

---

## RL78/G13

R01AN0459EJ0200

Rev. 2.00

2014.03.24

### Serial Array Unit (UART Communication)

---

#### Introduction

This application note explains how to use UART communication through the serial array unit (SAU). ASCII characters transmitted from the device on the opposite side are analyzed to make responses.

#### 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 .....	6
5. Description of the Software .....	7
5.1 Operation Outline .....	7
5.2 List of Option Byte Settings.....	8
5.3 List of Constants.....	8
5.4 List of Variables.....	8
5.5 List of Functions .....	9
5.6 Function Specifications .....	9
5.7 Flowcharts .....	12
5.7.1 Initialization Function .....	12
5.7.2 System Function .....	13
5.7.3 I/O Port Setup .....	14
5.7.4 CPU Clock Setup .....	15
5.7.5 Serial Array Unit Setup .....	16
5.7.6 UART0 Setup.....	18
5.7.7 Main Function .....	29
5.7.8 UART0 Reception Status Initialization Function .....	31
5.7.9 UART0 Operation Start Function.....	32
5.7.10 INTSR0 Interrupt Service Routine.....	36
5.7.11 UART0 Receive Data Classification Function.....	37
5.7.12 UART0 Data Transmission Function.....	38
5.7.13 UART0 Reception Error Interrupt Function .....	39
5.7.14 UART0 Reception Error Classification Function .....	40
5.7.15 INTST0 Interrupt Service Routine .....	41
5.7.16 UART0 Transmission End Processing Function.....	42
6. Sample Code.....	43
7. Documents for Reference .....	43
Revision Record .....	44
General Precautions in the Handling of MPU/MCU Products.....	45

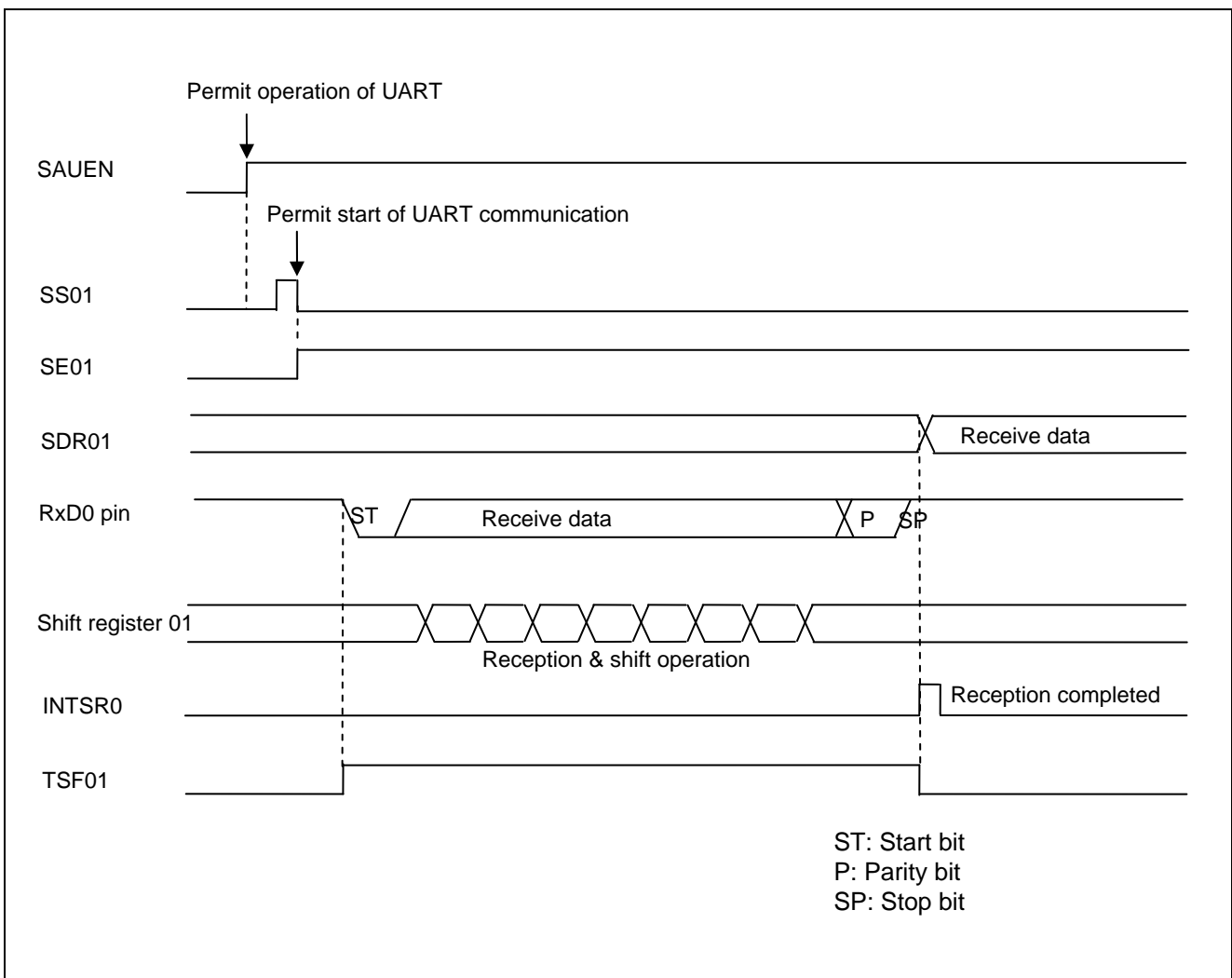
### 1. Specifications

In this application note, UART communication is performed through the serial array unit (SAU). ASCII characters transmitted from the device on the opposite side are analyzed to make responses.

Table 1.1 shows the peripheral function to be used and its use. Figures 1.1 and 1.2 illustrate UART communication operation.

**Table 1.1 Peripheral Function to be Used and its Use**

Peripheral Function	Use
Serial array unit 0	Perform UART communication using the TxD0 pin (transmission) and the RxD0 pin (reception).



**Figure 1.1 UART Reception Timing Chart**

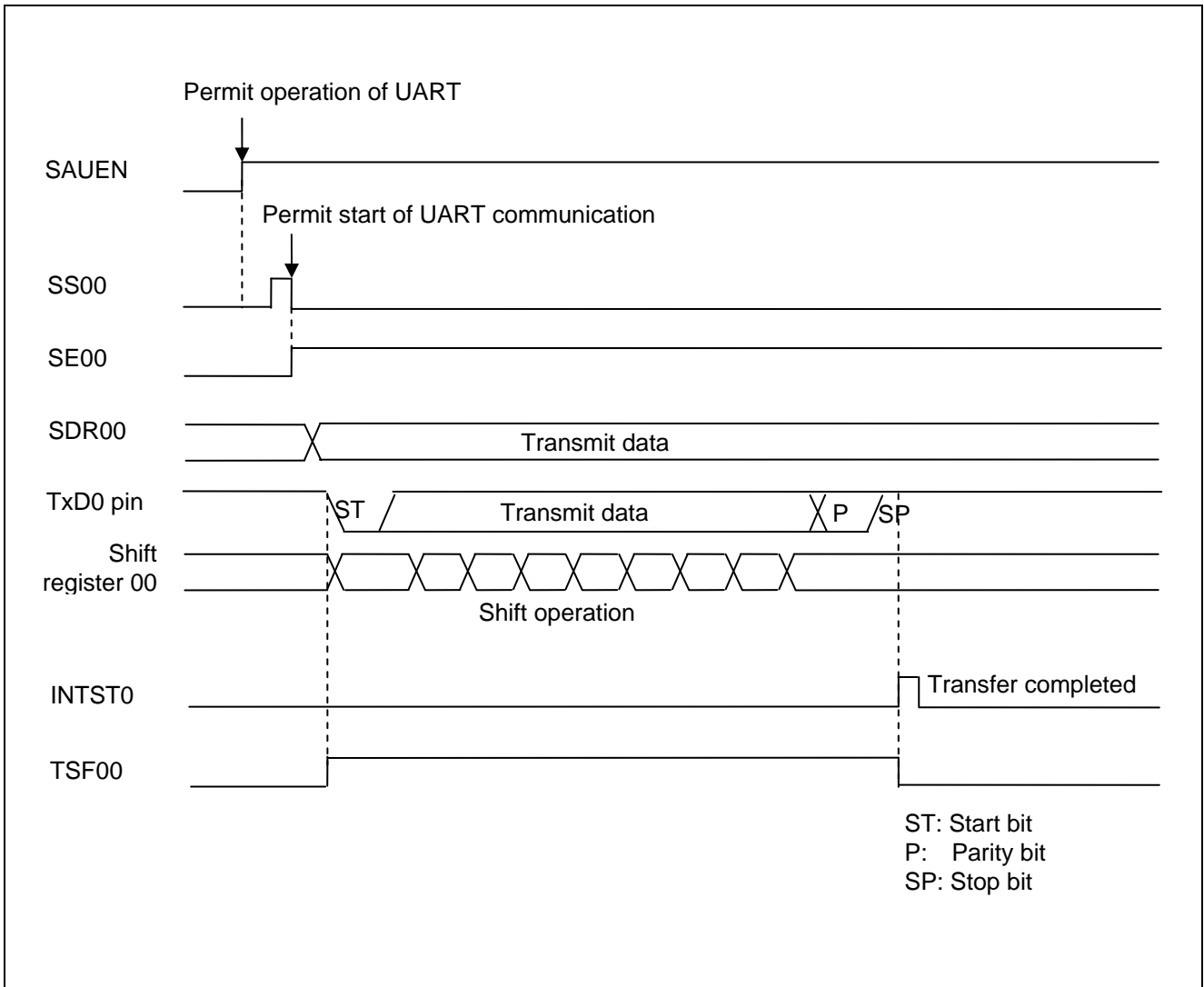


Figure 1.2 UART Transmission Timing Chart

## 2. Operation Check Conditions

The sample code contained 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+)	CubeSuite + V1.00.01
C compiler (CubeSuite+)	CA78K0R V1.20
Integrated development environment (e2studio)	e2studio V2.0.1.3
C compiler (e2studio)	KPIT GNURL78-ELF Toolchain V13.02
Integrated development environment (IAR)	IAR Embedded Workbench for Renesas RL78 V1.30.2
C compiler (IAR)	IAR C/C++ Compiler for Renesas RL78 V1.30.2.50607

## 3. Related Application Note

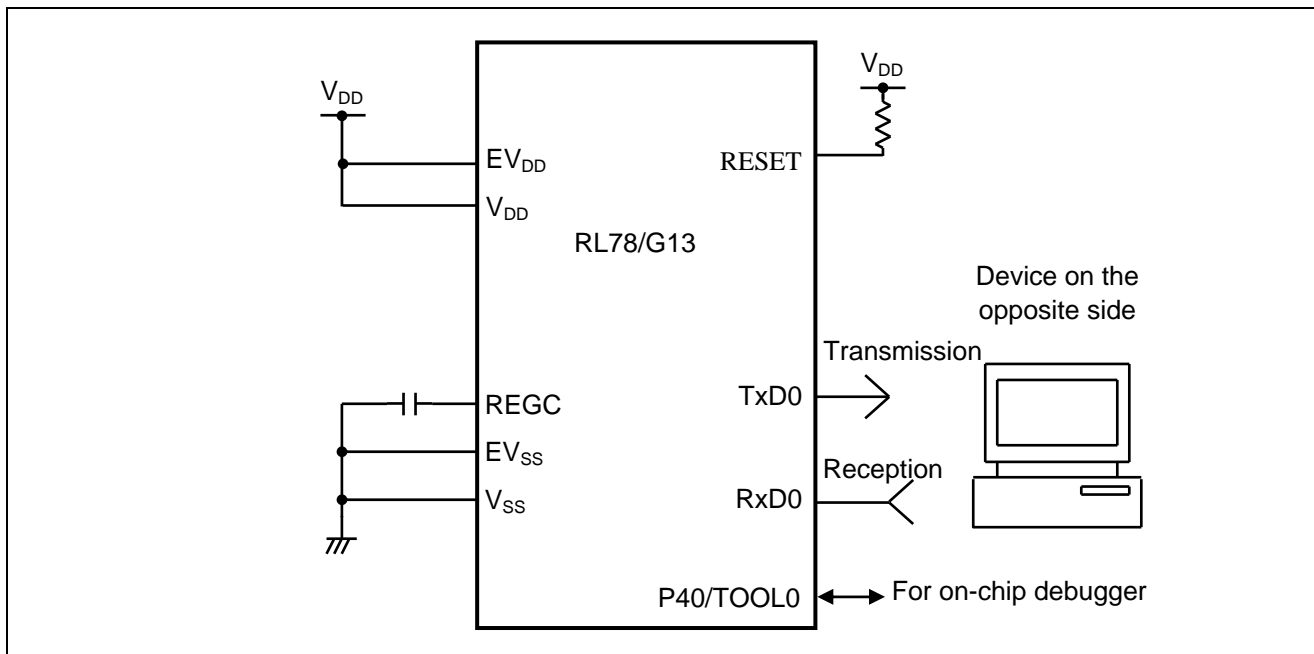
The application note that is related to this application note is listed below for reference.

RL78/G13 Initialization (R01AN0451EJ0100) 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**

- Caution:
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 function.

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

Pin Name	I/O	Description
P12/SO00/TxD0/TOOLTxD	Output	Data transmission pin
P11/SI00/RxD0/TOOLRxD/SDA00	Input	Data reception pin

## 5. Description of the Software

### 5.1 Operation Outline

This sample code transmits, to the device on the opposite side, the data corresponding to that received from the device. If an error occurs, it transmits to the device the data corresponding to the error. Tables 5.1 and 5.2 show the correspondence between transmit data and receive data.

