3.2 Notes on the Clock Oscillator Stabilization Times

This section discusses the main clock, the PLL wait control register, and the oscillator stabilization time. The oscillator stabilization time is shown in the sample code.

Main Clock Oscillator Stabilization Time (64-Pin Versions)

Figure 3.1 shows our approach to the main clock oscillator stabilization time and table 3.5 lists the setting values and methods for calculating the main clock wait control and oscillator stabilization time.

The main clock wait control register (MOSCWTCR) is set to a value greater than the main clock oscillator stabilization time recommended by the oscillator element manufacturer. Also, the main clock oscillator stabilization time is set to a value greater than the sum of the main clock oscillator stabilization time recommended by the oscillator element manufacturer, the wait time set in the MOSCWTCR register), and 16,384 cycles.

Since the main clock oscillator stabilization time used in the sample code is 4.2 ms, the value set by the sample code in the wait control register is 0Dh (about 8.19 ms) and the oscillator stabilization period is about 13.416 ms.

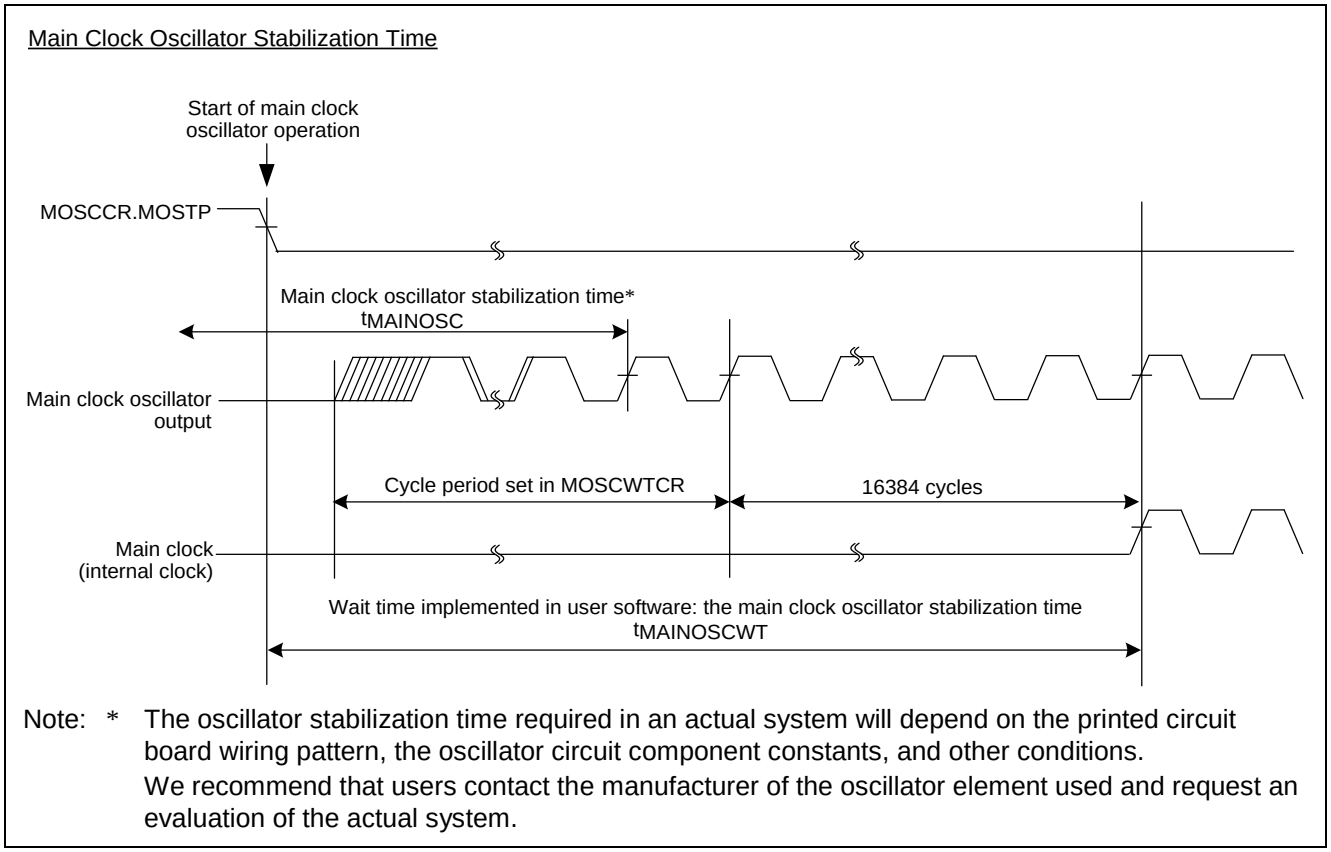


Figure 3.1 Main Clock Oscillator Stabilization Time

Table 3.5 Setting Values and Methods for Calculating the Main Clock Wait Control and Oscillator Stabilization Time

