

---

# RX210, RX21A, and RX220 Groups

R01AN1017EJ0101

Rev. 1.01

July 1, 2014

## Efficient Allocation of the DTC Vector Table

---

### Abstract

This document describes how to efficiently allocate the DTC vector table of the data transfer controller (DTC) 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 .....	3
2. Operation Confirmation Conditions .....	6
3. Reference Application Note.....	6
4. Hardware .....	7
4.1 Pins Used.....	7
5. Software .....	8
5.1 Operation Overview .....	9
5.2 Section Structure .....	10
5.3 File Composition .....	10
5.4 Option-Setting Memory.....	11
5.5 Constants .....	12
5.6 Structure/Union List .....	13
5.7 Variables .....	14
5.8 Functions.....	14
5.9 Function Specifications .....	15
5.10 Flowcharts.....	17
5.10.1 Main Processing .....	17
5.10.2 Port Initialization .....	17
5.10.3 Peripheral Function Initialization .....	18
5.10.4 Low Power Consumption Initialization.....	18
5.10.5 CMT Initialization .....	19
5.10.6 DTC Initialization .....	20
6. Applying This Application Note to the RX21A or RX220 Group.....	21
7. Sample Code.....	22
8. Reference Documents.....	22

### 1. Specifications

The pattern output from PORT1 to the LED is changed every 500 ms and the pattern output from PORTA is changed every 250 ms using the DTC.

This application note describes two methods to allocate the DTC vector table listed in Table 1.1.

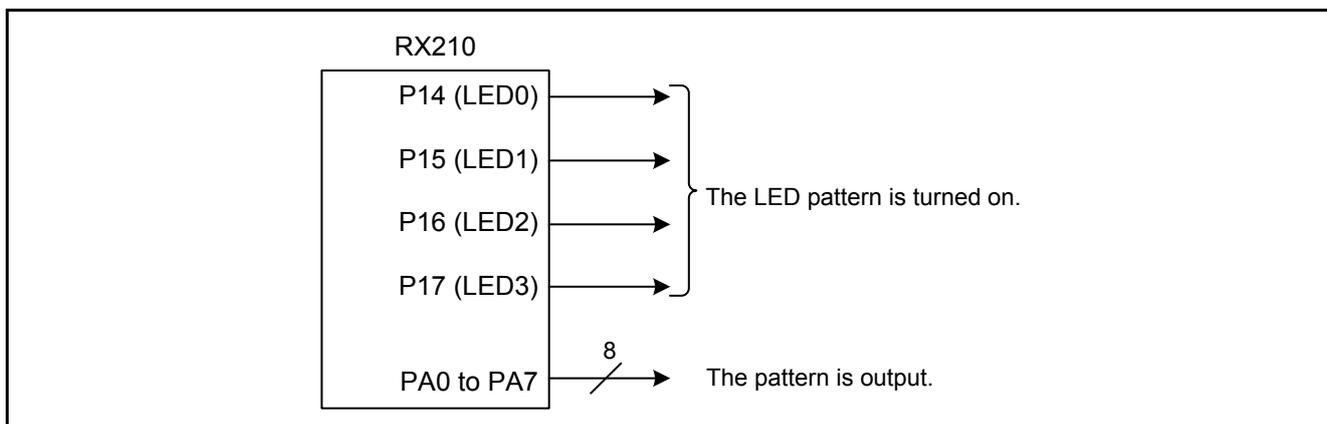
Table 1.2 lists the Peripheral Functions and Their Applications, Figure 1.1 shows a Connection Example, Figure 1.2 shows the Memory Map when the DTC Vector Table is Allocated to the ROM Area (Sample Code 1), and Figure 1.3 shows Memory Map when the DTC Vector Table is Allocated to the RAM Area (Sample Code 2).

**Table 1.1 DTC Vector Table Allocations**

Sample Code	Outline	Description
Sample code 1	Allocate the DTC vector table to the ROM area	Set all DTC vectors (4 x 256 bytes used). Add new section to allocate the DTC vector table.
Sample code 2	Allocate the DTC vector table to the RAM area	Set required DTC vectors only. Allocate DTC transfer data to blank areas. Specify the absolute address of the DTC vector table with #pragma.

**Table 1.2 Peripheral Functions and Their Applications**

Peripheral Function	Application
DTC	Transfer output data to ports.
Channel 0 in the compare match timer (CMT0)	DTC activation source (the pattern output to the LED is changed every 500 ms).
Channel 3 in the compare match timer (CMT3)	DTC activation source (the pattern output from PORTA is changed every 250 ms).



