

---

# RL78/G13

R01AN0737EJ0100

Rev. 1.00

## DMA Controller (UART Sequential Reception)

---

Feb. 02, 2012

### Introduction

This application note explains how to use the RL78/G13 DMA controller for sequential reception through the UART. The sample application covered in this application note stores data that is received using the UART communication in the on-chip RAM sequentially through the DMA controller.

### Target Device

RL78/G13

When applying the sample program covered in this application note to another microcomputer, modify the program according to the specifications for the target microcomputer and conduct an extensive evaluation of the modified program.

**Contents**

1. Specifications .....	3
2. Operation Check Conditions .....	5
3. Related Application Note.....	5
4. Description of the Hardware .....	6
4.1 Hardware Configuration Example .....	6
4.2 List of Pins to be Used .....	7
5. Description of the Software .....	8
5.1 Operation Outline .....	8
5.2 List of Option Byte Settings.....	9
5.3 List of Constants.....	9
5.4 List of Variables .....	10
5.5 List of Functions.....	10
5.6 Function Specifications .....	10
5.7 Flowcharts .....	12
5.7.1 Initialization Function.....	13
5.7.2 System Initialization Function .....	14
5.7.3 I/O Port Setup .....	15
5.7.4 CPU Clock Setup .....	17
5.7.5 Serial Array Unit 0 Setup .....	18
5.7.6 UART0 Setup .....	19
5.7.7 DMA Controller Initialization .....	20
5.7.8 Main Processing.....	26
5.7.9 UART0 Communication Start Processing.....	27
5.7.10 DMA Transfer Enable Processing .....	30
5.7.11 DMA0 transfer End Interrupt Processing .....	33
5.7.12 UART0 Reception Error Interrupt Processing .....	34
5.7.13 DMA0 Software Trigger Transfer Start Processing.....	35
6. Sample Code .....	36
7. Documents for Reference .....	36
Revision Record .....	37
General Precautions in the Handling of MPU/MCU Products .....	38

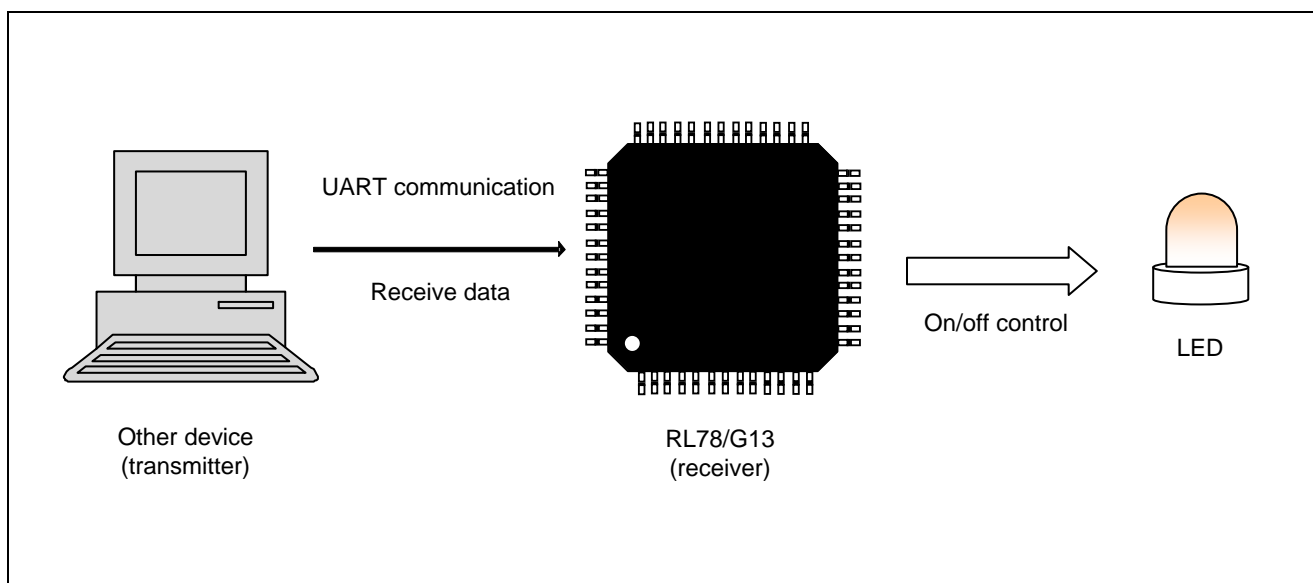
### 1. Specifications

This application note explains how to use the RL78/G13 DMA controller for sequential reception through the UART. The sample application covered in this application note receives ASCII data (one byte × five times) from an other device through UART communication using the serial array unit (SAU). The DMA controller is used to transfer the data from an SFR to the on-chip RAM. The receive data is stored in the on-chip RAM as triggered by the UART reception end interrupt. When the number of bytes to be transferred specified by the application (five times) is reached, the application regards the receive data as an ASCII code representing a five-digit decimal number and converts it into a numeric value. When the accumulated value of the receive data exceeds a threshold (100,000), the application turns on the LED.

Table 1.1 lists the peripheral functions and their uses. Figure 1.1 shows the outline of the operation. Figure 1.2 shows the timing chart of the DMA controller.

**Table 1.1 Peripheral Functions to be Used and their Uses**

Peripheral Function	Use
DMA controller	Transfers data received through the UART to the on-chip RAM.
Serial array unit	Used for UART reception.



**Figure 1.1 Outline of Operation**

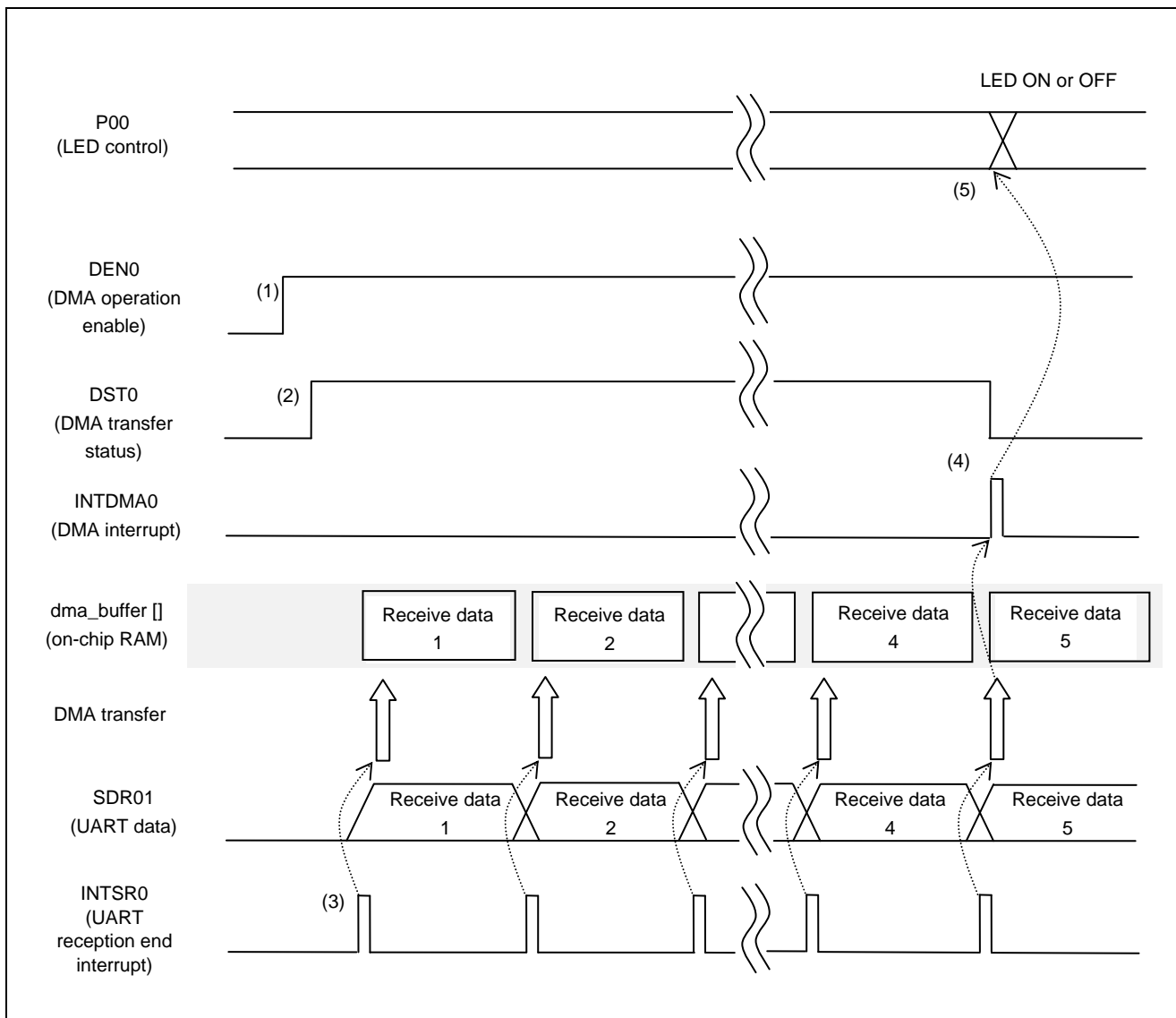


Figure 1.2 DMA Controller Timing Chart

- (1) Set DEN0 to 1 to enable the operation of DMA.
- (2) After making settings for the DMA controller, set DST0 to 1 to place the DAM controller in the DMA trigger wait mode.
- (3) Transfer the value of the serial data register 01 (SDR01) to dma\_buffer (on-chip RAM) using the UART reception transfer end interrupt (INTSR0) as a DMA trigger.
- (4) When the number of times of DMA transfer reaches the specified number (five), the DMA controller exits the DMA trigger wait mode (DST0 = 0) and a DMA interrupt (INTDMA0) is generated.
- (5) After a DMA interrupt (INTDMA0) occurs, the receive data stored in dma\_buffer is converted into a numeric value. Control the output at P00 according to the accumulated value of the receive data to turn on or off the LED.

## 2. Operation Check Conditions

The sample code described in this application note has been checked under the conditions listed in the table below.

**Table 2.1 Operation Check Conditions**

Item	Description
Microcontroller used	RL78/G13 (R5F100LEA)
Operating frequency	<ul style="list-style-type: none"><li>High-speed on-chip oscillator (HOCO) clock: 32 MHz</li><li>CPU/peripheral hardware clock: 32 MHz</li></ul>
Operating voltage	5.0 V (can run on a voltage range of 2.9 V to 5.5 V.) LVD operation ( $V_{LVI}$ ): Reset mode 2.81 V (2.76 V to 2.87 V)
Integrated development environment	CubeSuite + V1.00.01 from Renesas Electronics Corp.
C compiler	CA78K0R V1.20 from Renesas Electronics Corp.

## 3. Related Application Note

The application notes that are related to this application note are listed below for reference.

