RL78/G10

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/G10

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

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. A 16-byte ring buffer for making responses is implemented, making it possible to receive the next data during response transmission.

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).
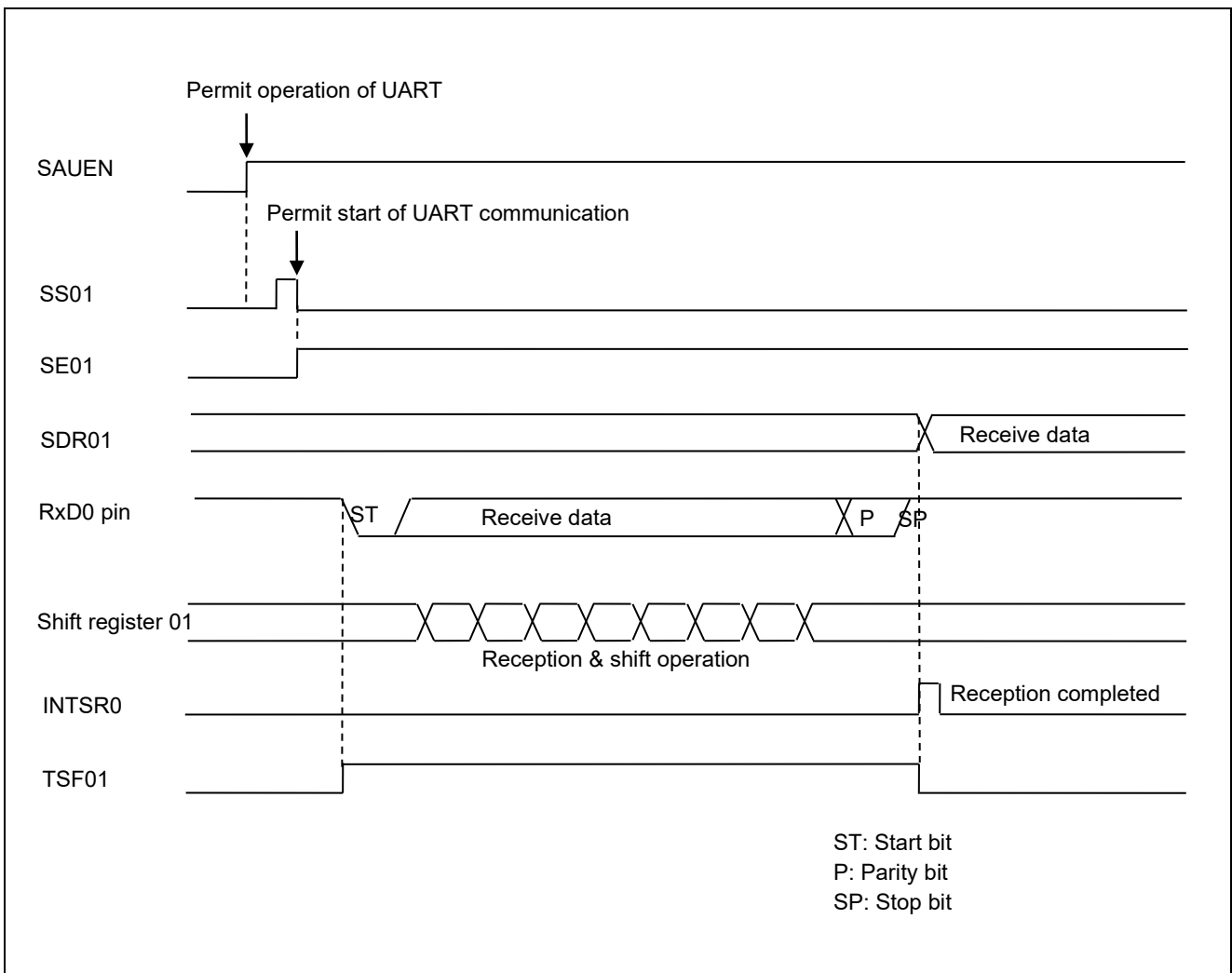
Figure 1.1   UART Reception Timing Chart
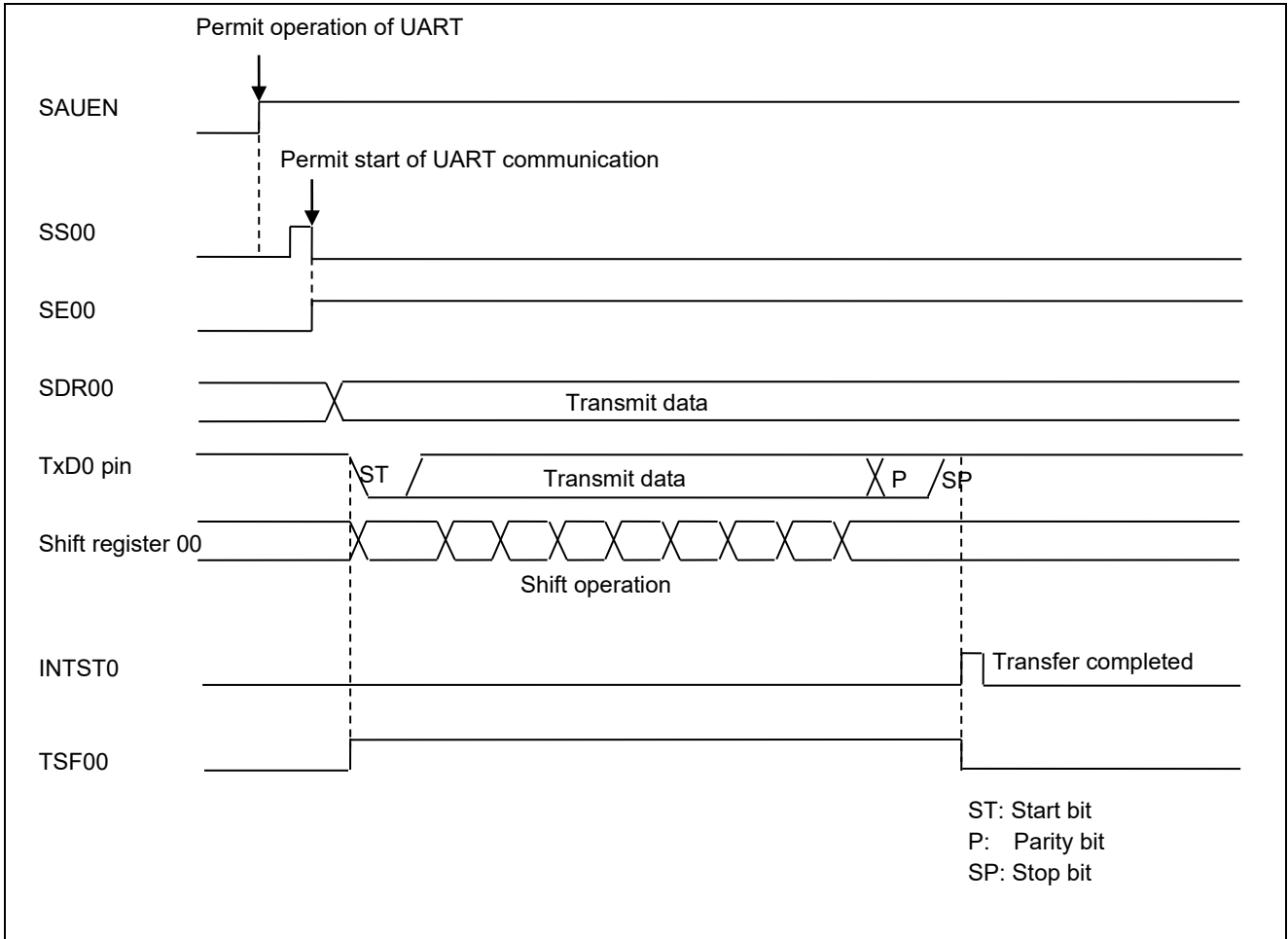
Figure 1.2   UART Transmission Timing Chart

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/G10 (R5F10Y16ASP)
Operating frequency	High-speed on-chip oscillator (HOCO) clock: 20 MHz CPU/peripheral hardware clock: 20 MHz
Operating voltage	5.0 V (Operation is possible over a voltage range of 2.9 to 5.5 V.) SPOR operation: VSPDR=2.82V, VSPOR=2.88V (reset occurrence: VDD < 2.82V, reset release: VDD>=2.88V)
Integrated development environment (CubeSuite+)	CubeSuite + E1.03.00k01 from Renesas Electronics Corp.
Assembler (CubeSuite+)	RA78K0R V1.70 from Renesas Electronics Corp.
Integrated development environment (e2studio)	e2studio V2.0.0.16 from Renesas Electronics Corp.
Assembler (e2studio)	KPIT GNURL78-ELF Toolchain V13.02 from Renesas Electronics Corp.
Board to be used	RL78/G10 target board (QB-R5F10Y16A-TB)

3.   Related Application Note

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

RL78/G10 Initialization (R01AN1454E) 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
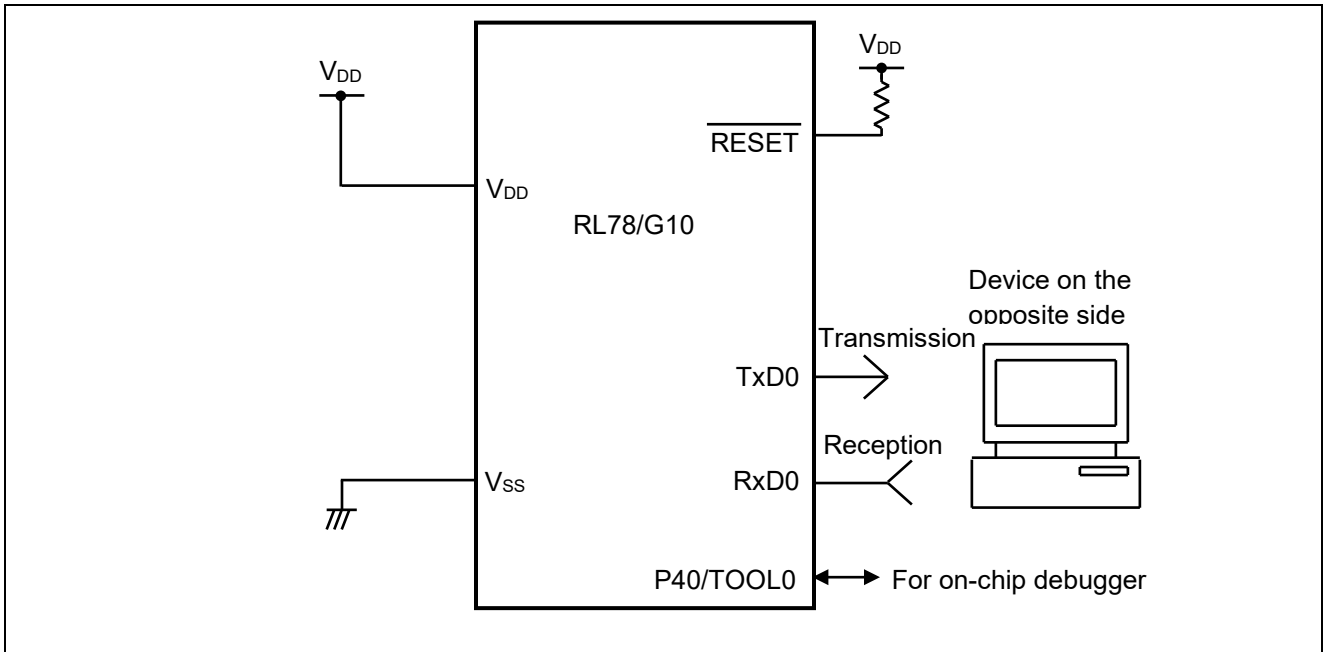
Figure 4.1   Hardware Configuration

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. 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
P00/SO00/TXD0/INTP1	Output	Data transmission pin
P01/ANI0/SI00/RXD0/SDA00/KR2	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).
- Select interrupt priority level 2 or 1 for INTSR0 and for INTSRE0. Select the low interrupt priority level (level 3) for INTST0.

(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 INTSR0 occurs, the system checks the buffer status. If the buffer has enough space, the system accepts receive data, and stores a response message associated with this receive data into the buffer. When INTSRE0 occurs, it also checks the buffer status. If the buffer has enough space, the system performs error processing and stores error code into the buffer.
- When the HALT mode is canceled, the system checks the buffer status. If the buffer is empty, the system returns to the HALT mode. If the buffer contains data, the system sends a response message associated with this data. While it is sending the response message, it executes the HALT instruction and waits for the transmission end interrupt (INTST0), reception end interrupt (INTSR0), and error interrupt (INTSRE0). Upon completion of the message transmission, it returns to the buffer status check process.