**Table 5.1 Correspondence between Receive Data and Transmit Data**

Receive Data	Response (Transmit) Data
T (54H)	O (4FH), K (4BH), "CR" (0DH), "LF" (0AH)
t (74H)	o (6FH), k (6BH), "CR" (0DH), "LF" (0AH)
Other than above	U (55H), C (43H), "CR" (0DH), "LF" (0AH)

**Table 5.2 Correspondence between Error and Transmit Data**

Error	Response (Transmit) Data
Parity error	P (50H), E (45H), "CR" (0DH), "LF" (0AH)
Framing error	F (46H), E (45H), "CR" (0DH), "LF" (0AH)
Overrun error	O (4FH), E (45H), "CR" (0DH), "LF" (0AH)

(1) Perform initial setting of UART.

<UART Setting Conditions>

- Use SAU0 channels 0 and 1 as UART.
- Use the P12/TxD0 pin and the P11/RxD0 pin for data output and data input, respectively.
- The data length is 8 bits.
- Set the data transfer direction to LSB first.
- Use even parity as the parity setting.
- Set the receive data level to standard.
- Set the transfer rate to 9600 bps.
- Use reception end interrupt (INTSR0), transmission end interrupt (INTST0), and error interrupt (INTSRE0).
- Set the interrupt priority orders of INTSR0, INTST0, and INTSRE0 to low priority.

(2) After the system is made to enter a UART communication wait state by using the serial channel start register, a HALT instruction is executed. Processing is performed in response to reception end interrupt (INTSR0) and error interrupt (INTSRE0).

- When an INTSR0 occurs, the received data is taken in and the data corresponding to the received data is transmitted. When an INTSRE0 occurs, error handling is performed to transmit the data corresponding to the error.
- After data transmission, a HALT instruction is executed again to wait for reception end interrupt (INTSR0) and error interrupt (INTSRE0).

## 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	01101110B	Disables the watchdog timer. (Stops counting after the release from the reset state.)
000C1H/010C1H	01111111B	LVD reset mode, 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
MessageOK[4]	"OK\r\n"	Response message to reception of "T".
Messageok[4]	"ok\r\n"	Response message to reception of "t".
MessageUC[4]	"UC\r\n"	Response message to reception of characters other than "T" or "t".
MessageFE[4]	"FE\r\n"	Response message to a framing error.
MessagePE[4]	"PE\r\n"	Response message to a parity error.
MessageOE[4]	"OE\r\n"	Response message to an overrun error.

## 5.4 List of Variables

Table 5.3 lists the global variable that is used by this sample program.

**Table 5.3 Global Variable**

Type	Variable Name	Contents	Function Used
uint8_t	g_Uart0RxBuf	Receive data buffer	main()
uint8_t	gp_Uart0TxAddress	Transmit data pointer	R_UART0_Send, R_UART0_Interrupt_Send
uint16_t	g_Uart0TxCnt	Transmit data number counter	R_UART0_Send, R_UART0_Interrupt_Send
uint8_t	gp_Uart0RxAddress	Receive data pointer	R_UART0_Receive, R_UART0_Interrupt_Receive, R_UART0_Interrupt_Error
uint16_t	g_Uart0RxCnt	Receive data number counter	R_UART0_Receive, R_UART0_Interrupt_Receive
uint16_t	g_Uart0RxLen	Receive data number	R_UART0_Receive, R_UART0_Interrupt_Receive



## 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_Start	UART0 operation start
R_UART0_Receive	UART0 reception status initialization function
R_UART0_Send	UART0 data transmission function
R_UART0_Interrupt_Receive	UART0 reception end interrupt handling
R_UART0_Callback_ReceiveEnd	UART0 receive data classification function
R_UART0_Interrupt_Error	UART0 error interrupt handling
R_UART0_Callback_Error	UART0 reception error classification function
R_UART0_Interrupt_Send	UART0 transmission end interrupt handling
R_UART0_Callback_SendEnd	UART0 transmission end processing function

## 5.6 Function Specifications

This section describes the specifications for the functions that are used in this sample program.

### [Function Name] R\_UART0\_Start

Synopsis	UART0 operation start
Header	r_cg_macrodriver.h, r_cg_serial.h, and r_cg_userdefine.h
Declaration	void R_UART0_Start(void)
Explanation	Starts operation of channel 0 of serial array units 0 and 1 to make the system enter a communication wait state.
Arguments	None
Return value	None
Remarks	None

### [Function Name] R\_UART0\_Receive

Synopsis	UART0 reception status initialization function
Header	r_cg_macrodriver.h, r_cg_serial.h, r_cg_userdefine.h
Declaration	MD_STATUS R_UART0_Receive(uint8_t *rxbuf, uint16_t rxnum)
Explanation	Makes initial setting for UART0 reception.
Arguments	uint8_t *rxbuf : [Receive data buffer address] uint16_t rxnum : [Receive data buffer size]
Return value	[MD_OK]: Reception setting is completed [MD_ARGERROR]: Reception setting failed
Remarks	None

**[Function Name] R\_UART0\_Send**


---

Synopsis	UART0 data transmission function	
Header	r_cg_macrodriver.h, r_cg_serial.h, r_cg_userdefine.h	
Declaration	MD_STATUS R_UART0_Send(uint8_t* txbuf, uint16_t txnum)	
Explanation	Makes initial setting for UART0 transmission, and starts data transmission.	
Arguments	uint8_t *txbuf	: [Transmit data buffer address]
	uint16_t txnum	: [Transmit data buffer size]
Return value	[MD_OK]: Transmission setting is completed [MD_ARGERROR]: Transmission setting failed	
Remarks	None	

**[Function Name] R\_UART0\_Interrupt\_Receive**


---

Synopsis	UART0 reception end interrupt handling	
Header	r_cg_macrodriver.h, r_cg_serial.h, and r_cg_userdefine.h	
Declaration	__interrupt void R_UART0_Interrupt_Receive(void)	
Explanation	Makes a response (data transmission) corresponding to received data.	
Arguments	None	
Return value	None	
Remarks	None	

**[Function Name] R\_UART0\_Interrupt\_Error**


---

Synopsis	UART error interrupt function	
Header	r_cg_macrodriver.h, r_cg_serial.h, and r_cg_userdefine.h	
Declaration	__interrupt void R_UART0_Interrupt_Error(void)	
Explanation	Transmits the data corresponding to a detected error.	
Arguments	None	
Return value	None	
Remarks	None	

**[Function Name] R\_UART0\_Callback\_ReceiveEnd**


---

Synopsis	UART0 receive data classification function	
Header	r_cg_macrodriver.h, r_cg_serial.h, and r_cg_userdefine.h	
Declaration	void R_UART0_Callback_ReceiveEnd(void)	
Explanation	Clears the reception error flag.	
Arguments	None	
Return value	None	
Remarks	None	

**[Function Name] R\_UART0\_Callback\_Error**


---

Synopsis	UART0 reception error classification function	
Header	r_cg_macrodriver.h, r_cg_serial.h, and r_cg_userdefine.h	
Declaration	void R_UART0_Callback_Error(uint8_t err_type)	
Explanation	Makes flag setting for transmission of the data corresponding to an error.	
Arguments	err_type	: Error type
Return value	None	
Remarks	None	

---

**[Function Name] R\_UART0\_Interrupt\_Send**

---

Synopsis	UART0 transmission end interrupt handling
Header	r_cg_macrodriver.h, r_cg_serial.h, and r_cg_userdefine.h
Declaration	__interrupt void R_UART0_Interrupt_Send(void)
Explanation	Transmits a specified number of pieces of data.
Arguments	None
Return value	None
Remarks	None

---

**[Function Name] R\_UART0\_Callback\_SendEnd**

---

Synopsis	UART0 transmission end processing function
Header	r_cg_macrodriver.h, r_cg_serial.h, r_cg_userdefine.h
Declaration	void R_UART0_Callback_SendEnd(void)
Explanation	Makes transmission end flag setting.
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.

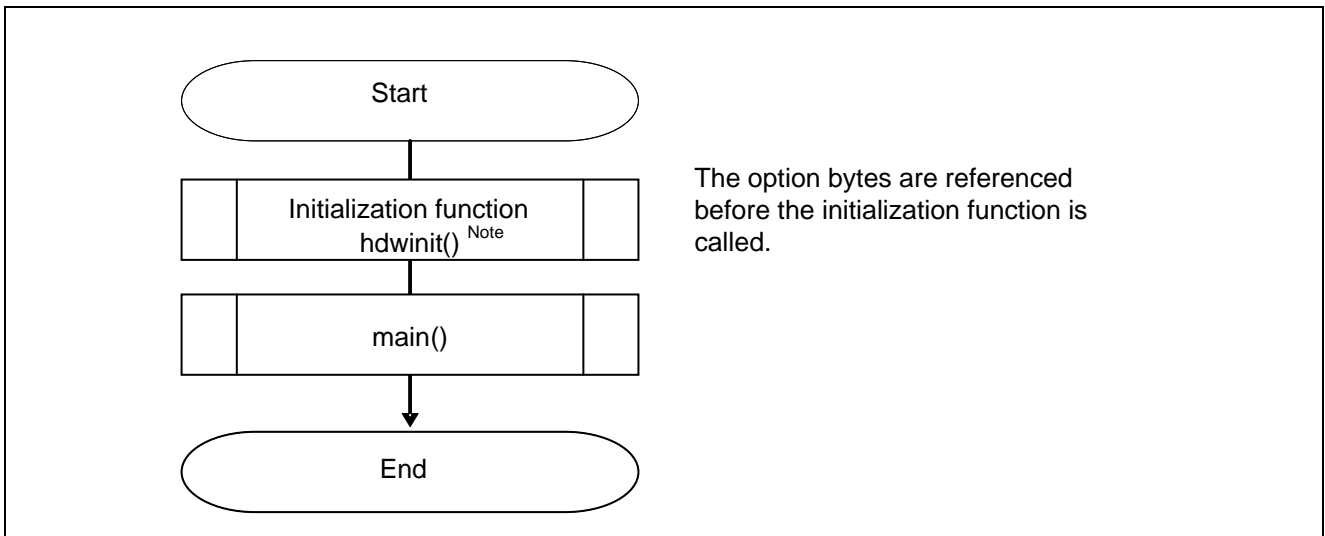


Figure 5.1 Overall Flow

#### 5.7.1 Initialization Function

Figure 5.2 shows the flowchart for the initialization function.

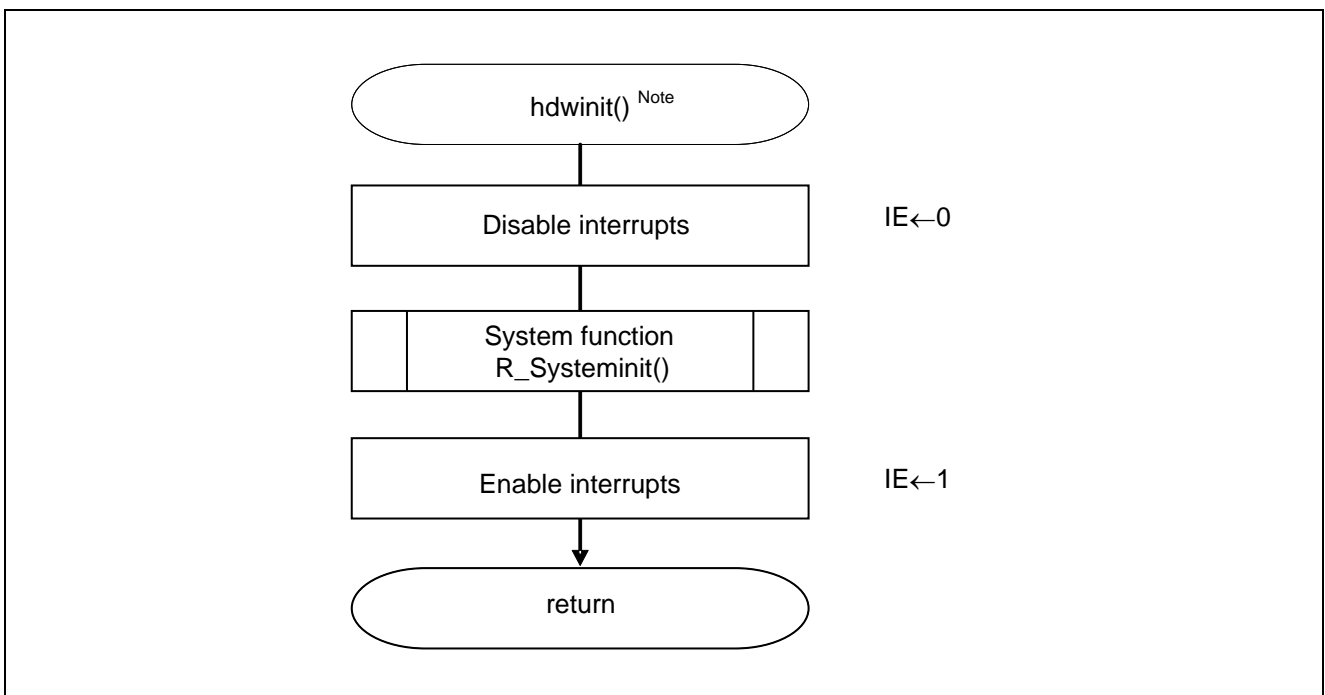


Figure 5.2 Initialization Function

Note: This function is used in the Renesas CubeSuite+ sample code. The function `__low_level_init` initializes the system in the IAR Workbench IDE-based sample code instead.

5.7.2 System Function

Figure 5.3 shows the flowchart for the system function.