**Figure 1.1 Connection Example**

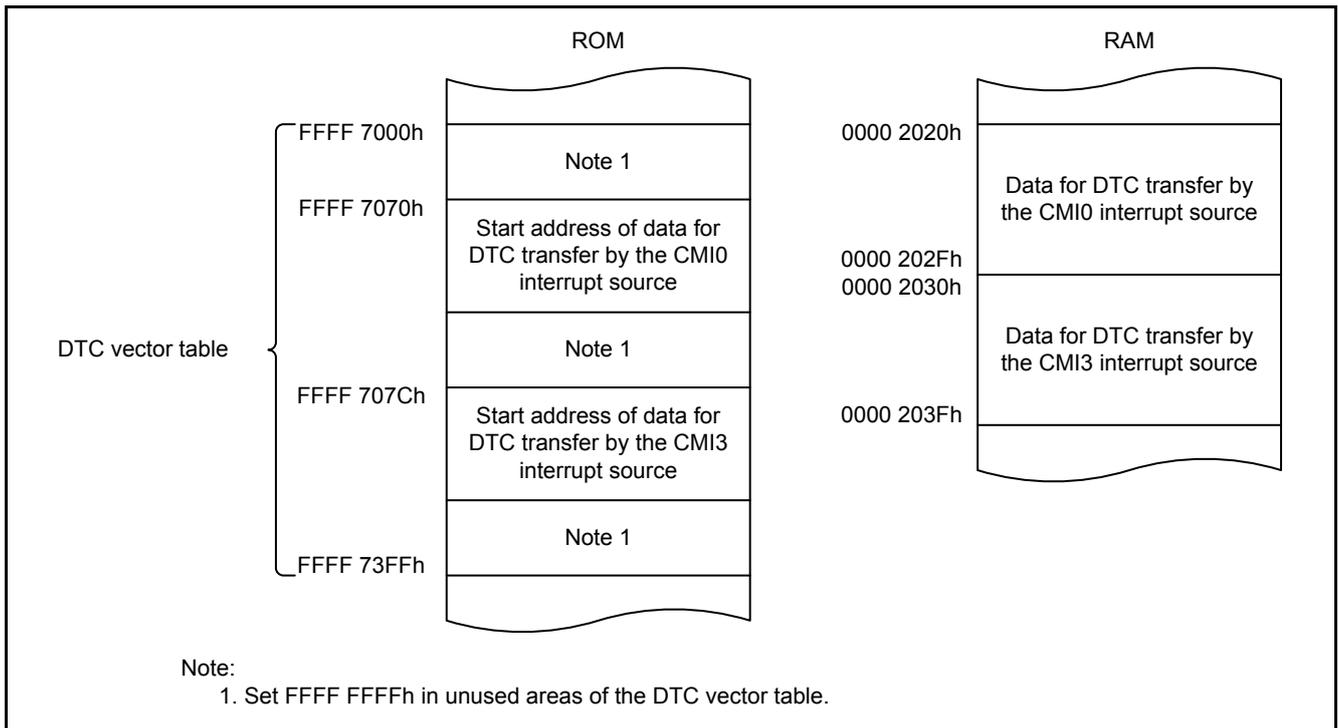


Figure 1.2 Memory Map when the DTC Vector Table is Allocated to the ROM Area (Sample Code 1)

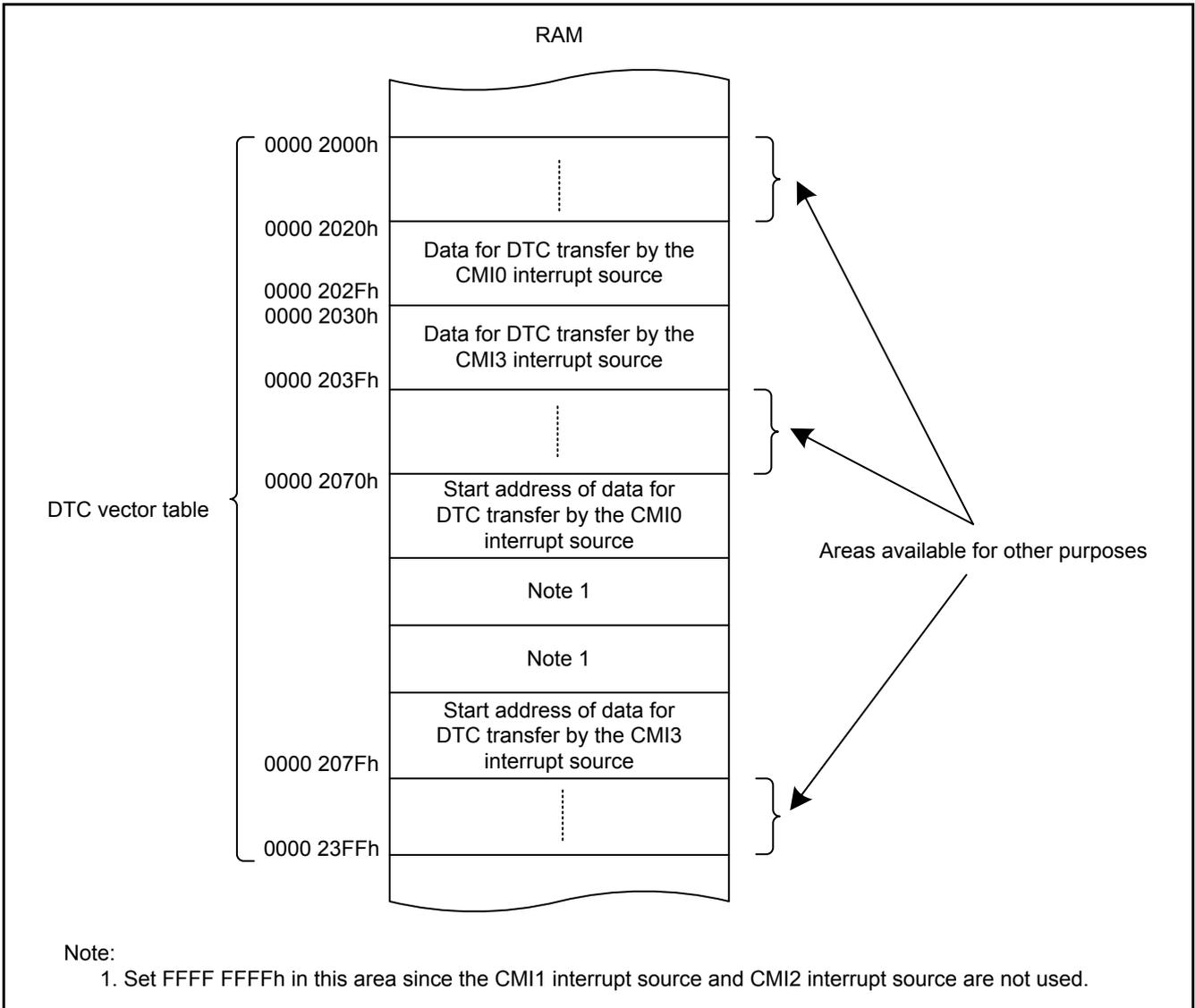


Figure 1.3 Memory Map when the DTC Vector Table is Allocated to the RAM Area (Sample Code 2)

## 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	- Main clock: 20 MHz - PLL: 100 MHz (main clock divided by 2 and multiplied by 10) - System clock (ICLK): 50 MHz (PLL divided by 2) - Peripheral module clock B (PCLKB): 25 MHz (PLL divided by 4)
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 no.: 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)
- 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 Pins Used

Table 4.1 lists the Pins Used and Their Functions.