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	01101110B	Disables the watchdog timer. (Stops counting after the release from the reset state.)

000C1H	01111111B	SPOR detection voltage
		  When reset occurs: VDD<2.82V
		  When reset is released: VDD >=2.88V

000C2H	11100000B	HOCO: 20 MHz
000C3H	10000101B	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
TMSGOK	'OK', 0DH, 0AH	Response message to reception of "T".
TMSGOK2	'ok', 0DH, 0AH	Response message to reception of "t".
TMSGUC	'UC', 0DH, 0AH	Response message to reception of characters other than "T" or "t".
TMSGFE	'FE', 0DH, 0AH	Response message to a framing error.
TMSGPE	'PE', 0DH, 0AH	Response message to a parity error.
TMSGOE	'OE', 0DH, 0AH	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
8 bits × 16 arrays	RDATABUF	Ring buffer for response messages	SPUTDATA, SGETDATA
8 bits	RSETPNT	Data storage pointer to the ring buffer	SPUTDATA
8 bits	RGETPNT	Pointer for reading data from the ring buffer	SGETDATA
8 bits	RDATACNT	Number of data bytes in the ring buffer	main, SPUTDATA, SGETDATA, IINTSR0, IINTSRE0
8 bits	RUASTT	Transmission status flag (counter)	SUART0SEND, IINTST0

5.5   List of Functions (Subroutines)

Table 5.4 lists the functions (subroutines) that are used by this sample program.

Table 5.4   List of Functions (Subroutines)

Function Name	Outline
SSTARTUART0	UART0 operation enable processing
SSENDERROR	Error response message transmission processing
SUART0SEND	UART0 data transmission function
SPUTDATA	Processing to store response messages into the ring buffer
SGETDATA	Processing to read the oldest response message from the ring buffer
IINTSR0	UART0 reception end interrupt processing
IINTSRE0	UART0 reception error interrupt processing
IINTST0	UART0 transmission end interrupt processing

5.6   Function Specifications

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

[Function Name] STARTUART0

Synopsis	UART0 operation enable processing
Explanation	Starts operation of channels 0 and 1 of serial array unit 0 to make the system enter a communication wait state. Enables reception end interrupts and reception error interrupts.
Argument	None
Return value	None
Remarks	None

[Function Name] SSENDERROR

Synopsis	Error response message transmission processing	
Explanation	Transmits a message associated with error information which was read from the ring buffer.	
Argument	A	: Error information
Return value	None	
Remarks	Performs transmission of four characters multiplied by three types at maximum.	

[Function Name] SUART0SEND

Synopsis	UART0 data transmission function	
Explanation	Transmits data (a response) from UART0. Transmission occurs on the basis of an interrupt.	
Argument	HL	: [Response data storage address]
Return value	None	
Remarks	Waits while interrupts are enabled until transmission of four characters is completed.	

[Function Name] SPUTDATA

Synopsis	Processing to store response messages into the ring buffer	
Explanation	Stores a message associated with receive data into the ring buffer. Also, stores reception error status information into the ring buffer. These operations are conducted if the ring buffer has enough space.	
Argument	A	: [Response message to be stored]
Return value	CY	: [1: Buffer full, 0: Storage operation completed]
Remarks	None	

[Function Name] SGETDATA

Synopsis	Processing to read the oldest response message from the ring buffer	
Explanation	Reads the oldest data from the ring buffer if there are response messages in this buffer.	
Argument	None	
Return value	CY	: [1: No data, 0: Read completed]
	A	: [Read data]
Remarks	None	

[Function Name] IINTSR0

Synopsis	UART0 reception end interrupt processing
Explanation	Checks the receive data and stores a response message associated with this data into the ring buffer if the ring buffer has enough space. Does nothing if the ring buffer is full.
Argument	None
Return value	None
Remarks	None

[Function Name] IINTSRE0

Synopsis	UART0 reception error interrupt processing
Explanation	Clears the reception end interrupt request. Stores the UART0 reception error status into the ring buffer if the ring buffer has enough space. Does nothing if the ring buffer is full.
Argument	None
Return value	None
Remarks	None

[Function Name] IINTST0

Synopsis	UART0 transmission end interrupt processing
Explanation	Decrements the number of send data bytes (RUASTT) by 1. Sends the next data if there is still any data remaining. Disables transmission end interrupts if there is no data remaining.
Argument	None
Return value	None (RUASTT indicates the number of remaining data bytes. 0 indicates the transmission is completed.)
Remarks	None

5.7   Flowcharts

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

```
                    ┌─────────────────────┐
                    (        Start         )
                    └─────────────────────┘
                              │                      The option bytes are referenced before the
                              │                      CPU initialization function is called.
                    ┌─────────────────────┐
                    │ CPU Initialization function │
                    │     RESET_START     │
                    └─────────────────────┘
                              │
                              ▼
                    ┌─────────────────────┐
                    (         End          )
                    └─────────────────────┘
```
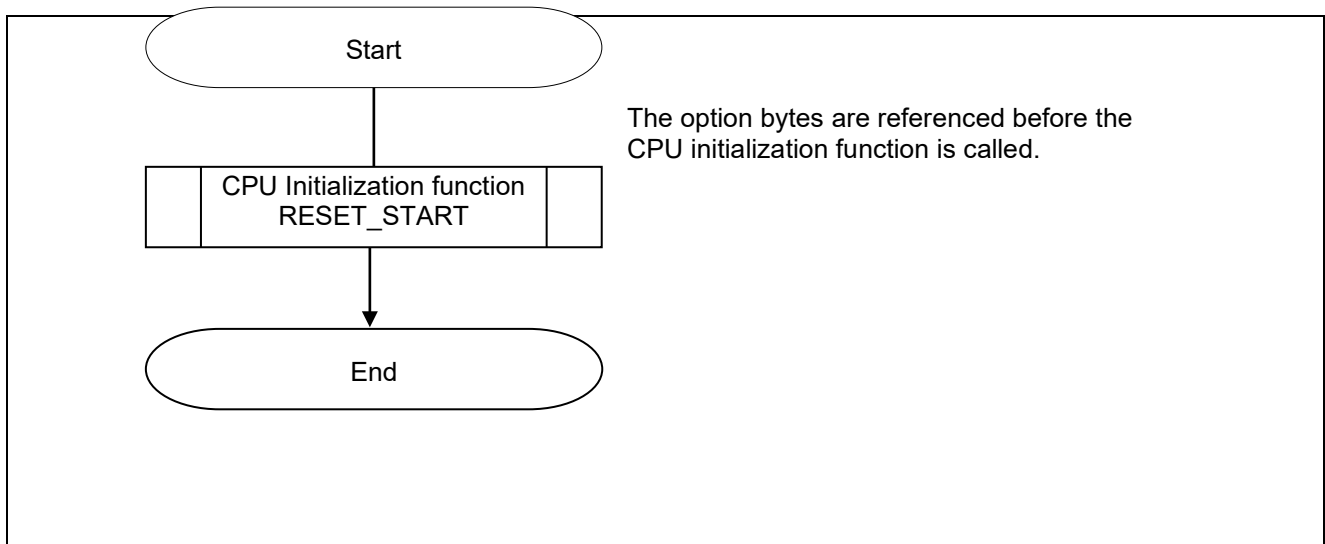
Figure 5.1   Overall Flow

5.7.1 CPU Initialization Function

Figure 5.2 shows the flowchart for the CPU initialization function.
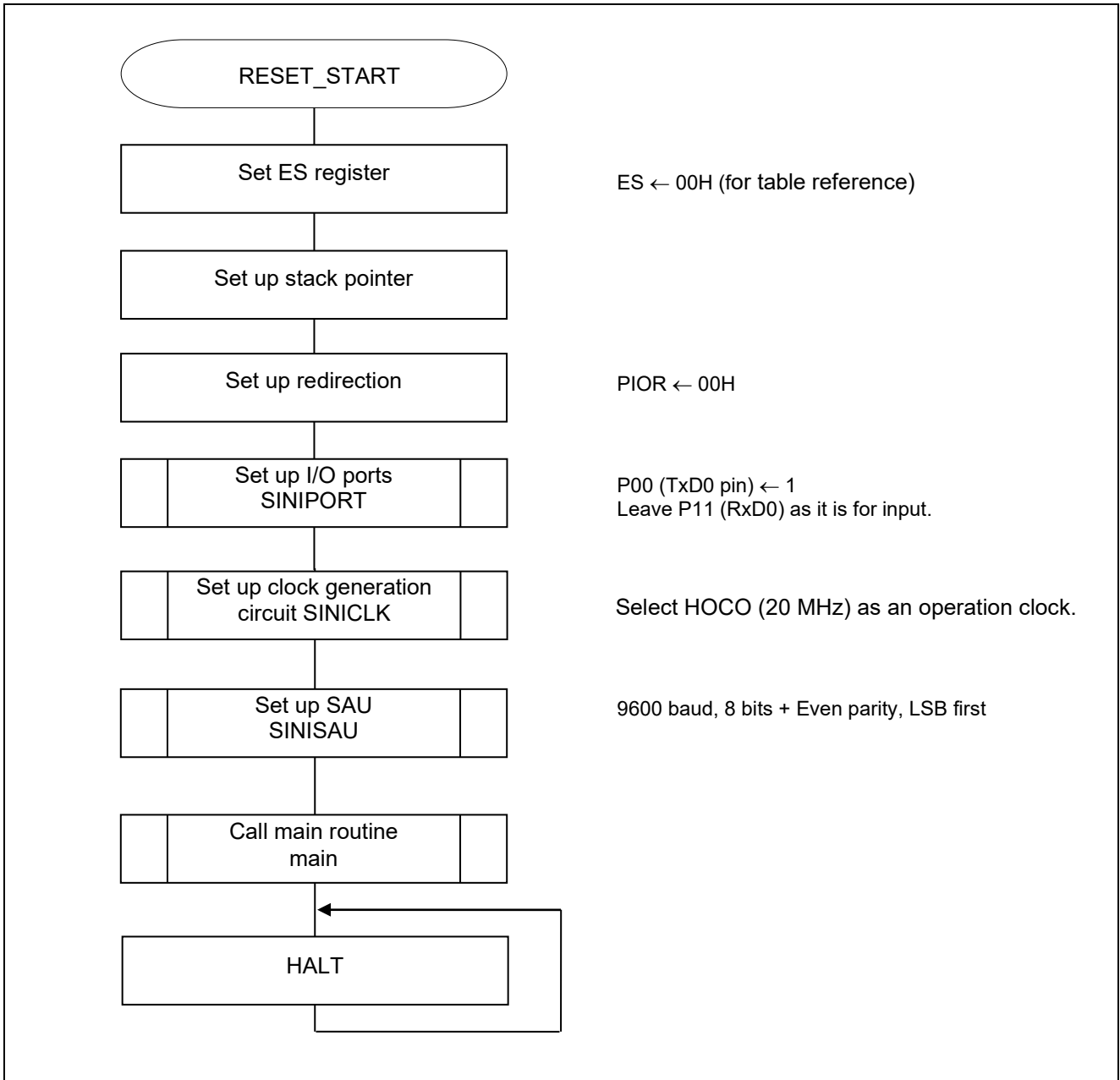
Figure 5.2   CPU Initialization Function