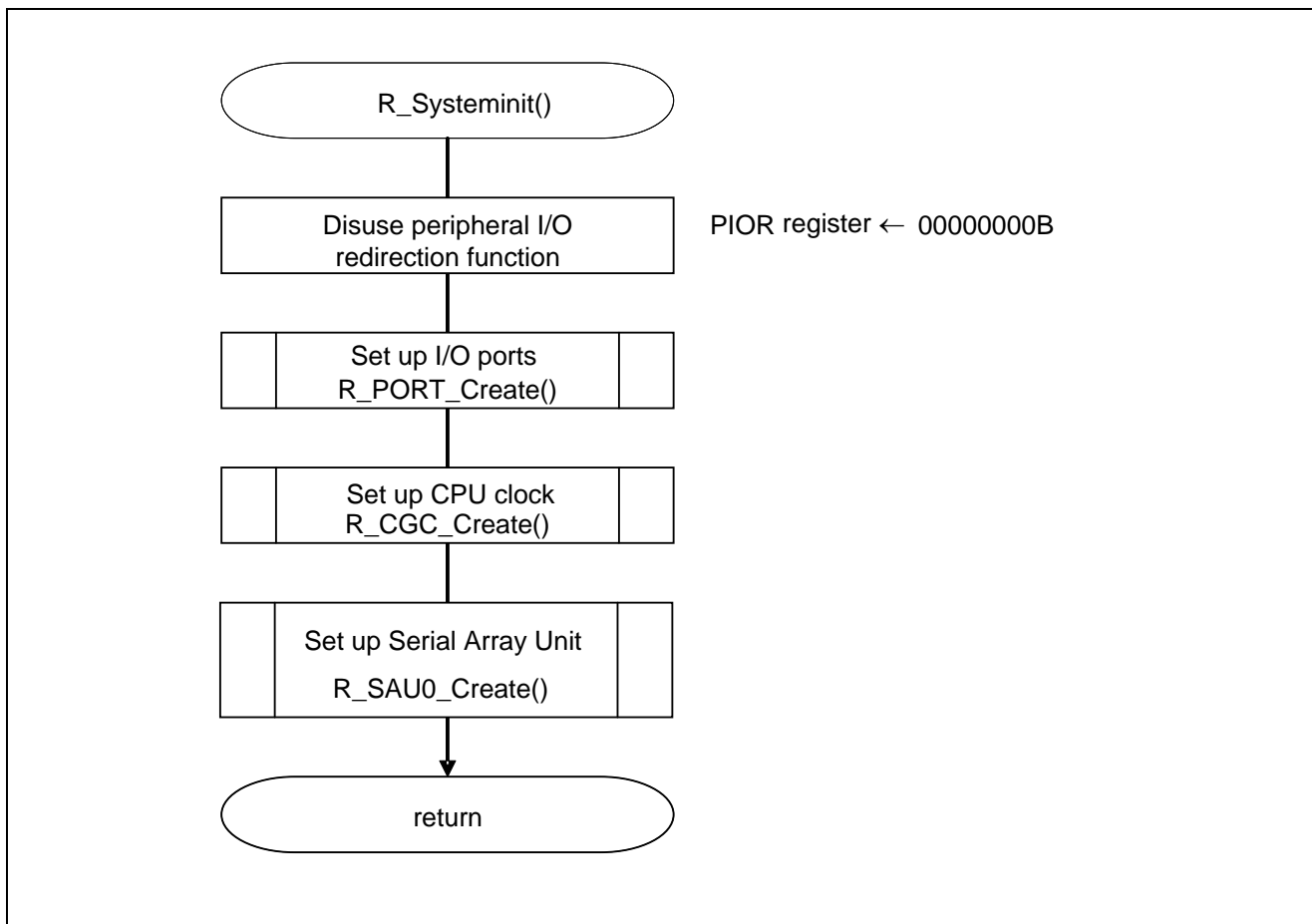
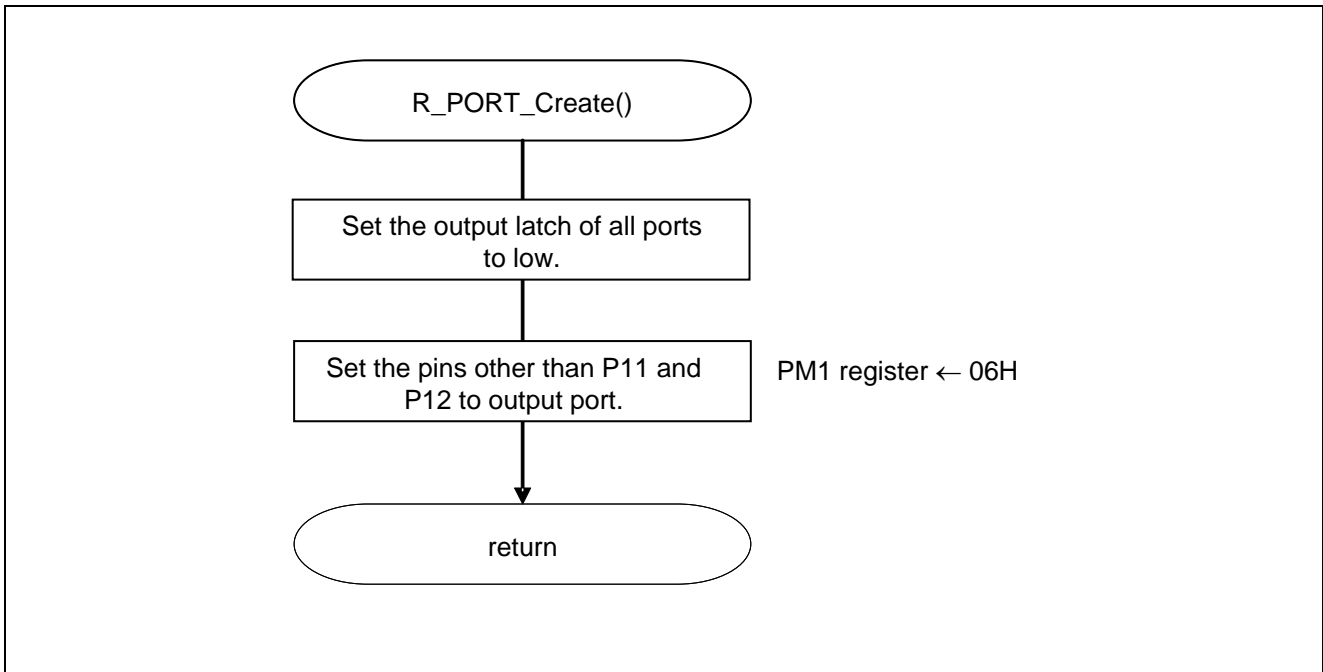


Figure 5.3 System Function

### 5.7.3 I/O Port Setup

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



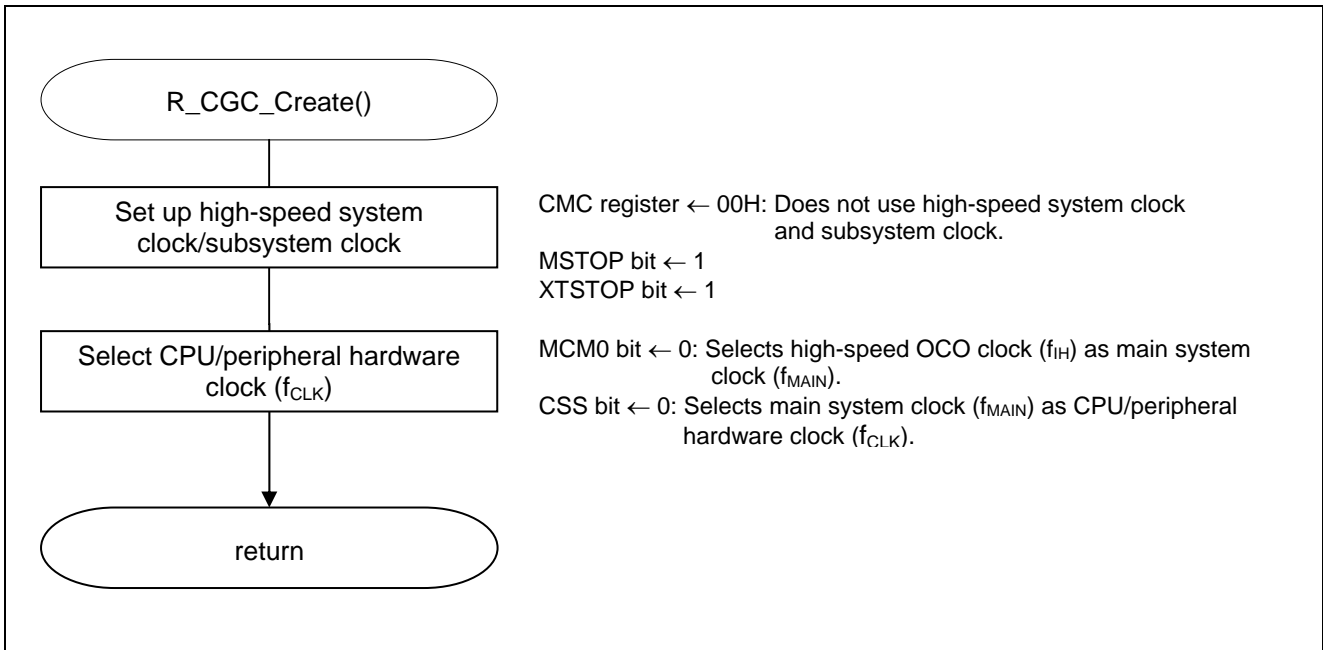
**Figure 5.4 I/O Port Setup**

Note: Refer to the section entitled "Flowcharts" in RL78/G13 Initialization Application Note (R01AN0451EJ0100) for the configuration of the unused ports.

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

### 5.7.4 CPU Clock Setup

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

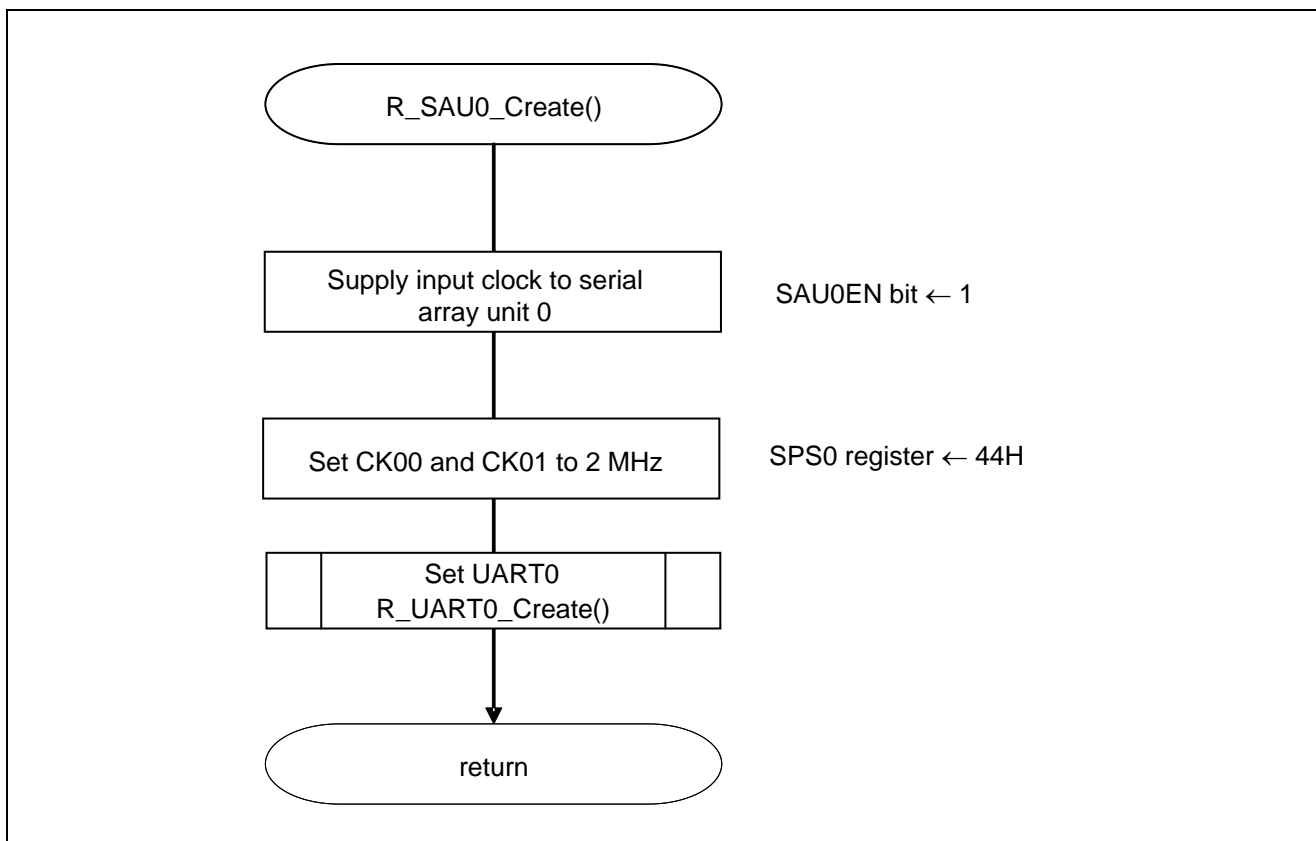


**Figure 5.5 CPU Clock Setup**

**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 (R01AN0451EJ0100).

**5.7.5 Serial Array Unit Setup**

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



**Figure 5.6 Serial Array Unit Setup**



Start supplying clock to the SAU

- Peripheral enable register 0 (PER0)  
Clock supply

Symbol: PER0

7	6	5	4	3	2	1	0
RTCCEN	IICA1EN	ADCEN	IICA0EN	SAU1EN	SAU0EN	TAU1EN	TAU0EN
x	0	x	x	x	1	0	x

Bit 2

SAU0EN	Input clock control for serial array unit 0
0	Stops supply of input clock.
1	<b>Starts supply of input clock.</b>