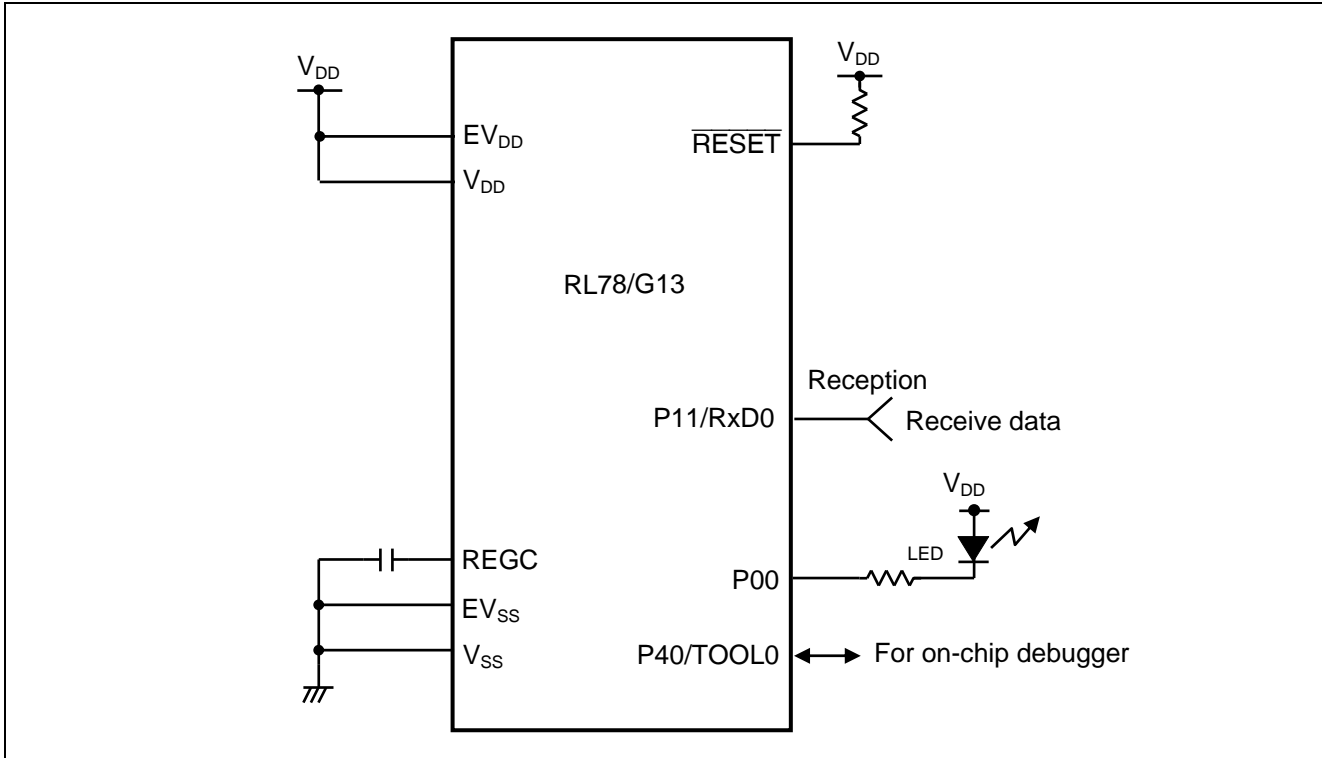
RL78/G13 Initialization (R01AN0451E) Application Note

RL78/G13 Serial Array Unit (UART Communication) (R01AN0459E) Application Note

## 4. Description of the Hardware

### 4.1 Hardware Configuration Example

Figure 4.1 shows an example of hardware configuration that is used for this application note.



**Figure 4.1 Hardware Configuration**

- Cautions:
1. The purpose of this circuit is only to provide the connection outline and the circuit is simplified accordingly. When designing and implementing an actual circuit, provide proper pin treatment and make sure that the hardware's electrical specifications are met (connect the input-only ports separately to  $V_{DD}$  or  $V_{SS}$  via a resistor).
  2. Connect any pins whose name begins with  $EV_{SS}$  to  $V_{SS}$  and any pins whose name begins with  $EV_{DD}$  to  $V_{DD}$ , respectively.
  3.  $V_{DD}$  must be held at not lower than the reset release voltage ( $V_{LVI}$ ) that is specified as LVD.

## 4.2 List of Pins to be Used

Table 4.1 lists the pins to be used and their functions.

**Table 4.1 Pins to be Used and their Functions**

Pin Name	I/O	Description
P11/RxD0	Input	Serial data reception pin
P00	Output	LED lighting control port

## 5. Description of the Software

### 5.1 Operation Outline

The sample application covered in this application note uses the DMA controller to store the data that is received through the UART in the on-chip RAM. Upon the completion of the transfer of 5-byte data by the DMA controller, the application converts the receive data stored in the on-chip RAM to a 5-digit numeric value. It turns on the LED when the accumulated value of the receive data exceeds 100,000 and turns off otherwise.

(1) Initialize the DMA controller.

<Conditions for Setting>

- Set the DMA transfer direction to "SFR to on-chip RAM."
- Use the UART reception transfer end interrupt request (INTSR0) as the DMA startup source.
- Set the transfer data size to 8 bits.
- Select the SDR01 address (0x000FFF12) as the address of the transfer source SFR.
- Set the transfer destination RAM address to the start address of variable dma\_buffer [].
- Set the number of times of transfer to five.

(2) Initialize the SAU0.

<Conditions for setting>

- Use the SAU0 in UART mode, receive only.
- Use even parity as the parity setting.
- Set the data transfer order to LSB first.
- Set the data length to 8 bits.
- Set the receive data level to standard.
- Set the transfer rate to 9600 bps
- Use the error interrupt (INTSRE0).
- Set the priority level of the INTSRE0 interrupt to low.
- Use the UART reception transfer end interrupt (INTSR0) as the interrupt source.

(3) Set operation start trigger of channel 1 (SS01 bit) to 1 to place the UART in the communication wait status. Mask the transfer end interrupt (SRMK0 = 1) to disable INTSR0 interrupt processing. The vector interrupt processing for error interrupts (INTSRE0) is enabled to prevent overrun errors due to reception errors.

(4) Place the DMA controller in the trigger wait mode.

(5) Execute the HALT instruction to turn on the HALT mode and wait for a DMA transfer end interrupt (INTDMA0).

(6) The DMA controller updates the receive data on each occurrence of a transfer end interrupt request (INTSR0).



- (7) When an error interrupt (INTSRE0) occurs, the error status is stored within the vector interrupt processing and DMA transfer is carried out via a software trigger.
- (8) The HALT mode is exited on the occurrence of a DMA transfer end interrupt (INTDMA0) (even if a UART reception error occurs, a DMA0 transfer end interrupt (INTDMA0) is generated after the transfer of 5-byte data including the byte that is received when the error occurred is completed). Subsequently, the error status is examined. If no UART reception error has occurred, the processing described in step (9) is performed. If a UART reception error has occurred, the error status is cleared and the processing described in step (11) is performed.
- (9) Convert the receive data (5 bytes) into a numeric value regarding it as a 5-digit decimal number represented in ASCII code.
- (10) When the accumulated value computed in step (9) exceeds 100,000, the accumulated value is cleared and the LED which is connected to P00 is turned on. Otherwise, the accumulated value is retained and the LED connected to P00 is turned off.
- (11) Initialize the DMA setting and wait for a trigger again.
- (12) Turn on the HALT mode again and wait for DMA transfer end interrupt (INTDMA0).

## 5.2 List of Option Byte Settings

Table 5.1 summarizes the settings of the option bytes.

**Table 5.1 Option Byte Settings**

Address	Value	Description
000C0H/010C0H	11101111B	Disables the watchdog timer. (Stops counting after the release from the reset status.)
000C1H/010C1H	01111111B	LVD reset mode which uses 2.81 V (2.76 V to 2.87 V)
000C2H/010C2H	11101000B	HS mode, HOCO: 32 MHz
000C3H/010C3H	10000100B	Enables the on-chip debugger.

## 5.3 List of Constants

Table 5.2 lists the constants that are used in this sample program.

**Table 5.2 Constants for the Sample Program**

Constant	Setting	Description
_0002_SAU_CH1_START_TRG_ON	0x0002	Setting to enable the operation of serial channel enable status register 0 (SE0)
TOTAL_LIMIT	100000	Upper limit of total received values
_0005_DMA0_BYTE_COUNT	0x0005	Number of times of DMA transfer
P_LED	P0.0	LED lighting control port

## 5.4 List of Variables

Table 5.3 lists the global variables that are used by this sample program.

**Table 5.3 Global Variables**

Type	Variable Name	Contents	Function Used
uint8_t	g_DmaUartError	UART reception error variable	R_DMAC0_Create_UserInit R_DMAC0_Interrupt R_UART0_Interrupt_Error
uint8_t	dma_buffer[5]	Data receive buffer	R_DMAC0_Create_UserInit R_DMAC0_Interrupt
uint32_t	rx_total	Receive data accumulated value buffer	R_DMAC0_Create_UserInit R_DMAC0_Interrupt

## 5.5 List of Functions

Table 5.4 lists the functions that are used in this sample program.

**Table 5.4 Functions**

Function Name	Outline
R_UART0_MaskStart	UART0 communication start processing (interrupt masking)
R_DMAC0_Start	DMA0 transfer enabling
R_DMAC0_Interrupt	DMA0 transfer end interrupt
R_UART0_Interrupt_Error	UART0 reception error interrupt
R_DMAC0_Set_SoftwareTriggerOn	DMA0 software trigger transfer start processing

## 5.6 Function Specifications

Shown below are the functions that are used in this sample program.

[Function Name] R\_UART0\_MaskStart

Synopsis	UART0 communication start processing (interrupt masking)
Header	r_cg_serial.h
Declaration	void R_UART0_MaskStart(void)
Explanation	This function starts UART communication with the UART0 transfer end interrupt (INTSR0) masked.
Arguments	None
Return value	None
Remarks	Although R_UART0_Start which serves a similar purpose is available, this function is used because it is necessary to mask the required interrupt.

---

[Function Name] R\_DMACH0\_Start

---

Synopsis	DMA0 transfer enabling
Header	r_cg_dmac.h
Declaration	void R_DMACH0_Start(void)
Explanation	This function starts controlling DMA transfer. It performs the following processing: <ul style="list-style-type: none"> <li>• Clears the end interrupt request.</li> <li>• Enables DMA transfer end interrupts.</li> <li>• Enables the DMA transfer and switches the CPU to the DMA transfer wait status.</li> </ul>
Arguments	None
Return value	None
Remarks	None

---

[Function Name] R\_DMACH0\_Interrupt

---

Synopsis	DMA0 transfer end interrupt
Header	r_cg_dmac.h
Declaration	__interrupt void R_DMACH0_Interrupt(void)
Explanation	This function performs the interrupt processing when the specified number of DMA transfer have been performed. The function adds the receive data to the past accumulated value. When the specified count of 100,000 is exceeded, the function clears the accumulated value and turns on the LED that is connected to P00. Subsequently, the function restarts the DMA.
Arguments	None
Return value	None
Remarks	None

---

[Function Name] R\_UART0\_Interrupt\_Error

---

Synopsis	UART0 reception error interrupt
Header	None
Declaration	__interrupt void R_UART0_Interrupt_Error(void)
Explanation	This function clears the error status and reads the contents of the SDR01 register.
Arguments	None
Return value	None
Remarks	None

---

[Function Name] R\_DMACH0\_Set\_SoftwareTriggerOn

---

Synopsis	DMA0 software trigger transfer start processing
Header	None
Declaration	void R_DMACH0_Set_SoftwareTriggerOn(void)
Explanation	This function sets up the DMA0 software trigger and starts DMA transfer (once).
Arguments	None
Return value	None
Remarks	None

## 5.7 Flowcharts

Figure 5.1 shows the overall flow of the sample program described in this application note.