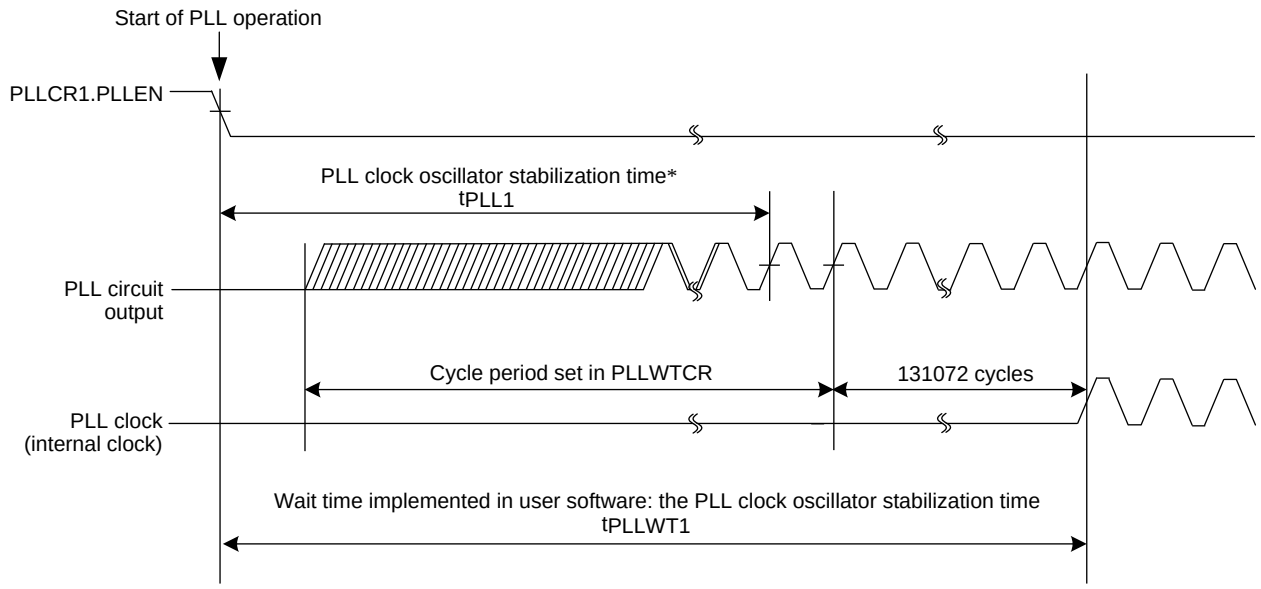
	Calculation	Sample Code Setting Value
Wait control register (MOSCWTCR.MSTS)	A value greater than the main clock oscillator stabilization time recommended by the oscillator element manufacturer	64-pin versions: 0Dh (about 8.19 ms) 144-pin versions: 0Dh (about 10.92 ms)
Oscillator stabilization time (tMAINOSCWT)	When n is the wait time selected with the MOSWTCR.MSTS bits: $tMAINOSC + \frac{n + 16384}{fMAIN}$	64-pin versions: 13.416 ms 144-pin versions: 16.487 ms

Notes on PLL Oscillator Stabilization Time (when the PLL Is Started after the Main Clock Oscillator Stabilization Time Has Elapsed) (64-Pin Versions)

Figure 3.2 shows our approach to the PLL oscillator stabilization time and table 3.6 lists the setting values and methods for calculating the PLL wait control and oscillator stabilization time settings.

The PLL wait control register (PLLWTCR) is set to a value greater than tPLL1 (a maximum of 500 µs). Also, the PLL oscillator stabilization time is set to the sum of tPLL1 (500 µs), the wait time set in the PLLWTCR, and 131072 cycles.

Since the PLL oscillator stabilization time is a maximum of 500 µs, the sample code sets the wait control register to 0Ah (about 681.6 µs) and the oscillator stabilization time is about 1.865 ms.



Note: * See the Electrical Characteristics section in the User's Manual: Hardware.

Figure 3.2 PLL Oscillator Stabilization Time

Table 3.6 Setting Values and Methods for Calculating the PLL Clock Wait Control and Oscillator Stabilization Time

	Calculation	Sample Code Setting Value
Wait control register (PLLWTCR.PSTS)	tPLL1 (a maximum of 500 µs)	64-pin versions: 0Ah (about 681.6 µs) 144-pin versions: 0Ah (about 910.2 µs)
Oscillator stabilization time (tPLLWT1)	When n is the wait time selected with the PLLWTCR.PSTS: $tPLL1 + \frac{n + 131072}{fPLL}$	64-pin versions: about 1.865 ms 144-pin versions: about 2.320 ms

3.4 File Composition

Table 3.7 lists the files used in the sample code. Files generated by the integrated development environment are not included in this table.

Table 3.7 Files Used in the Sample Code

File Name	Outline	Remarks
main.c	Main processing	
r_init_stop_module.c	Stops the peripheral functions that operate after a reset	
r_init_stop_module.h	Header file for r_init_stop_module.c	
r_init_non_existent_port.c	Initializes the ports that do not exist	
r_init_non_existent_port.h	Header file for r_init_non_existent_port.c	
r_init_clock.c	Initializes the clocks	
r_init_clock.h	Header file for r_init_clock.c	

3.5 Option-Setting Memory

Table 3.8 lists the option-setting memory configured in the sample code. When necessary, set a value suited to the user system.

Table 3.8 Option-Setting Memory Configured in the Sample Code

Symbol	Address	Setting Value	Contents
OFS0	FFFF FF8Fh to FFFF FF8Ch	FFFF FFFFh	After a reset, the IWDT is stopped. After a reset, the WDT is stopped.
OFS1	FFFF FF8Bh to FFFF FF88h	FFFF FFFFh	After a reset, voltage monitoring reset 0 is ignored.
MDES	FFFF FF83h to FFFF FF80h	FFFF FFFFh	Little endian

3.6 Constants

Table 3.9 lists the constants used in the sample code, and tables 3.10 to 3.15 list the pin counts for the products used. The values indicated by asterisks (*) are the ones set in the sample code.

Table 3.9 Constants Used in the Sample Code

Constant Name	Setting Value	Contents
WAIT_TIME_FOR_MAIN_OSCILLATION*1	13,416,000 L 16,487,000 L*	64-pin: the main clock oscillator stabilization time (ns) 144-pin: the main clock oscillator stabilization time (ns)
WAIT_TIME_FOR_PLL_OSCILLATION*1	1,865,000 L 2.320,000 L*	64-pin: the PLL oscillator stabilization time (ns) 144-pin: the PLL oscillator stabilization time (ns)
MSTP_STATE_DMACDTC*2	MODULE_STOP_DISABLE	Clears the module stop state for the DMAC and DTC modules.
MSTP_STATE_RAM0*2	MODULE_STOP_DISABLE	Clears the module stop state for the RAM0 modules.
PIN_SIZE*3	64	Number of pins for the product used
FOR_CMT0_TIME	1/LOCO(143.75kHz)*32	Counting period (ns) for the timer (CMT0) used to wait the oscillator stabilization time (LOCO = PCLKB = 143.75 kHz (maximum), PCLKB divided by 32)
MODULE_STOP_ENABLE	1	Module stop state
MODULE_STOP_DISABLE	0	Module stop disabled state

Notes: 1. Modify the settings in r_init_clock.h to correspond to the actual system.
2. Modify the settings in r_init_stop_module.h to correspond to the actual system.
3. Modify the settings in r_init_no_existent_port.h to correspond to the actual system.