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

Select serial clock

- Serial clock select register 0 (SPS0)  
Operation clock setting

Symbol: SPS0

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	0	0	0	0	0	0	0	PRS 013	PRS 012	PRS 011	PRS 010	PRS 003	PRS 002	PRS 001	PRS 000
0	0	0	0	0	0	0	0	0	1	0	0	0	1	0	0

Bits 7 to 0

PRS On3	PRS On2	PRS On1	PRS On0	Operation clock (CK00) selection (n = 0, 1)					
				$f_{CLK}$ = 2 MHz	$f_{CLK}$ = 5 MHz	$f_{CLK}$ = 10 MHz	$f_{CLK}$ = 20 MHz	$f_{CLK}$ = 32 MHz	
0	0	0	0	$f_{CLK}$	2 MHz	5 MHz	10 MHz	20 MHz	32 MHz
0	0	0	1	$f_{CLK}/2$	1 MHz	2.5 MHz	5 MHz	10 MHz	16 MHz
0	0	1	0	$f_{CLK}/2^2$	500 kHz	1.25 MHz	2.5 MHz	5 MHz	8 MHz
0	0	1	1	$f_{CLK}/2^3$	250 kHz	625 kHz	1.25 MHz	2.5 MHz	4 MHz
<b>0</b>	<b>1</b>	<b>0</b>	<b>0</b>	<b><math>f_{CLK}/2^4</math></b>	<b>125 kHz</b>	<b>312.5 kHz</b>	<b>625 kHz</b>	<b>1.25 MHz</b>	<b>2 MHz</b>
0	1	0	1	$f_{CLK}/2^5$	62.5 kHz	156.2 kHz	312.5 kHz	625 kHz	1 MHz
0	1	1	0	$f_{CLK}/2^6$	31.25 kHz	78.1 kHz	156.2 kHz	312.5 kHz	500 kHz
0	1	1	1	$f_{CLK}/2^7$	15.62 kHz	39.1 kHz	78.1 kHz	156.2 kHz	250 kHz
1	0	0	0	$f_{CLK}/2^8$	7.81 kHz	19.5 kHz	39.1 kHz	78.1 kHz	125 kHz
1	0	0	1	$f_{CLK}/2^9$	3.91 kHz	9.76 kHz	19.5 kHz	39.1 kHz	62.5 kHz
1	0	1	0	$f_{CLK}/2^{10}$	1.95 kHz	4.88 kHz	9.76 kHz	19.5 kHz	31.25 kHz
1	0	1	1	$f_{CLK}/2^{11}$	977 Hz	2.44 kHz	4.88 kHz	9.76 kHz	15.6 kHz
Other than above.				Setting prohibited.					

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

5.7.6 UART0 Setup

Figures 5.7, 5.8, and 5.9 show the flowcharts for setting up UART0.

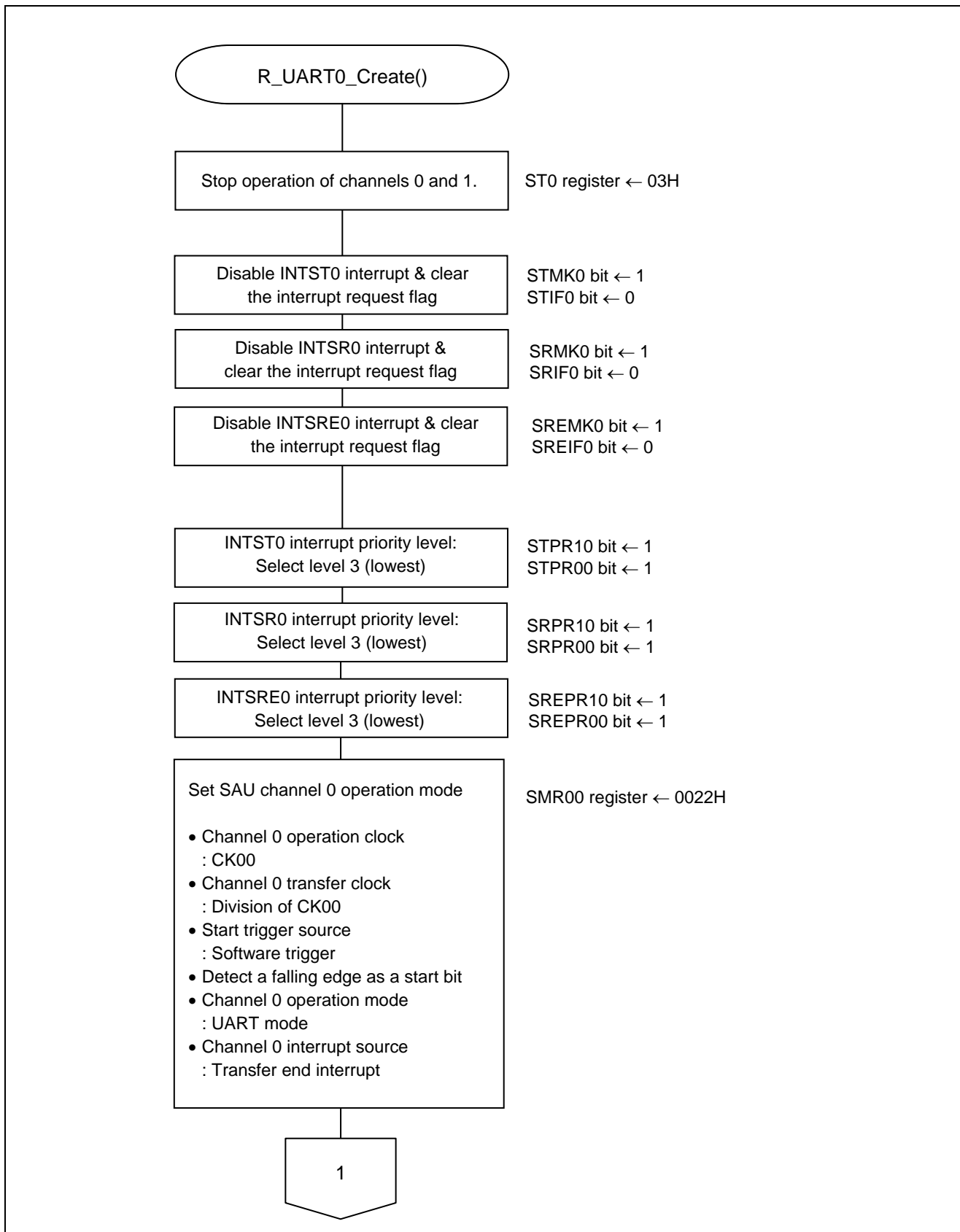


Figure 5.7 UART0 Setup (1/3)

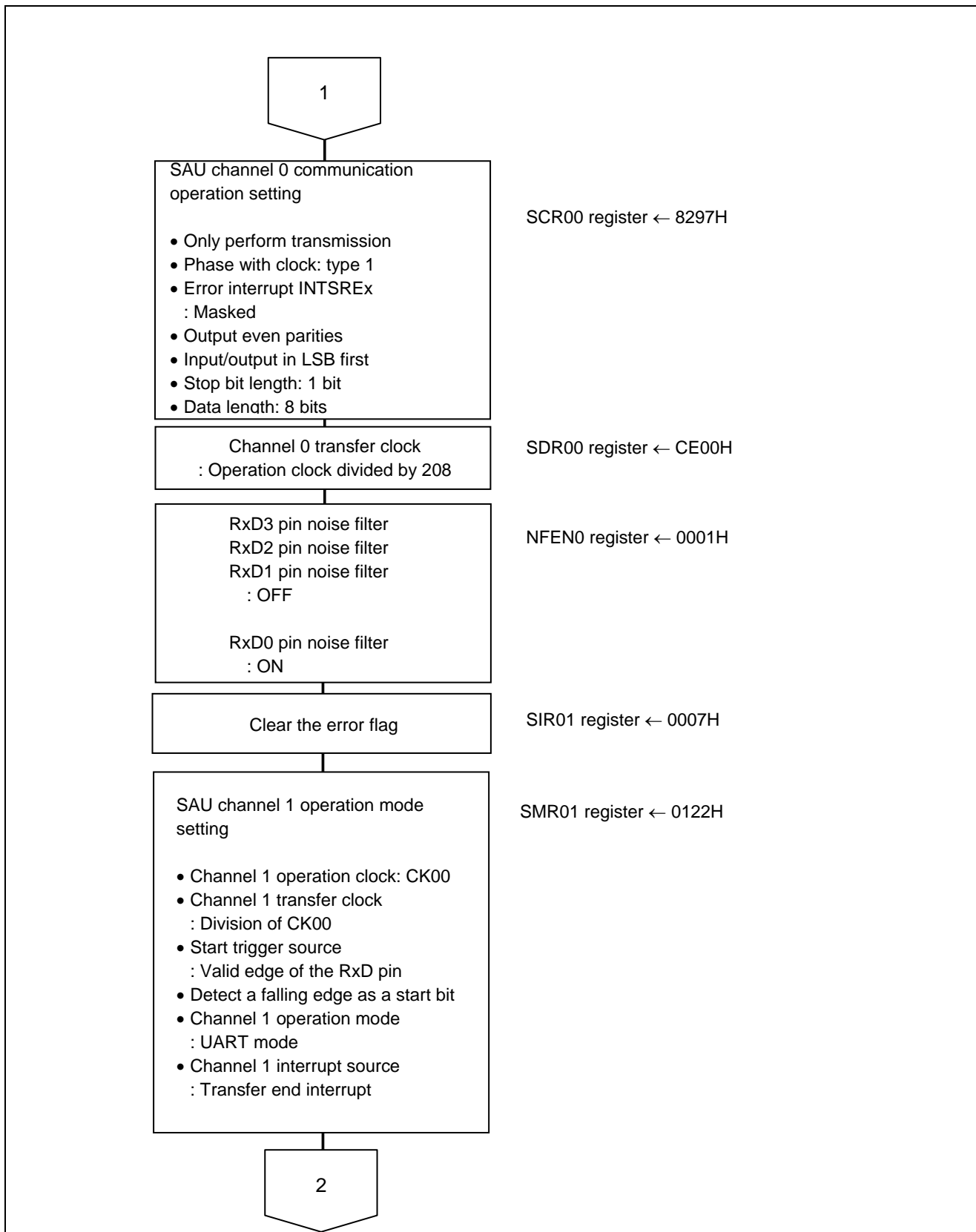


Figure 5.8 UART0 Setup (2/3)

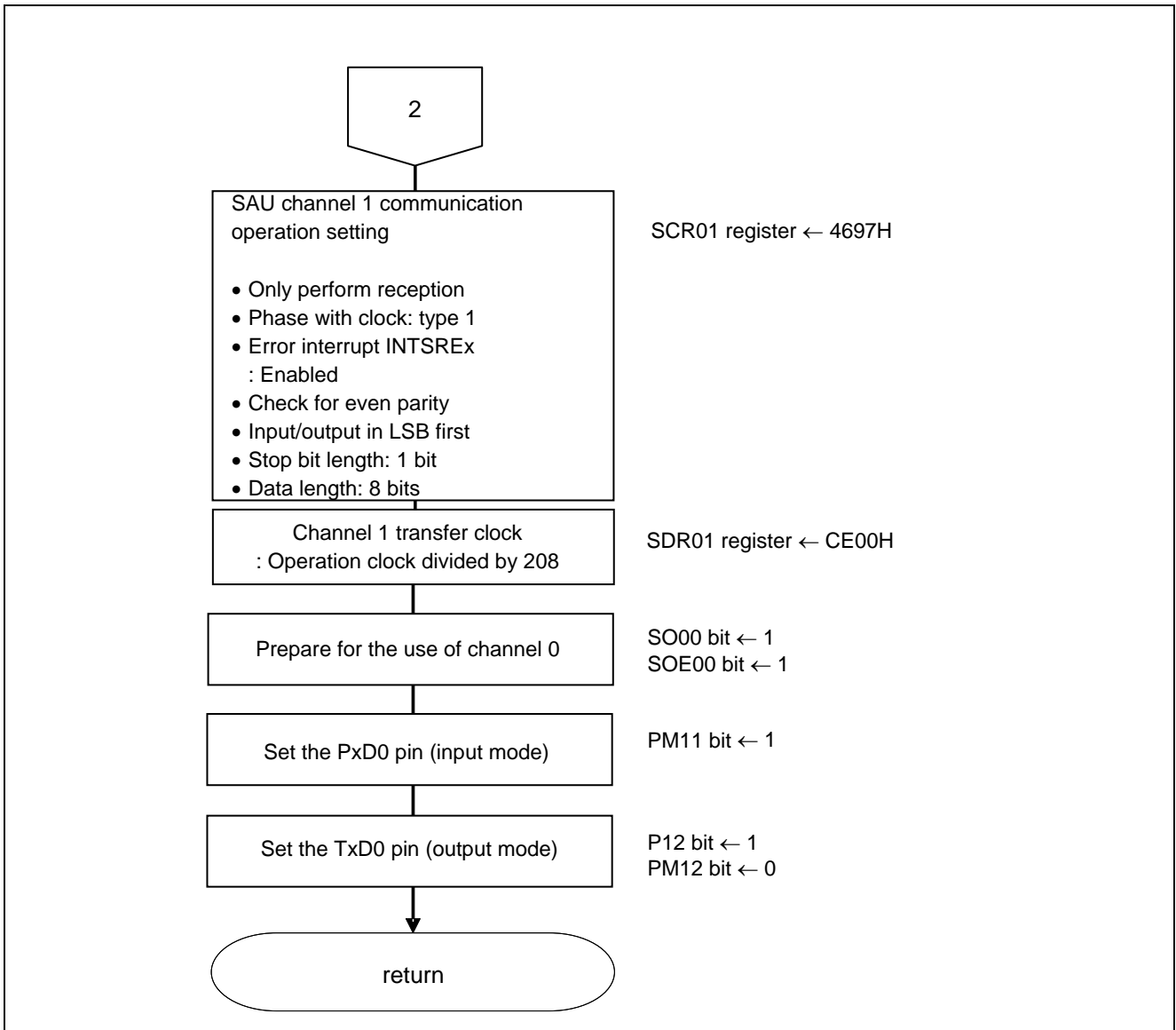


Figure 5.9 UART0 Setup (3/3)

## Transmission channel operation mode setting

- Serial mode register 00 (SMR00)
  - Interrupt source
  - Operation mode
  - Transfer clock selection
  - $f_{MCK}$  selection

Symbol: SMR00

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CKS00	CCS00	0	0	0	0	0	0	0	0	1	0	0	MD002	MD001	MD000
0	0	0	0	0	0	0	0	0	0	1	0	0	0	1	0

## Bit 15

CKS00	Channel 0 operation clock ( $f_{MCK}$ ) selection
0	<b>Prescaler output clock CK00 configured by the SPS0 register</b>
1	Prescaler output clock CK01 configured by the SPS0 register

## Bit 14

CCS00	Channel 0 transfer clock (TCLK) selection
0	<b>Clock obtained by dividing the operation clock <math>f_{MCK}</math> specified by the CKS00 bit.</b>
1	Clock input from the SCK pin.