**Table 4.1 Pins Used and Their Functions**

Pin Name	I/O	Function
P14	Output	Output data to turn on or turn off LED0
P15	Output	Output data to turn on or turn off LED1
P16	Output	Output data to turn on or turn off LED2
P17	Output	Output data to turn on or turn off LED3
PA0 to PA7	Output	Output a pattern

## 5. Software

Allocate the data table of the LED output pattern and data table of the PORTA output pattern to the ROM area in advance.

When the CMT0 compare match interrupt (CMI0) request is generated, DTC transfer is activated and the LED output pattern is transferred to PORT1. Then the values in the data table are sequentially transferred every time the CMI0 interrupt request is generated.

Also when the CMT3 compare match interrupt (CMI3) request is generated, DTC transfer is activated and the PORTA output pattern is transferred to PORTA. Then the values in the data table are sequentially transferred every time the CMI3 interrupt request is generated.

Settings for the peripheral functions used are as follows:

### CMT

- Count clock: PCLKB/512
- CMT0 compare match: In 500 ms intervals
- CMT3 compare match: In 250 ms intervals

### DTC

- Activation source: Compare match interrupts for CMT0 and CMT3
- DTC address mode: Full-address mode

Setting for DTC transfer triggered by the CMI0 interrupt request

- Transfer mode: Repeat transfer mode
- Transfer source address addressing mode: SAR value is incremented after data transfer
- Transfer source address: ROM area (LED output data)
- Transfer destination address addressing mode: Address in the DAR register is fixed
- Transfer destination address: PORT1 (LED)
- Data transfer unit: 8 bits
- Number of transfers: 4 times
- Chain transfer: Disabled

Setting for DTC transfer triggered by CMI3 interrupt request

- Transfer mode: Repeat transfer mode
- Transfer source address addressing mode: SAR value is incremented after data transfer
- Transfer source address: ROM area (PORTA output data)
- Transfer destination address addressing mode: Address in the DAR register is fixed
- Transfer destination address: PORTA
- Data transfer unit: 8 bits
- Number of transfers: 8 times
- Chain transfer: Disabled

### 5.1 Operation Overview

- (1) After initialization, when setting bits CMTSTR0.STR0 and CMTSTR1.STR3 to 1, CMT0 and CMT3 start counting.
- (2) When the CMI3 interrupt request is generated, write the first byte of PORTA output data (01h) to the PORTA.PODR register using DTC transfer. Then every time the CMI3 interrupt request is generated, PORTA output data is sequentially transferred to the PORTA.PODR register using DTC transfer.
- (3) When the CMI0 interrupt request is generated, write the first byte of LED output data (LED0 turned on) to the PORT1.PODR register using DTC transfer. Then every time the CMI0 interrupt request is generated, the LED output data is sequentially transferred to the PORT1.PODR register and output to the LED using DTC transfer.
- (4) After DTC transfers with the CMI3 interrupt source have been performed for the specified number of times (8 times in the sample codes), when the CMI3 interrupt request is generated again, perform DTC transfer from the first byte of PORTA output data.
- (5) After DTC transfers with the CMI0 interrupt source have been performed for the specified number of times (4 times in the sample codes), when the CMI0 interrupt request is generated again, perform DTC transfer from the first byte of LED output data.

Figure 5.1 shows the Operation Timing Diagram. (1) to (5) in the figure correspond to (1) to (5) in the description above.