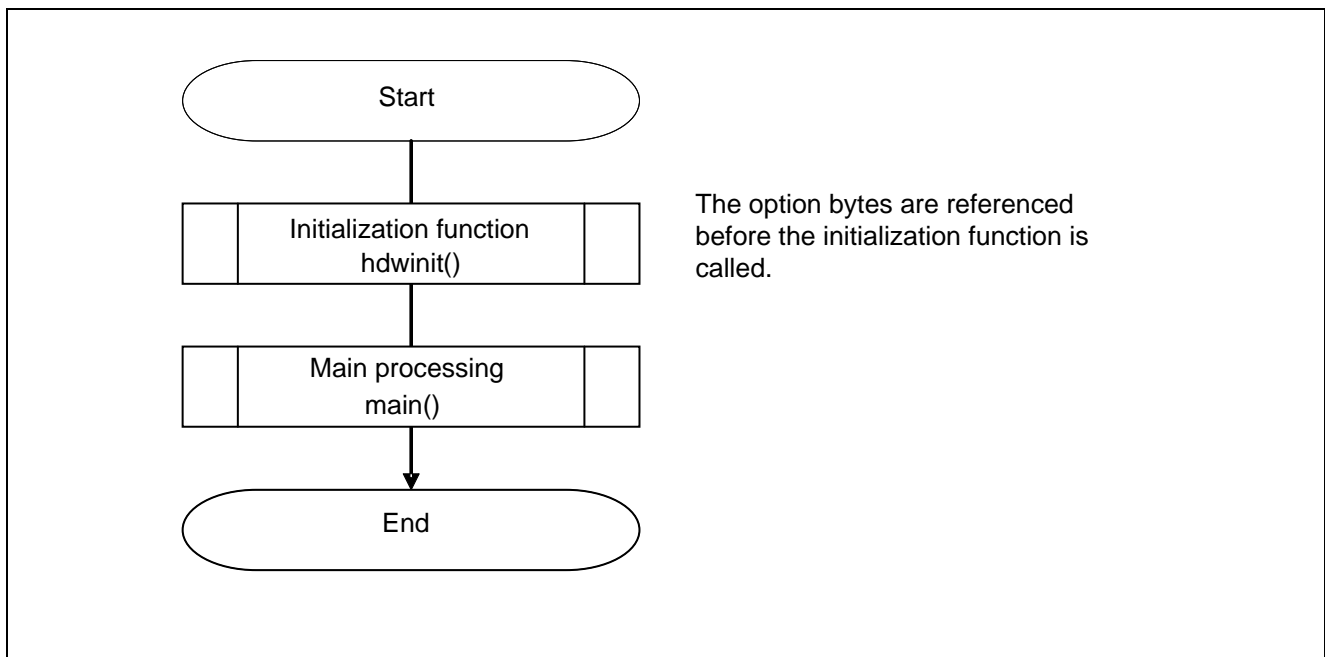


Figure 5.1 Overall Flow

### 5.7.1 Initialization Function

Figure 5.2 shows the flowchart for the initialization function.

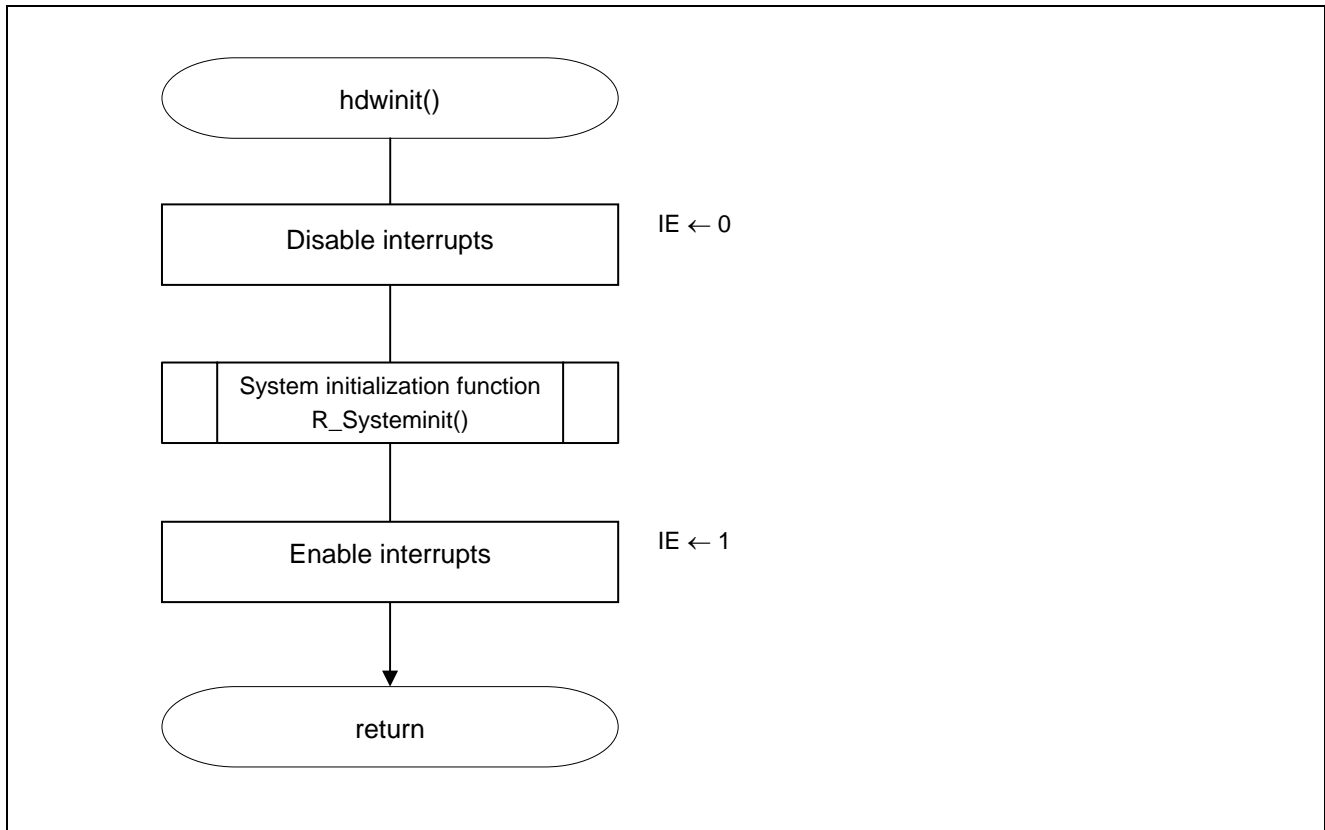
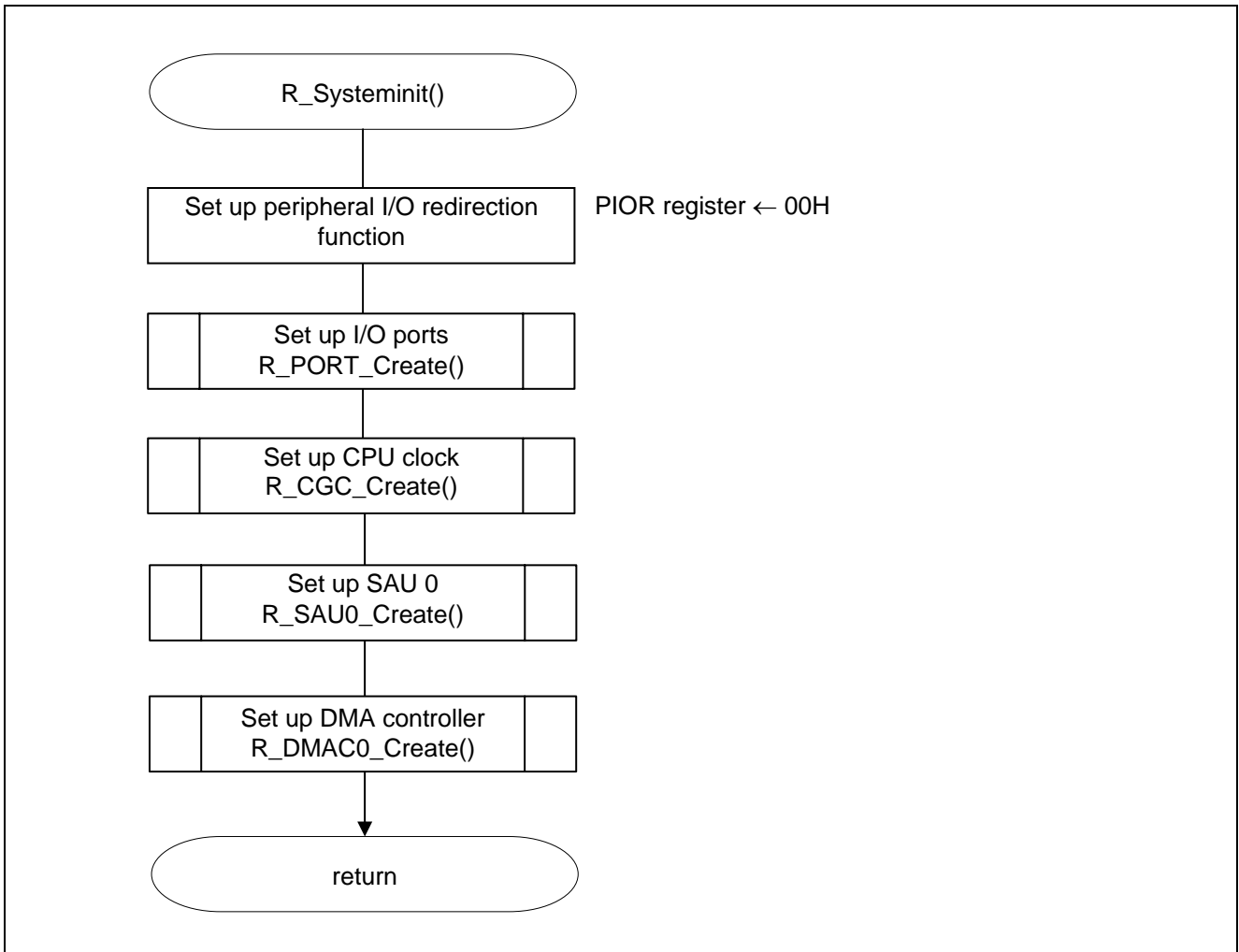