Table 3.10 Constants for 144-Pin Versions (PIN_SIZE = 144)

Constant Name	Setting Value	Contents
DEF_P0PDR	0x00	Setting value for the port P0 direction register
DEF_P1PDR	0x00	Setting value for the port P1 direction register
DEF_P2PDR	0x00	Setting value for the port P2 direction register
DEF_P3PDR	0x00	Setting value for the port P3 direction register
DEF_P7PDR	0x00	Setting value for the port P7 direction register
DEF_P8PDR	0x00	Setting value for the port P8 direction register
DEF_P9PDR	0x00	Setting value for the port P9 direction register
DEF_PAPDR	0x00	Setting value for the port PA direction register
DEF_PBPDR	0x00	Setting value for the port PB direction register
DEF_PDPDR	0x00	Setting value for the port PD direction register
DEF_PEPDR	0x00	Setting value for the port PE direction register
DEF_PFPDR	0x00	Setting value for the port PF direction register
DEF_PGPDR	0x00	Setting value for the port PG direction register

Table 3.11 Constants for 120-Pin Versions (PIN_SIZE = 120)

Constant Name	Setting Value	Contents
DEF_P0PDR	0x3C	Setting value for the port P0 direction register
DEF_P1PDR	0x10	Setting value for the port P1 direction register
DEF_P2PDR	0x00	Setting value for the port P2 direction register
DEF_P3PDR	0x30	Setting value for the port P3 direction register
DEF_P7PDR	0x00	Setting value for the port P7 direction register
DEF_P8PDR	0x00	Setting value for the port P8 direction register
DEF_P9PDR	0x00	Setting value for the port P9 direction register
DEF_PAPDR	0x40	Setting value for the port PA direction register
DEF_PBPDR	0x00	Setting value for the port PB direction register
DEF_PDPDR	0x00	Setting value for the port PD direction register
DEF_PEPDR	0x00	Setting value for the port PE direction register
DEF_PFPDR	0x10	Setting value for the port PF direction register
DEF_PGPDR	0x00	Setting value for the port PG direction register

Table 3.12 Constants for 112-Pin Versions (PIN_SIZE = 112)

Constant Name	Setting Value	Contents
DEF_P0PDR	0x0C	Setting value for the port P0 direction register
DEF_P1PDR	0x18	Setting value for the port P1 direction register
DEF_P2PDR	0x60	Setting value for the port P2 direction register
DEF_P3PDR	0x30	Setting value for the port P3 direction register
DEF_P7PDR	0x00	Setting value for the port P7 direction register
DEF_P8PDR	0x00	Setting value for the port P8 direction register
DEF_P9PDR	0x00	Setting value for the port P9 direction register
DEF_PAPDR	0x40	Setting value for the port PA direction register
DEF_PBPDR	0x00	Setting value for the port PB direction register
DEF_PDPDR	0x00	Setting value for the port PD direction register
DEF_PEPDR	0x00	Setting value for the port PE direction register
DEF_PFPDR	0x03	Setting value for the port PF direction register
DEF_PGPDR	0x40	Setting value for the port PG direction register

Table 3.13 Constants for 100-Pin Versions (PIN_SIZE = 100)

Constant Name	Setting Value	Contents
DEF_P0PDR	0x3C	Setting value for the port P0 direction register
DEF_P1PDR	0x1C	Setting value for the port P1 direction register
DEF_P2PDR	0x60	Setting value for the port P2 direction register
DEF_P3PDR	0x30	Setting value for the port P3 direction register
DEF_P7PDR	0x00	Setting value for the port P7 direction register
DEF_P8PDR	0x00	Setting value for the port P8 direction register
DEF_P9PDR	0x00	Setting value for the port P9 direction register
DEF_PAPDR	0x40	Setting value for the port PA direction register
DEF_PBPDR	0x00	Setting value for the port PB direction register
DEF_PDPDR	0x00	Setting value for the port PD direction register
DEF_PEPDR	0x00	Setting value for the port PE direction register
DEF_PFPDR	0x1F	Setting value for the port PF direction register
DEF_PGPDR	0x7F	Setting value for the port PG direction register

Table 3.14 Constants for 64-Pin Versions (PIN_SIZE = 64)

Constant Name	Setting Value	Contents
DEF_P0PDR	0x3C	Setting value for the port P0 direction register
DEF_P1PDR	0x1C	Setting value for the port P1 direction register
DEF_P2PDR	0x63	Setting value for the port P2 direction register
DEF_P3PDR	0x30	Setting value for the port P3 direction register
DEF_P7PDR	0x00	Setting value for the port P7 direction register
DEF_P9PDR	0x61	Setting value for the port P9 direction register
DEF_PAPDR	0x43	Setting value for the port PA direction register
DEF_PBPDR	0x00	Setting value for the port PB direction register
DEF_PDPDR	0x07	Setting value for the port PD direction register

Table 3.15 Constants for 48-Pin Versions (PIN_SIZE = 48)

Constant Name	Setting Value	Contents
DEF_P0PDR	0x3F	Setting value for the port P0 direction register
DEF_P1PDR	0x1F	Setting value for the port P1 direction register
DEF_P2PDR	0x63	Setting value for the port P2 direction register
DEF_P3PDR	0x3E	Setting value for the port P3 direction register
DEF_P7PDR	0x00	Setting value for the port P7 direction register
DEF_P9PDR	0x7F	Setting value for the port P9 direction register
DEF_PAPDR	0x73	Setting value for the port PA direction register
DEF_PBPDR	0x80	Setting value for the port PB direction register
DEF_PDPDR	0x07	Setting value for the port PD direction register

3.7 Functions

Table 3.16 lists the functions used in the sample code.

Table 3.16 Functions Used in the Sample Code

Function Name	Outline
main	Main processing
R_INIT_StopModule	Stops the peripheral functions that operate after a reset
R_INIT_NonExistentPort	Initializes the ports that do not exist
R_INIT_Clock	Initializes the clocks
CGC_oscillation_main	Main clock oscillation setting
CGC_oscillation_PLL	PLL clock oscillation setting
cmt0_wait	Software wait using CMT0

3.8 Function Specifications

The following tables list the sample code function specifications.

main	
Outline	Main processing
Header	None
Declaration	void main(void)
Description	This function calls the function that stops the peripheral functions that operate after a reset, the function that initializes the nonexistent ports, and the function that initializes the clocks.
Arguments	None
Return Value	None

R_INIT_StopModule	
Outline	Stops the peripheral functions that operate after a reset
Header	r_init_stop_module.h
Declaration	void R_INIT_StopModule (void)
Description	This function transitions to the module stop state.
Arguments	None
Return Value	None
Remarks	The sample code does not perform the transition to the module stop state.