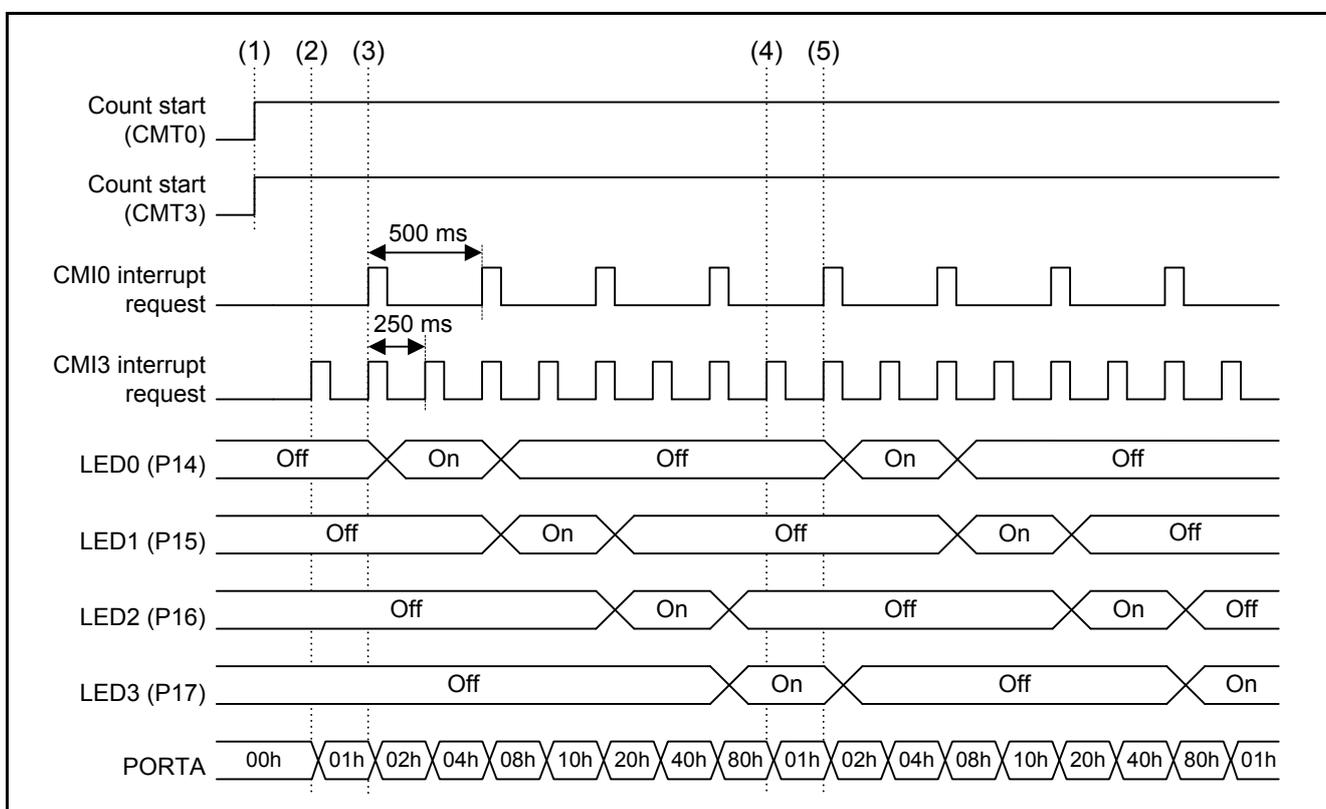


Figure 5.1 Operation Timing Diagram

#### Notes on embedding the sample codes

When embedding the sample codes from this application note in the actual system, note the following:

As DTC data transfer is performed in 8-bit units, when the DTC transfer is triggered by the CMI0 interrupt request and output data is written to the PORT1.PODR register for LED0 to LED3 (P14 to P17), P10 to P13 are also set to 0. Therefore do not use P10 to P13 as output ports.

## 5.2 Section Structure

Table 5.1 lists the Section Changed in Sample Code 1.

Refer to the latest RX Family C/C++ Compiler Package user's manual for adding, changing, or deleting sections.

**Table 5.1 Section Changed in Sample Code 1**

Section Name	Change	Address	Description
DTC_SECTION	Addition	FFFF 7000h	DTC vector table (ROM area)

## 5.3 File Composition

Table 5.2 lists the Files Used in Sample Code 1 and Table 5.3 lists the Files Used in Sample Code 2. Files generated by the integrated development environment are not included in this table.

**Table 5.2 Files Used in Sample Code 1**

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	
dtc_vector_table.c	DTC vector table	

**Table 5.3 Files Used in Sample Code 2**

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	

## 5.4 Option-Setting Memory

Table 5.4 lists the Option-Setting Memory Configured in Sample Codes 1 and 2. When necessary, set values suited to the user system.

**Table 5.4 Option-Setting Memory Configured in Sample Codes 1 and 2**

Symbol	Address	Setting Value	Contents
OFS0	FFFF FF8Fh to FFFF FF8Ch	FFFF FFFFh	IWDT is stopped after reset WDT is stopped after reset
OFS1	FFFF FF8Bh to FFFF FF88h	FFFF FFFFh	Voltage monitoring 0 reset is disabled after reset HOCO oscillation is disabled after reset
MDES	FFFF FF83h to FFFF FF80h	FFFF FFFFh	Little endian

## 5.5 Constants

Table 5.5 lists the Constants Used in Sample Code 1 and Table 5.6 lists the Constants Used in Sample Code 2.

**Table 5.5 Constants Used in Sample Code 1**

Constant Name	Setting Value	Contents
P_LED	PORT1.PODR.BYTE	LED output pin
LED_OFF	F0h	LED0 to LED3 all turned off
LED0_ON	E0h	LED0 turned on, LED1 to LED3 turned off
LED1_ON	D0h	LED1 turned on, LED0, LED2, and LED3 turned off
LED2_ON	B0h	LED2 turned on, LED0, LED1, and LED3 turned off
LED3_ON	70h	LED3 turned on, LED0 to LED2 turned off
PD_LED	PORT1.PDR.BYTE	I/O select bit of the LED output pin
LED_OUTPUT	F0h	LED output setting
ADDR_CMT0_DTC_TBL	0000 2020h	Start address of data for DTC transfer triggered by the CMI0 interrupt request
ADDR_CMT3_DTC_TBL	0000 2030h	Start address of data for DTC transfer triggered by the CMI3 interrupt request

**Table 5.6 Constants Used in Sample Code 2**

Constant Name	Setting Value	Contents
P_LED	PORT1.PODR.BYTE	LED output pin
LED_OFF	F0h	LED0 to LED3 all turned off
LED0_ON	E0h	LED0 turned on, LED1 to LED3 turned off
LED1_ON	D0h	LED1 turned on, LED0, LED2, and LED3 turned off
LED2_ON	B0h	LED2 turned on, LED0, LED1, and LED3 turned off
LED3_ON	70h	LED3 turned on, LED0 to LED2 turned off
PD_LED	PORT1.PDR.BYTE	I/O select bit of the LED output pin
LED_OUTPUT	F0h	LED output setting
ADDR_CMT0_DTC_TBL	0000 2020h	Start address of data for DTC transfer triggered by the CMI0 interrupt request
ADDR_CMT3_DTC_TBL	0000 2030h	Start address of data for DTC transfer triggered by the CMI3 interrupt request
ADDR_DTC_VECT_TOP	0000 2000h	Start address of the DTC vector address
ADDR_CMT0_DTC_VECT	0000 2070h	DTC vector address of CMT0 (CMI0)
ADDR_CMT1_DTC_VECT	0000 2074h	DTC vector address of CMT1 (CMI1)
ADDR_CMT2_DTC_VECT	0000 2078h	DTC vector address of CMT2 (CMI2)
ADDR_CMT3_DTC_VECT	0000 207Ch	DTC vector address of CMT3 (CMI3)

## 5.6 Structure/Union List

Figure 5.2 shows the Structure/Union Used in Sample Codes 1 and 2.