Figure 5.2 Initialization Function

**5.7.2 System Initialization Function**

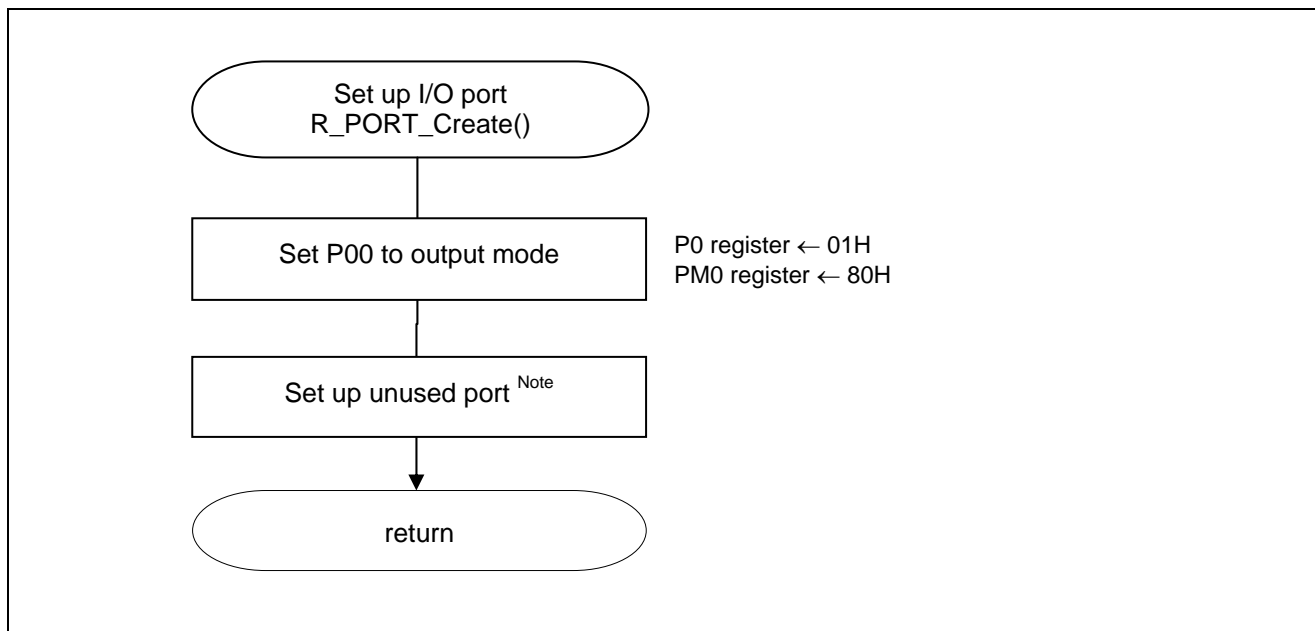
Figure 5.3 shows the flowchart for the system initialization function.



**Figure 5.3 System Initialization Function**

### 5.7.3 I/O Port Setup

Figure 5.4 shows the flowchart for setting up the I/O ports.



**Figure 5.4 I/O Port Setup**

Note: Refer to the section entitled "Flowcharts" in RL78/G13 Initialization Application Note (R01AN0451E) for the configuration of the unused ports.

Caution: Provide proper treatment for unused pins so that their electrical specifications are met. Connect each of any unused input-only ports to  $V_{DD}$  or  $V_{SS}$  via a separate resistor.

Setting up the LED port

- Port register 0 (P0)
- Port mode register 0 (PM0)

Symbol: P0

7	6	5	4	3	2	1	0
P07	P06	P05	P04	P03	P02	P01	P00
0	0	0	0	0	0	0	<b>1</b>

Bit 0

P00	P00 pin output data control (in output mode)
0	Output 0
<b>1</b>	<b>Output 1</b>

Symbol: PM0

7	6	5	4	3	2	1	0
PM07	PM06	PM05	PM04	PM03	PM02	PM01	PM00
1	0	0	0	0	0	0	<b>0</b>

Bit 0

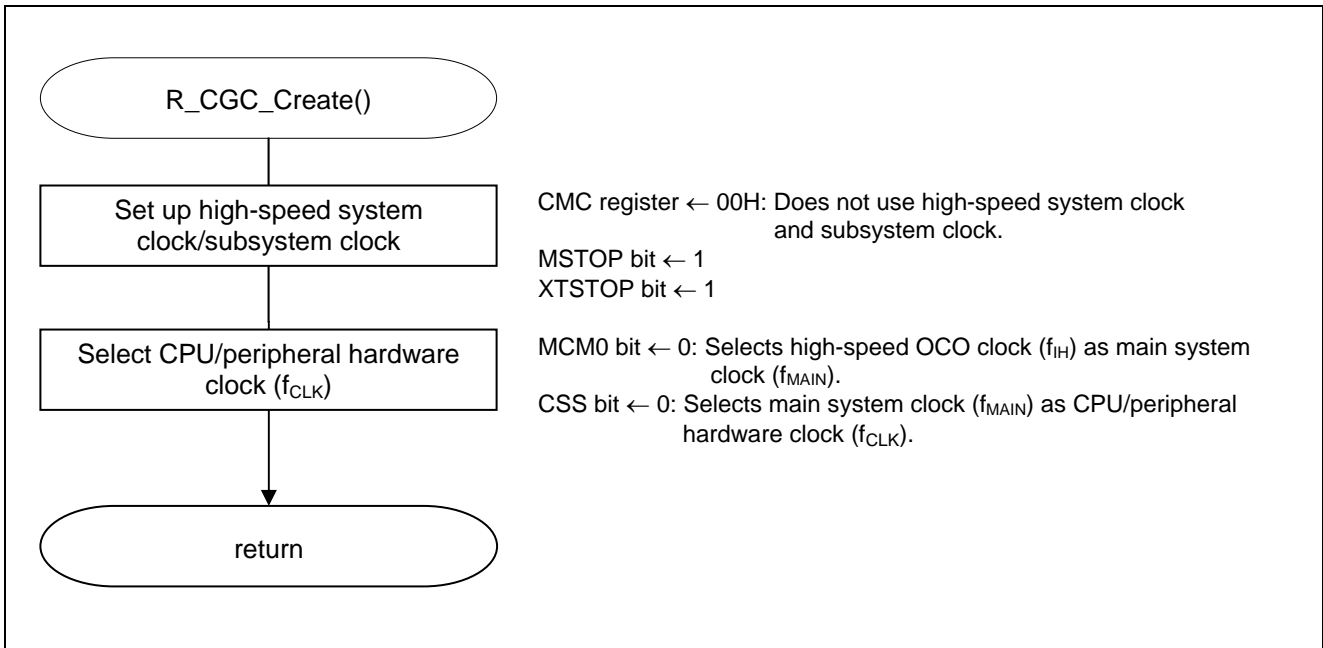
PM00	P00 pin I/O mode selection
<b>0</b>	<b>Output mode (output buffer on)</b>
1	Input mode (output buffer off)

Caution: For details on the register setup procedures, refer to RL78/G13 User's Manual: Hardware.



### 5.7.4 CPU Clock Setup

Figure 5.5 shows the flowchart for setting up the CPU clock.

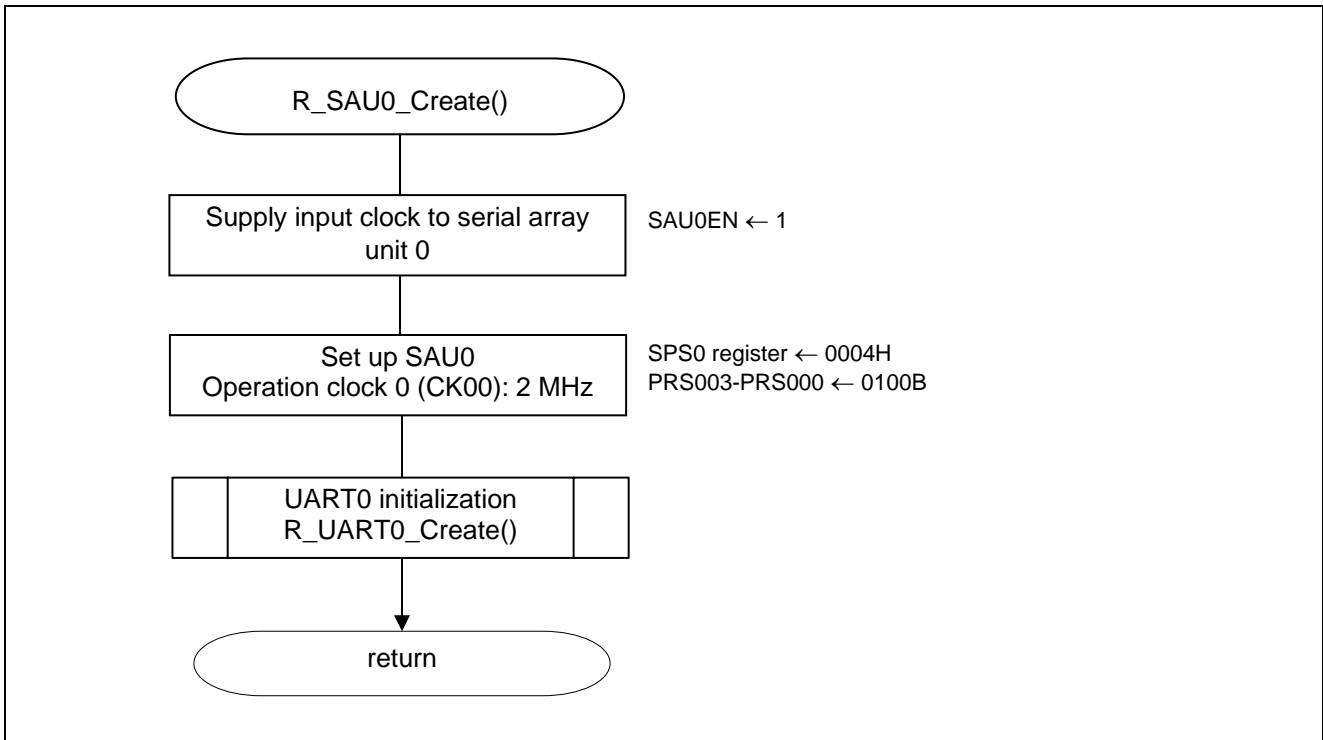


**Figure 5.5 CPU Clock Setup**

**Caution:** For details on the procedure for setting up the CPU clock (`R_CGC_Create()`), refer to the section entitled "Flowcharts" in RL78/G13 Initialization Application Note (R01AN0451E).

### 5.7.5 Serial Array Unit 0 Setup

Figure 5.6 shows the flowchart for setting up the serial array unit 0.



**Figure 5.6 Serial Array Unit 0 Setup**

**Caution:** For details on the procedure for setting up the SAU0 (`R_SAU0_Create()`), refer to the section entitled "Flowcharts" in RL78/G13 Serial Array Unit (UART Communication) Application Note (R01AN0459E).

5.7.6 UART0 Setup

Figure 5.7 shows the flowchart for setting up the UART0.