R_INIT_NonExistentPort	
Outline	Initializes the ports that do not exist
Header	r_init_non_existent_port.h
Declaration	void R_INIT_NonExistentPort(void)
Description	This function initializes the port direction registers for the port pins that do not exist for versions with fewer than 144 pins.
Arguments	None
Return Value	None
Remarks	The sample code performs the settings for 64-pin versions (PIN_SIZE = 64). After this function is called, if the application writes data in byte units to a PDR register that includes ports that do not exist, for the ports that do not exist, the application should write 1 for the direction control bit and 0 for the port output data storage bit.

R_INIT_Clock	
Outline	Initializes the clocks
Header	r_init_clock.h
Declaration	void R_INIT_Clock(void)
Description	This function initializes the clocks.
Arguments	None
Return Value	None
Remarks	In the sample code, the PLL is used as the system clock.

CGC_oscillation_main	
Outline	Main clock oscillation setting
Header	r_init_clock.h
Declaration	void CGC_oscillation_main (void)
Description	This function sets MOSCWTCR and then starts the main clock. After that, it waits for the main clock oscillator stabilization time to elapse.
Arguments	None
Return Value	None

CGC_oscillation_PLL	
Outline	PLL oscillation setting
Header	r_init_clock.h
Declaration	void CGC_oscillation_PLL (void)
Description	This function sets the PLL input divisor, the frequency multiplier value, and the PLLWTCR register. It then starts the PLL clock. After that, it waits for the PLL oscillator stabilization time to elapse.
Arguments	None
Return Value	None
Remarks	This processing may be omitted if the PLL is not used as the system clock.

cmt0_wait	
Outline	Software wait using CMT0
Header	r_init_clock.h
Declaration	static void cmt0_wait (uint32_t cnt)
Description	This function is used to wait for an oscillator stabilization time to elapse.
Arguments	uint32_t cnt: Oscillator stabilization time cnt = (oscillator stabilization time (ns)*1) ÷ FOR_CMT0_TIME*2)
Return Value	None
Remarks	Notes: 1. The oscillator stabilization time depends on the oscillator element used. Set this value using the method described in section 3.3.2. 2. This is the calculated value FOR_CMT0_TIME - 143.75 kHz (maximum). The actual wait time will differ depending on the LOCO frequency.

3.9 Flowcharts

3.9.1 Main Processing

Figure 3.3 shows the main processing.

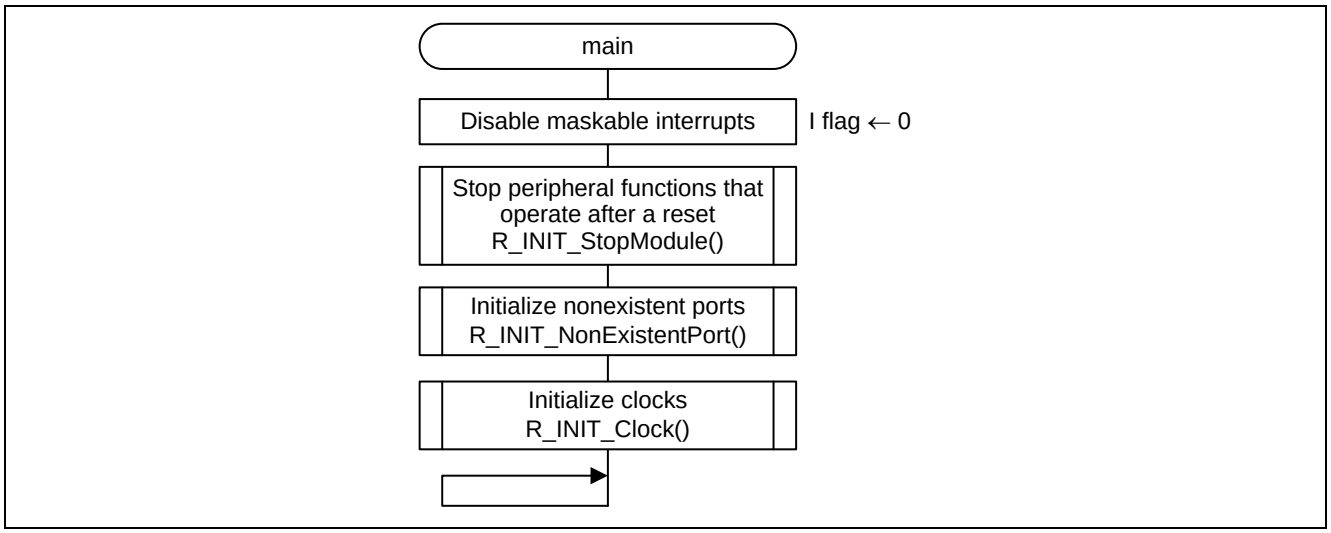
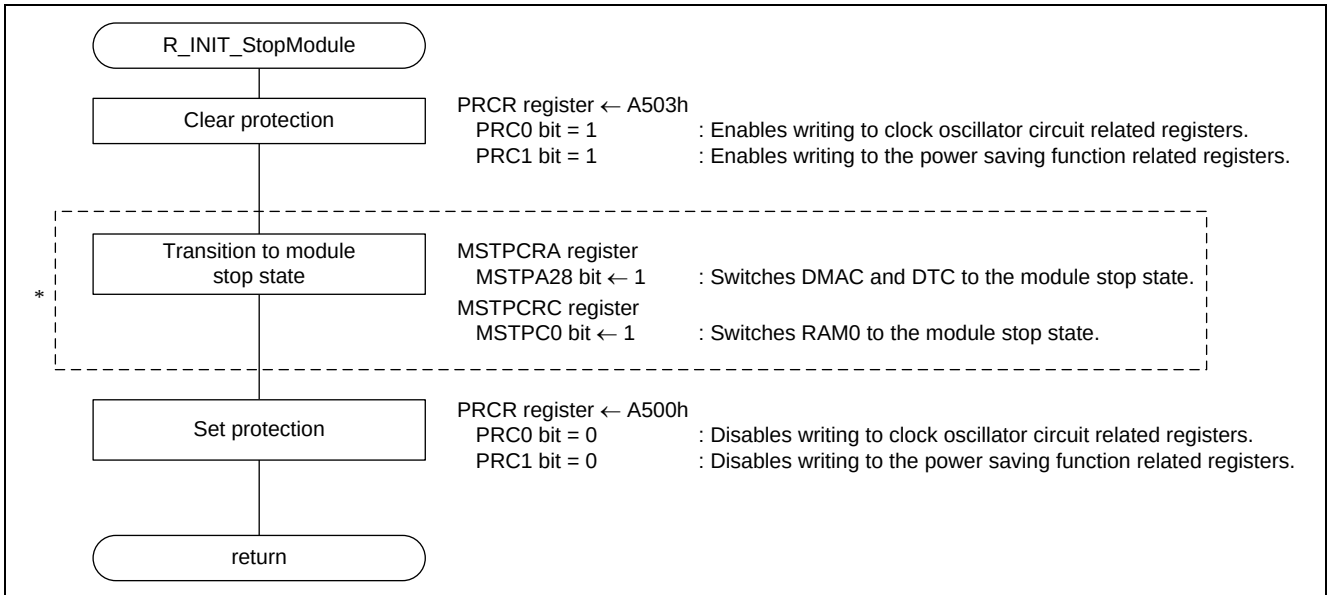