5.7.2　I/O Port Setup

Figure 5.3 shows the flowchart for I/O port setup.
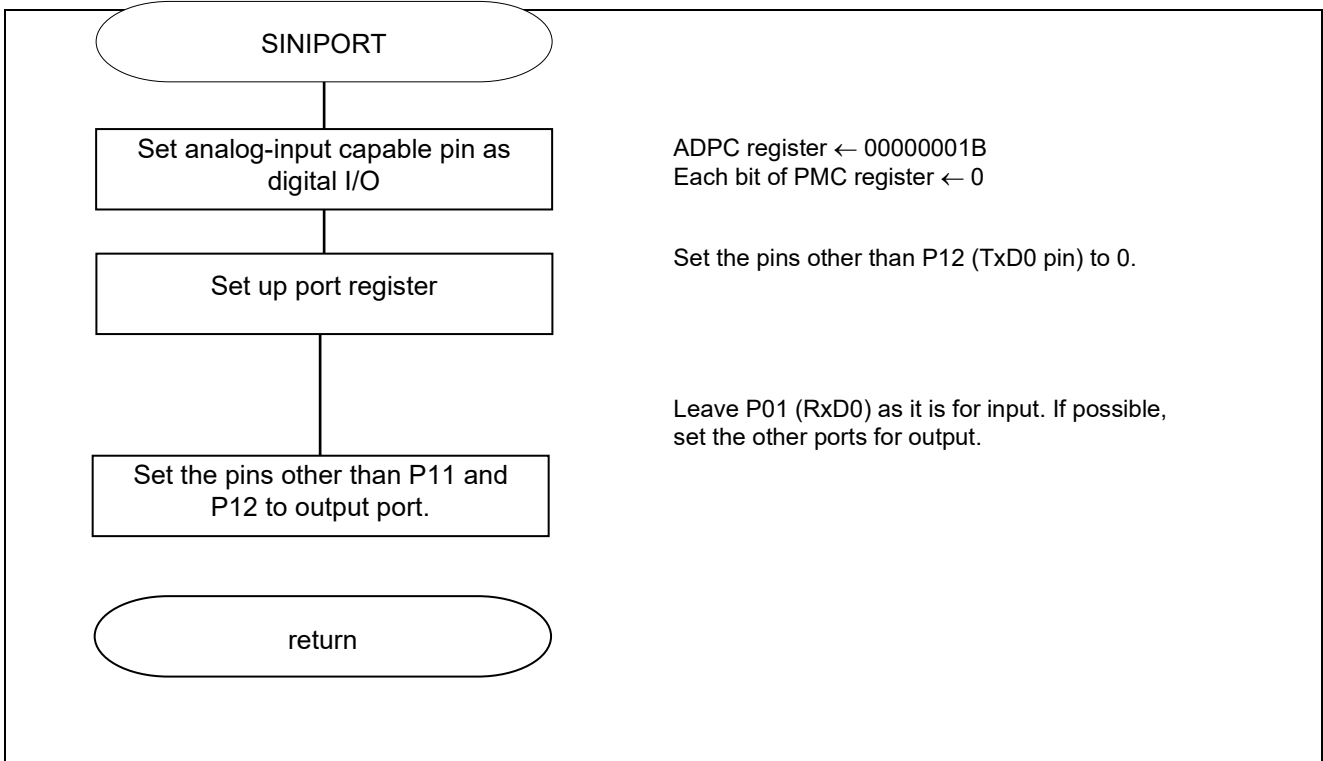
Figure 5.3　I/O Port Setup

Note:　Refer to the section entitled "Flowcharts" in RL78/G10 Initialization Application Note (R01AN1454E) 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.

5.7.3　Clock Generation Circuit Setup

Figure 5.4 shows the flowchart for clock generation circuit setup.
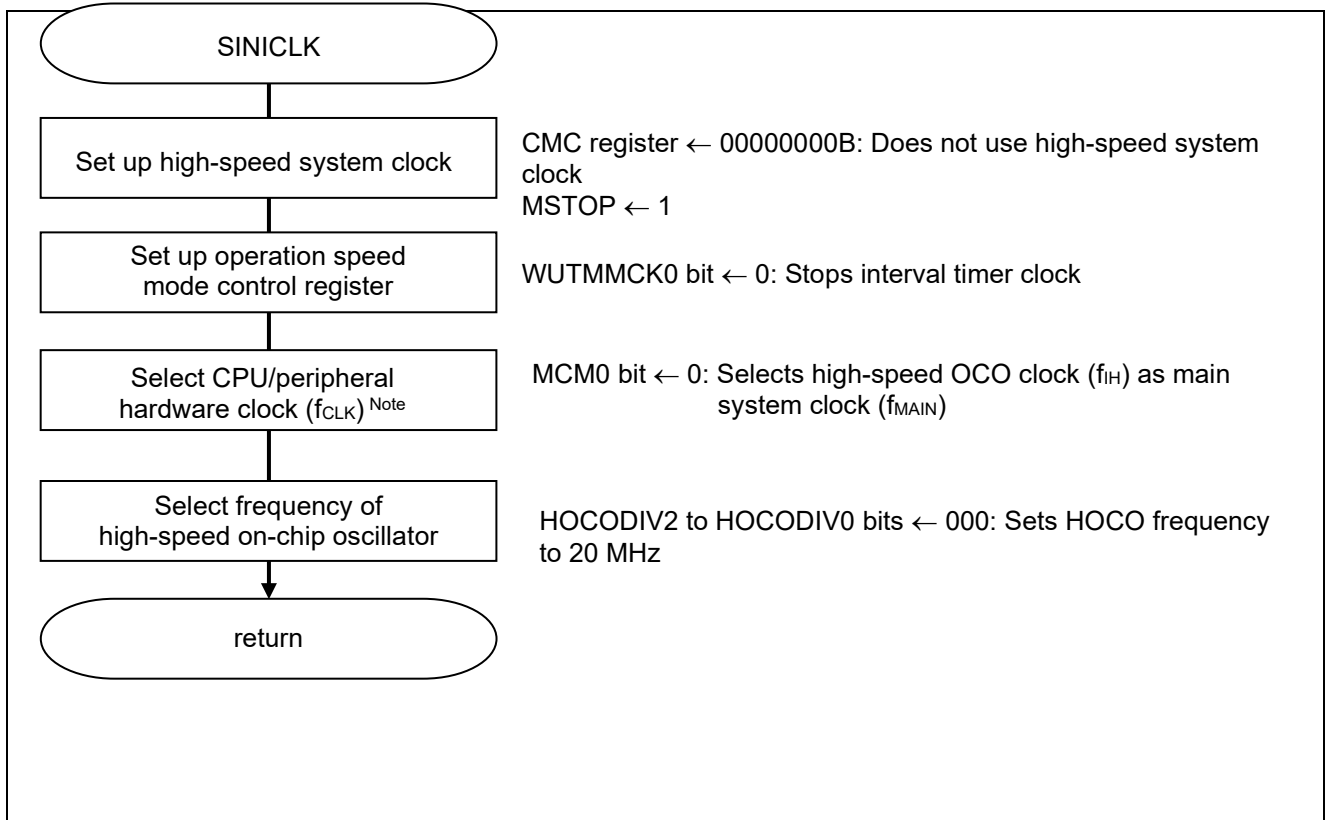
Figure 5.4　Clock Generation Circuit Setup

Note　　16-pin products only.

Caution:　For details on the procedure for setting up the clock generation circuit (SINICLK), refer to the section entitled "Flowcharts" in RL78/G10 Initialization Application Note (R01AN1454E).

5.7.4 SAU Setup

Figure 5.5 shows the flowchart for setting up the SAU.

Figure 5.5   SAU Setup (1/2)

A

| Set transfer speed for channels 0 and 1 | SDR00H and SDR01H registers ← 40H (Operation clock (1.25 MHz) divided by 130.) |

| Set output level to non-inverted state | SOL0 register ← 00H (Output: non-inverted) |

| Set initial value of channel 0 output (TxD0) | SO0 register ← 0001H (Initial level: high) |

| Enable channel 0 output | SOE0L register ← 01H (Enable channel 0 output) |

| Set noise filter | NFEN0 register ← 01H (Turn the RxD0 signal noise filter on.) |

| Clear the error flag | SIR01 register ← 07H |

| Set up interrupt mask | STMK0 bit ← 1: Disable transmission end interrupt
SRMK0 bit ← 1: Disable reception end interrupt
SREMK0 bit ← 1: Disable reception error interrupt |

| Set up interrupt priority | Select interrupt priority level 3 for INTST0, interrupt priority level 2 for INTSR0, and interrupt priority level 1 for INTSRE0.

PR00L register ← 11101111B
PR10L register ← 11011111B |

| Set up port | PM0.0 bit ← 0: TxD0 output
PM0.1 bit ← 1: RxD0 input |

return

Figure 5.5   SAU Setup (2/2)

Start supplying clock to the SAU

- Peripheral enable register 0 (PER0)
  Clock supply

  Symbol: PER0

7	6	5	4	3	2	1	0
TMKAE Note	0	ADCEN	IICA0EN	SAU1EN	SAU0EN	0	TAU0EN
X	0	x	x	x	**1**	0	x

  Bit 5

SAU0EN	Input clock control for serial array unit 0
0	Stops supply of input clock.
1	**Starts supply of input clock.**

  Note   16-pin products only

Select serial clock

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

  Symbol: SPS0

7	6	5	4	3	2	1	0
PRS 013	PRS 012	PRS 011	PRS 010	PRS 003	PRS 002	PRS 001	PRS 000
0	0	0	0	**0**	**1**	**0**	**0**

  Bits 7 to 0

PRS 0n3	PRS 0n2	PRS 0n1	PRS 0n0		Operation clock (CK0n) selection (n = 0, 1)				
					f_{CLK} = 1.25MHz	f_{CLK} = 2.5MHz	f_{CLK} = 5MHz	f_{CLK} = 10MHz	f_{CLK} = 20MHz
0	0	0	0	f_{CLK}	1.25MHz	2.5MHz	5MHz	10MHz	20MHz
0	0	0	1	$f_{CLK}/2$	625kHz	1.25MHz	2.5MHz	5MHz	10MHz
0	0	1	0	$f_{CLK}/2^2$	313kHz	625kHz	1.25MHz	2.5MHz	5MHz
0	0	1	1	$f_{CLK}/2^3$	156kHz	313kHz	625kHz	1.25MHz	2.5MHz
0	**1**	**0**	**0**	**$f_{CLK}/2^4$**	**78kHz**	**156kHz**	**313kHz**	**625kHz**	**1.25MHz**
0	1	0	1	$f_{CLK}/2^5$	39kHz	78kHz	156kHz	313kHz	625kHz
0	1	1	0	$f_{CLK}/2^6$	19.5kHz	39kHz	78kHz	156kHz	313kHz
0	1	1	1	$f_{CLK}/2^7$	9.8kHz	19.5kHz	39kHz	78kHz	156kHz
1	0	0	0	$f_{CLK}/2^8$	4.9kHz	9.8kHz	19.5kHz	39kHz	78 kHz
1	0	0	1	$f_{CLK}/2^9$	2.5kHz	4.9kHz	9.8kHz	19.5kHz	39kHz
1	0	1	0	$f_{CLK}/2^{10}$	1.22kHz	2.5kHz	4.9kHz	9.8kHz	19.5kHz
1	0	1	1	$f_{CLK}/2^{11}$	625Hz	1.22kHz	2.5kHz	4.9kHz	9.8kHz
1	1	0	0	$f_{CLK}/2^{12}$	313Hz	625Hz	1.22kHz	2.5kHz	4.9kHz
1	1	0	1	$f_{CLK}/2^{13}$	152Hz	313Hz	625Hz	1.22kHz	2.5kHz
1	1	1	0	$f_{CLK}/2^{14}$	78Hz	152Hz	313Hz	625Hz	1.22kHz
1	1	1	1	$f_{CLK}/2^{15}$	39Hz	78Hz	152Hz	313Hz	625Hz

Caution: For details on the register setup procedures, refer to RL78/G10 User's Manual: Hardware.

R01AN1455EJ0210   Rev.2.10
Sep.30.22
     Page 17 of 45

RENESAS

Transmission channel operation mode setting

- Serial mode register 00 (SMR00H, SMR00L)
  Interrupt source
  Operation mode
  Transfer clock selection
  f_{MCK} selection

Symbol: SMR00H, SMR00L

	SMR00H								SMR00L						
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CKS 00	CCS 00	0	0	0	0	0	STS 00	0	0	1	0	0	MD 002	MD 001	MD 000
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	**Prescaler output clock CK00 configured by the SPS0 register**
1	Prescaler output clock CK01 configured by the SPS0 register

Bit 14

CCS00	Channel 0 transfer clock (TCLK) selection
0	**Clock obtained by dividing the operation clock f_{MCK} specified by the CKS00 bit**
1	Clock input from the SCK pin

Bit 8

STS00	Selection of start trigger factor
0	**Only the software trigger is valid.**
1	Valid edge of the RxD pin (selected for UART reception)

Bit 6

SIS000	Controls inversion of level of receive data of channel 0 in UART mode
0	**Falling edge is detected as the start bit.**
1	Rising edge is detected as the start bit.

Bits 2 and 1

MD002	MD001	Channel 0 operation mode setting
0	0	CSI mode
0	**1**	**UART mode**
1	0	Simplified I^{2}C mode
1	1	Setting prohibited

Bit 0

MD000	Channel 0 interrupt source selection
0	**Transfer end interrupt**
1	Buffer empty interrupt

Caution: For details on the register setup procedures, refer to RL78/G10 User's Manual: Hardware.