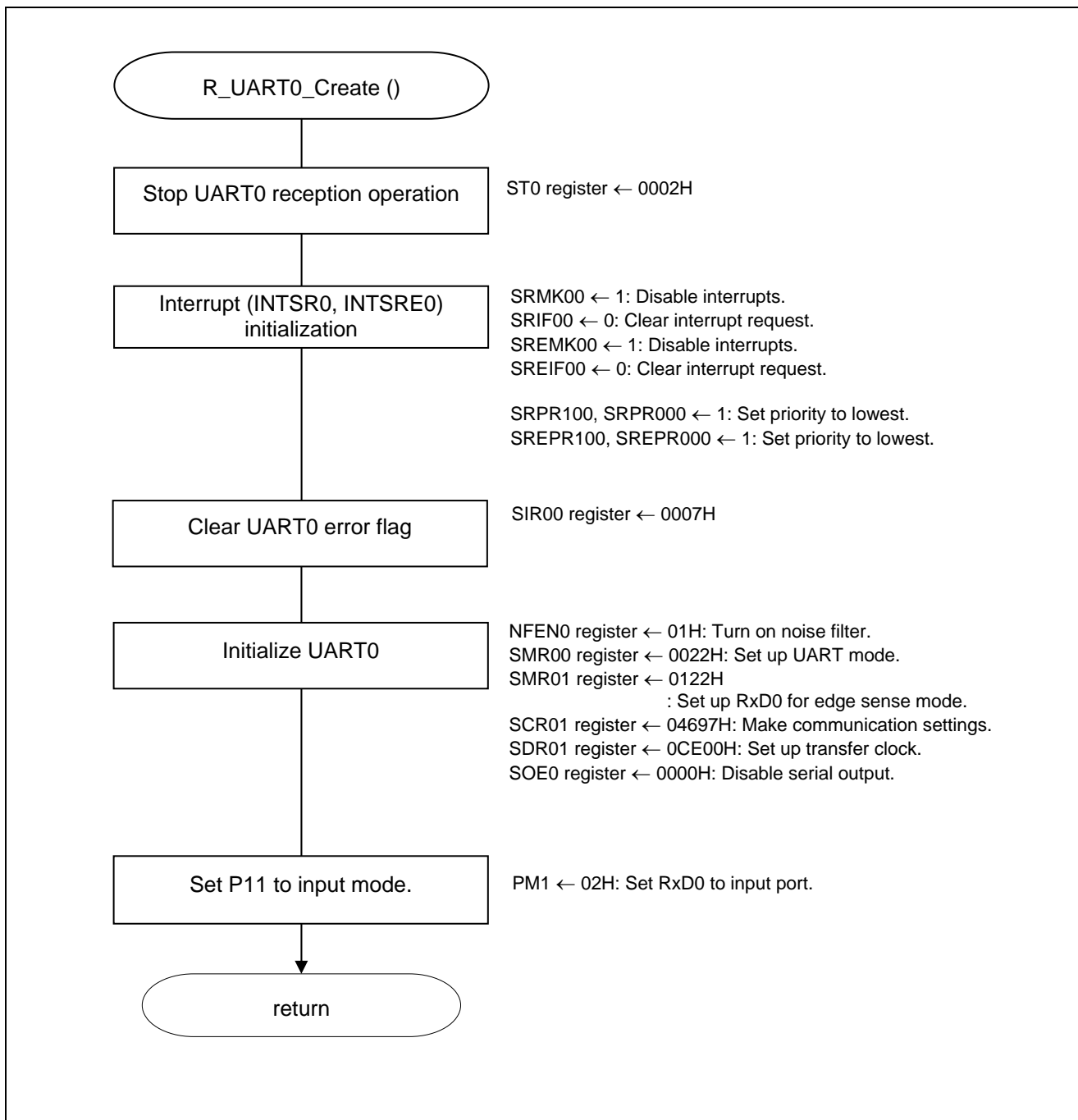


Figure 5.7 UART0 Setup

Caution: For details on the procedure for setting up the UART0 (R\_UART0\_Create ()), refer to the section entitled "Flowcharts" in RL78/G13 Serial Array Unit (UART Communication) Application Note (R01AN0459E).

### 5.7.7 DMA Controller Initialization

Figure 5.8 shows the flowchart for initializing the DMA controller.

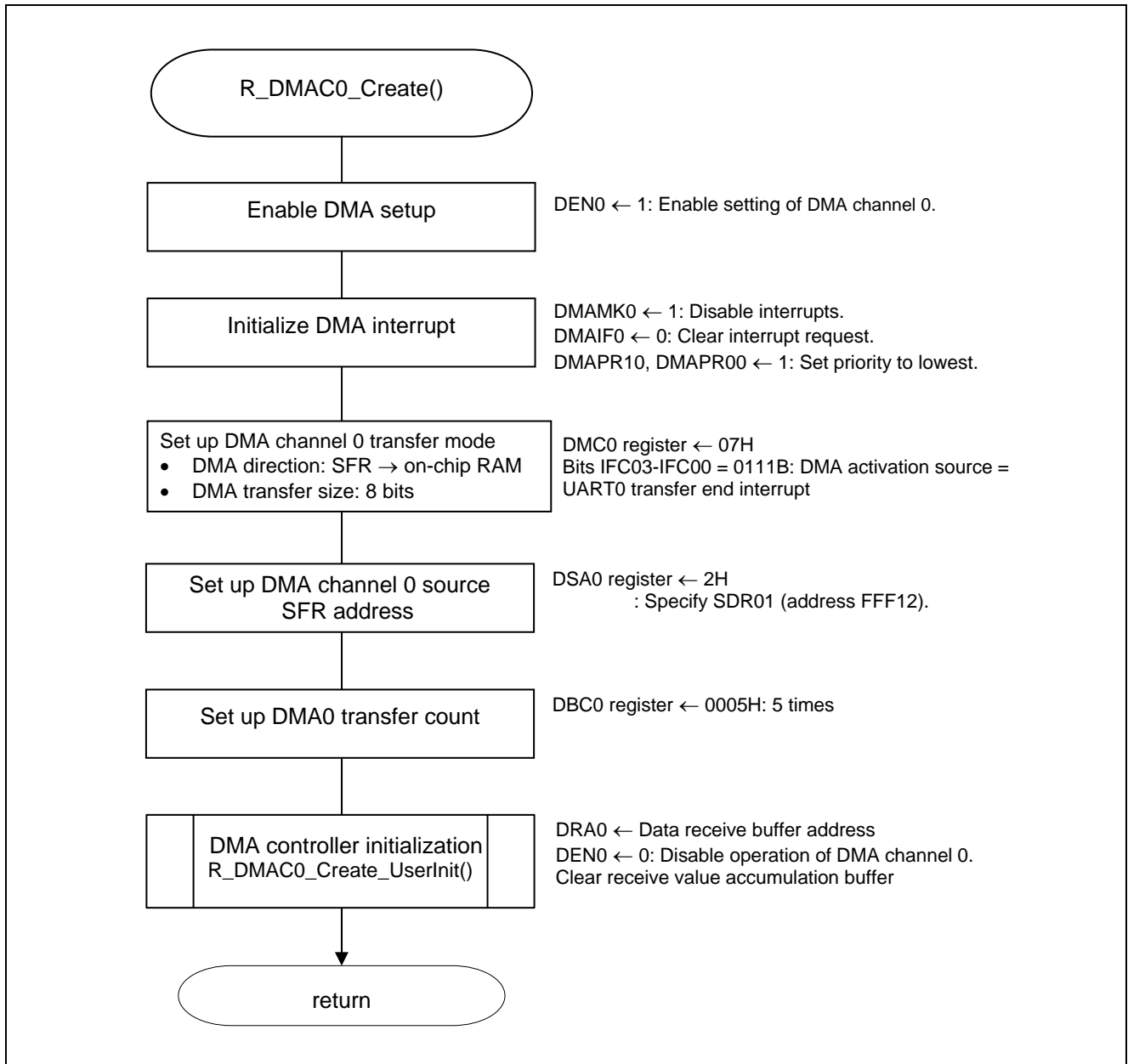


Figure 5.8 DMA Controller Initialization

## Disabling DMA channel 0

- DMA operation control register (DRC0)

Symbol: DRC0

7	6	5	4	3	2	1	0
DEN0	0	0	0	0	0	0	DST0
1/0	0	0	0	0	0	0	0

## Bit 7

DEN0	DMA operation enable flag
0	<b>Disables operation of DMA channel 0 (stops operating clock of DMA).</b> Disables DMA setup processing.
1	Enables operation of DMA channel 0. <b>Enables DMA setup processing.</b>

## Bit 0

DST0	DMA transfer mode flag
0	<b>DMA transfer of DMA channel 0 is completed.</b>
1	DMA transfer of DMA channel 0 is not completed (still under execution).

## Initializing DMA transfer end interrupts

- Interrupt request flag register (IFOH)  
Clear interrupt request flag.
- Interrupt mask flag register (MK0H)  
Clear interrupt mask.
- Priority specification flag register (PR00H, PR10H)  
Interrupt level = Level 3 (lowest level)

Symbol: IFOH

7	6	5	4	3	2	1	0
SREIF0 TMIF01H	SRIF0 CSIF01 IICIF01	STIF0 CSIF00 IICIF00	DMAIF1	DMAIF0	SREIF2 TMIF11H	SRIF2 CSIF21 IICIF21	STIF2 CSIF20 IICIF20
x	x	x	x	0	x	x	x

## Bit 3

DMAIF0	Interrupt request flag
0	<b>No interrupt request signal is generated</b>
1	Interrupt request is generated, interrupt request status

Caution: For details on the register setup procedures, refer to RL78/G13 User's Manual: Hardware.

Symbol: MK0H

7	6	5	4	3	2	1	0
SREMK0	SRMK0	STMK0			SREMK2	SRMK2	STMK2
TMMK01	CSIMK01	CSIMK00	DMAMK1	DMAMK0	TMMK11	CSIMK21	CSIMK20
H	IICMK01	IICMK00			H	IICMK21	IICMK20
x	x	x	x	<b>1</b>	x	x	x

Bit 3

DMAMK0	Interrupt processing control
0	Enables interrupt processing.
<b>1</b>	<b>Disables interrupt processing.</b>

Symbol: PR00H

7	6	5	4	3	2	1	0
SREPR00	SRPR00	STPR00			SREPR02	SRPR02	STPR02
TMPR001	CSIPR001	CSIPR000	DMAPR01	DMAPR00	TMPR011	CSIPR021	CSIPR020
H	IICPR001	IICPR000			H	IICPR021	IICPR020
x	x	x	x	<b>1</b>	x	x	x

Symbol: PR10H

7	6	5	4	3	2	1	0
SREPR10	SRPR10	STPR10			SREPR12	SRPR12	STPR12
TMPR101	CSIPR101	CSIPR100	DMAPR11	DMAPR10	TMPR111	CSIPR121	CSIPR120
H	IICPR101	IICPR100			H	IICPR121	IICPR120
x	x	x	x	<b>1</b>	x	x	x

Bit 3

DMAPR10	DMAPR00	Priority level selection
0	0	Specify level 0 (highest level)
0	1	Specify level 1
1	0	Specify level 2
<b>1</b>	<b>1</b>	<b>Specify level 3 (lowest level)</b>

Caution: For details on the register setup procedures, refer to RL78/G13 User's Manual: Hardware.

## Setting up DMA channel 0 transfer mode

- DMA mode control register (DMC0)  
Use no software trigger.  
Set DMA transfer direction to SFR to on-chip RAM.  
Set transfer data size to 8 bits.  
Specify DMA transfer on DMA startup request.  
Select UART0 transfer end interrupt as DMA startup source.

