RL78/G15

Wireless Communication with the XBee ZB 2SC and SHT4x

Introduction

This application note describes a sample program to control the XBee ZB S2C* on the RL78/G15 to perform wireless communication. Humidity and temperature data acquired from the SHT4x (humidity and temperature sensor) is communicated wirelessly. This application note also describes how to control the SHT4x on the RL78/G15.

Additionally, we will explain the sample program for the module (hereinafter referred to as the receiving wireless module) that controls XBee ZB S2C, receives humidity and temperature data transmitted from the transmission wireless module, and displays it on the LCD module. This application note also describes the method of controlling the LCD module with RL78/G15.

* Digi, Digi International, the Digi logo, XBee, XBee ZB S2C are trademarks or registered trademarks of Digi International Inc. in the United States and other countries worldwide.

Target Device

RL78/G15

Contents

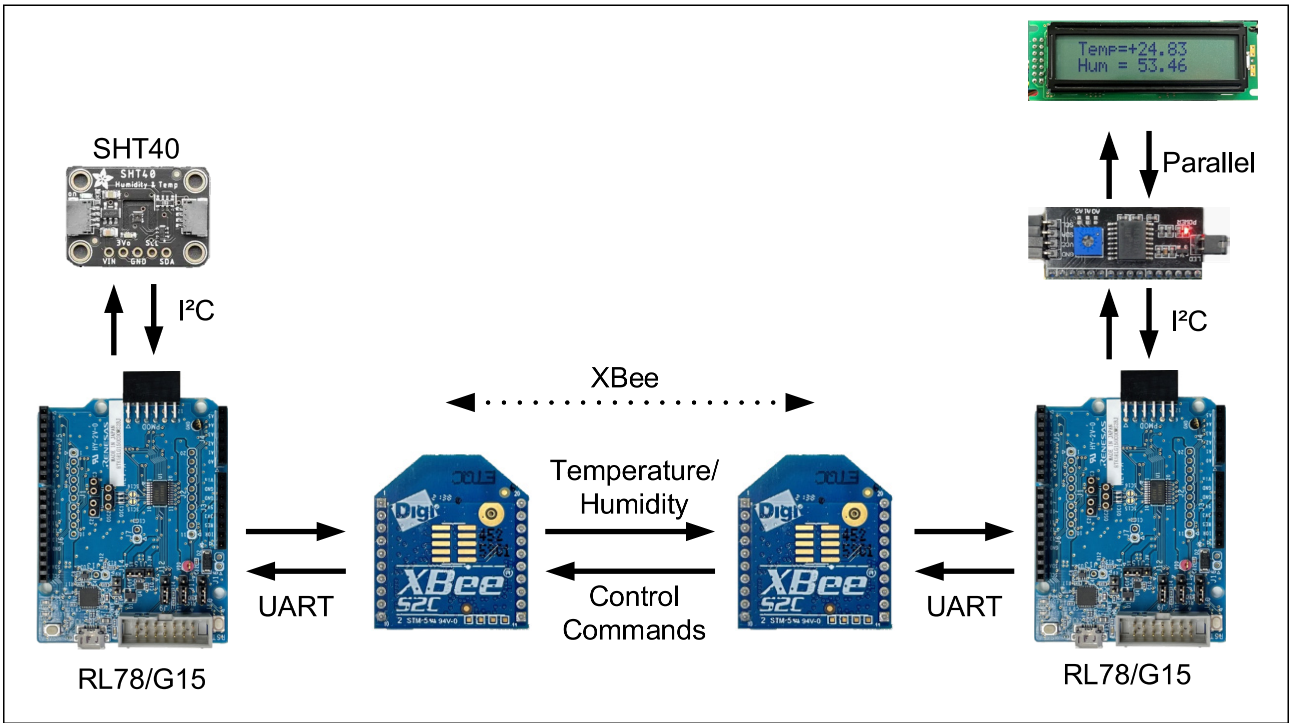
1. Overview

This application note describes how to connect the RL78/G15 with the XBee ZB S2C and wirelessly communicate humidity and temperature data acquired from the SHT40. The sample program uses the XBee ZB S2C by the UART function incorporated into the RL78/G15 to wirelessly communicate humidity and temperature data.

In addition, this application note describes how to acquire and calculate humidity and temperature data from the SHT40 humidity and temperature sensor on the RL78/G15. The sample program controls the SHT40 by the built-in I2C driver of the RL78/G15 and acquires humidity and temperature data from the SHT40. Additionally, this sample program calculates acquired data on the RL78/G15.

Additionally, we will explain how to connect the RL78/G15 with the receiving wireless module and the LCD module. We will also demonstrate the process of displaying humidity and temperature data received via XBee ZB S2C on the LCD module.

Figure 1-1 Schematic Diagram of the Entire System



2. Operation Confirmation Conditions

The operation of the sample program provided with this application note has been tested under the following conditions.