Reception channel operation mode setting

- Serial mode register 01 (SMR01H, SMR01L)
  Interrupt source
  Operation mode
  Transfer clock selection
  f_{MCK} selection

Symbol: SMR01H, SMR01L

		SMR01H								SMR01L					
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CKS 01	CCS 01	0	0	0	0	0	STS 01	0	SIS 010	1	0	0	MD 012	MD 011	MD 010
0	**0**	0	0	0	0	0	**1**	0	**0**	1	0	0	**0**	**1**	**0**

Bit 15

CKS01	Channel 1 operation clock (f_{MCK}) 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_{MCK} 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 I2C 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/G10 User's Manual: Hardware.

Transmission channel communication operation setting

- Serial communication operation setting register 00 (SCR00H, SCR00L)
  Data length setting, data transfer order, error interrupt signal mask availability, and operation mode

Symbol: SCR00H, SCR00L

	SCR00H								SCR00L						
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 15 and 14

TXE00	RXE00	Channel 0 operation mode setting
0	0	Communication prohibited
0	1	Reception only
1	**0**	**Transmission only**
1	1	Both transmission and reception

Bit 10

EOC00	Error interrupt signal (INTSRE0) mask availability selection
0	**Error interrupt INTSRE0 is masked**
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
1	**0**	**Even parity is output**	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
1	**Input and output in LSB first**

Bits 5 and 4

SLC001	SLC000	Stop bit setting in UART mode
0	0	No stop bit
0	**1**	**Stop bit length = 1 bit**
1	0	Stop bit length = 2 bits
1	1	Setting prohibited

Caution:　For details on the register setup procedures, refer to RL78/G10
User's Manual: Hardware.

Symbol: SCR00H, SCR00L

	SCR00H									SCR00L					
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
1	0	7-bit data length
1	**1**	**8-bit data length**
Others		Setting prohibited

Caution: For details on the register setup procedures, refer to RL78/G10 User's Manual: Hardware.

Reception channel communication operation setting

- Serial communication operation setting register 01 (SCR01H, SCR01L)
  Data length setting, data transfer order, error interrupt signal mask availability, and operation mode

Symbol: SCR01H, SCR01L

SCR01H ... SCR01L

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
0	**1**	**0**	**0**	**0**	**1**	**1**	**0**	**1**	**0**	**0**	**1**	**0**	**1**	**1**	**1**

Bits 15 and 14

TXE01	RXE01	Channel 1 operation mode setting
0	0	Communication prohibited
0	**1**	**Reception only**
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 (INTSRE0) mask availability selection
0	Error interrupt INTSRE0 is masked
1	**Generation of error interrupt INTSRE0 is enabled**

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
1	**0**	Even parity is output	**Check is made for even parity**
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
1	**Input and output in LSB first**

Bits 5 and 4

SLC011	SLC010	Stop bit setting in UART mode
0	0	No stop bit
0	**1**	**Stop bit length = 1 bit**
1	0	Stop bit length = 2 bits
1	1	Setting prohibited

Caution: For details on the register setup procedures, refer to RL78/G10 User's Manual: Hardware.

Symbol: SCR01H, SCR01L

		SCR01H								SCR01L					
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
0	**1**	**0**	**0**	**0**	**1**	**1**	**0**	**1**	**0**	**0**	**1**	**0**	**1**	**1**	**1**

Bits 1 and 0

DLS011	DLS010	Data length setting in CSI mode
0	1	9-bit data length
1	0	7-bit data length
1	**1**	**8-bit data length**
Others		Setting prohibited

Transmission channel transfer clock setting

- Serial data register 00 (SDR00H, SDR00L)

    Transfer clock frequency: $f_{MCK}/130$ ($\approx$ 9600 Hz)

    Symbol: SDR00

			SDR00H								SDR00L				
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
1	**0**	**0**	**1**	**1**	**0**	**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	**0**	**0**	**0**	**0**	**0**	**0**	**$f_{MCK}/130$**
. .	. .	. .	. .	. .	. .	. .	. .
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/G10 User's Manual: Hardware.

Reception transfer clock setting
- Serial data register 01 (SDR01H, SDR01L)
  Transfer clock frequency: $f_{MCK}/130$ ($\approx$ 9600 Hz)

Symbol: SDR01H, SDR01L

			SDR01H								SDR01L				
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
1	0	0	1	1	0	1	0	x	x	x	x	x	x	x	x
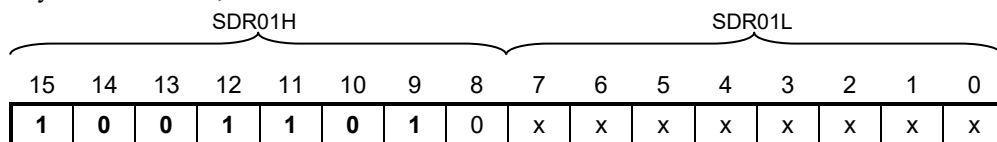
Bits 15 to 9

			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	$f_{MCK}/8$
.	.	.	.	.	.	.	.
.	.	.	.	.	.	.	.
1	**0**	**0**	**0**	**0**	**0**	**0**	**$f_{MCK}/130$**
.	.	.	.	.	.	.	.
.	.	.	.	.	.	.	.
1	1	1	1	1	1	0	$f_{MCK}/254$
1	1	1	1	1	1	1	$f_{MCK}/256$

Output level setting
- Serial output level register 0 (SOL0/SOL0L)
  Output: Non-inverted

Symbol: SOL0

7	6	5	4	3	2	1	0
0	0	0	0	0	0	0	SOL00
0	0	0	0	0	0	0	**0**
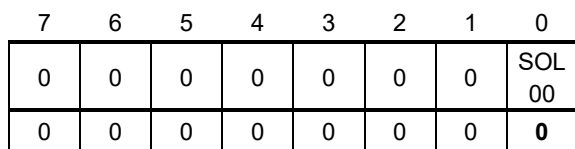
Bit 0

SOL00	Selects inversion of the level of the transmit data of channel n in UART mode
0	**Communication data is output as it is.**
1	Communication data is inverted and output.

Caution: For details on the register setup procedures, refer to RL78/G10 User's Manual: Hardware.

Initial output level setting

- Serial output register 0 (SO0)
  Initial output: 1

Symbol: SO0

7	6	5	4	3	2	1	0
0	0	0	0	0	0	SO 01	SO 00
0	0	0	0	0	0	x	1

Bit 0

SO00	Channel 0 serial data output
0	Serial data output value is "0"
1	**Serial data output value is "1"**

Enabling of data output on target channel

- Serial output enable register 0 (SOE0/SOE0L)
  Output enable

Symbol: SOE0

7	6	5	4	3	2	1	0
0	0	0	0	0	0	SOE 01	SOE 00
0	0	0	0	0	0	x	1

Bit 0

SOE00	Channel 0 serial output enable/stop
0	Serial communication output is stopped
1	**Serial communication output is enabled**

Enabling of noise filter

- Noise filter enable register 00 (NFEN0)
  Turn the noise filter for the RxD0 pin on.

Symbol: SOE0

7	6	5	4	3	2	1	0
0	0	0	0	0	0	0	SNFEN00
0	0	0	0	0	0	0	1

Bit 0

SNFEN00	Use of noise filter of RxD0 pin (RxD0/P01)
0	Noise filter OFF
1	**Noise filter ON**

Caution: For details on the register setup procedures, refer to RL78/G10 User's Manual: Hardware.

Clearing of the error flag

- Serial flag clear trigger register 01 (SIR01)
  Clear the error flag.

Symbol: SIR01

7	6	5	4	3	2	1	0
0	0	0	0	0	FECT01	PECT01	OVCT01
0	0	0	0	0	**1**	**1**	**1**

Bit 2

FECT01	Clear trigger of framing error flag of channel 1
0	Not cleared
1	**Clears the FEF01 bit of the SSR01 register.**

Bit 1

PECT01	Clear trigger of parity error flag of channel 1
0	Not cleared
1	**Clears the PEF01 bit of the SSR01 register.**

Bit 0

OVCT01	Clear trigger of overrun error flag of channel 1
0	Not cleared
1	**Clears the OVF01 bit of the SSR01 register.**

Configuring the interrupt mask

- Interrupt mask flag register 0H (MK0L)
  Disable interrupt processing.
- Priority order specification flag registers (PR00L, PR10L)
  Specify the interrupt priority.

Symbol: MK0L (For 10-pin products)

7	6	5	4	3	2	1	0
TMMK00	TMMK01H	SREMK0	SRMK0	STMK0 CSIMK00 IICMK00	PMK1	PMK0	WDTIMK
x	x	**1**	**1**	**1**	x	x	x

SREMK0	SRMK0	STMK0	Interrupt processing control
0	0	0	Interrupt processing enabled
1	**1**	**1**	**Interrupt processing disabled**

Caution: For details on the register setup procedures, refer to RL78/G10 User's Manual: Hardware.

Symbol: PR00L (For 10-pin products)

7	6	5	4	3	2	1	0
TMPR000	TMPR001H	SREPR00	SRPR00	STPR00 CSIPR000 IICPR000	PPR01	PPR00	WDTIPR0
x	x	**1**	**0**	**1**	x	x	x

Symbol: PR10H (For 10-pin products)

TMPR100	TMPR101	SREPR10	SRPR00	STPR10 CSIPR100 IICPR100	PPR11	PPR10	WDTIPR1
x	x	**0**	**1**	**1**	x	x	xx

Bits 2 to 0

xxPR1x	xxPR0x	Priority level selection
0	0	Selects level 0 (high priority level)
0	**1**	**Selects level 1**
1	**0**	**Selects level 2**
1	**1**	**Selects level 3 (low priority level)**

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**

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/G10 User's Manual: Hardware.

5.7.5   Main Processing
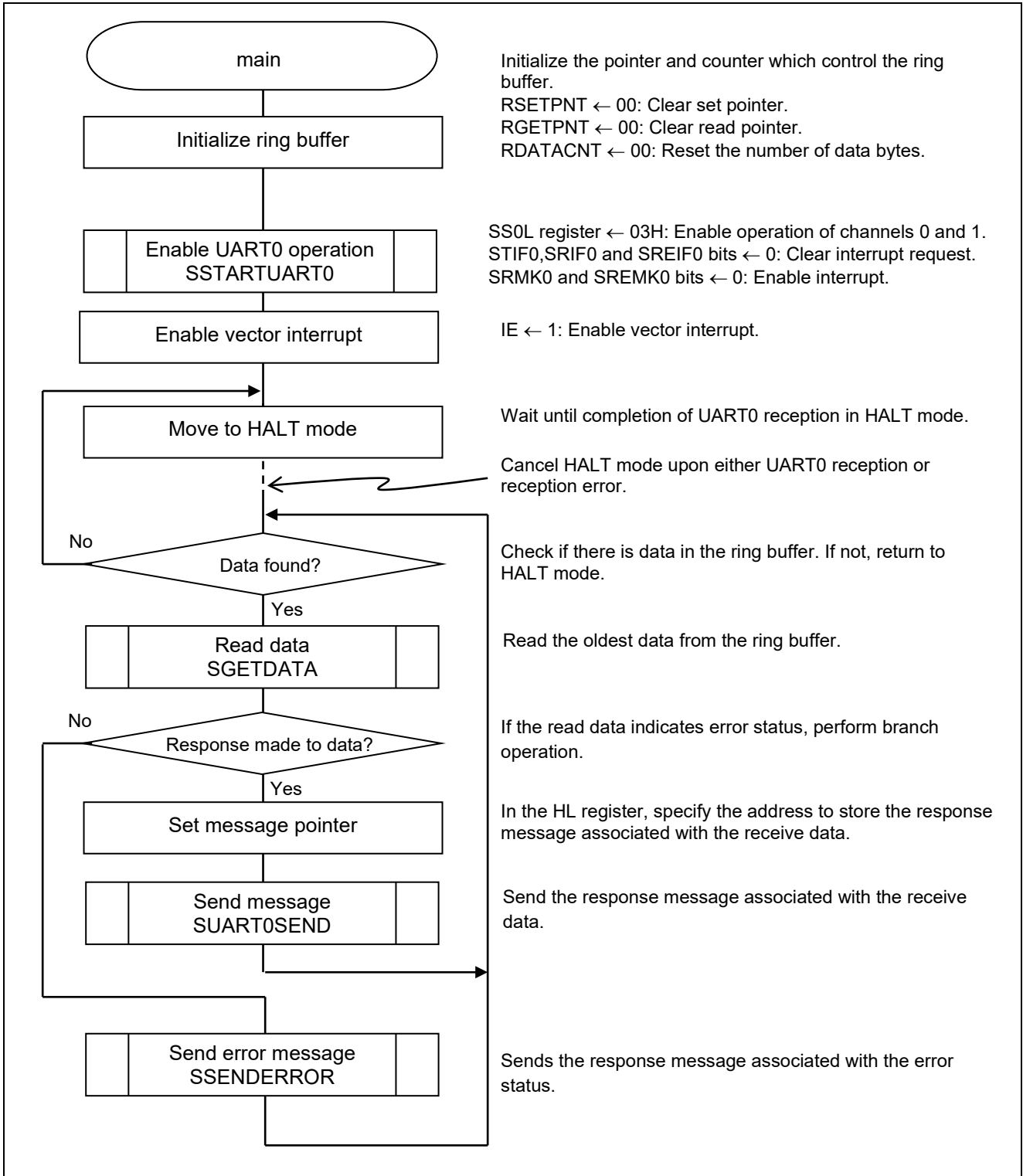
Figure 5.6 shows the flow chart for main processing.

The flow chart contains the following process steps and annotations:

main

Initialize ring buffer
Initialize the pointer and counter which control the ring buffer.
RSETPNT ← 00: Clear set pointer.
RGETPNT ← 00: Clear read pointer.
RDATACNT ← 00: Reset the number of data bytes.