Symbol: DMC0

7	6	5	4	3	2	1	0
STG0	DRS0	DS0	DWAIT0	IFC03	IFC02	IFC01	IFC00
0	0	0	0	0	1	1	1

Bit 7

STG0	DMA transfer start software trigger
0	No trigger operation
1	DMA transfer is started when DMA operation is enabled (DEN0 = 1).

Bit 6

DRS0	Selection of DMA transfer direction
0	SFR to on-chip RAM
1	On-chip RAM to SFR

Bit 5

DS0	Specification of transfer data size for DMA transfer
0	8 bits
1	16 bits

Bit 4

DWAIT0	Pending of DMA transfer
0	Executes DMA transfer upon DMA start request (no held pending).
1	Holds DMA start request pending if any.

Bits 3 to 0

IFC03	IFC02	IFC01	IFC00	Selection of DMA start source	
				Trigger Signal	Trigger contents
0	0	0	0	—	Disables DMA transfer by interrupt. (Only software trigger is enabled.)
0	0	0	1	INTAD	A/D conversion end interrupt
0	0	1	0	INTTM00	End of timer channel 0 count end or capture end interrupt
0	0	1	1	INTTM01	End of timer channel 1 count end or capture end interrupt
0	1	0	0	INTTM02	End of timer channel 2 count end or capture end interrupt
0	1	0	1	INTTM03	End of timer channel 3 count end or capture end interrupt
0	1	1	0	INTST0/INTCSI00	UART0 transmission transfer end or buffer empty interrupt/CSI00 transfer end or buffer empty interrupt
<b>0</b>	<b>1</b>	<b>1</b>	<b>1</b>	<b>INTSR0/INTCSI01</b>	<b>UART0 reception transfer end interrupt/CSI01 transfer end or buffer empty interrupt</b>
1	0	0	0	INTST1/INTCSI10	UART1 transmission transfer end or buffer empty interrupt/CSI10 transfer end or buffer empty interrupt
1	0	0	1	INTSR1/INTCSI11	UART1 reception transfer end interrupt/CSI11 transfer end or buffer empty interrupt
1	0	1	0	INTST2/INTCSI20	UART2 transmission transfer end or buffer empty interrupt/CSI20 transfer end or buffer empty interrupt
1	0	1	1	INTSR2/INTCSI21	UART2 reception transfer end interrupt/CSI21 transfer end or buffer empty interrupt
Other than above				Setting prohibited	

Caution: For details on the register setup procedures, refer to RL78/G13 User's Manual: Hardware.



## Setting up DMA channel 0 transfer SFR

- DMA SFR address register 0 (DSA0)  
Set up the source SFR of DMA transfer.

Symbol: DSA0

7	6	5	4	3	2	1	0
0	0	0	1	0	0	1	0

Set the lower 8 bits of RXD0/SIO01 (SFR address: 0x000FFF12).

Caution: For details on the register setup procedures, refer to RL78/G13 User's Manual: Hardware.

## Setting up DMA channel 0 destination RAM address

- DMA RAM address register 0 (DRA0)  
Set up the RAM address of DMA transfer destination.

Symbol: DRA0

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

Set the start address of array dma\_buffer[.].

Caution: For details on the register setup procedures, refer to RL78/G13 User's Manual: Hardware.

## Setting up DMA channel 0 transfer count

- DMA byte count register 0 (DBC0)  
Specify DMA transfer count.

Symbol: DBC0

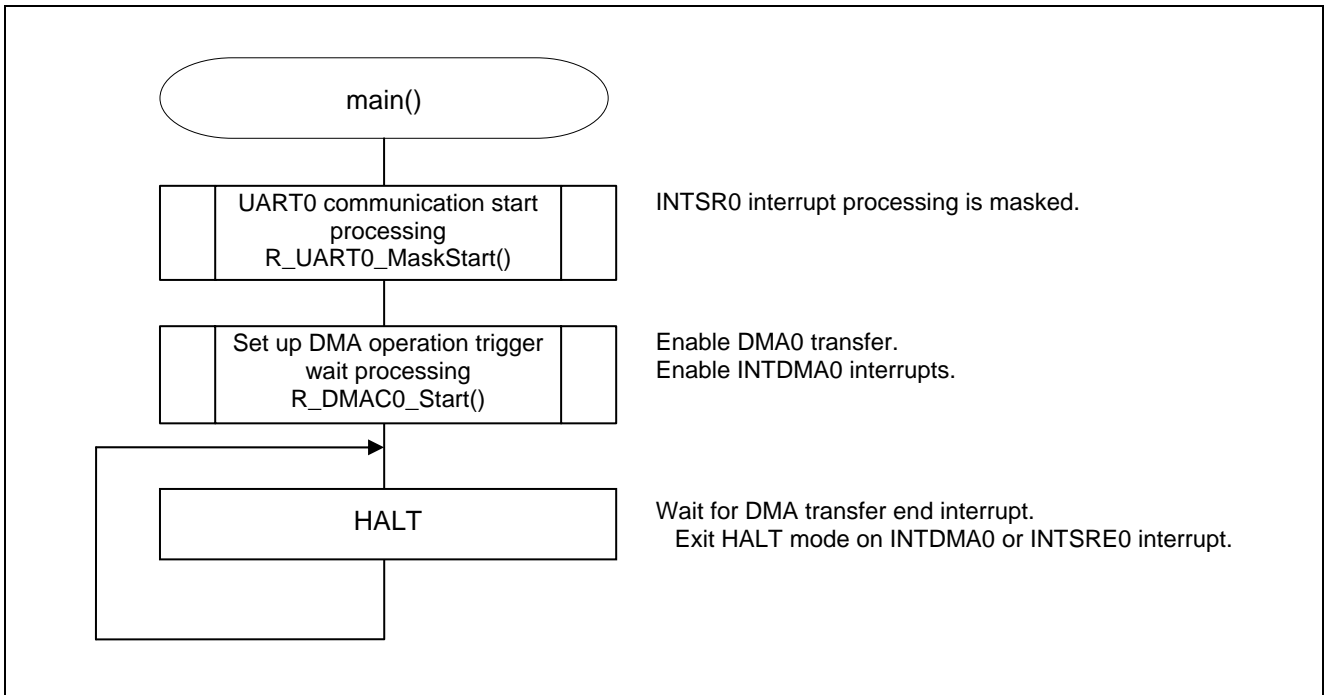
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1

Set the number of times of DMA transfer to 5.

Caution: For details on the register setup procedures, refer to RL78/G13 User's Manual: Hardware.

**5.7.8 Main Processing**

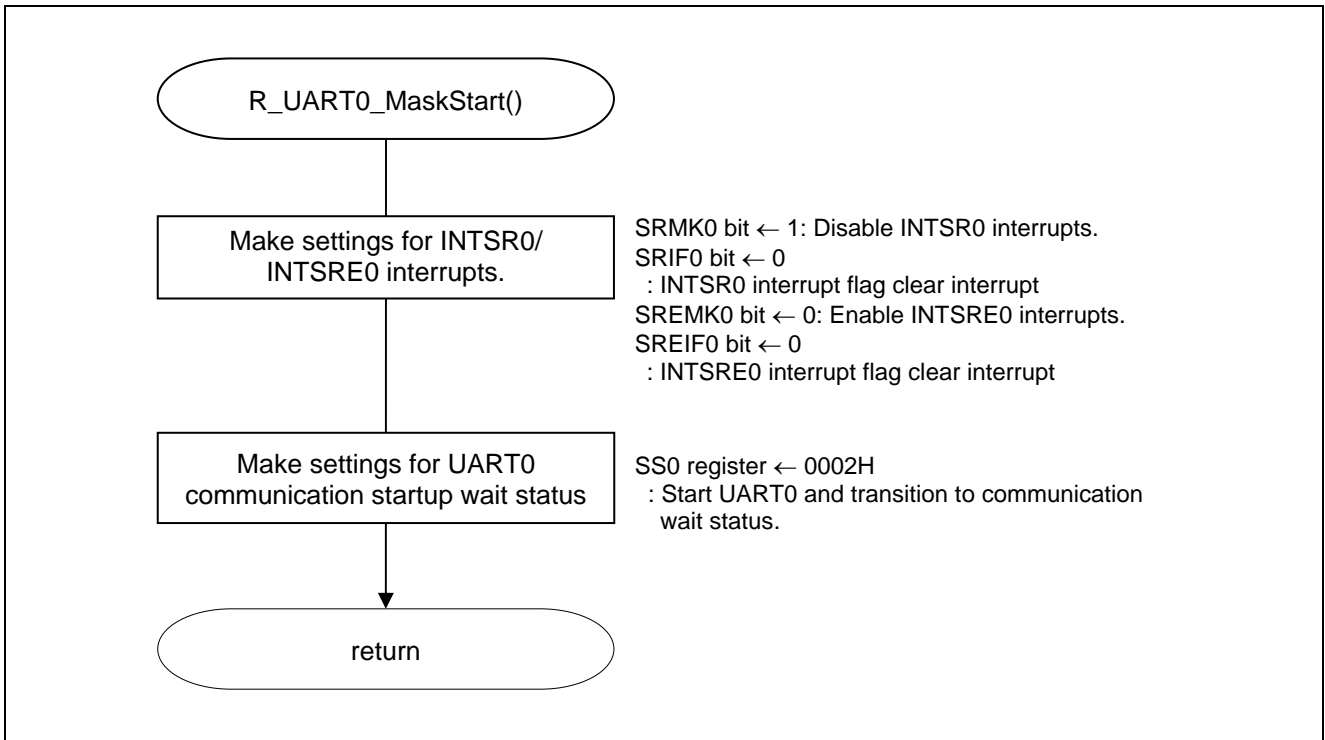
Figure 5.9 shows the flowchart for main processing.



**Figure 5.9 Main Processing**

**5.7.9 UART0 Communication Start Processing**

Figure 5.10 shows the flowchart for UART0 communication start processing.



**Figure 5.10 UART0 Communication Start Processing**

Preparing for enabling DMA transfer end interrupts

- Interrupt request flag register (IF0H)  
Clear interrupt request flags.
- Interrupt mask flag register (MK0H)  
Clear interrupt masks.

Symbol: IF0H

7	6	5	4	3	2	1	0
SREIF0 TMIF01H	SRIF0 CSIF01 IICIF01	STIF0 CSIF00 IICIF00	DMAIF1	DMAIF0	SREIF2 TMIF11H	SRIF2 CSIF21 IICIF21	STIF2 CSIF20 IICIF20
0	0	x	x	x	x	x	x

Bit 7

SREIF0	Interrupt request flag
0	No interrupt request signal is generated
1	Interrupt request is generated, interrupt request status

Bit 6

SRIF0	Interrupt request flag
0	No interrupt request signal is generated
1	Interrupt request is generated, interrupt request status

Symbol: MK0H

7	6	5	4	3	2	1	0
SREMK0 TMMK01H	SRMK0 CSIMK01 IICMK01	STMK0 CSIMK00 IICMK00	DMAMK1	DMAMK0	SREMK2 TMMK11H	SRMK2 CSIMK21 IICMK21	STMK2 CSIMK20 IICMK20
0	1	x	x	x	x	x	x

Bit 7

SREMK0	Interrupt processing control
0	Enables interrupt processing.
1	Disables interrupt processing.