Table 2-1 Operation Confirmation Conditions

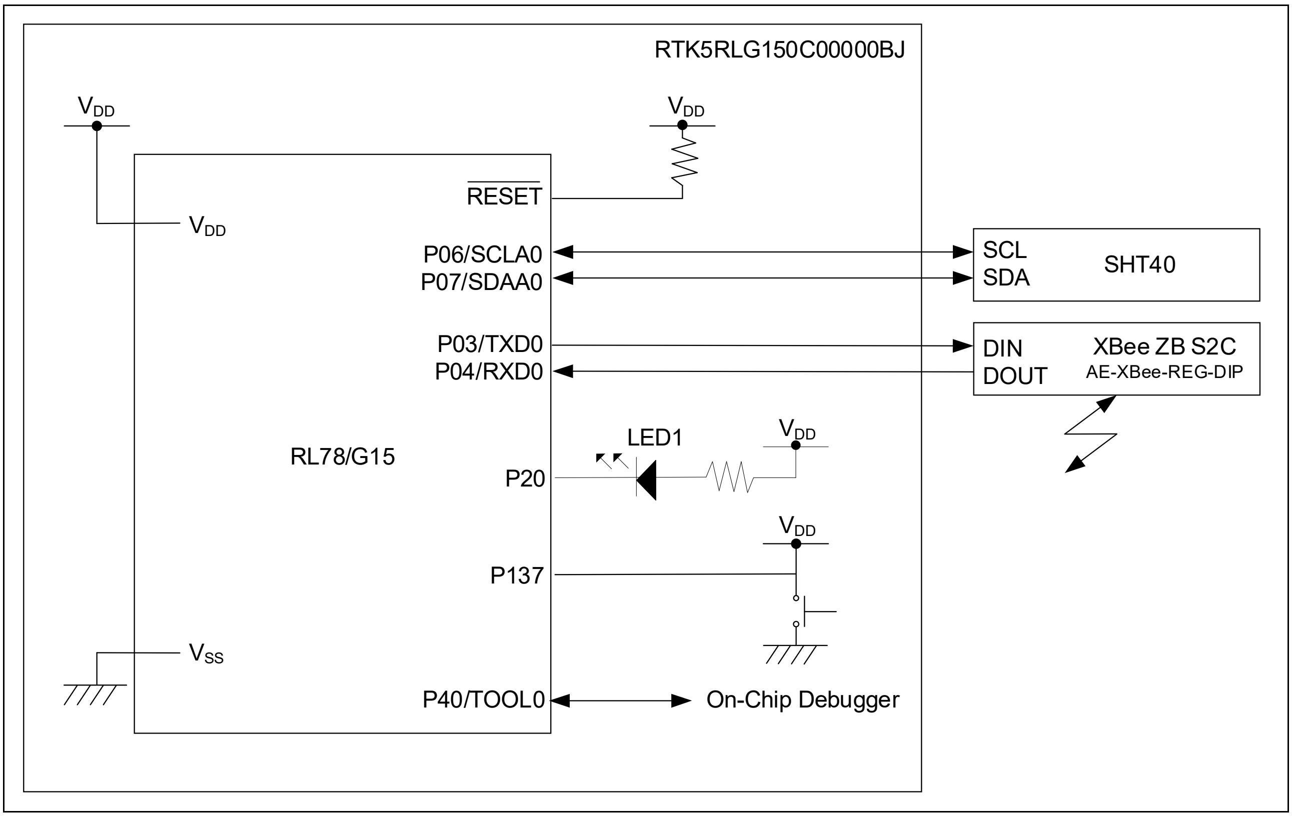
Item	Description
MCU used	RL78/G15 （R5F12068）
Operating frequency	16 MHz
Memory size (memory used)	・Transmitting Wireless Module ROM: 8 KB (5,138 bytes used by this APN) RAM 1KB (120 bytes used by this APN) ・Receiving Wireless Module ROM: 8 KB (2,434 bytes used by this APN) RAM 1KB (116 bytes used by this APN)
Operating voltage	3.3V SPOR operation : Reset mode At rising edge TYP. 2.57 V (2.44 V to 2.68 V) At falling edge TYP. 2.52 V (2.40 V to 2.62 V)
Integrated development environment (e2studio)	Renesas Electronics Corp. e2 studio V2025-10
C compiler (e2studio)	Renesas Electronics Corp. C Compiler Package for RL78 Family [CC-RL] V1.15.01
Integrated development environment (CS+)	Renesas Electronics Corp. CS+ for CC V8.14.0
C compiler (CS+)	Renesas Electronics Corp. C Compiler Package for RL78 Family [CC-RL] V1.15.01
Integrated development environment (IAR)	IAR Systems Corp. IAR Embedded Workbench for Renesas RL78 V5.20.2
C compiler (IAR)	IAR Systems Corp. IAR C/C++ Compiler for Renesas RL78 V 9.4.3.1558
Smart configurator (SC)	Renesas Electronics Corp. Renesas Smart Configurator for RL78 V1.15.0
Board support package (BSP)	Renesas Electronics Corp. BSP v1.91
Board used	RL78/G15 Fast Prototyping Board (RTK5RLG150C00000BJ)
Temperature/Humidity Sensor Modules	SHT40
Data transmission module	XBee ZB S2C
Serial to Parallel Conversion Module for LCD Module	PCF8574
LCD Module	LCD1602B

3. Hardware Descriptions

3.1 Example of Hardware Configuration

The following shows an example of the hardware configuration used in the application note.