The bit field members are allocated from upper bits using the function of #pragma to specify the bit field sequence. Refer to the latest RX Family C/C++ Compiler Package user's manual for details

```
/* **** DTC Transfer information data **** */
#pragma bit_order left
#pragma unpack
struct st_dtc_full{
    union{
        unsigned long LONG;
        struct{
            unsigned long MRA_MD      :2;
            unsigned long MRA_SZ      :2;
            unsigned long MRA_SM      :2;
            unsigned long              :2;
            unsigned long MRB_CHNE    :1;
            unsigned long MRB_CHNS    :1;
            unsigned long MRB_DISEL   :1;
            unsigned long MRB_DTS     :1;
            unsigned long MRB_DM      :2;
            unsigned long              :2;
            unsigned long              :16;
        }BIT;
    }MR;
    void * SAR;
    void * DAR;
    struct{
        unsigned long CRA:16;
        unsigned long CRB:16;
    }CR;
};
#pragma bit_order
#pragma packoption
```

**Figure 5.2 Structure/Union Used in Sample Codes 1 and 2**

## 5.7 Variables

Table 5.7 lists the Global Variables for Sample Codes 1 and 2, Table 5.8 lists the const Variables for Sample Code 1, and Table 5.9 lists the const Variables for Sample Code 2.

**Table 5.7 Global Variables for Sample Codes 1 and 2**

Type	Variable Name	Contents	Function Used
st_dtc_full	cmt0_dtc_tbl	DTC transfer data table when the CMI0 interrupt is requested	dtc_init
st_dtc_full	cmt3_dtc_tbl	DTC transfer data table when the CMI3 interrupt is requested	dtc_init

**Table 5.8 const Variables for Sample Code 1**

Type	Variable Name	Contents	Function Used
unsigned char	dtc_output_data0	LED output data (transfer source of DTC transfer triggered by the CMI0 interrupt request)	dtc_init
unsigned char	dtc_output_data1	PORTA output data (transfer source of DTC transfer triggered by the CMI3 interrupt request)	dtc_init
void*	Relocatable_Vectors	DTC vector table	dtc_init

**Table 5.9 const Variables for Sample Code 2**

Type	Variable Name	Contents	Function Used
unsigned char	dtc_output_data0	LED output data (transfer source of DTC transfer triggered by the CMI0 interrupt request)	dtc_init
unsigned char	dtc_output_data1	PORTA output data (transfer source of DTC transfer triggered by the CMI3 interrupt request)	dtc_init

## 5.8 Functions

Table 5.10 lists the Functions for Sample Codes 1 and 2.

**Table 5.10 Functions for Sample Codes 1 and 2**

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 initialization
lpc_init	Low power consumption initialization
cmt_init	CMT initialization
dtc_init	DTC initialization

## 5.9 Function Specifications

The following tables list the sample code function specifications.

main	
<b>Outline</b>	Main processing
<b>Header</b>	None
<b>Declaration</b>	void main(void)
<b>Description</b>	After initialization, start the CMT count.
<b>Arguments</b>	None
<b>Return Value</b>	None
port_init	
<b>Outline</b>	Port initialization
<b>Header</b>	None
<b>Declaration</b>	void port_init(void)
<b>Description</b>	Initialize ports.
<b>Arguments</b>	None
<b>Return Value</b>	None
R_INIT_StopModule	
<b>Outline</b>	Stop processing for active peripheral functions after a reset
<b>Header</b>	r_init_stop_module.h
<b>Declaration</b>	void R_INIT_StopModule(void)
<b>Description</b>	Configure the setting to enter the module-stop state.
<b>Arguments</b>	None
<b>Return Value</b>	None
<b>Remarks</b>	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	
<b>Outline</b>	Nonexistent port initialization
<b>Header</b>	r_init_non_existent_port.h
<b>Declaration</b>	void R_INIT_NonExistentPort(void)
<b>Description</b>	Initialize port direction registers for ports that do not exist in products with less than 100 pins.
<b>Arguments</b>	None
<b>Return Value</b>	None
<b>Remarks</b>	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.

<b>R_INIT_Clock</b>	
<b>Outline</b>	Clock initialization
<b>Header</b>	r_init_clock.h
<b>Declaration</b>	void R_INIT_Clock(void)
<b>Description</b>	Initialize the clock.
<b>Arguments</b>	None
<b>Return Value</b>	None
<b>Remarks</b>	The sample code selects processing which uses PLL as the system clock without using the sub-clock. For details on this function, refer to the Initial Setting application note for the product used.
<b>peripheral_init</b>	
<b>Outline</b>	Peripheral function initialization
<b>Header</b>	None
<b>Declaration</b>	void peripheral_init(void)
<b>Description</b>	Initialize peripheral functions used.
<b>Arguments</b>	None
<b>Return Value</b>	None
<b>lpc_init</b>	
<b>Outline</b>	Low power consumption initialization
<b>Header</b>	None
<b>Declaration</b>	void lpc_init(void)
<b>Description</b>	Cancel the module stop state for peripheral functions used.
<b>Arguments</b>	None
<b>Return Value</b>	None
<b>cmt_init</b>	
<b>Outline</b>	CMT initialization
<b>Header</b>	None
<b>Declaration</b>	void cmt_init(void)
<b>Description</b>	Configure the setting to use the CMT.
<b>Arguments</b>	None
<b>Return Value</b>	None
<b>dtc_init</b>	
<b>Outline</b>	DTC initialization
<b>Header</b>	None
<b>Declaration</b>	void dtc_init(void)
<b>Description</b>	Configure the setting to use the DTC.
<b>Arguments</b>	None
<b>Return Value</b>	None

### 5.10 Flowcharts

#### 5.10.1 Main Processing

Figure 5.3 shows the Main Processing.