Figure 3.3 Main Processing

3.9.2 Stops the Peripheral Functions that Operate after a Reset

Figure 3.4 shows the flowchart for stopping peripheral functions that operate after a reset.



Note: * In the sample code, the modules go to the module stop cleared state.
To switch the peripheral modules to the module state, Set the value of the corresponding #define MSTP_STATE_<module name> constant to 1.

Figure 3.4 Stops the Peripheral Functions that Operate after a Reset

3.9.3 Initializes the Nonexistent Ports

Figure 3.5 shows the flowchart for initializing ports that do not exist.

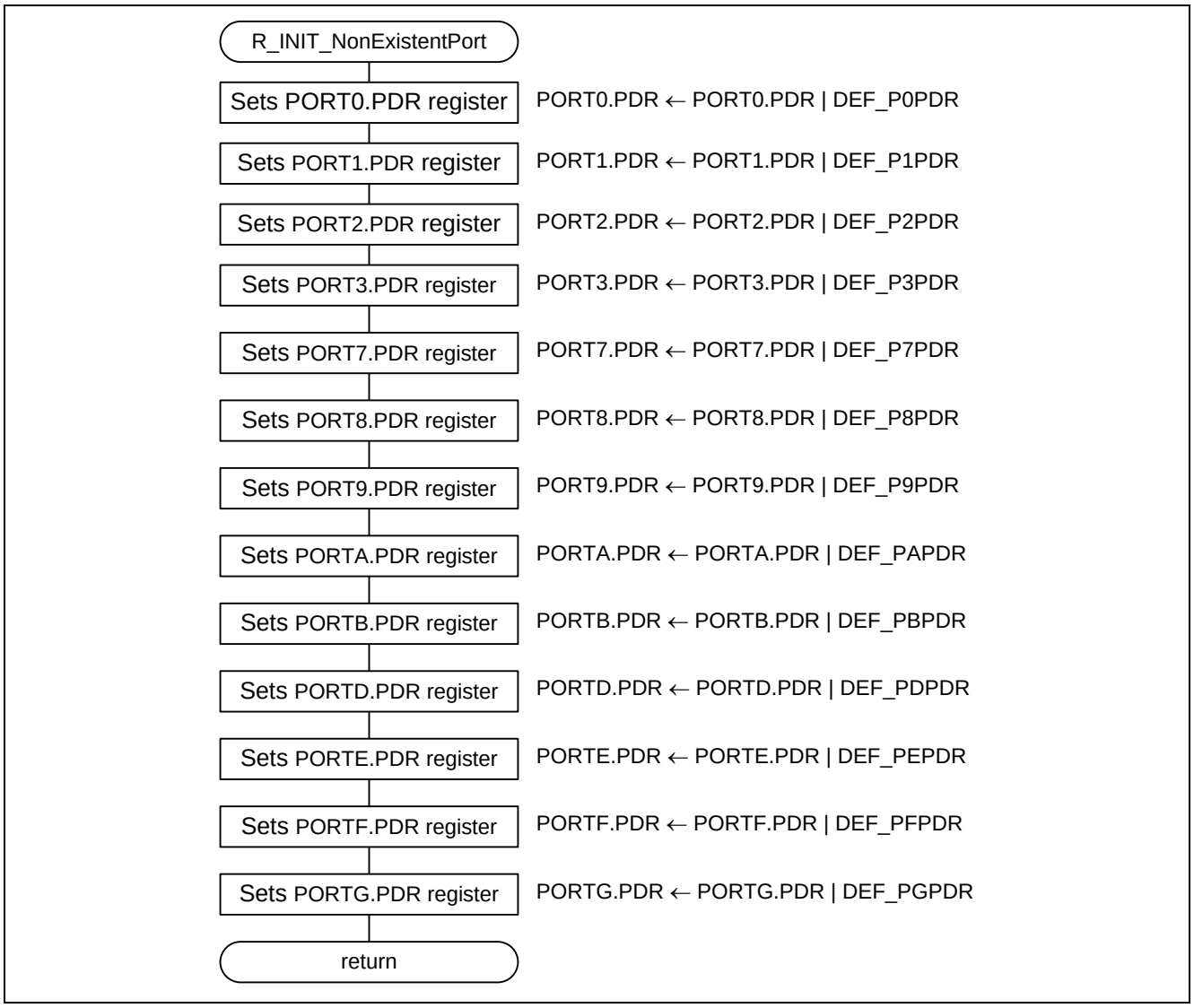
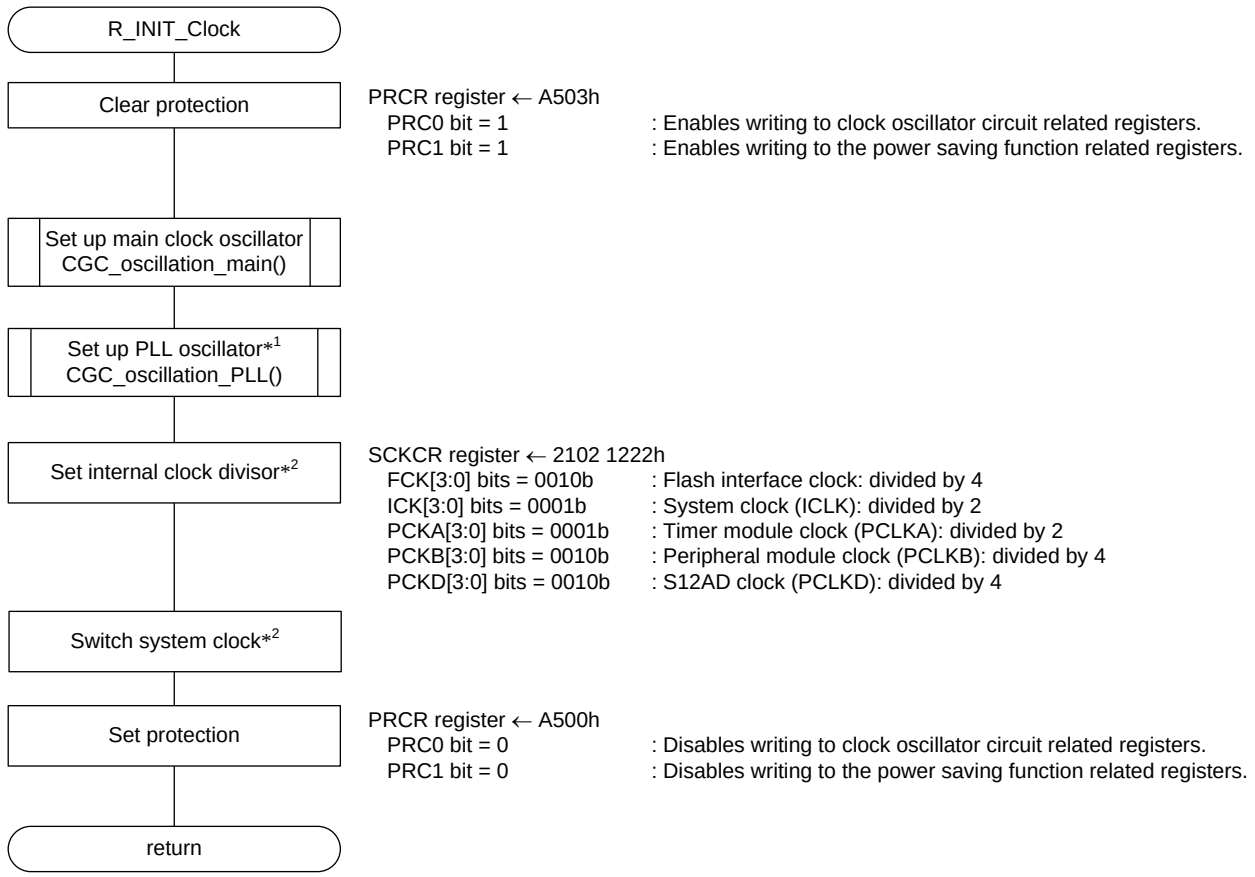