Enable UART0 operation SSTARTUART0
SS0L register ← 03H: Enable operation of channels 0 and 1.
STIF0,SRIF0 and SREIF0 bits ← 0: Clear interrupt request.
SRMK0 and SREMK0 bits ← 0: Enable interrupt.

Enable vector interrupt
IE ← 1: Enable vector interrupt.

Move to HALT mode
Wait until completion of UART0 reception in HALT mode.
Cancel HALT mode upon either UART0 reception or reception error.

Data found?
No / Yes
Check if there is data in the ring buffer. If not, return to HALT mode.

Read data SGETDATA
Read the oldest data from the ring buffer.

Response made to data?
No / Yes
If the read data indicates error status, perform branch operation.

Set message pointer
In the HL register, specify the address to store the response message associated with the receive data.

Send message SUART0SEND
Send the response message associated with the receive data.

Send error message SSENDERROR
Sends the response message associated with the error status.

Figure 5.6   Main Processing

5.7.6   UART0 Operation Enable Function

Figure 5.7 shows the flowchart for the UART0 operation enable function.
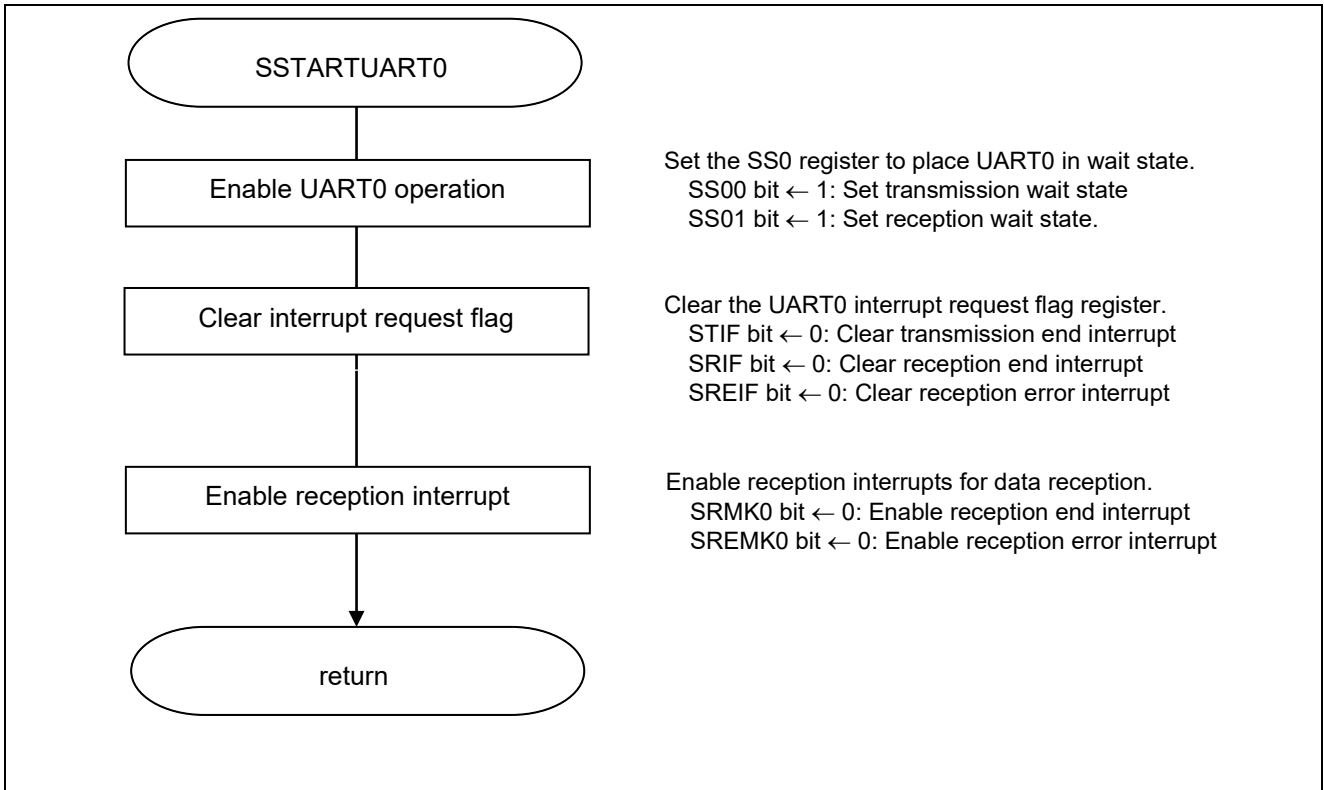

Figure 5.7   UART0 Operation Enable Function

Transition to a communication wait state

- Serial channel start register 0 (SS0/SS0L)
  Start operation.

  Symbol: SS0

7	6	5	4	3	2	1	0
0	0	0	0	0	0	SS01	SS00
0	0	0	0	0	0	1 Note	1

  Bits 3 to 0

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

  Note:　For UART reception, wait for 4 fCLK clock cycles or more before setting SS0n to 1, after setting the RXE0n bit of the SCR0n register to 1.

Interrupt setting

- Interrupt request flag register (IF0L)
  Clear the interrupt request flag
- Interrupt mask flag register (MK0L)
  Cancel interrupt mask

  Symbol: IF0L (for 10-pin products)

7	6	5	4	3	2	1	0
TMIF00	TMIF01H	SREIF0	SRIF0	STIF0 CSIIF00 IICIF00	PIF1	PIF0	WDTIIF
x	x	**0**	**0**	**0**	x	x	x

SREIF0	SRIF0	STIF0	Interrupt request flag
0	**0**	**0**	**No interrupt request signal is generated**
1	1	1	Interrupt request is generated, interrupt request status

  Symbol: MK0L (10-pin products)

7	6	5	4	3	2	1	0
TMMK00	TMMK01H	SREMK0	SRMK0	STMK0 CSIMK00 IICMK00	PMK1	PMK0	WDTIMK
x	x	**0**	**0**	**1**	x	x	x

SREMK0	SRMK0	STMK0	Interrupt processing control
0	**0**	0	**Enables interrupt processing.**
1	1	1	**Disable interrupt processing**

Caution:　For details on the register setup procedures, refer to RL78/G10 User's Manual: Hardware.

5.7.7 Error Response Message Transmission Function

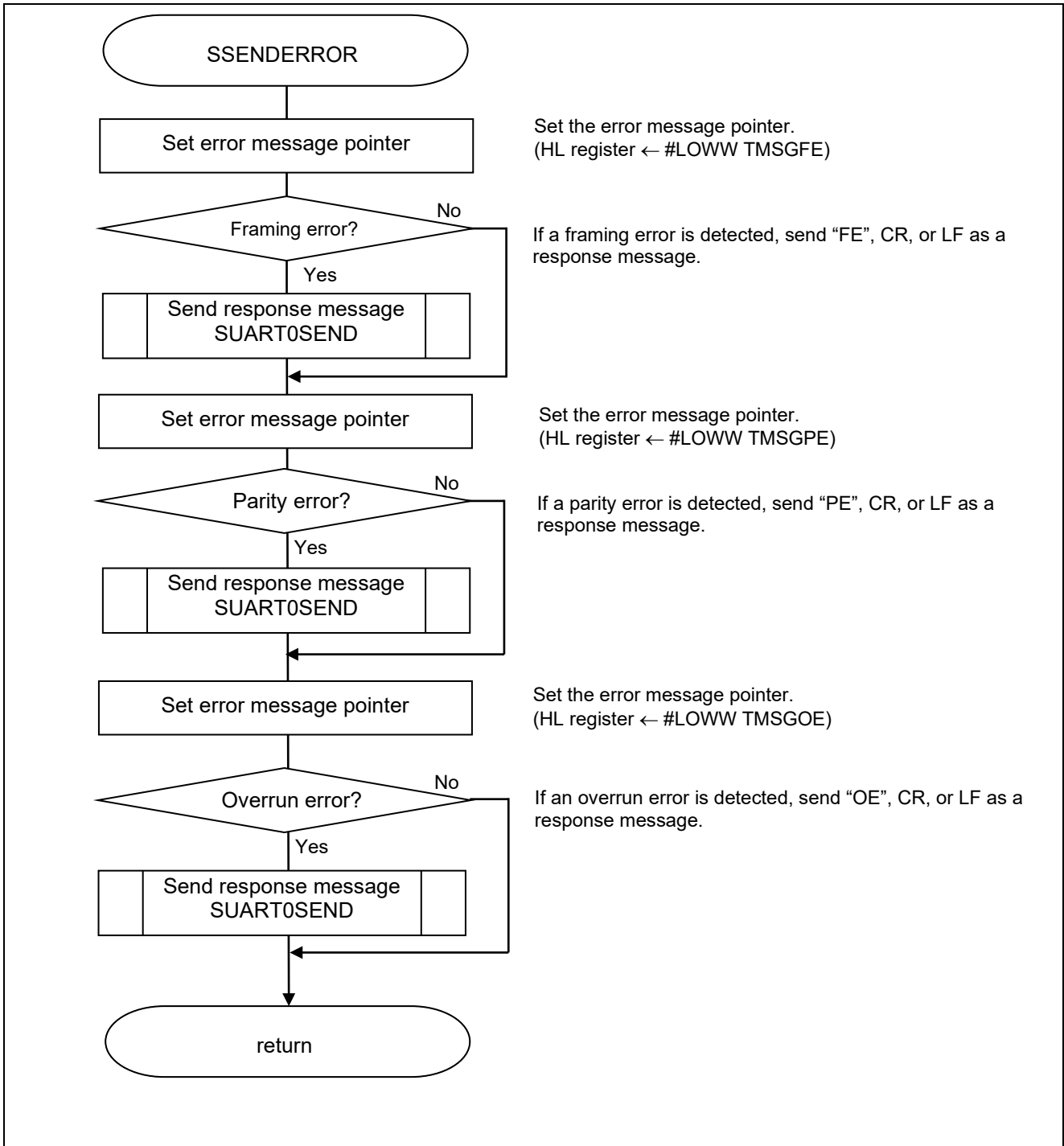Figure 5.8 shows the flowchart for the error response message transmission function.