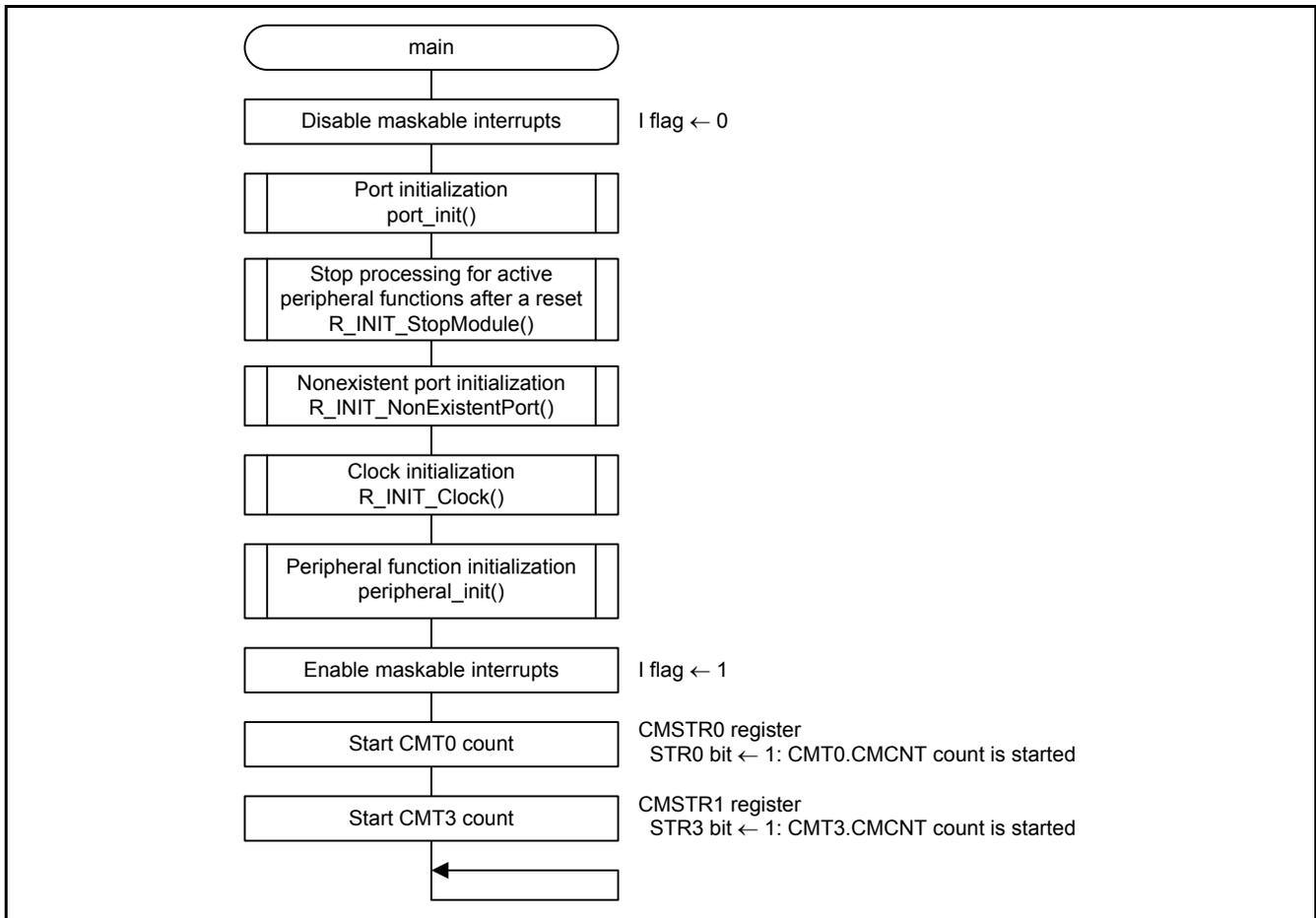


Figure 5.3 Main Processing

#### 5.10.2 Port Initialization

Figure 5.4 shows the Port Initialization.

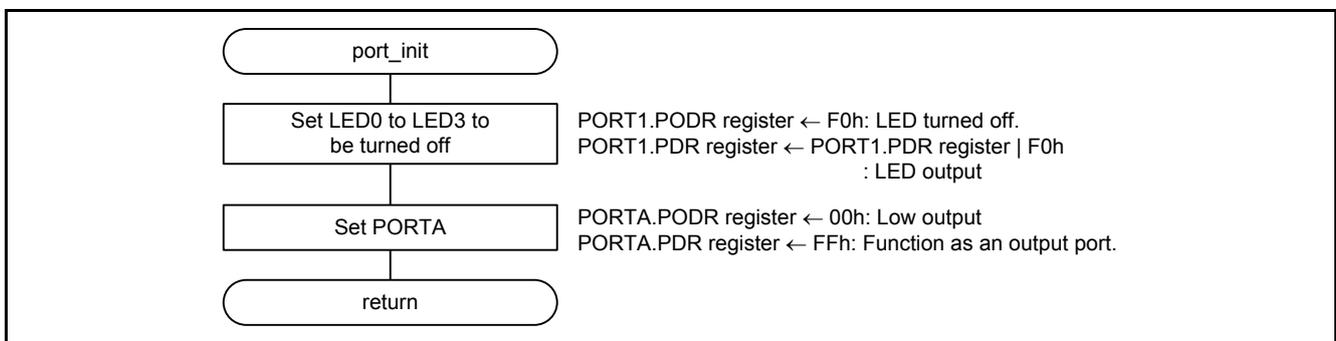


Figure 5.4 Port Initialization

### 5.10.3 Peripheral Function Initialization

Figure 5.5 shows Peripheral Function Initialization.