## Bits 2 and 1

MD002	MD001	Channel 0 operation mode setting
0	0	CSI mode
0	1	<b>UART mode</b>
1	0	Simplified I <sup>2</sup> C mode
1	1	Setting prohibited

## Bit 0

MD000	Channel 0 interrupt source selection
0	<b>Transfer end interrupt</b>
1	Buffer empty interrupt

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

Transmission channel communication operation setting

- Serial communication operation setting register 00 (SCR00)  
Data length setting, data transfer order, error interrupt signal mask availability, and operation mode

Symbol: SCR00

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TXE 00	RXE 00	DAP 00	CKP 00	0	EOC 00	PTC 001	PTC 000	DIR 00	0	SLC 001	SLC 000	0	1	DLS 001	DLS 000
<b>1</b>	<b>0</b>	<b>0</b>	<b>0</b>	0	<b>0</b>	<b>1</b>	<b>0</b>	<b>1</b>	0	<b>0</b>	<b>1</b>	0	1	<b>1</b>	<b>1</b>

Bits 15 and 14

TXE00	RXE00	Channel 0 operation mode setting
0	0	Communication prohibited
0	1	Reception Only
<b>1</b>	<b>0</b>	<b>Transmission only</b>
1	1	Both transmission and reception

Bit 10

EOC00	Error interrupt signal (INTSREx (x = 0, 1)) mask availability selection
<b>0</b>	<b>Error interrupt INTSREx is masked</b>
1	Generation of error interrupt INTSREx is enabled

Bits 9 and 8

PTC001	PTC000	Parity bit setting in UART mode	
		Transmission	Reception
0	0	No parity bit is output	Data is received without parity
0	1	0 parity is output	No parity check is made
<b>1</b>	<b>0</b>	<b>Even parity is output</b>	Check is made for even parity
1	1	Odd parity is output	Check is made for odd parity

Bit 7

DIR00	Selection of data transfer order in CSI and UART modes
0	Input and output in MSB first
<b>1</b>	<b>Input and output in LSB first</b>

Bits 5 and 4

SLC001	SLC000	Stop bit setting in UART mode
0	0	No stop bit
<b>0</b>	<b>1</b>	<b>Stop bit length = 1 bit</b>
1	0	Stop bit length = 2 bits
1	1	Setting prohibited

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

Symbol: SCR00

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TXE 00	RXE 00	DAP 00	CKP 00	0	EOC 00	PTC 001	PTC 000	DIR 00	0	SLC 001	SLC 000	0	1	DLS 001	DLS 000
1	0	0	0	0	0	1	0	1	0	0	1	0	1	1	1

Bits 1 and 0

DLS001	DLS000	Data length setting in CSI mode
0	1	9-bit data length
1	0	7-bit data length
1	1	<b>8-bit data length</b>
Others		Setting prohibited

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

Transmission channel transfer clock setting

- Serial data register 00 (SDR00)  
Transfer clock frequency:  $f_{MCK}/208$  ( $\approx 9600$  Hz)

Symbol: SDR00

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

Bits 15 to 9

SDR00[15:9]							Transfer clock setting by dividing operation clock ( $f_{MCK}$ )
0	0	0	0	0	0	0	$f_{MCK}/2$
0	0	0	0	0	0	1	$f_{MCK}/4$
0	0	0	0	0	1	0	$f_{MCK}/6$
0	0	0	0	0	1	1	$f_{MCK}/8$
.	.	.	.	.	.	.	.
.	.	.	.	.	.	.	.
1	1	0	0	1	1	1	<b><math>f_{MCK}/208</math></b>
.	.	.	.	.	.	.	.
.	.	.	.	.	.	.	.
1	1	1	1	1	1	0	$f_{MCK}/254$
1	1	1	1	1	1	1	$f_{MCK}/256$

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

Reception channel operation mode setting

- Serial mode register 01 (SMR01)
  - Interrupt source
  - Operation mode
  - Transfer clock selection
  - f<sub>MCK</sub> selection

Symbol: SMR01

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CKS01	CCS01	0	0	0	0	0	STS01	0	SIS010	1	0	0	MD012	MD011	MD010
0	0	0	0	0	0	0	1	0	0	1	0	0	0	1	0

Bit 15

CKS01	Channel 1 operation clock (f <sub>MCK</sub> ) selection
0	Prescaler output clock CK00 configured by the SPS0 register
1	Prescaler output clock CK01 configured by the SPS0 register

Bit 14

CCS01	Channel 1 transfer clock (TCLK) selection
0	Clock obtained by dividing the operation clock f <sub>MCK</sub> specified by the CKS01 bit
1	Clock input from the SCK pin

Bit 8

STS01	Start trigger source selection
0	Only software trigger is valid
1	Valid edge of the RxD pin (selected during UART reception)

Bit 6

SIS010	Control of receive data level inversion on channel 1 in UART mode
0	Falling edge is detected as a start bit
1	Rising edge is detected as a start bit

Bits 2 and 1

MD012	MD011	Channel 1 operation mode setting
0	0	CSI mode
0	1	UART mode
1	0	Simplified I <sup>2</sup> C mode
1	1	Setting prohibited

Bit 0

MD010	Channel 1 interrupt source selection
0	Transfer end interrupt
1	Buffer empty interrupt

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



## Reception channel communication operation setting

- Serial communication operation setting register 01 (SCR01)  
Data length setting, data transfer order, error interrupt signal mask availability, and operation mode

Symbol: SCR01

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TXE 01	RXE 01	DAP 01	CKP 01	0	EOC 01	PTC 011	PTC 010	DIR 01	0	SLC 011	SLC 010	0	1	DLS 011	DLS 010
<b>0</b>	<b>1</b>	<b>0</b>	<b>0</b>	0	<b>1</b>	<b>1</b>	<b>0</b>	<b>1</b>	0	<b>0</b>	<b>1</b>	0	1	<b>1</b>	<b>1</b>

Bits 15 and 14

TXE01	RXE01	Channel 1 operation mode setting
0	0	Communication prohibited
<b>0</b>	<b>1</b>	<b>Reception only</b>
1	0	Transmission only
1	1	Both transmission and reception

For UART reception, wait for 4  $f_{CLK}$  clock cycles or more before setting SS01 to 1, after setting the RXE01 bit of the SCR01 register to 1.

Bit 10

EOC01	Error interrupt signal (INTSRE1) mask availability selection
0	Error interrupt INTSRE1 is masked
<b>1</b>	<b>Generation of error interrupt INTSRE1 is enabled</b>

Bits 9 and 8

PTC011	PTC010	Parity bit setting in UART mode	
		Transmission	Reception
0	0	No parity bit is output	Data is received without parity
0	1	0 parity is output	No parity check is made
<b>1</b>	<b>0</b>	Even parity is output	<b>Check is made for even parity</b>
1	1	Odd parity is output	Check is made for odd parity

Bit 7

DIR01	Selection of data transfer order in CSI and UART modes
0	Input and output in MSB first
<b>1</b>	<b>Input and output in LSB first</b>

Bits 5 and 4

SLC011	SLC010	Stop bit setting in UART mode
0	0	No stop bit
<b>0</b>	<b>1</b>	<b>Stop bit length = 1 bit</b>
1	0	Stop bit length = 2 bits
1	1	Setting prohibited

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

Symbol: SCR01

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TXE 01	RXE 01	DAP 01	CKP 01	0	EOC 01	PTC 011	PTC 010	DIR 01	0	SLC 011	SLC 010	0	1	DLS 011	DLS 010
<b>0</b>	<b>1</b>	<b>0</b>	<b>0</b>	0	<b>1</b>	<b>1</b>	<b>0</b>	<b>1</b>	0	<b>0</b>	<b>1</b>	0	1	<b>1</b>	<b>1</b>

Bits 1 and 0

DLS011	DLS010	Data length setting in CSI mode
0	1	9-bit data length
1	0	7-bit data length
<b>1</b>	<b>1</b>	<b>8-bit data length</b>
others		Setting prohibited

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

Reception transfer clock setting

- Serial data register 01 (SDR01)  
Transfer clock frequency:  $f_{MCK}/208$  ( $\approx 9600$  Hz)

Symbol: SDR01

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

Bits 15 to 9

SDR01[15:9]							Transfer clock setting by dividing operation clock ( $f_{MCK}$ )
0	0	0	0	0	0	0	$f_{MCK}/2$
0	0	0	0	0	0	1	$f_{MCK}/4$
0	0	0	0	0	1	0	$f_{MCK}/6$
0	0	0	0	0	1	1	$f_{MCK}/8$
.	.	.	.	.	.	.	.
.	.	.	.	.	.	.	.
<b>1</b>	<b>1</b>	<b>0</b>	<b>0</b>	<b>1</b>	<b>1</b>	<b>1</b>	<b><math>f_{MCK}/208</math></b>
.	.	.	.	.	.	.	.
.	.	.	.	.	.	.	.
1	1	1	1	1	1	0	$f_{MCK}/254$
1	1	1	1	1	1	1	$f_{MCK}/256$

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

## Initial output level setting

- Serial output register 0 (SO0)

Initial output: 1

Symbol: SO0

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	0	0	0	CKO 03	CKO 02	CKO 01	CKO 00	0	0	0	0	SO 03	SO 02	SO 01	SO 00
0	0	0	0	x	x	x	x	0	0	0	0	x	x	x	1

Bit 0

SO00	Channel 0 serial data output
0	Serial data output value is "0"
1	<b>Serial data output value is "1"</b>

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

## Enabling of data output on target channel

- Serial output enable register 0 (SOE0)

Output enable

Symbol: SOE0

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	SOE 03	SOE 02	SOE 01	SOE 00
0	0	0	0	0	0	0	0	0	0	0	0	x	x	x	1

Bit 0

SOE00	Channel 0 serial output enable/stop
0	Serial communication output is stopped
1	<b>Serial communication output is enabled</b>

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

## Port setting

- Port register 1 (P1)
- Port mode register 1 (PM1)  
Port setting for each of transmit data and receive data.

Symbol: P1

7	6	5	4	3	2	1	0
P17	P16	P15	P14	P13	P12	P11	P10
x	x	x	x	x	1	x	x

## Bit 2

P12	Output data control (in output mode)
0	0 is output
1	1 is output

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

Symbol: PM1

7	6	5	4	3	2	1	0
PM17	PM16	PM15	PM14	PM13	PM12	PM11	PM10
x	x	x	x	x	0	1	x

## Bit 2

PM12	P12 I/O mode selection
0	Output mode (output buffer is on)
1	Input mode (output buffer is off)

## Bit 1

PM11	P11 I/O mode selection
0	Output mode (output buffer is on)
1	Input mode (output buffer is off)

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

5.7.7 Main Function

Figures 5.10 and 5.11 show the flowchart for the main function.