Figure 5.8   Error Response Message Transmission Function

5.7.8   UART0 Data Transmission Function

Figure 5.9 shows the flowchart for the UART0 data transmission function.
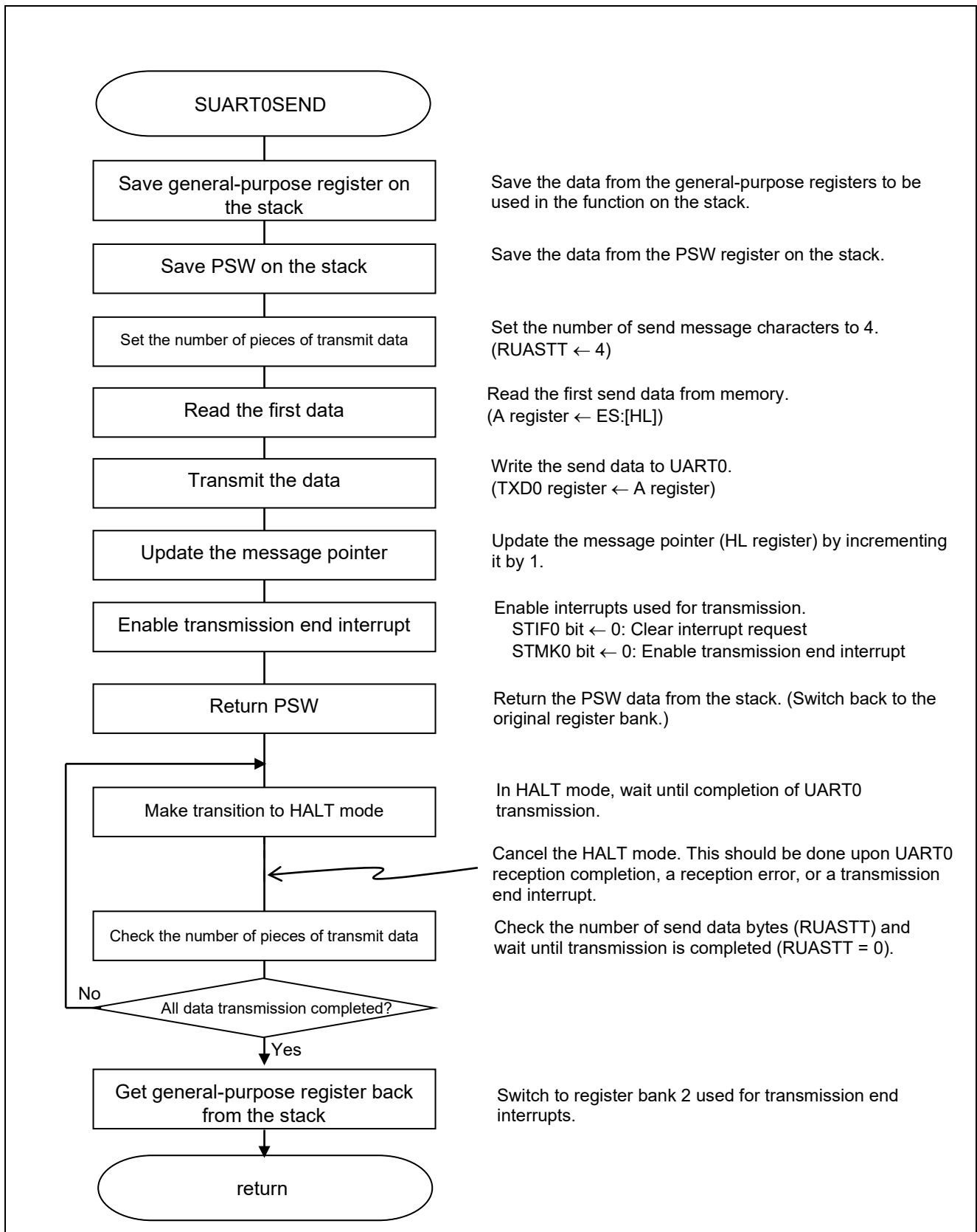

Figure 5.9   UART0 Data Transmission Function

The flowchart contents:

SUART0SEND

Save general-purpose register on the stack — Save the data from the general-purpose registers to be used in the function on the stack.

Save PSW on the stack — Save the data from the PSW register on the stack.

Set the number of pieces of transmit data — Set the number of send message characters to 4. (RUASTT ← 4)

Read the first data — Read the first send data from memory. (A register ← ES:[HL])

Transmit the data — Write the send data to UART0. (TXD0 register ← A register)

Update the message pointer — Update the message pointer (HL register) by incrementing it by 1.

Enable transmission end interrupt — Enable interrupts used for transmission.
  STIF0 bit ← 0: Clear interrupt request
  STMK0 bit ← 0: Enable transmission end interrupt

Return PSW — Return the PSW data from the stack. (Switch back to the original register bank.)

Make transition to HALT mode — In HALT mode, wait until completion of UART0 transmission.

Cancel the HALT mode. This should be done upon UART0 reception completion, a reception error, or a transmission end interrupt.

Check the number of pieces of transmit data — Check the number of send data bytes (RUASTT) and wait until transmission is completed (RUASTT = 0).

All data transmission completed? — No / Yes

Get general-purpose register back from the stack — Switch to register bank 2 used for transmission end interrupts.

return

Interrupt setting

- Interrupt request flag register (IF0L)
  Clear the interrupt request flag
- Interrupt mask flag register (MK0L)
  Cancel interrupt mask

Symbol: IF0L (for 10-pin products)

7	6	5	4	3	2	1	0
TMIF00	TMIF01H	SREIF0	SRIF0	STIF0 CSIIF00 IICIF00	PIF1	PIF0	WDTIIF
x	x	x	x	**0**	x	x	x

Bit 0

STIF0	Interrupt request flag
0	**No interrupt request signal is generated**
1	Interrupt request is generated, interrupt request status

Symbol: MK0L (for 10-pin products)

7	6	5	4	3	2	1	0
TMMK00	TMMK01H	SREMK0	SRMK0	STMK0 CSIMK00 IICMK00	PMK1	PMK0	WDTIMK
X	x	x	x	**0**	x	x	x

Bit 0

STMK0	Interrupt processing control
0	**Enables interrupt processing.**
1	Disables interrupt processing.

Caution: For details on the register setup procedures, refer to RL78/G10 User's Manual: Hardware.

5.7.9   Ring Buffer Write Processing

Figure 5.10 shows the flowchart for the ring buffer write processing.
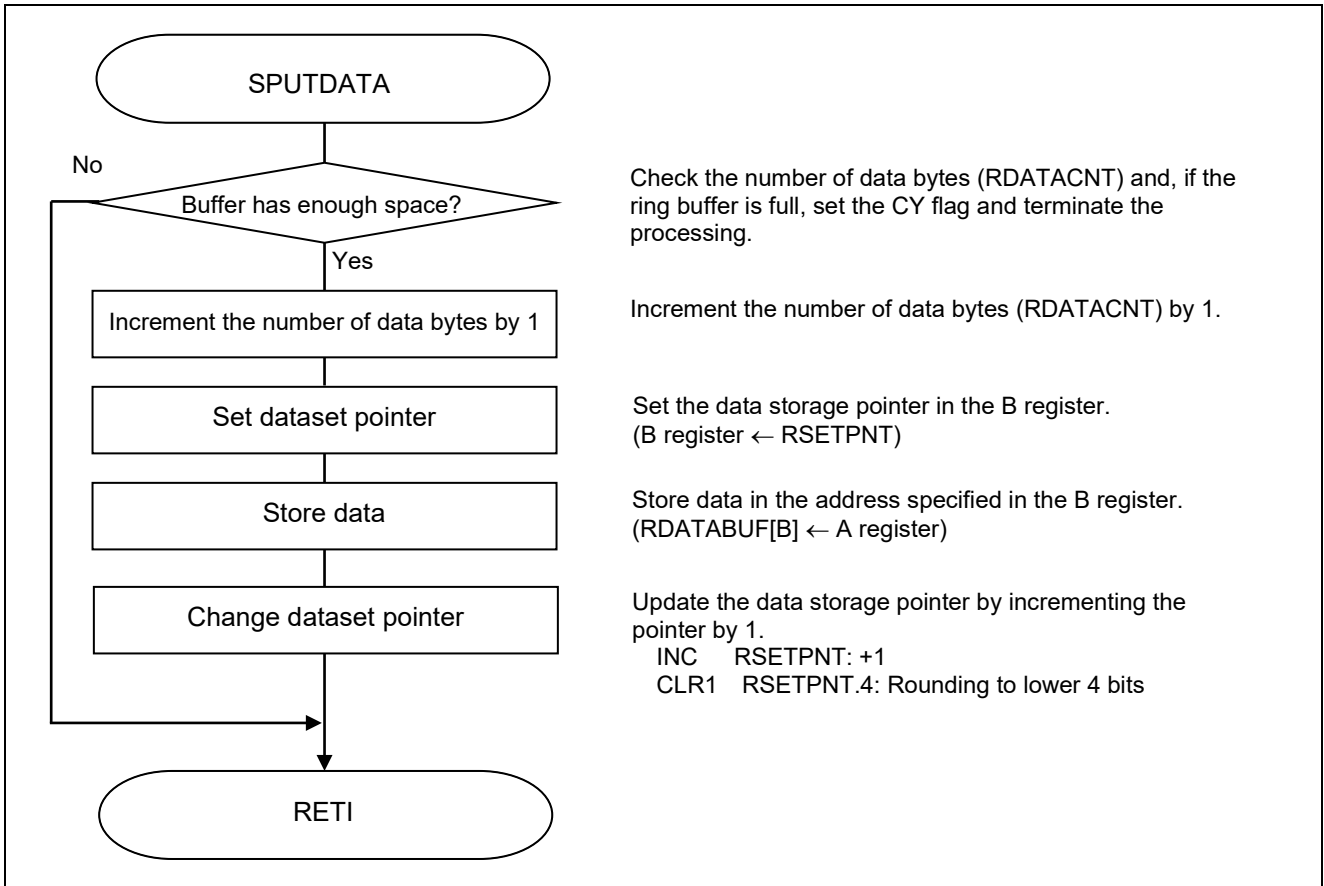
Figure 5.10   Ring Buffer Write Processing

5.7.10   Ring Buffer Read Processing

Figure 5.11 shows the flowchart for the ring buffer read processing.
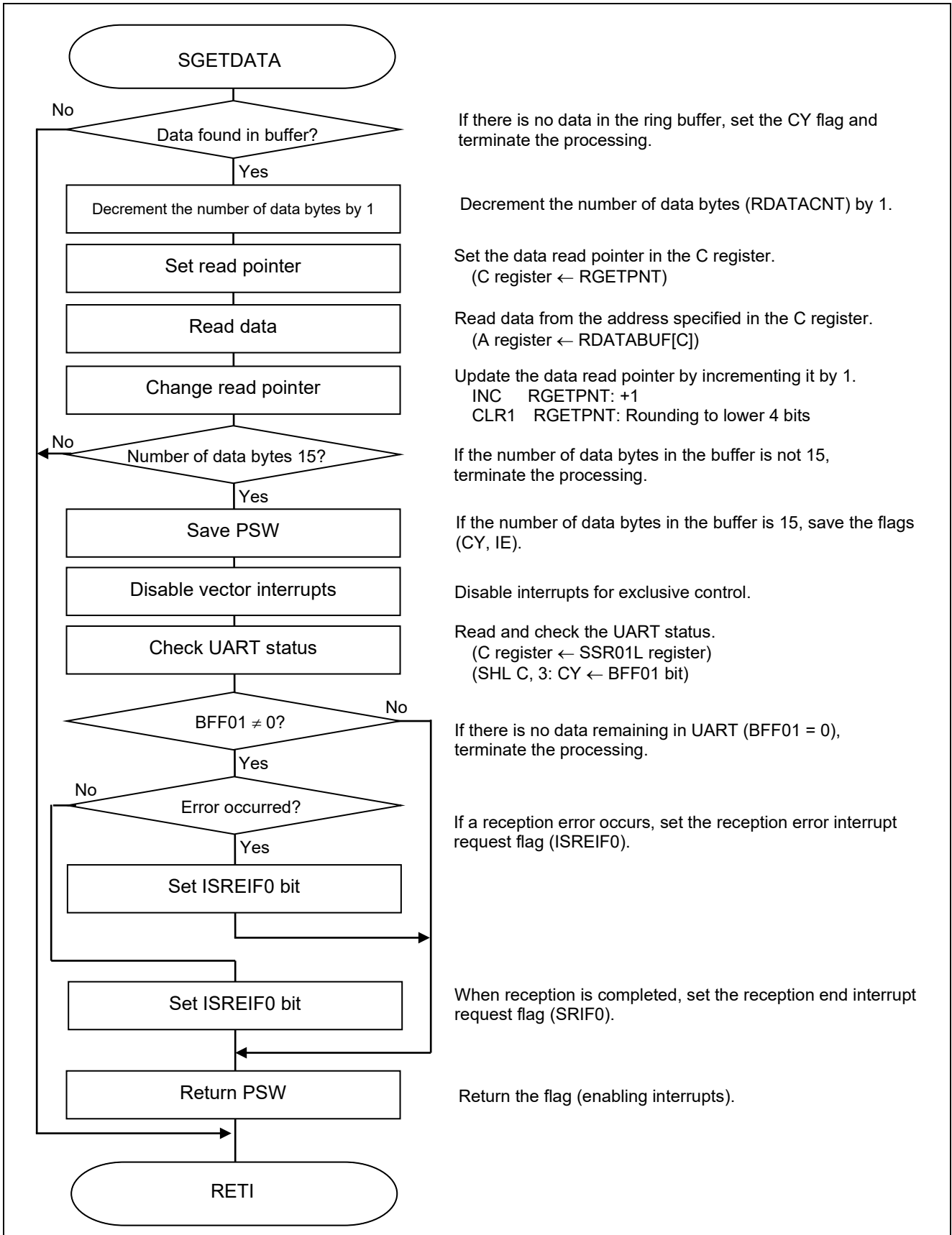
Figure 5.11   Ring Buffer Read Processing

Reception status check
- Serial status register 01 (SSR01)
  Read error status.