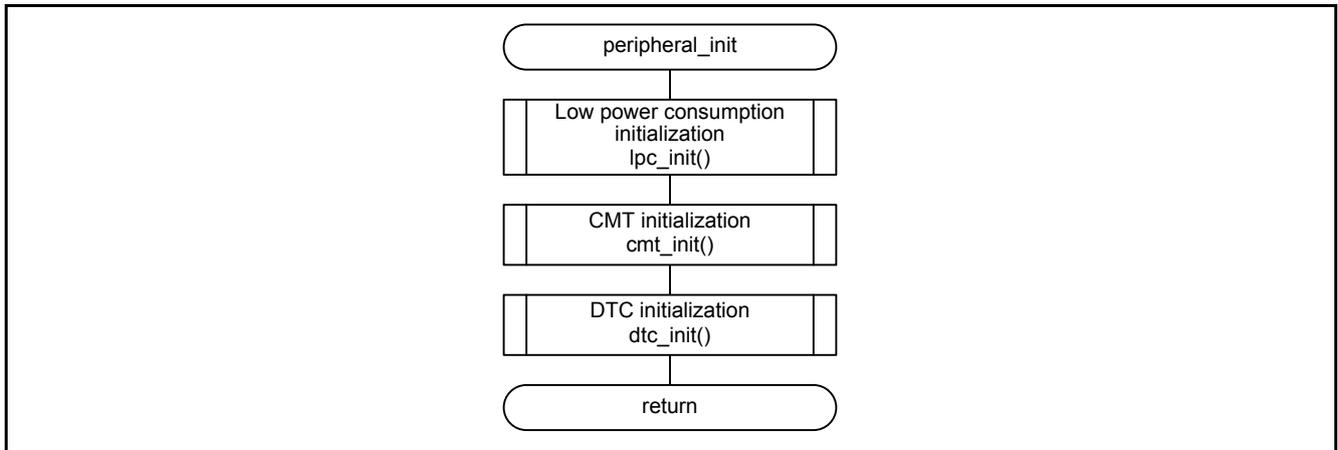


Figure 5.5 Peripheral Function Initialization

### 5.10.4 Low Power Consumption Initialization

Figure 5.6 shows the Low Power Consumption Initialization.

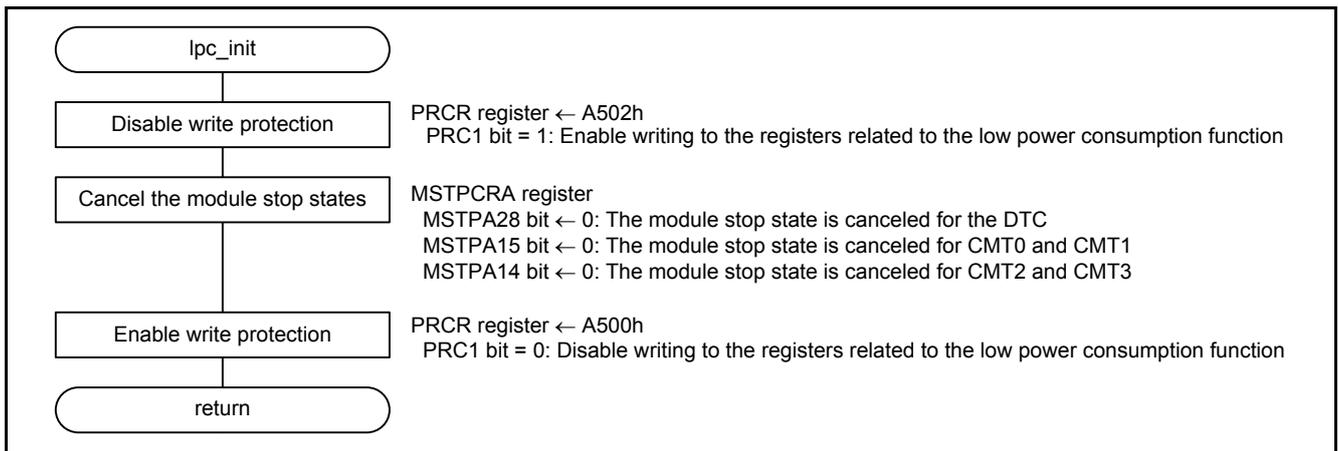


Figure 5.6 Low Power Consumption Initialization

5.10.5 CMT Initialization

Figure 5.7 shows the CMT Initialization.