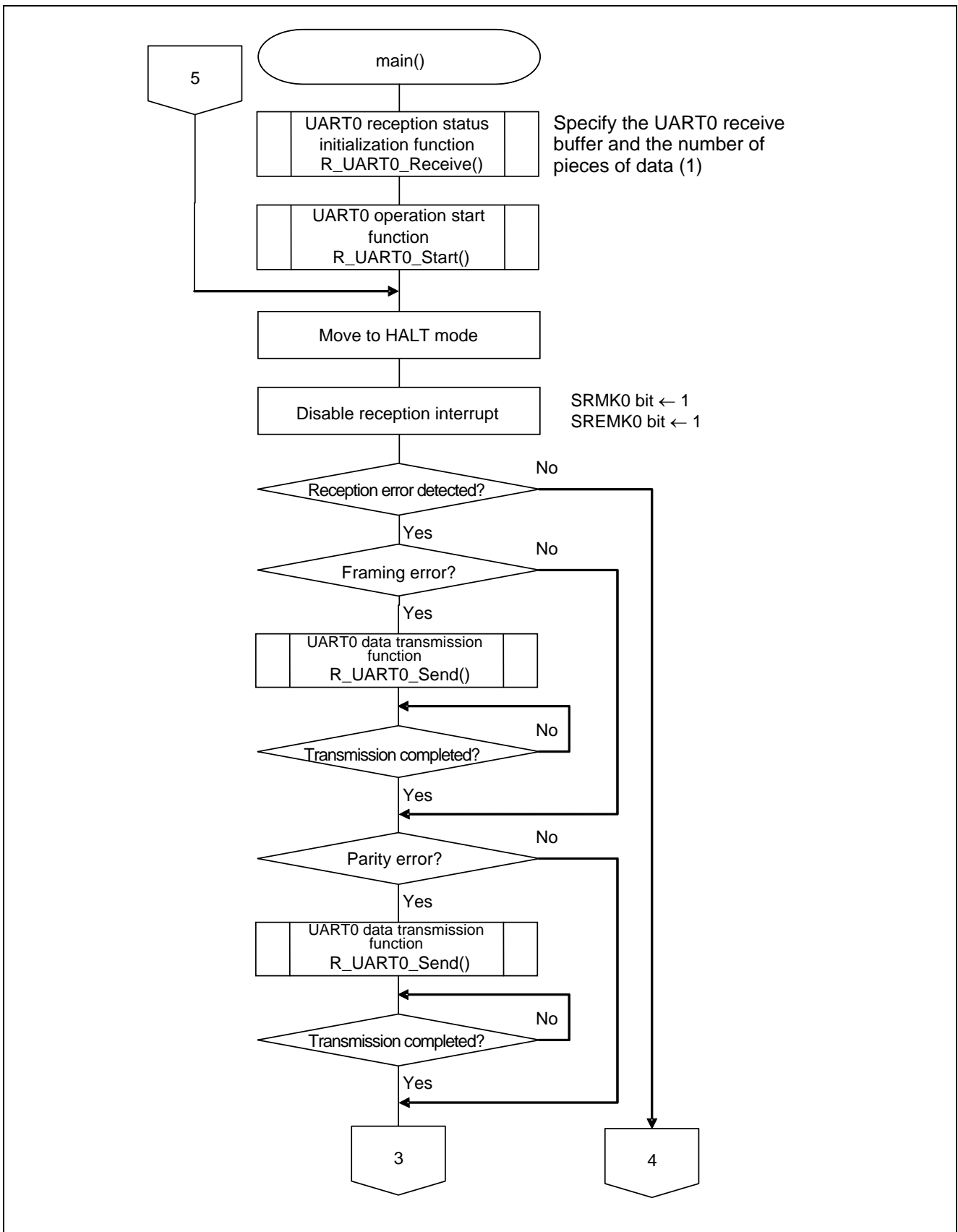


Figure 5.10 Main Function (1/2)

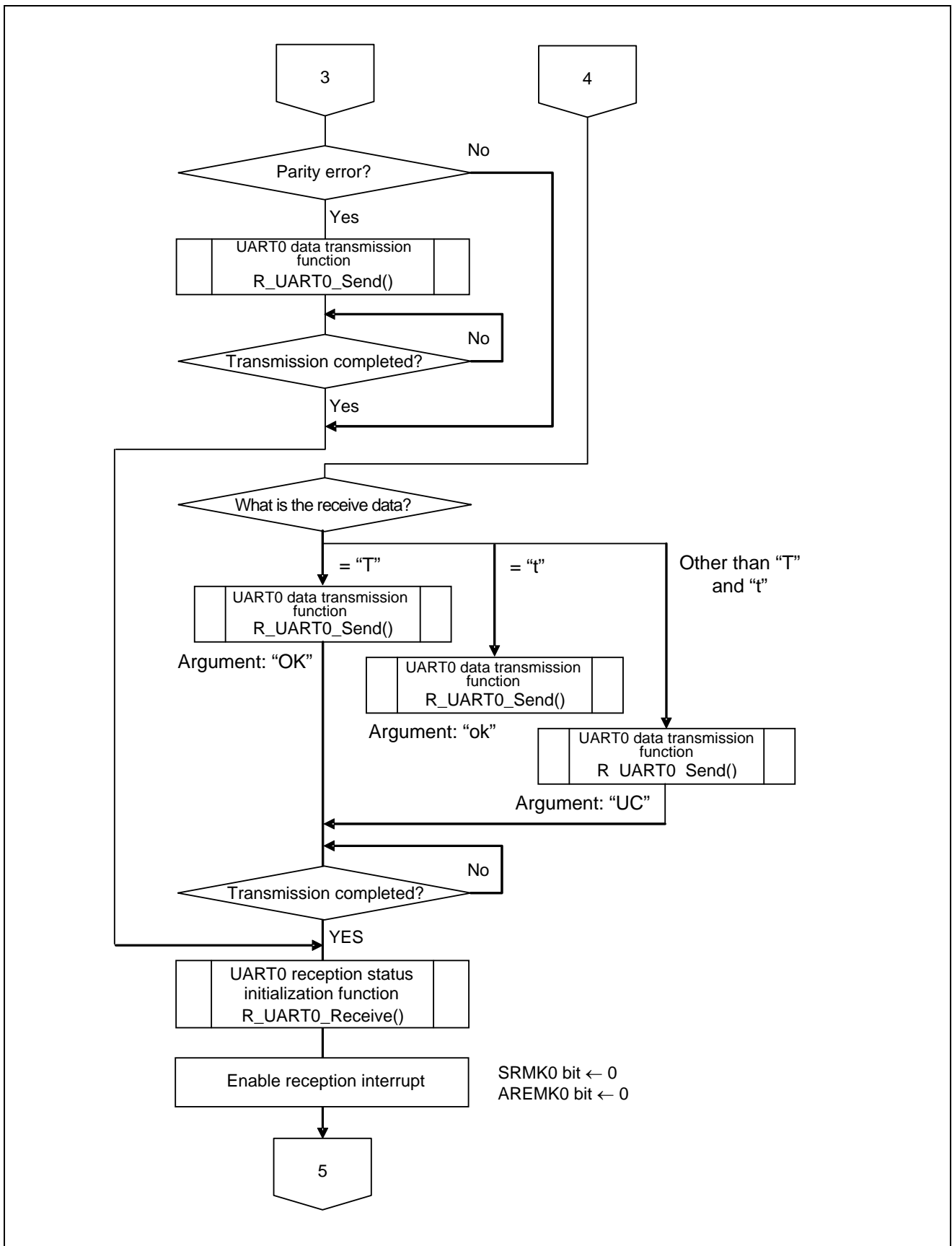
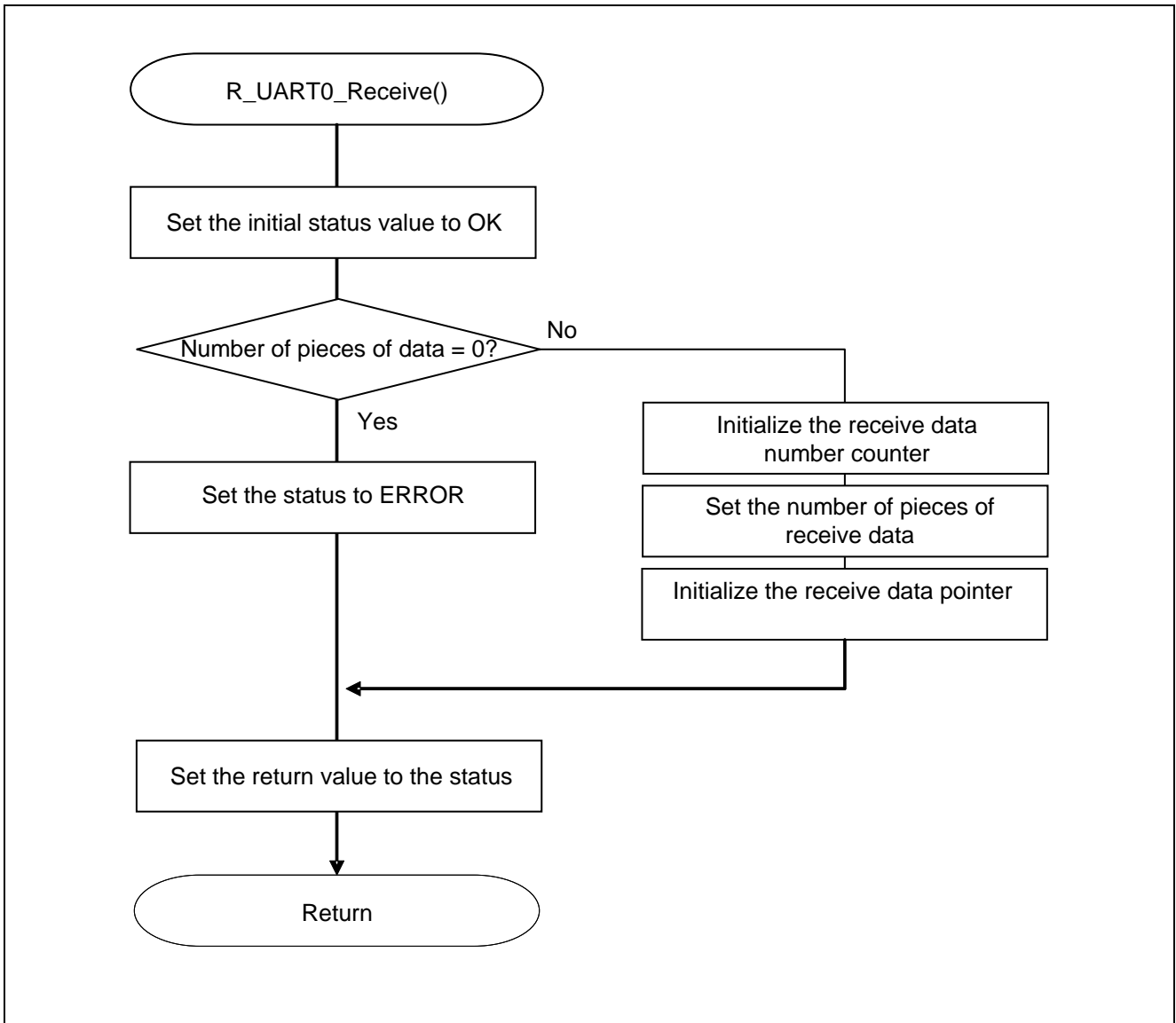


Figure 5.11 Main Function (2/2)

**5.7.8 UART0 Reception Status Initialization Function**

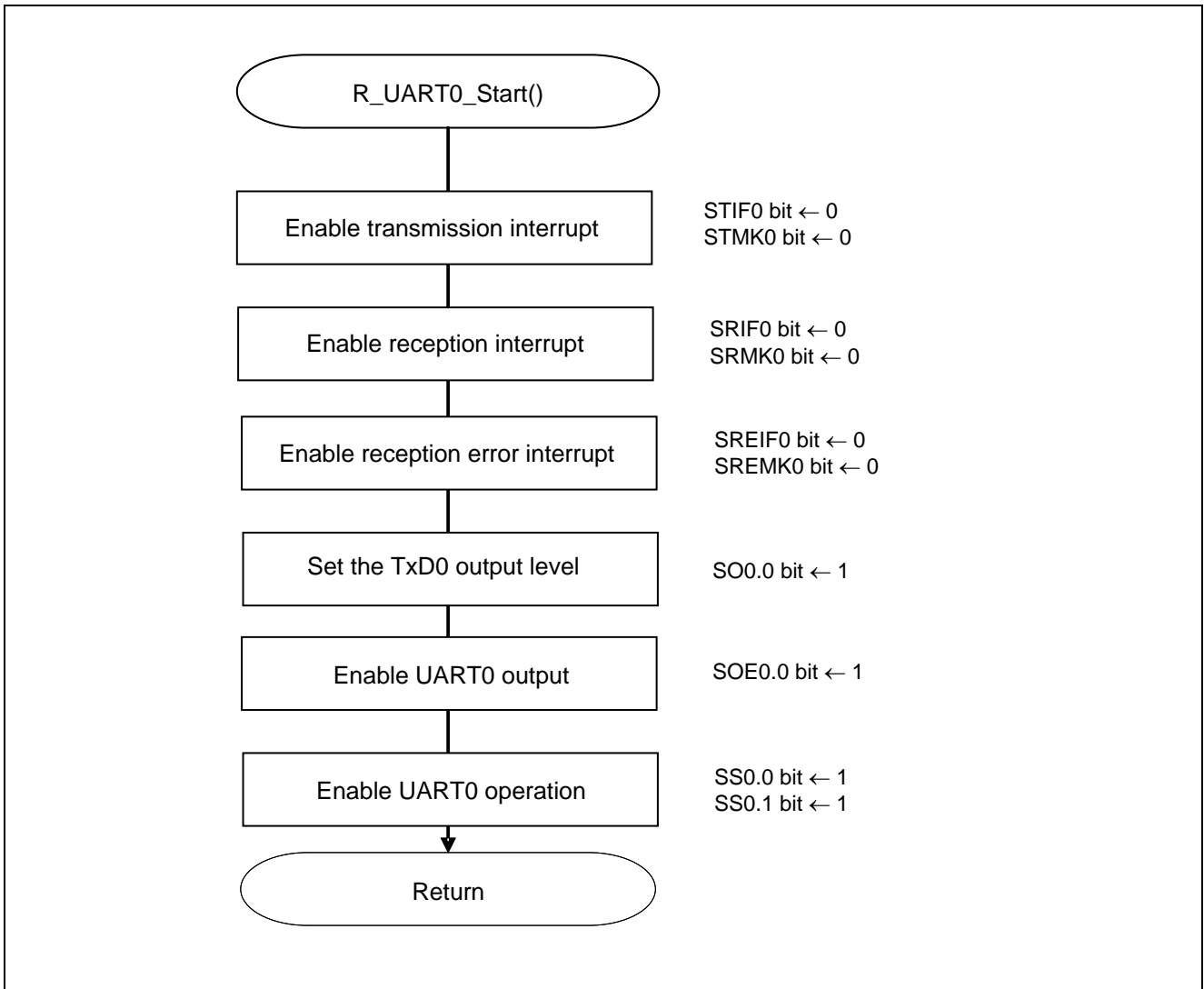
Figure 5.12 shows the flowchart for the UART0 reception status initialization function.