Figure 3.5 Initializes the Nonexistent Ports

3.9.4 Initializes the Clocks

Figure 3.6 shows the flowchart for initializing the clocks.



Notes: 1. The PLL oscillator setup processing may be omitted if the PLL is not used.
2. The internal clock divisors and the system clock switching should be modified according to the specifications of the actual system.

Figure 3.6 Initializes the Clocks

Figures 3.7 and 3.8 show the flowcharts for the main clock oscillator settings and the PLL oscillator settings.

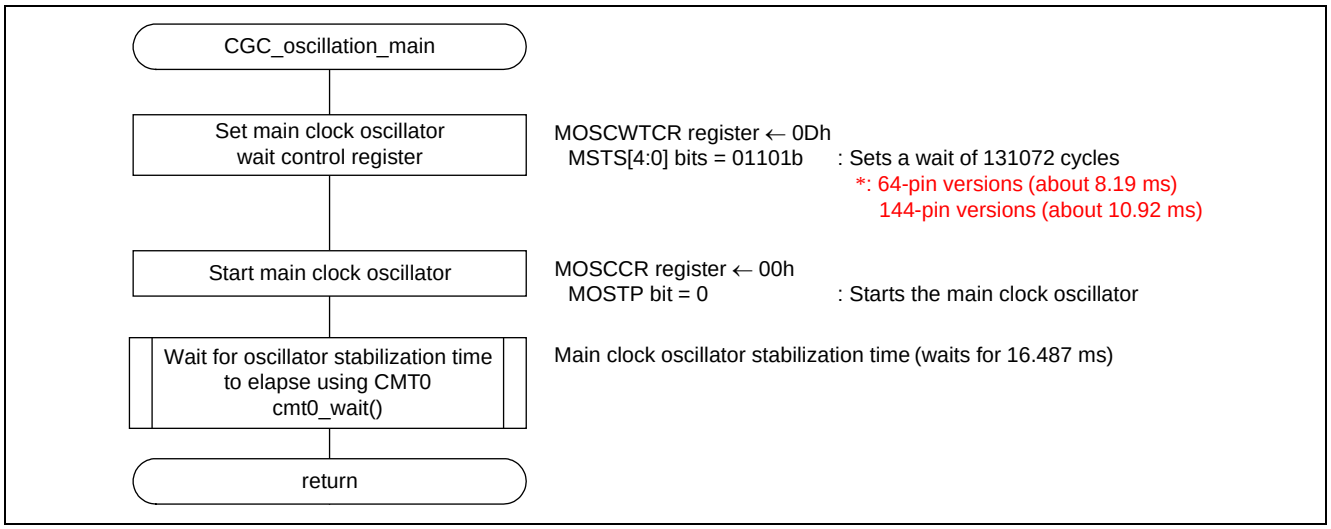


Figure 3.7 Main Clock Oscillation Setting

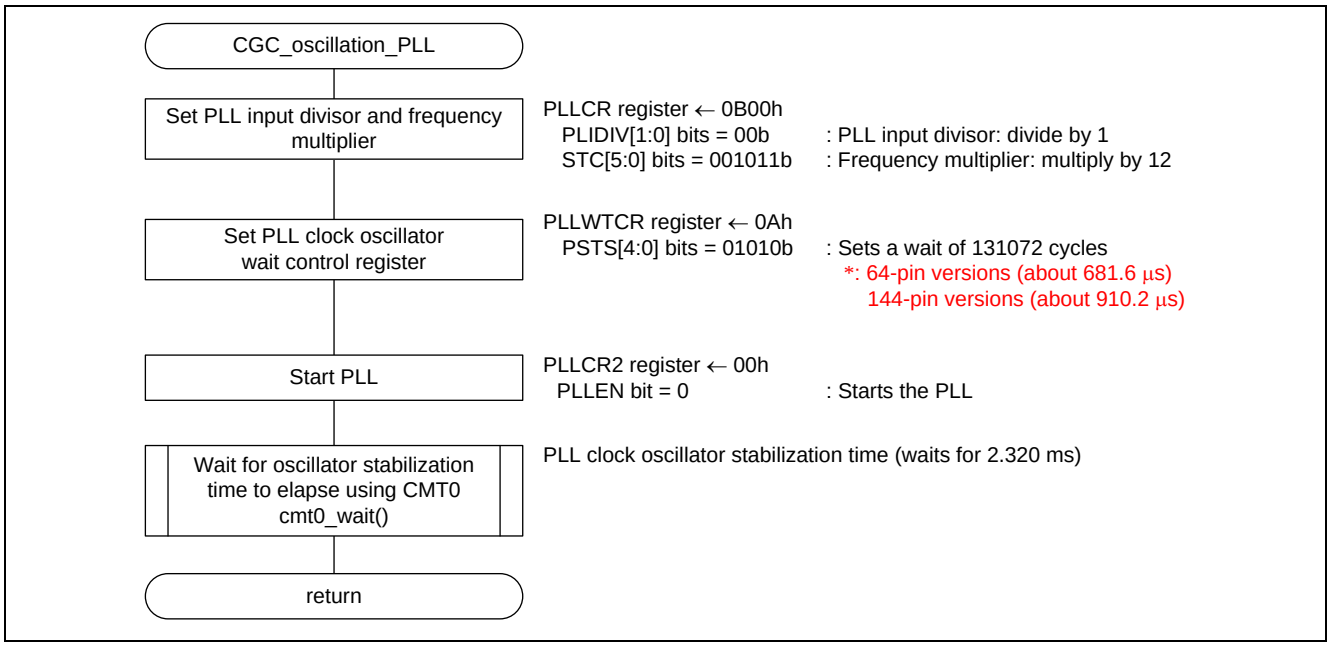
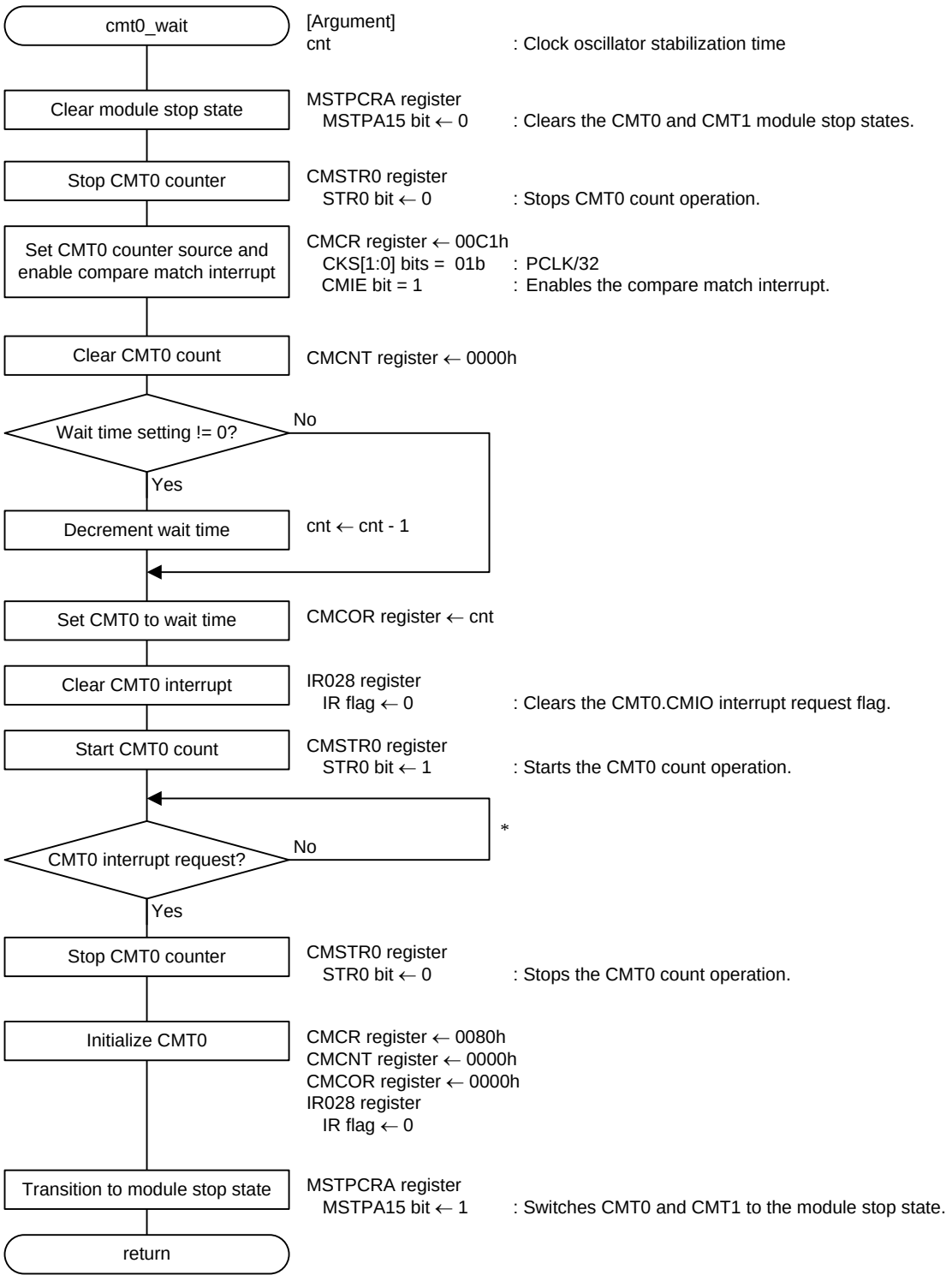


Figure 3.8 PLL Oscillation Setting

3.9.5 Software Wait Using CMT0

Figure 3.9 shows the flowchart for software wait using CMT0.