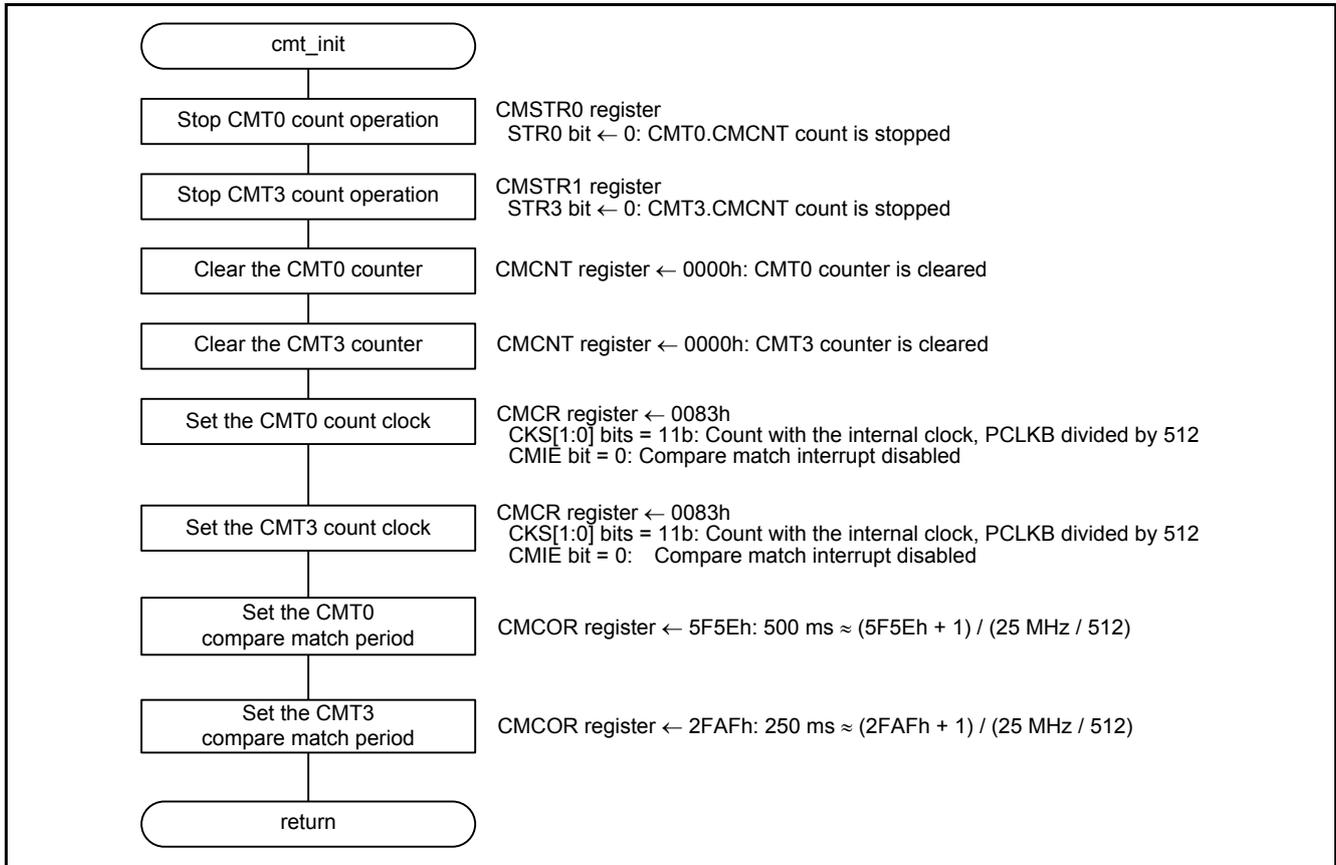


Figure 5.7 CMT Initialization

5.10.6 DTC Initialization

Figure 5.8 shows the DTC Initialization.

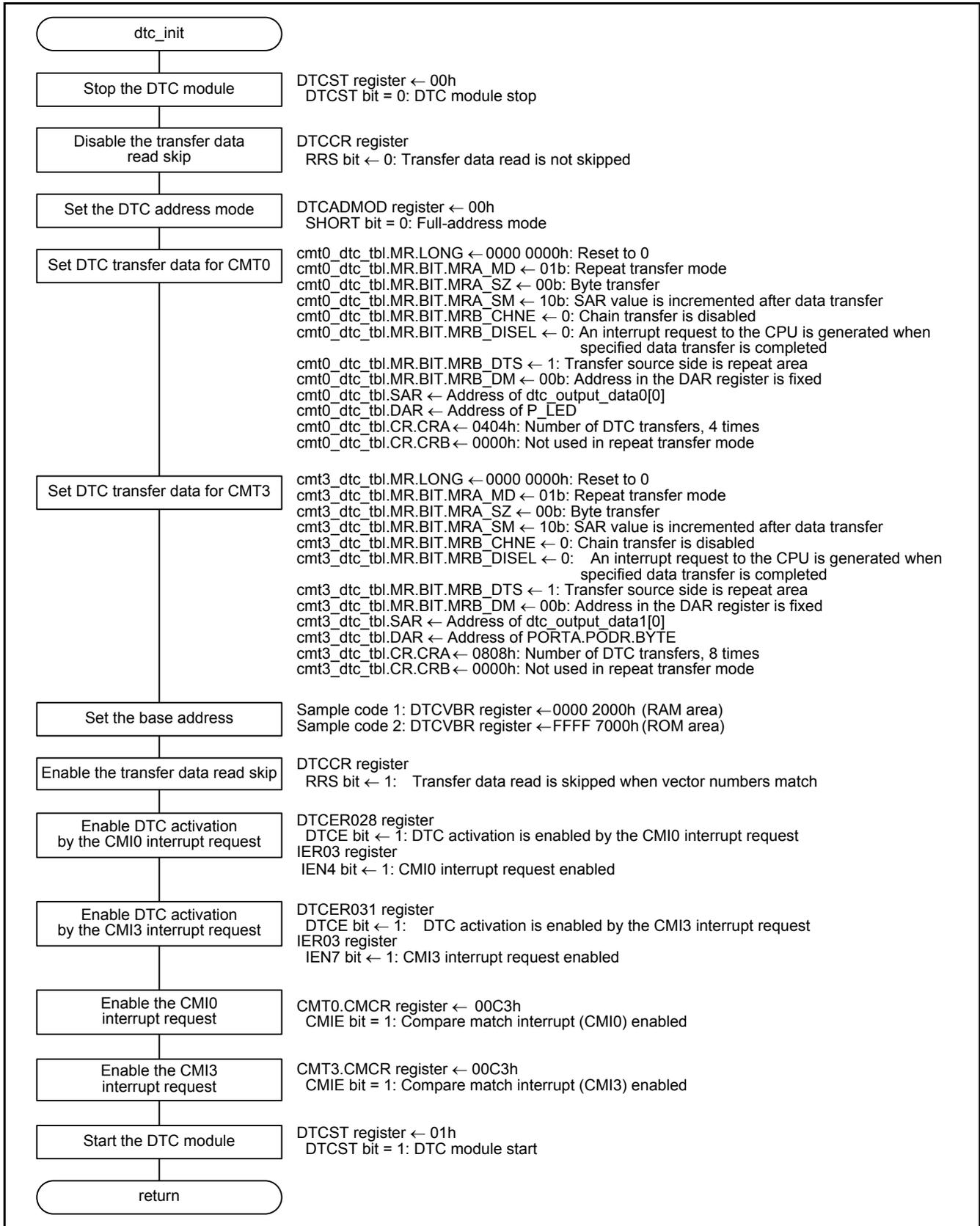


Figure 5.8 DTC Initialization

## 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.

## 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/>

<b>REVISION HISTORY</b>	RX210, RX21A, and RX220 Groups Application Note Efficient Allocation of the DTC Vector Table
-------------------------	---

Rev.	Date	Description	
		Page	Summary
1.00	Apr. 1, 2013	—	First edition issued
1.01	July 1, 2014	1	Products: Added the RX21A and RX220 Groups.
		6	2. Operation Confirmation Conditions: - Changed the Contents of the MCU used. - Changed the product part no. in the Contents of the Board used.
		6	3. Reference Application Notes: Added the Initial Setting application notes for the RX21A and RX220 Groups.
		15, 16	Modified the description of reference application note in the following functions: R_INIT_StopModule, R_INIT_NonExistentPort, and R_INIT_Clock.
		21	6. Applying This Application Note to the RX21A or RX220 Group: Added.
		22	8. Reference Documents: Added the User's Manual: Hardware for the RX21A and RX220 Groups.

All trademarks and registered trademarks are the property of their respective owners.

## 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 products 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.



### 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, Jin 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