**Figure 5.12 UART0 Reception Status Initialization Function**

**5.7.9 UART0 Operation Start Function**

Figure 5.13 shows the flowchart for the UART0 operation start function.



**Figure 5.13 UART0 Operation Start Function**



## Interrupt setting

- Interrupt request flag register (IF0H)  
Clear the interrupt request flag
- Interrupt mask flag register (MK0H)  
Cancel interrupt mask

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	0	x	x	x	x	x

## Bit 7

SREIF0	Interrupt request flag
0	<b>No interrupt request signal is generated</b>
1	Interrupt request is generated, interrupt request status

## Bit 6

SRIF0	Interrupt request flag
0	<b>No interrupt request signal is generated</b>
1	Interrupt request is generated, interrupt request status

## Bit 5

STIF0	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 TMMK01H	SRMK0 CSIMK01 IICMK01	STMK0 CSIMK00 IICMK00	DMAMK1	DMAMK0	SREMK2 TMMK11H	SRMK2 CSIMK21 IICMK21	STMK2 CSIMK20 IICMK20
0	0	0	x	x	x	x	x

Bit 7

SREMK0	Interrupt processing control
0	<b>Enables interrupt processing.</b>
1	Disables interrupt processing.

Bit 6

SRMK0	Interrupt processing control
0	<b>Enables interrupt processing.</b>
1	Disables interrupt processing.

Bit 5

STMK0	Interrupt processing control
0	<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.

Transition to communication wait state

- Serial channel start register 0 (SS0)  
Operation start

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 <sup>Note</sup>	x	1 <sup>Note</sup>	1

Bits 3 to 0

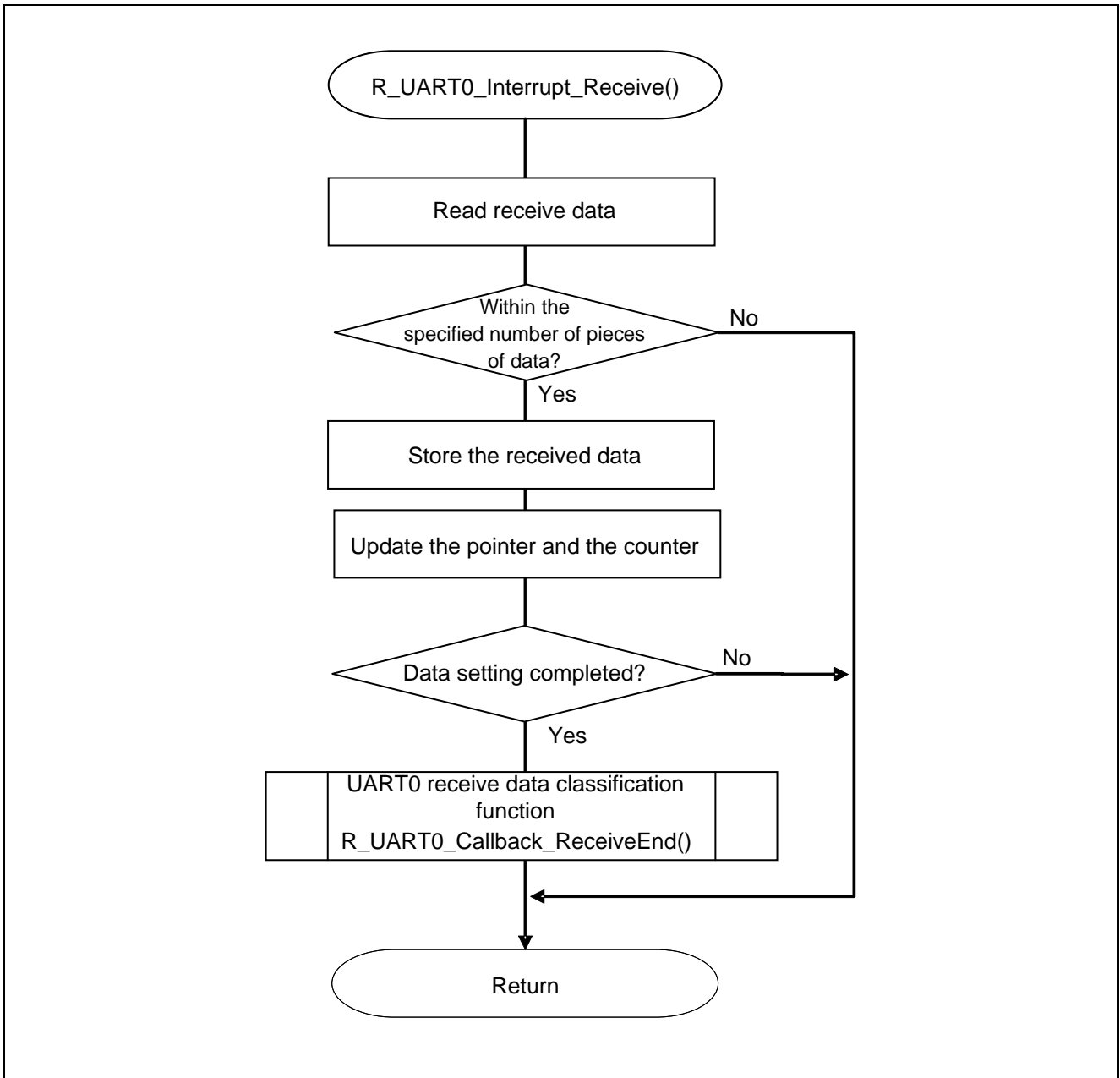
SS0n	Channel n operation start trigger
0	Trigger operation is not performed
1	<b>SE0n is set to 1, and a communication wait state is entered.</b>

Note For UART reception, wait for 4  $f_{CLK}$  clock cycles or more before setting SS0n to 1, after setting the RXE0n bit of the SCR0n register to 1.

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

**5.7.10 INTSR0 Interrupt Service Routine**

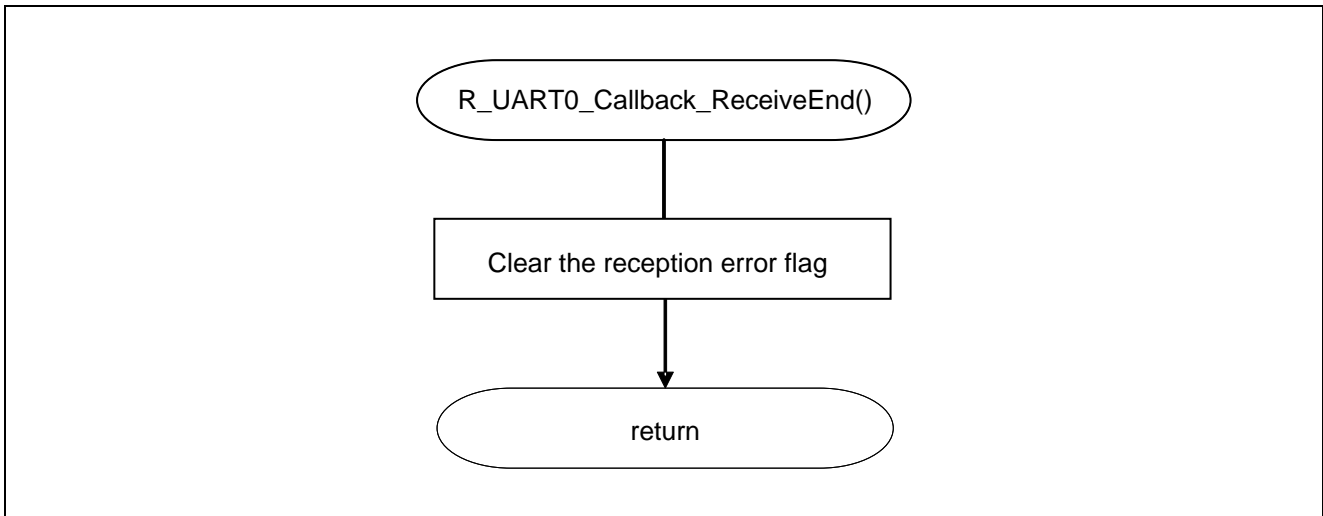
Figure 5.14 shows the flowchart for the INTSR0 interrupt service routine.



**Figure 5.14 INTSR0 Interrupt Service Routine**

### 5.7.11 UART0 Receive Data Classification Function

Figure 5.15 shows the flowchart for the UART0 receive data classification function.



**Figure 5.15** UART0 Receive Data Classification Function

5.7.12 UART0 Data Transmission Function

Figure 5.16 shows the flowchart for the UART0 data transmission function.

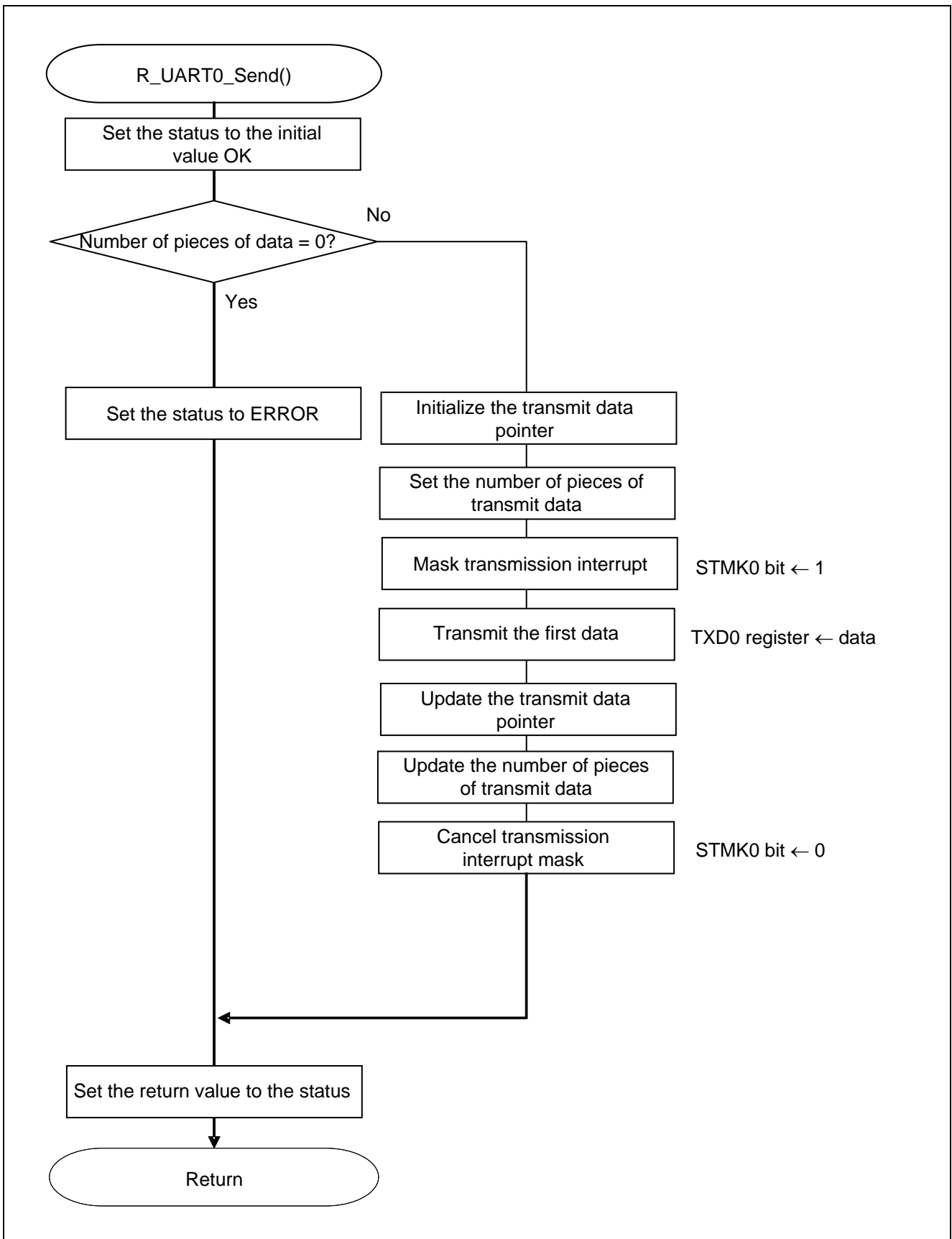
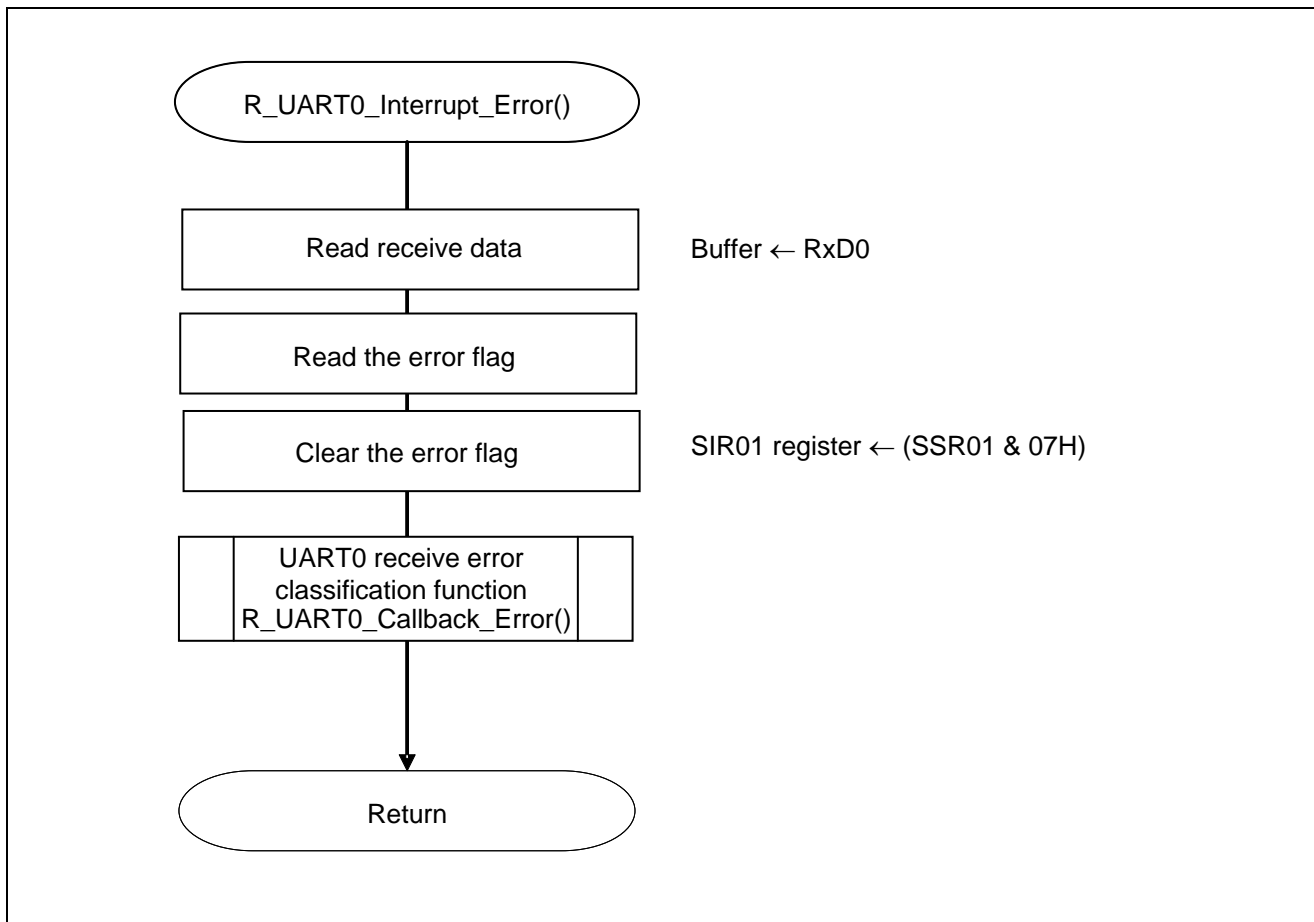