Bit 6

SRMK0	Interrupt processing control
0	Enables interrupt processing.
1	Disables interrupt processing.

Caution: For details on the register setup procedures, refer to RL78/G13 User's Manual: Hardware.

## Starting serial channel 0

- Serial channel start register 0 (SS0)  
Start serial channel 0 communication/counting.

Symbol: SS0

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	0	0	0	0	0	0	0	0	0	0	0	SS03	SS02	SS01	SS00
0	0	0	0	0	0	0	0	0	0	0	0	x	x	<b>1</b>	x

Bit 1

SS01	Operation start trigger of channel 1
0	No trigger operation
<b>1</b>	<b>Sets the SE01 bit to 1 and enters the communication wait status.</b>

Caution: For details on the register setup procedures, refer to RL78/G13 User's Manual: Hardware.

### 5.7.10 DMA Transfer Enable Processing

Figure 5.11 shows the flowchart for the DMA transfer enable processing.

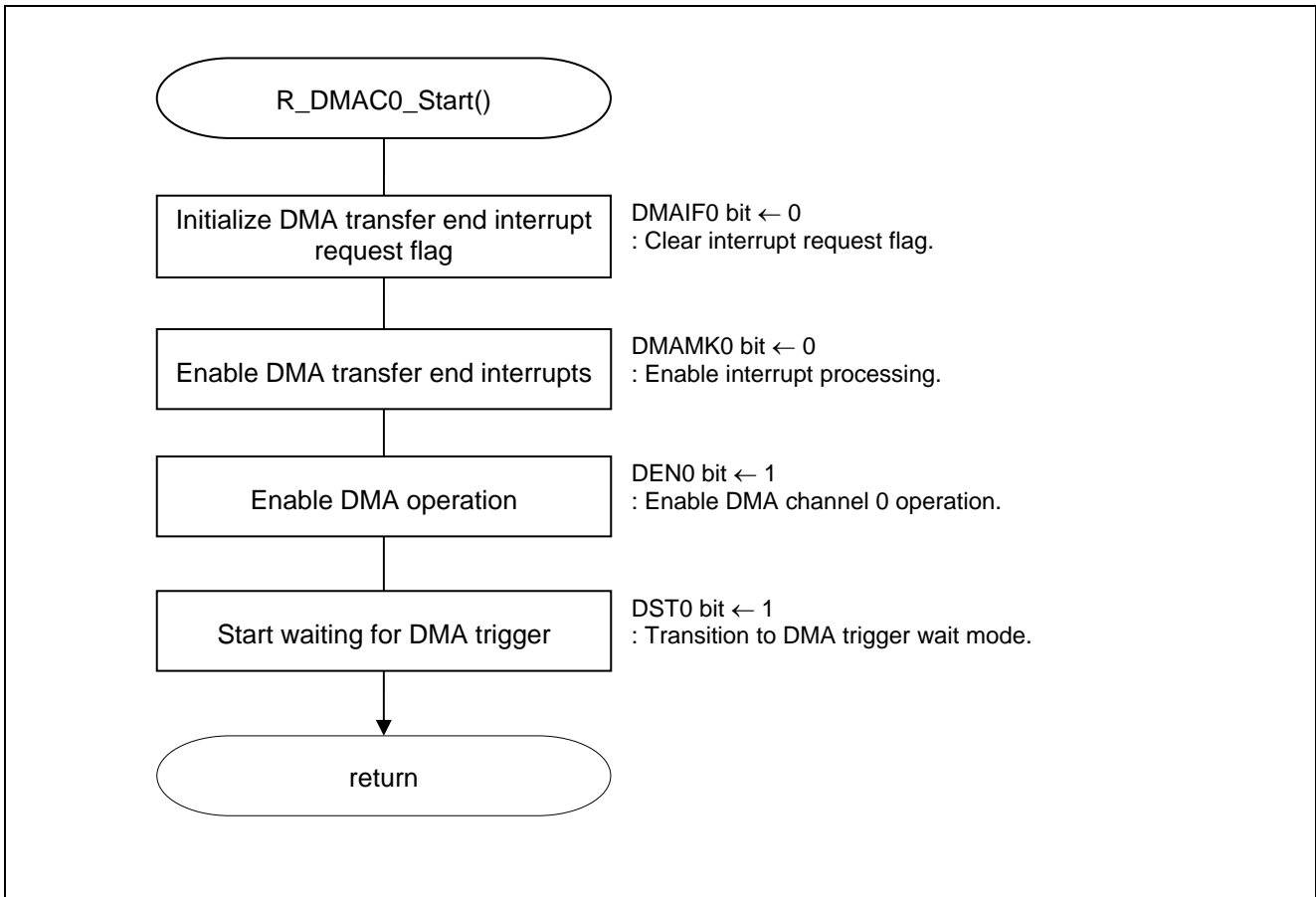


Figure 5.11 DMA Transfer Enable Processing

Preparing for DMA transfer end interrupt enable processing

- Interrupt request flag register (IF0H)  
Clear interrupt request flags.
- Interrupt mask flag register (MK0H)  
Clear interrupt masks.

Symbol: IF0H

7	6	5	4	3	2	1	0
SREIF0 TMIF01H	SRIF0 CSIF01 IICIF01	STIF0 CSIF00 IICIF00	DMAIF1	DMAIF0	SREIF2 TMIF11H	SRIF2 CSIF21 IICIF21	STIF2 CSIF20 IICIF20
x	x	x	x	<b>0</b>	x	x	x

Bit 3

DMAIF0	Interrupt request flag
<b>0</b>	<b>No interrupt request signal is generated</b>
1	Interrupt request is generated, interrupt request status

Symbol: MK0H

7	6	5	4	3	2	1	0
SREMK0 TMMK01H	SRMK0 CSIMK01 IICMK01	STMK0 CSIMK00 IICMK00	DMAMK1	DMAMK0	SREMK2 TMMK11H	SRMK2 CSIMK21 IICMK21	STMK2 CSIMK20 IICMK20
x	x	x	x	<b>0</b>	x	x	x

Bit 3

DMAMK0	Interrupt processing control
<b>0</b>	<b>Enables interrupt processing.</b>
1	Disables interrupt processing.

Caution: For details on the register setup procedures, refer to RL78/G13 User's Manual: Hardware.

Setting up DMA channel 0 operation trigger wait mode

- DMA operation control register (DRC0)

Symbol: DRC0

	7	6	5	4	3	2	1	0
DEN0	0	0	0	0	0	0	0	DST0
	1	0	0	0	0	0	0	1

Bit 7

DEN0	DMA operation enable flag
0	Disables operation of DMA channel 0 (stops operating clock of DMA). Disables DMA setup processing.
1	<b>Enables operation of DMA channel 0.</b> <b>Enables DMA setup processing.</b>

Bit 0

DST0	DMA transfer mode flag
0	DMA transfer of DMA channel 0 DMA is completed.
1	<b>DMA transfer of DMA channel 0 is not completed (still under execution).</b>

The DMA trigger wait mode is entered by setting DST0 to 1 after enabling DMA operation (DEN0 = 1).

Caution: For details on the register setup procedures, refer to RL78/G13 User's Manual: Hardware.



5.7.11 DMA0 transfer End Interrupt Processing

Figure 5.12 shows the flowchart for the DMA0 transfer end interrupt processing.

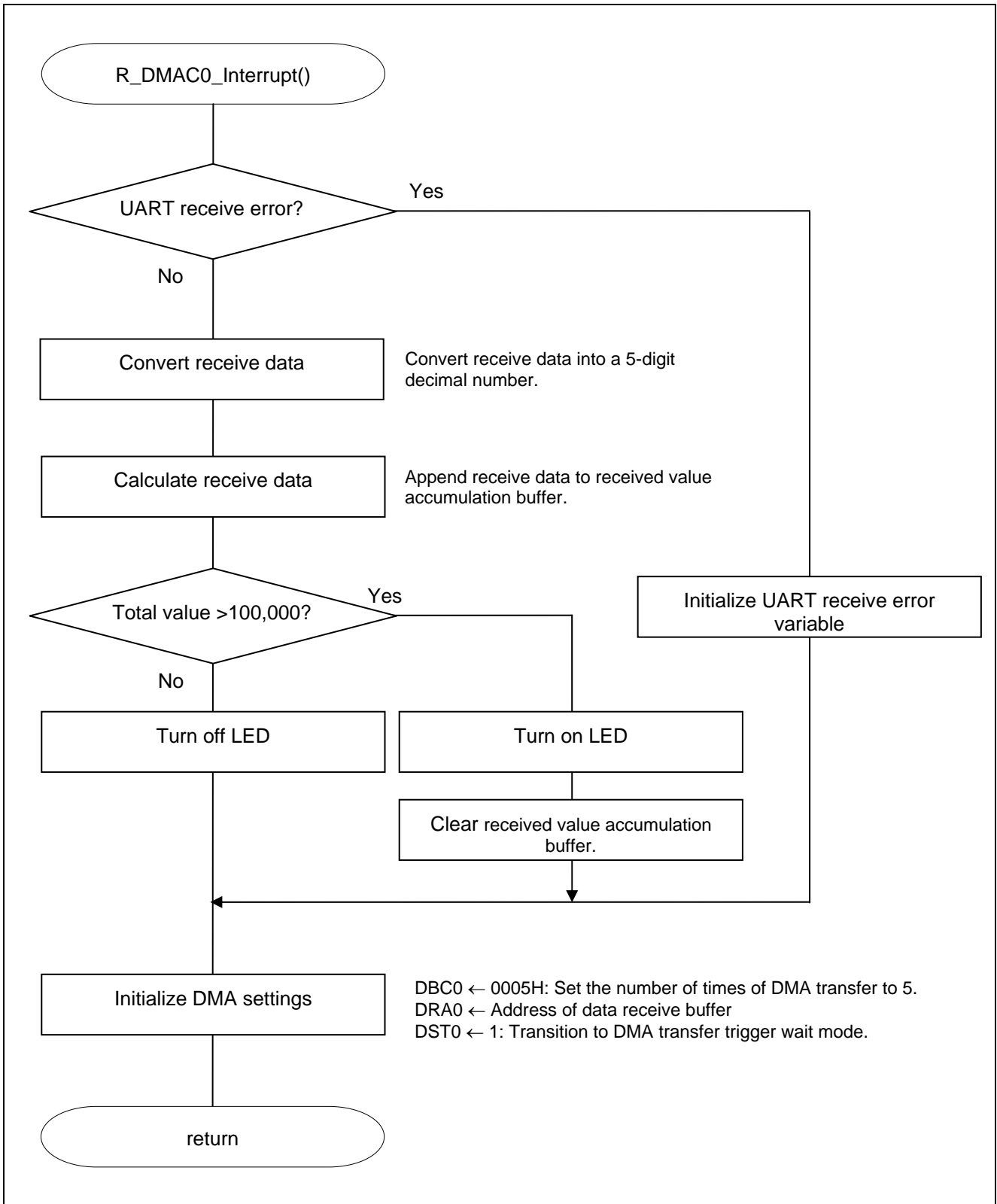
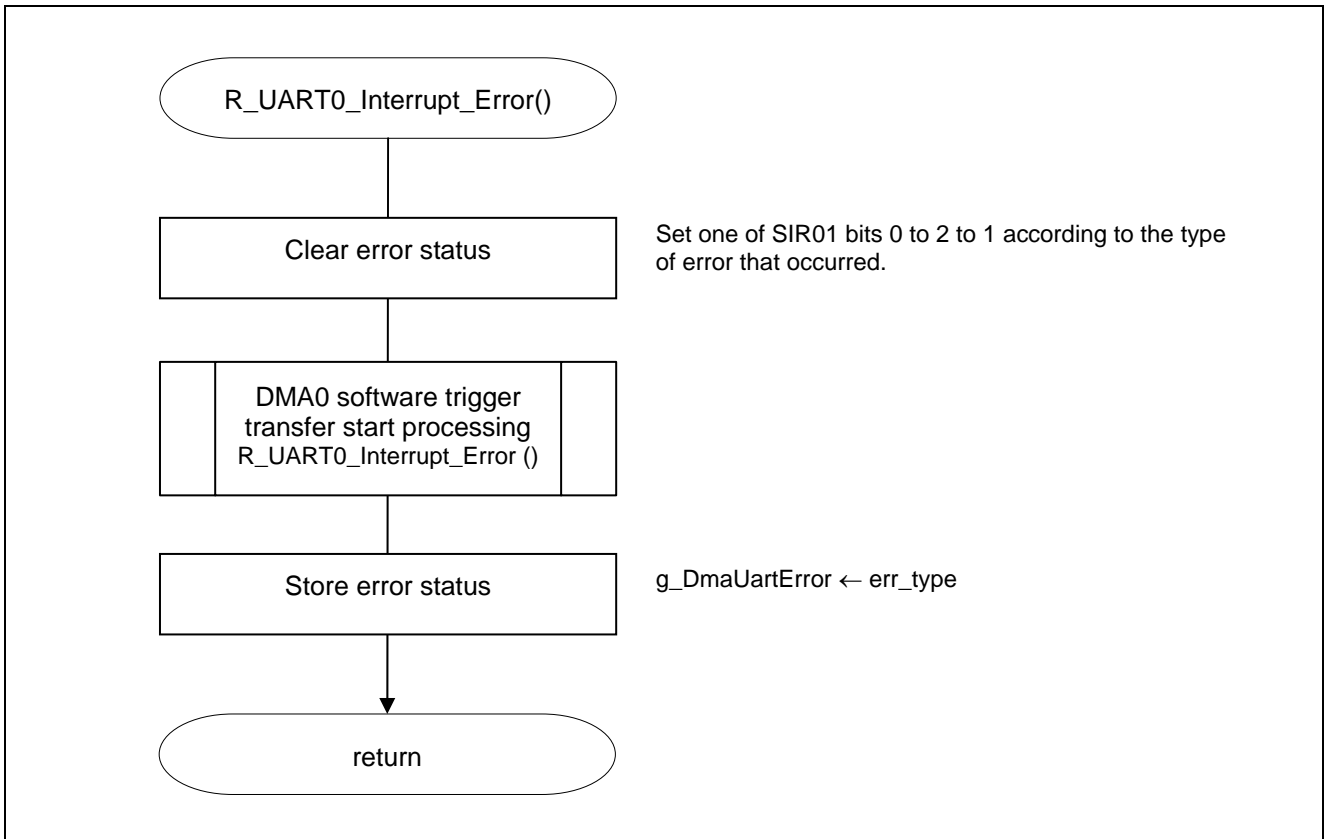