  Symbol: SSR01

7	6	5	4	3	2	1	0
0	TSF01	BFF01	0	0	FEF01	PEF01	OVF01
0	x	**0/1**	0	0	**0/1**	**0/1**	**0/1**

Bit 5

BFF01	Buffer register status indication flag of channel 01
0	**Valid data is not stored in the SDRmn register.**
1	**Valid data is stored in the SDRmn register.**

Bit 2

FEF01	Framing error detection flag of channel 01
0	**No error occurs.**
1	**An error occurs.**

Bit 1

PEF01	Parity error detection flag of channel 01
0	**No error occurs.**
1	**An error occurs.**

Bit 0

OVF01	Overrun error detection flag of channel 01
0	**No error occurs.**
1	**An error occurs.**

Interrupt setting
- Interrupt request flag register (IF0H)
  Setting interrupt request flag.

  Symbol: IF0L (For 10-pin products)

7	6	5	4	3	2	1	0
TMIF00	TMIF01	SREIF0	SRIF0	STIF0 CSIIF00 IICIF00	PIF1	PIF0	WDTIIF
x	x	X	x	**0**	x	x	0

Bits 2 and 1

SREIF0	SRIF0	Interrupt request flag
0	0	No interrupt request signal is generated.
1	**1**	**Interrupt request is generated, interrupt request status.**

Caution: For details on the register setup procedures, refer to RL78/G10 User's Manual: Hardware.

5.7.11   UART0 Reception End Interrupt Processing

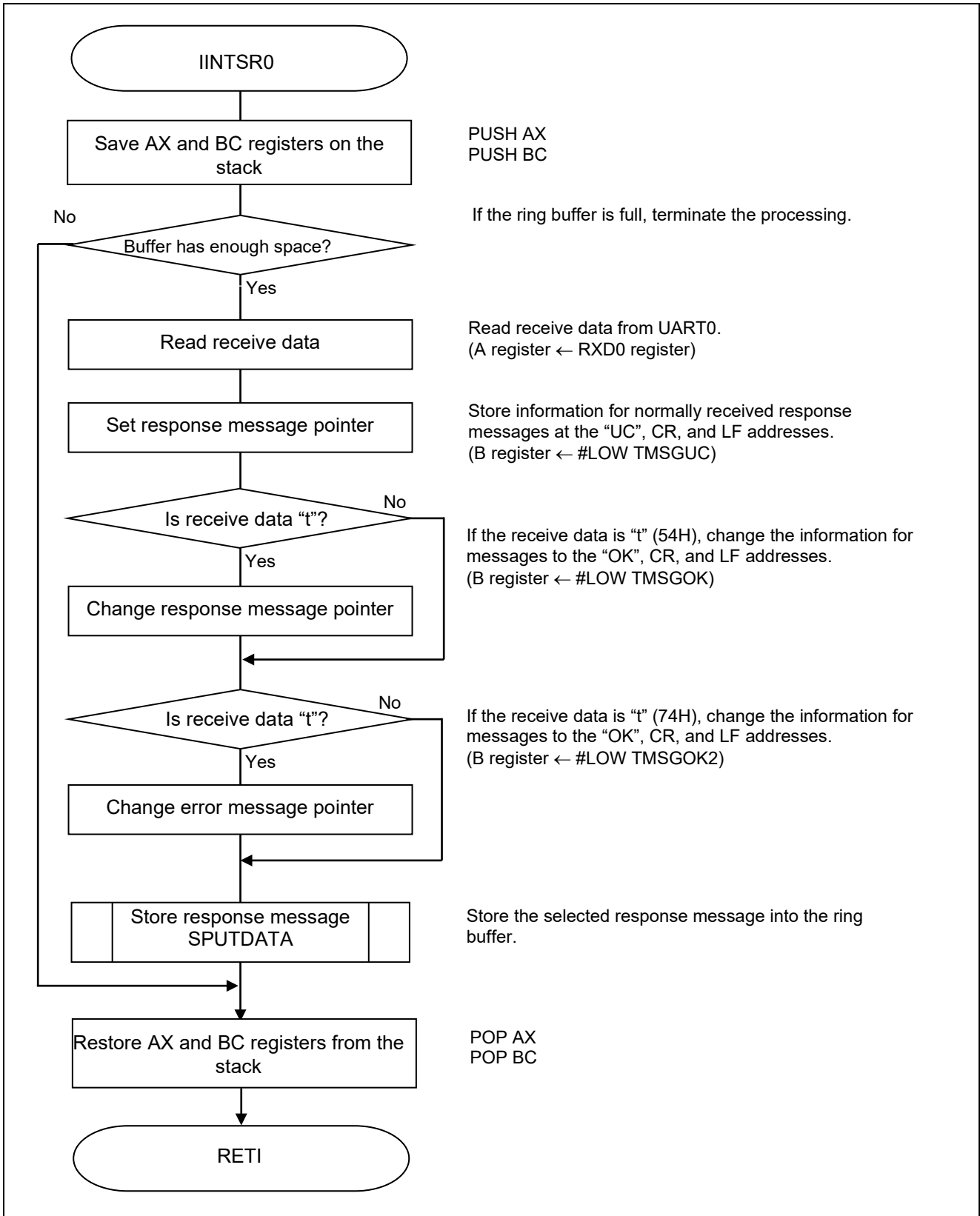Figure 5.12 shows the flowchart for the UART0 reception end interrupt processing.

Figure 5.12   UART0 Reception End Interrupt Processing

5.7.12 UART0 Reception Error Interrupt Processing

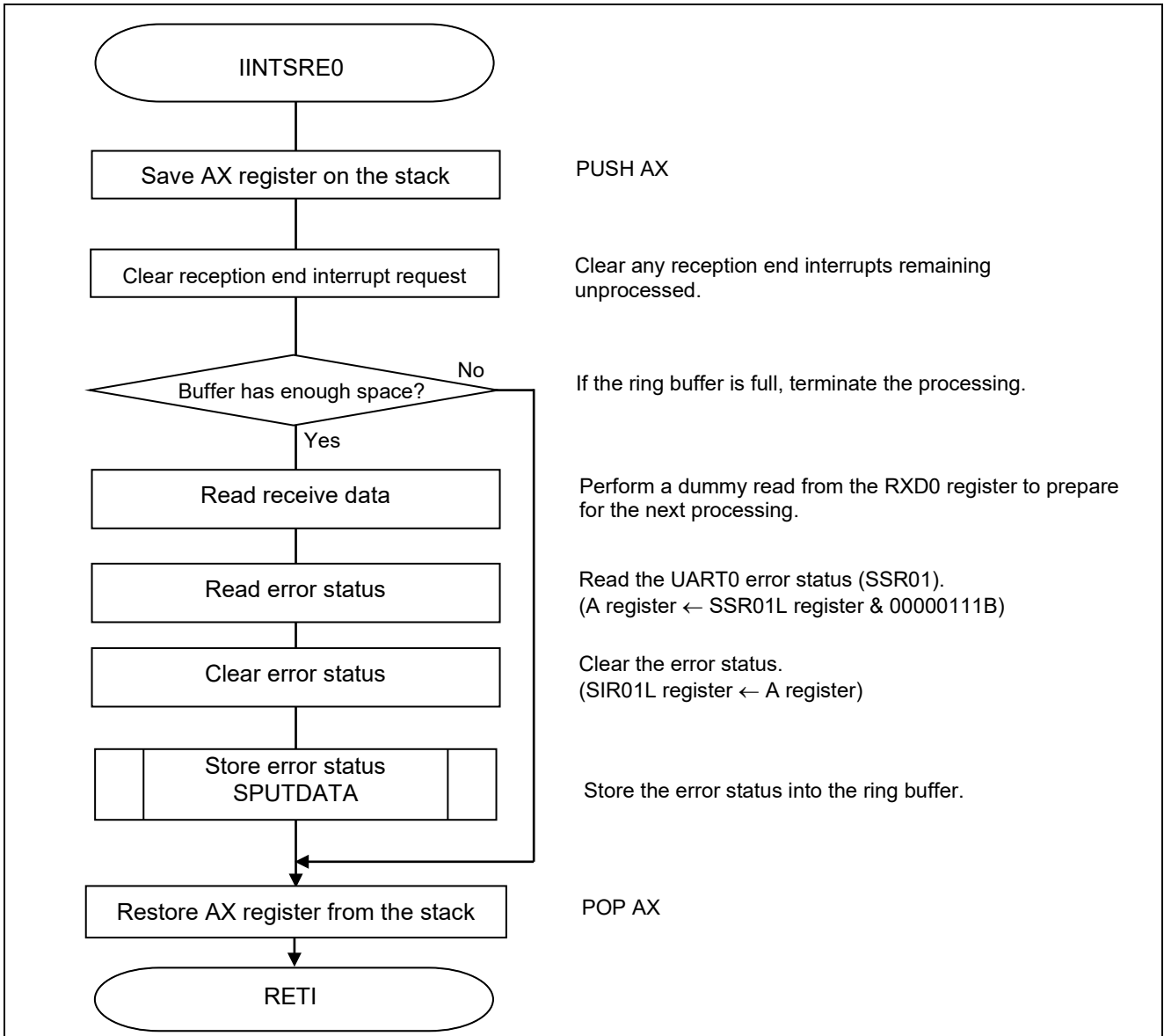Figure 5.13 shows the flowchart for the UART0 reception error interrupt processing.

```
                    ┌─────────────────────┐
                    │      IINTSRE0       │
                    └─────────────────────┘
                              │
                    ┌─────────────────────┐
                    │ Save AX register on │        PUSH AX
                    │     the stack       │
                    └─────────────────────┘
                              │
                    ┌─────────────────────┐        Clear any reception end interrupts remaining
                    │  Clear reception    │        unprocessed.
                    │ end interrupt request│
                    └─────────────────────┘
                              │
                                        No
                    ◇─────────────────────◇────    If the ring buffer is full, terminate the processing.
                     Buffer has enough space?
                    ◇─────────────────────◇
                              │ Yes
                    ┌─────────────────────┐        Perform a dummy read from the RXD0 register to prepare
                    │  Read receive data  │        for the next processing.
                    └─────────────────────┘
                              │
                    ┌─────────────────────┐        Read the UART0 error status (SSR01).
                    │  Read error status  │        (A register ← SSR01L register & 00000111B)
                    └─────────────────────┘
                              │
                    ┌─────────────────────┐        Clear the error status.
                    │  Clear error status │        (SIR01L register ← A register)
                    └─────────────────────┘
                              │
                    ┌──┬──────────────┬──┐
                    │  │ Store error  │  │          Store the error status into the ring buffer.
                    │  │   status     │  │
                    │  │  SPUTDATA    │  │
                    └──┴──────────────┴──┘
                              │
                    ┌─────────────────────┐        POP AX
                    │ Restore AX register │
                    │  from the stack     │
                    └─────────────────────┘
                              │
                    ┌─────────────────────┐
                    │        RETI         │
                    └─────────────────────┘
```

Figure 5.13   UART0 Reception Error Interrupt Processing

Reception error processing

- Serial status register 01 (SSR01)
  Read error status.
- Serial flag clear trigger register 01 (SIR01)
  Clear error status.

Symbol: SSR01

7	6	5	4	3	2	1	0
0	TSF 01	BFF 01	0	0	FEF 01	PEF 01	OVF 01
0	x	x	0	0	**0/1**	**0/1**	**0/1**

Bit 2

FEF01	Framing error detection flag of channel 01
0	**No error occurs.**
1	**An error occurs.**

Bit 1

PEF01	Parity error detection flag of channel 01
0	**No error occurs.**
1	**An error occurs.**

Bit 0

OVF01	Overrun error detection flag of channel 01
0	**No error occurs.**
1	**An error occurs.**

Symbol: SIR01

7	6	5	4	3	2	1	0
0	0	0	0	0	FECT 01	PEC T01	OVC T01
0	0	0	0	0	**0/1**	**0/1**	**0/1**

Bits 2 to 0

FECT01	PECT01	OVCT01	Clear trigger of error flag of channel 01
0	**0**	**0**	**Not cleared**
1	**1**	**1**	**Clears the error flag bit of the SSR01 register.**

Caution: For details on the register setup procedures, refer to RL78/G10 User's Manual: Hardware.