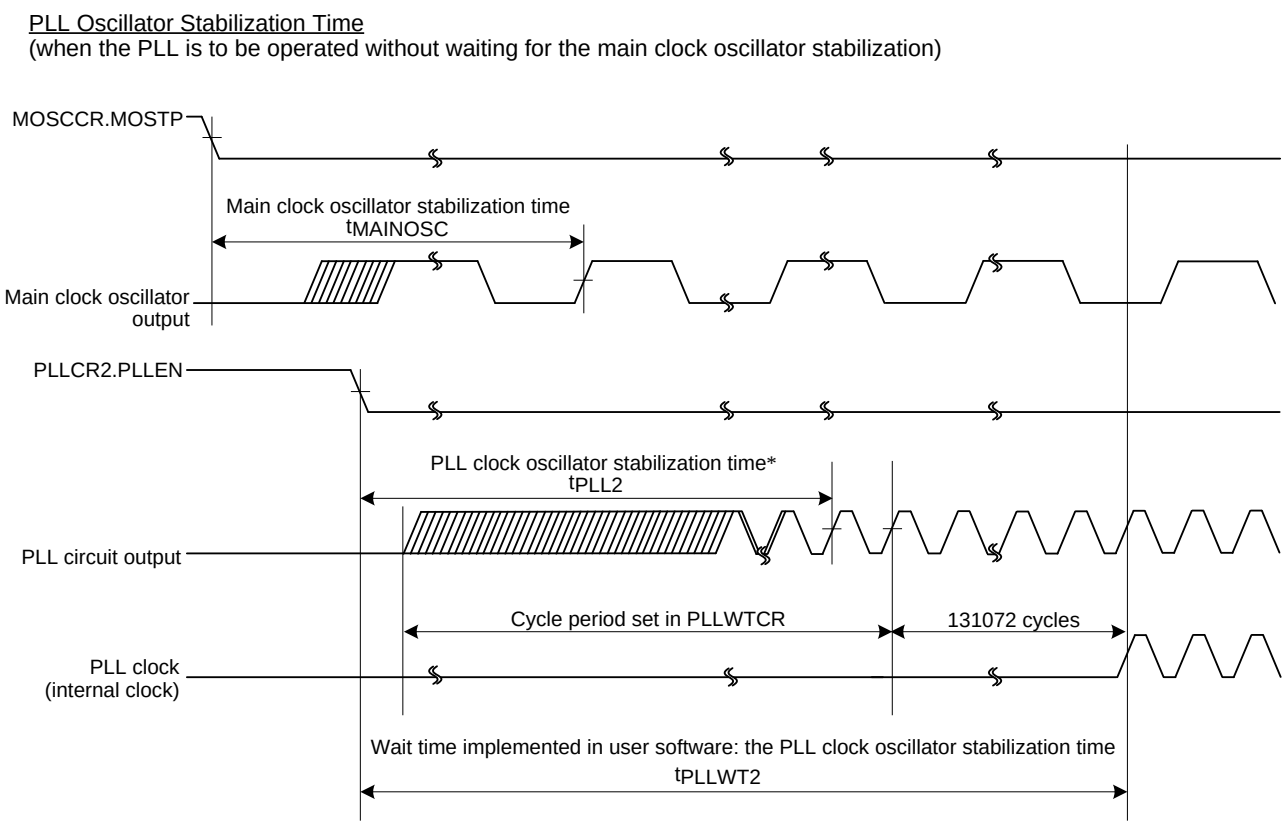
Note: * If the watchdog timer (WDT) and/or independent watchdog timer (IWDT) count operations are operation, the WDT and/or IWDT refresh operations should be performed.

Figure 3.9 Software Wait Using CMT0

4. Appendix

Notes on the PLL Oscillator Stabilization Time (when the PLL is to be operated without waiting for the main clock oscillator stabilization)

When the main clock and the PLL clock are both operated, applications can wait for the main clock and the PLL clock to stabilize together (i.e. the wait periods may overlap). Figure 4.1 shows our approach to waiting for both oscillator stabilization times together, and table 4.1 lists the calculation methods for the wait controller and oscillator stabilization time.



Note: * See the Electrical Characteristics section in the User's Manual: Hardware.

Figure 4.1 Approach to Waiting for Both the Main Clock and PLL Clock Oscillator Stabilization Times Together

Table 4.1 Setting Values and Methods for Calculating the PLL Clock Wait Control and Oscillator Stabilization Time

	Calculation
Wait control register (PLLWTCR.PSTS)	The sum of a value greater than main clock oscillator stabilization time recommended by the oscillator element manufacturer and a value greater than tPLL1 (which has a maximum value of 500 µs)
Oscillator stabilization time (tPLLWT2)	When n is the wait time selected with the PLLWTCR.PSTS: $tMAINOSC + tPLL1 + \frac{n + 131072}{fPLL}$

5. Sample Code

Sample code can be downloaded from the Renesas Electronics website.

6. Reference Documents

User's Manual: Hardware
RX63T Group User's Manual: Hardware Rev.2.00 (R01UH0238EJ0200)
(The latest version can be downloaded from the Renesas Electronics website.)

Technical Update/Technical News
(The latest version can be downloaded from the Renesas Electronics website.)

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

		Description	
Rev.	**Date**	**Page**	**Summary**
1.00	Feb. 15, 2013	—	First edition issued
1.01	Jul. 29, 2016	13	Removed part of the content of table 3.1, Option Setting Memory OFS1 Used in Sample Code

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.

   — 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 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