Figure 5.16 UART0 Data Transmission Function

**5.7.13 UART0 Reception Error Interrupt Function**

Figure 5.17 shows the flowchart for the UART0 reception error interrupt function.



**Figure 5.17 UART0 Reception Error Interrupt Function**

### 5.7.14 UART0 Reception Error Classification Function

Figure 5.18 shows the flowchart for the UART0 reception error classification function.

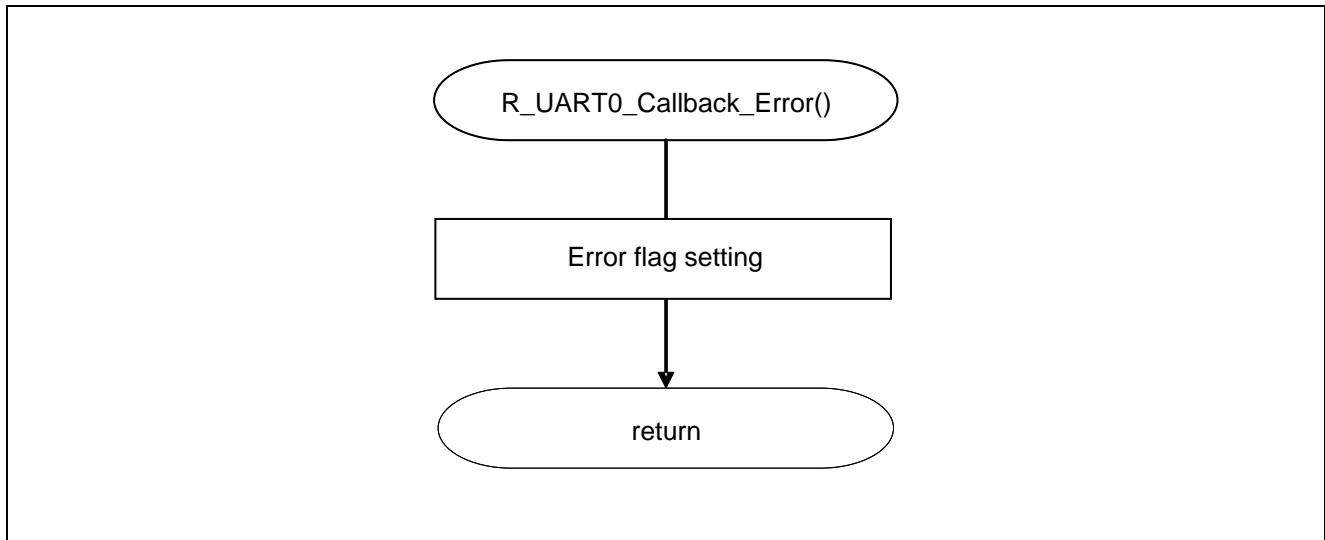
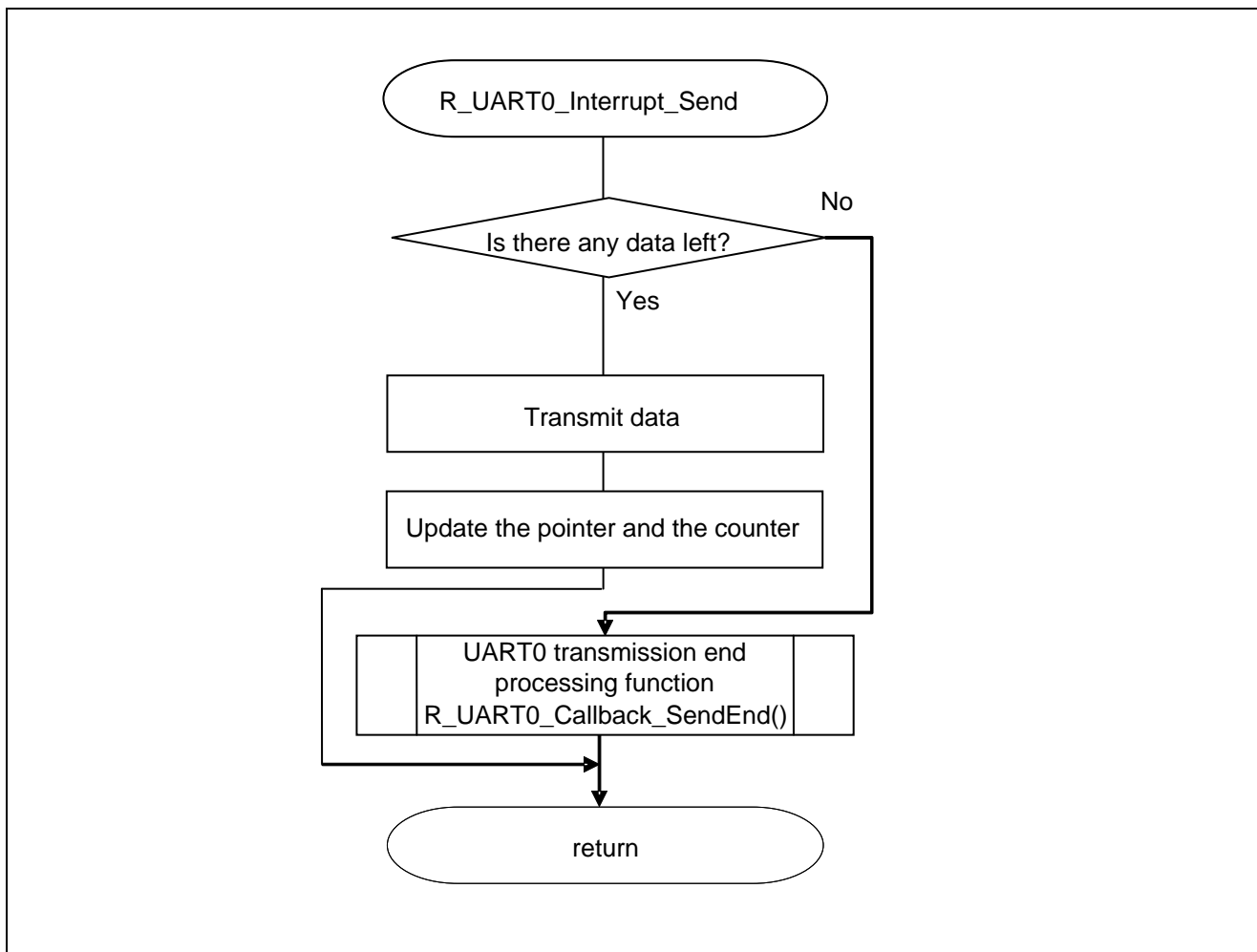


Figure 5.18 UART0 Reception Error Classification Function



**5.7.15 INTST0 Interrupt Service Routine**

Figure 5.19 shows the flowchart for the INTST0 interrupt service routine.



**Figure 5.19 INTST0 Interrupt Service Routine**

### 5.7.16 UART0 Transmission End Processing Function

Figure 5.20 shows the flowchart for the UART0 transmission end processing function.

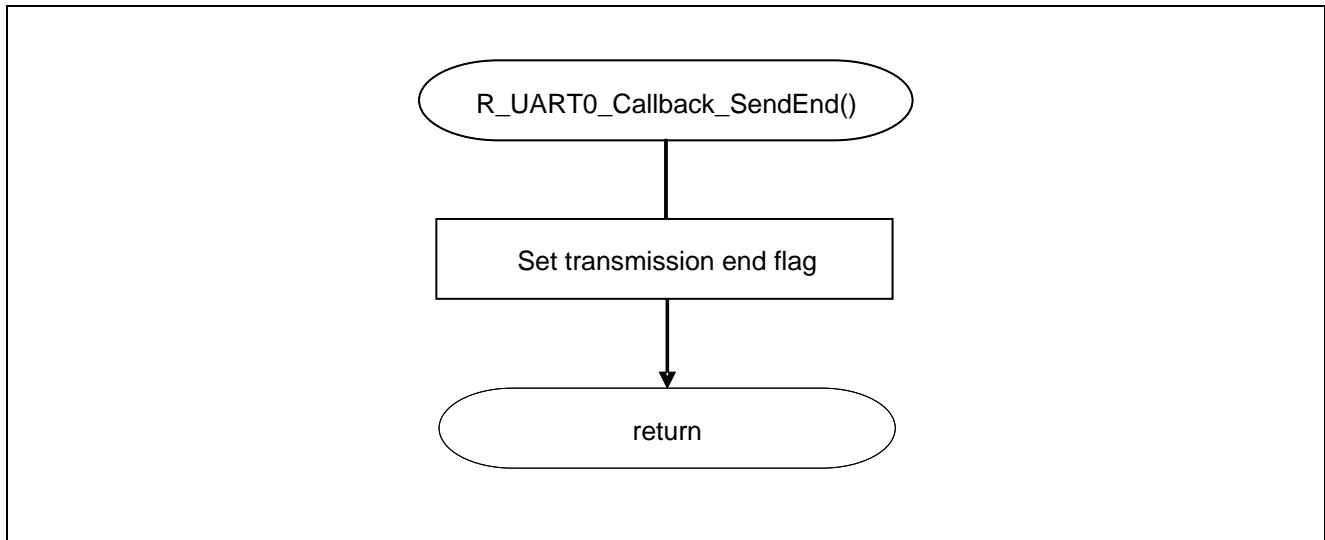


Figure 5.20 UART0 Transmission End Processing Function

## 6. Sample Code

The sample code is available on the Renesas Electronics Website.

## 7. Documents for Reference

RL78/G13 User's Manual: Hardware Rev.0.07 (R01UH0146EJ0007)

RL78 Family User's Manual: Software Rev.1.00 (R01US0015EJ0100)

(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 Serial Array Unit (UART Communication)
-----------------	---

Rev.	Date	Description	
		Page	Summary
1.00	2011.09.30	—	First edition issued
2.00	2014.03.24	5	Table 2.1: Added e2studio and IAR version information
		12	Figure 5.1, Figure 5.2: Added note Figure 5.2: Fixed typo in a function name
		13	Figure 5.3: Fixed typo in the function name

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. 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.



### 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: +1-408-586-6000, 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: +44-1628-651-700, Fax: +44-1628-651-804

**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 301, Tower A, Central Towers, 555 LanGao Rd., Putuo District, Shanghai, China  
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: +852-2886-9318, 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.**  
80 Bendemeer Road, Unit #05-02 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-gu, Seoul, 135-080, Korea  
Tel: +82-2-558-3737, Fax: +82-2-558-5141