5.7.13   UART0 Transmission End Interrupt Processing

Figure 5.14 shows the flowchart for the UART0 transmission end interrupt processing.
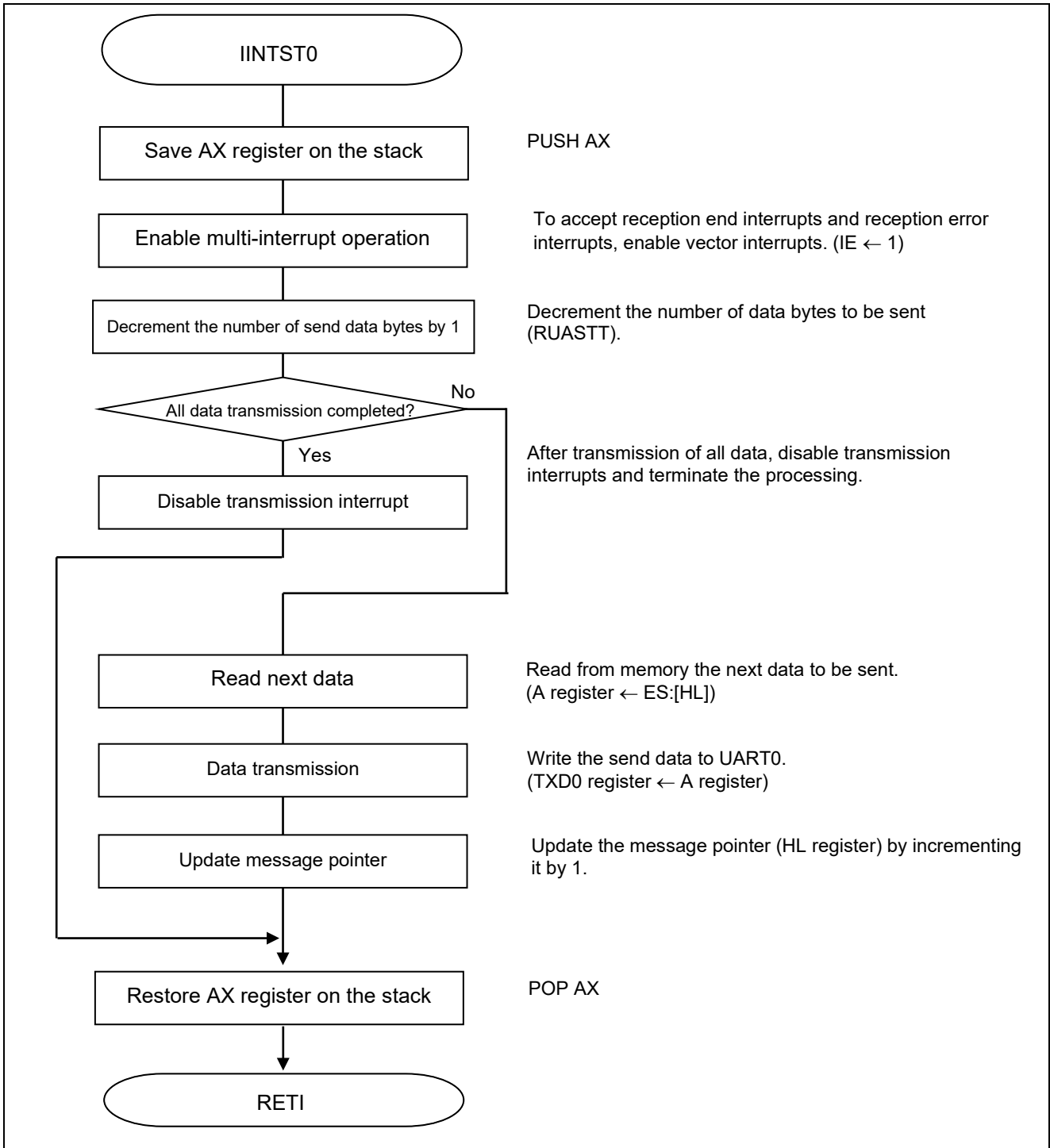
Figure 5.14   UART0 Transmission End Interrupt Processing

6.   Sample Code

The sample code is available on the Renesas Electronics Website.

7.   Documents for Reference

RL78/G10 User's Manual: Hardware (R01UH0384E)

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.)

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

Revision History

Rev.	Date	Description	
		Page	Summary
1.00	2013.01.24	-	First Edition
2.00	2013.12.31	5	e2studio and IAR information added in Table 2.1
2.10	2022.09.30	5	Delete IAR information from Table 2.1

General Precautions in the Handling of Microprocessing Unit and Microcontroller Unit Products

The following usage notes are applicable to all Microprocessing unit and Microcontroller unit products from Renesas. For detailed usage notes on the products covered by this document, refer to the relevant sections of the document as well as any technical updates that have been issued for the products.

1. Precaution against Electrostatic Discharge (ESD)

   A strong electrical field, when exposed to a CMOS device, can cause destruction of the gate oxide and ultimately degrade the device operation. Steps must be taken to stop the generation of static electricity as much as possible, and quickly dissipate it when it occurs. Environmental control must be adequate. When it is dry, a humidifier should be used. This is recommended to avoid using insulators that can easily build up static electricity. Semiconductor devices must be stored and transported in an anti-static container, static shielding bag or conductive material. All test and measurement tools including work benches and floors must be grounded. The operator must also be grounded using a wrist strap. Semiconductor devices must not be touched with bare hands. Similar precautions must be taken for printed circuit boards with mounted semiconductor devices.

2. Processing at power-on

   The state of the product is undefined at the time 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 time 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 time 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 time when power is supplied until the power reaches the level at which resetting is specified.

3. Input of signal during power-off state

   Do not input signals or an I/O pull-up power supply while the device is powered off. The current injection that results from input of such a signal or I/O pull-up power supply may cause malfunction and the abnormal current that passes in the device at this time may cause degradation of internal elements. Follow the guideline for input signal during power-off state as described in your product documentation.

4. 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 the 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.

5. Clock signals

   After applying a reset, only release the reset line after the operating clock signal becomes stable. When switching the clock signal during program execution, wait until the target clock signal is 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. Additionally, 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.

6. Voltage application waveform at input pin

   Waveform distortion due to input noise or a reflected wave may cause malfunction. If the input of the CMOS device stays in the area between V_{IL} (Max.) and V_{IH} (Min.) due to noise, for example, the device may malfunction. Take care to prevent chattering noise from entering the device when the input level is fixed, and also in the transition period when the input level passes through the area between V_{IL} (Max.) and V_{IH} (Min.).

7. Prohibition of access to reserved addresses

   Access to reserved addresses is prohibited. The reserved addresses are provided for possible future expansion of functions. Do not access these addresses as the correct operation of the LSI is not guaranteed.

8. Differences between products

   Before changing from one product to another, for example to a product with a different part number, confirm that the change will not lead to problems. The characteristics of a microprocessing unit or microcontroller unit products in the same group but having a different part number might differ in terms of 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 or any other use of the circuits, software, and information in the design of your product or system. Renesas Electronics disclaims any and all liability for any losses and damages incurred by you or third parties arising from the use of these circuits, software, or information.

2. Renesas Electronics hereby expressly disclaims any warranties against and liability for infringement or any other claims involving 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, including but not limited to, the product data, drawings, charts, programs, algorithms, and application examples.

3. 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 shall be responsible for determining what licenses are required from any third parties, and obtaining such licenses for the lawful import, export, manufacture, sales, utilization, distribution or other disposal of any products incorporating Renesas Electronics products, if required.

5. You shall not alter, modify, copy, or reverse engineer any Renesas Electronics product, whether in whole or in part. Renesas Electronics disclaims any and all liability for any losses or damages incurred by you or third parties arising from such alteration, modification, copying or reverse engineering.

6. Renesas Electronics products are classified according to the following two quality grades: "Standard" and "High Quality". The intended 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; industrial robots; etc.

    "High Quality": Transportation equipment (automobiles, trains, ships, etc.); traffic control (traffic lights); large-scale communication equipment; key financial terminal systems; safety control equipment; etc.

    Unless expressly designated as a high reliability product or a product for harsh environments in a Renesas Electronics data sheet or other Renesas Electronics document, Renesas Electronics products are not intended or 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 damage (space system; undersea repeaters; nuclear power control systems; aircraft control systems; key plant systems; military equipment; etc.). Renesas Electronics disclaims any and all liability for any damages or losses incurred by you or any third parties arising from the use of any Renesas Electronics product that is inconsistent with any Renesas Electronics data sheet, user's manual or other Renesas Electronics document.

7. No semiconductor product is absolutely secure. Notwithstanding any security measures or features that may be implemented in Renesas Electronics hardware or software products, Renesas Electronics shall have absolutely no liability arising out of any vulnerability or security breach, including but not limited to any unauthorized access to or use of a Renesas Electronics product or a system that uses a Renesas Electronics product. RENESAS ELECTRONICS DOES NOT WARRANT OR GUARANTEE THAT RENESAS ELECTRONICS PRODUCTS, OR ANY SYSTEMS CREATED USING RENESAS ELECTRONICS PRODUCTS WILL BE INVULNERABLE OR FREE FROM CORRUPTION, ATTACK, VIRUSES, INTERFERENCE, HACKING, DATA LOSS OR THEFT, OR OTHER SECURITY INTRUSION ("Vulnerability Issues"). RENESAS ELECTRONICS DISCLAIMS ANY AND ALL RESPONSIBILITY OR LIABILITY ARISING FROM OR RELATED TO ANY VULNERABILITY ISSUES. FURTHERMORE, TO THE EXTENT PERMITTED BY APPLICABLE LAW, RENESAS ELECTRONICS DISCLAIMS ANY AND ALL WARRANTIES, EXPRESS OR IMPLIED, WITH RESPECT TO THIS DOCUMENT AND ANY RELATED OR ACCOMPANYING SOFTWARE OR HARDWARE, INCLUDING BUT NOT LIMITED TO THE IMPLIED WARRANTIES OF MERCHANTABILITY, OR FITNESS FOR A PARTICULAR PURPOSE.

8. When using Renesas Electronics products, refer to the latest product information (data sheets, user's manuals, application notes, "General Notes for Handling and Using Semiconductor Devices" in the reliability handbook, etc.), and ensure that usage conditions are within the ranges specified by Renesas Electronics with respect to maximum ratings, operating power supply voltage range, heat dissipation characteristics, installation, etc. Renesas Electronics disclaims any and all liability for any malfunctions, failure or accident arising out of the use of Renesas Electronics products outside of such specified ranges.

9. Although Renesas Electronics endeavors to improve the quality and reliability of Renesas Electronics products, semiconductor products have specific characteristics, such as the occurrence of failure at a certain rate and malfunctions under certain use conditions. Unless designated as a high reliability product or a product for harsh environments in a Renesas Electronics data sheet or other Renesas Electronics document, Renesas Electronics products are not subject to radiation resistance design. You are responsible for implementing safety measures to guard against the possibility of bodily injury, injury or damage caused by fire, and/or danger to the public in the event of a failure or malfunction of Renesas Electronics products, 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 and impractical, you are responsible for evaluating the safety of the final products or systems 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. You are responsible for carefully and sufficiently investigating applicable laws and regulations that regulate the inclusion or use of controlled substances, including without limitation, the EU RoHS Directive, and using Renesas Electronics products in compliance with all these applicable laws and regulations. Renesas Electronics disclaims any and all liability for damages or losses occurring as a result of your noncompliance with applicable laws and regulations.

11. Renesas Electronics products and technologies shall 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 shall comply with any applicable export control laws and regulations promulgated and administered by the governments of any countries asserting jurisdiction over the parties or transactions.

12. It is the responsibility of the buyer or distributor of Renesas Electronics products, or any other party who distributes, disposes of, or otherwise sells or transfers the product to a third party, to notify such third party in advance of the contents and conditions set forth in this document.

13. This document shall not be reprinted, reproduced or duplicated in any form, in whole or in part, without prior written consent of Renesas Electronics.

14. Please contact a Renesas Electronics sales office if you have any questions regarding the information contained in this document or Renesas Electronics products.

(Note1) "Renesas Electronics" as used in this document means Renesas Electronics Corporation and also includes its directly or indirectly controlled subsidiaries.

(Note2) "Renesas Electronics product(s)" means any product developed or manufactured by or for Renesas Electronics.

(Rev.5.0-1   October 2020)

Corporate Headquarters

TOYOSU FORESIA, 3-2-24 Toyosu,
Koto-ku, Tokyo 135-0061, Japan

www.renesas.com

Trademarks

Renesas and the Renesas logo are trademarks of Renesas Electronics Corporation. All trademarks and registered trademarks are the property of their respective owners.

Contact information

For further information on a product, technology, the most up-to-date version of a document, or your nearest sales office, please visit:
www.renesas.com/contact/.