Figure 5.12 DMA0 Transfer End Interrupt Processing

**5.7.12 UART0 Reception Error Interrupt Processing**

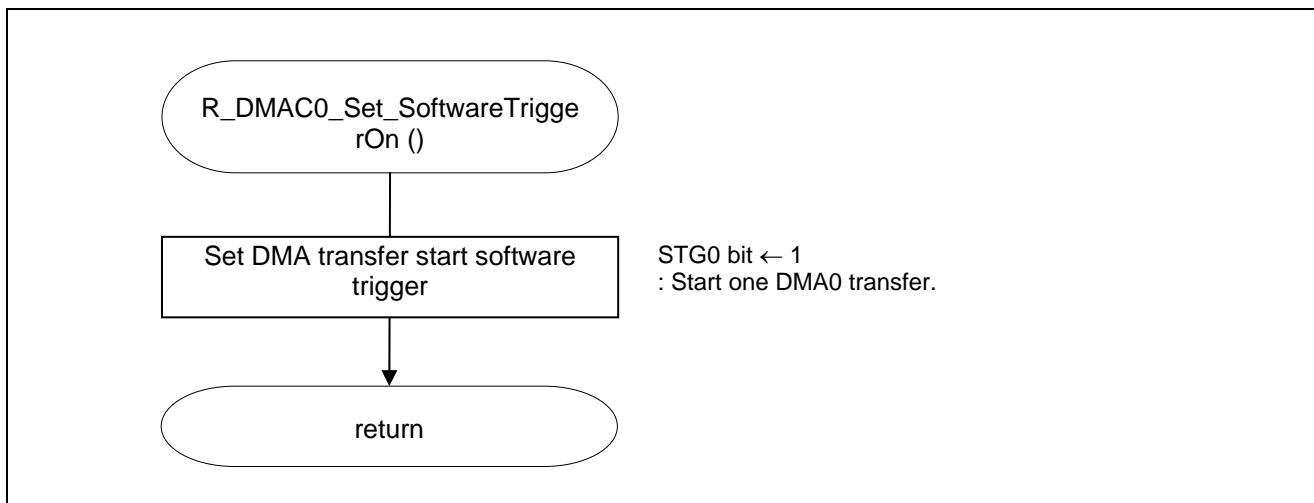
Figure 5.13 shows the flowchart for the UART0 reception error interrupt processing.



**Figure 5.13 UART0 Reception Error Interrupt Processing**

### 5.7.13 DMA0 Software Trigger Transfer Start Processing

Figure 5.14 shows the flowchart for the DMA0 software trigger transfer start processing.



**Figure 5.14 DMA0 Software Trigger Transfer Start Processing**

Setting up DMA channel 0 transfer start software trigger

- DMA mode control register (DMC0)  
Set up DMA transfer start software trigger.

Symbol: DMC0

7	6	5	4	3	2	1	0
STG0	DRS0	DS0	DWAIT0	IFC03	IFC02	IFC01	IFC00
1	0	0	0	0	1	1	1

Bit 7

STG0	DMA transfer start software trigger
0	No trigger operation
1	<b>DMA transfer is started when DMA transfer operation is enabled (DEN0 = 1).</b>

Caution: For details on the register setup procedures, refer to RL78/G13 User's Manual: Hardware.

## 6. Sample Code

The sample code is available on the Renesas Electronics Website.

## 7. Documents for Reference

RL78/G13 User's Manual: Hardware (R01UH0146E)

RL78 Family User's Manual: Software (R01US0015E)

(The latest versions of the documents are available on the Renesas Electronics Website.)

Technical Updates/Technical Brochures

(The latest versions of the documents are available on the Renesas Electronics Website.)

## Website and Support

Renesas Electronics Website

- <http://www.renesas.com/index.jsp>

Inquiries

- <http://www.renesas.com/contact/>

Revision Record	RL78/G13 DMA Controller (UART Sequential Reception)
-----------------	---

Rev.	Date	Description	
		Page	Summary
1.00	Feb. 02, 2012	—	First edition issued

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 manual, refer to the relevant sections of the manual. If the descriptions under General Precautions in the Handling of MPU/MCU Products and in the body of the manual differ from each other, the description in the body of the manual takes precedence.

### 1. Handling of Unused Pins

- Handle unused pins in accord 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 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 one with a different part number, confirm that the change will not lead to problems.
  - The characteristics of MPU/MCU in the same group but having different part numbers may differ because of the differences in internal memory capacity and layout pattern. When changing to products of different part numbers, implement a system-evaluation test for each of the products.

## Notice

1. All information included in this document is current as of the date this document is issued. Such information, however, is subject to change without any prior notice. Before purchasing or using any Renesas Electronics products listed herein, please confirm the latest product information with a Renesas Electronics sales office. Also, please pay regular and careful attention to additional and different information to be disclosed by Renesas Electronics such as that disclosed through our website.
2. 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.
3. You should not alter, modify, copy, or otherwise misappropriate any Renesas Electronics product, whether in whole or in part.
4. 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.
5. When exporting the 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. You should not use Renesas Electronics products or the 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. 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.
6. 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.
7. Renesas Electronics products are classified according to the following three quality grades: "Standard", "High Quality", and "Specific". The recommended applications for each Renesas Electronics product depends on the product's quality grade, as indicated below. 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 categorized as "Specific" without the prior written consent of Renesas Electronics. Further, you may not use any Renesas Electronics product for any application for which it is not intended without the prior written consent of Renesas Electronics. 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 an application categorized as "Specific" or for which the product is not intended where you have failed to obtain the prior written consent of Renesas Electronics. The quality grade of each Renesas Electronics product is "Standard" unless otherwise expressly specified in a Renesas Electronics data sheets or data books, etc.  
"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.  
"High Quality": Transportation equipment (automobiles, trains, ships, etc.); traffic control systems; anti-disaster systems; anti-crime systems; safety equipment; and medical equipment not specifically designed for life support.  
"Specific": Aircraft; aerospace equipment; submersible repeaters; nuclear reactor control systems; medical equipment or systems for life support (e.g. artificial life support devices or systems), surgical implantations, or healthcare intervention (e.g. excision, etc.), and any other applications or purposes that pose a direct threat to human life.
8. 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.
9. 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 system manufactured by you.
10. 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.
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.**  
2880 Scott Boulevard Santa Clara, CA 95050-2554, 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: +1-905-898-9441, 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: +49-211-65030, Fax: +49-211-6503-1327

**Renesas Electronics (China) Co., Ltd.**  
7th Floor, Quantum Plaza, No.27 ZhiChunLu Haidian District, Beijing 100083, P.R.China  
Tel: +86-10-8235-1155, Fax: +86-10-8235-7679

**Renesas Electronics (Shanghai) Co., Ltd.**  
Unit 204, 205, AZIA Center, No.1233 Lujiazui Ring Rd., Pudong District, Shanghai 200120, China  
Tel: +86-21-5877-1818, Fax: +86-21-6887-7858 / -7898

**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: +852-2886-9318, Fax: +852 2886-9022/9044

**Renesas Electronics Taiwan Co., Ltd.**  
13F, No. 363, Fu Shing North Road, Taipei, Taiwan  
Tel: +886-2-8175-9600, Fax: +886 2-8175-9670

**Renesas Electronics Singapore Pte. Ltd.**  
1 harbourFront Avenue, #06-10, Keppel Bay Tower, Singapore 098632  
Tel: +65-6213-0200, Fax: +65-6278-8001

**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.**  
11F., Samik Lavied' or Bldg., 720-2 Yeoksam-Dong, Kangnam-Ku, Seoul 135-080, Korea  
Tel: +82-2-558-3737, Fax: +82-2-558-5141