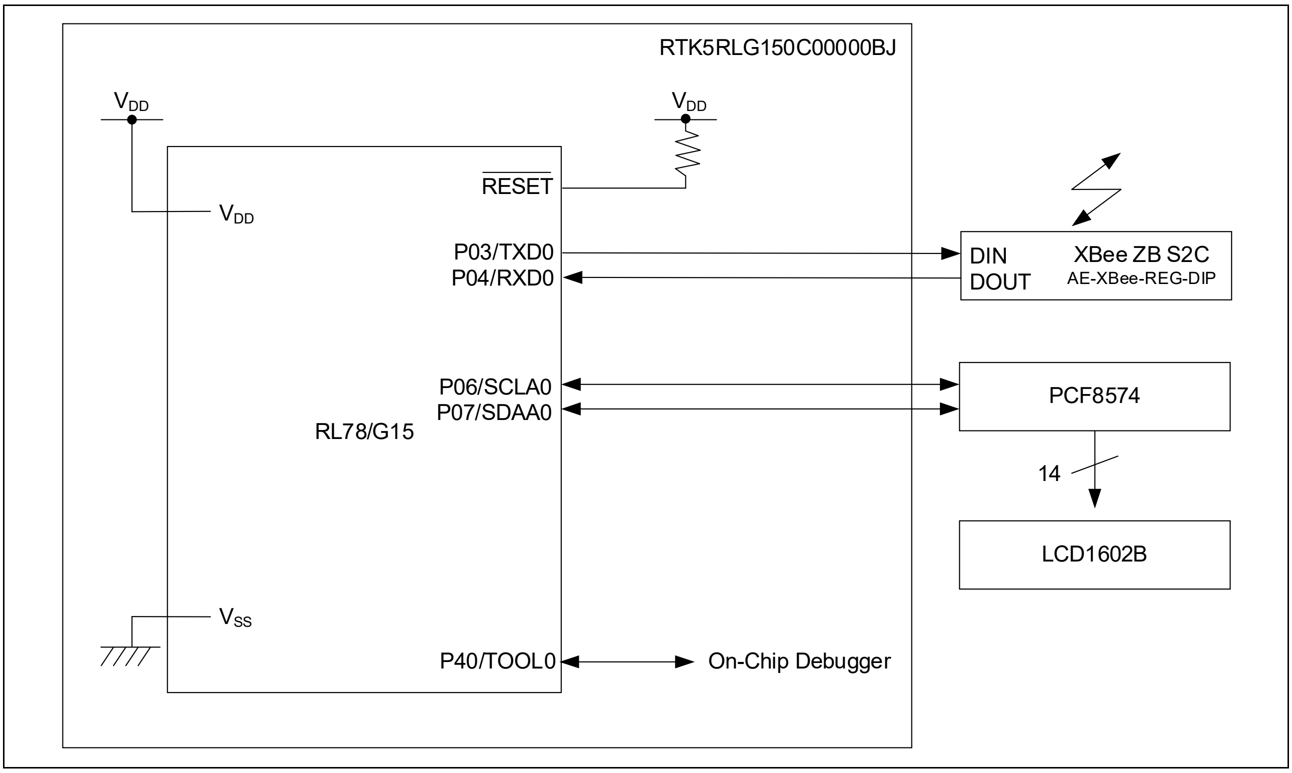
Figure 3-1 Hardware Configuration (Transmitting Wireless Module)



Note 1. This schematic circuit diagram is simplified to show the outline of connections. When creating circuits, design them so that they meet electrical characteristics by properly performing pin processing. (Connect input-only ports to V_{DD} or V_{SS} individually through a resistor.)

Note 2. V_{DD} must not be lower than the reset release voltage (V_{SPOR}) that is specified for the SPOR.

Figure 3-2 Hardware Configuration (Receiving Wireless Module)



Note 1. This schematic circuit diagram is simplified to show the outline of connections. When creating circuits, design them so that they meet electrical characteristics by properly performing pin processing. (Connect input-only ports to V_{DD} or V_{SS} individually through a resistor.)

Note 2. V_{DD} must not be lower than the reset release voltage (V_{SPOR}) that is specified for the SPOR.

3.2 List of Pins to be Used

The following lists the pins to be used and their functions.

Table 3-1 Pins to be Used and Their Functions (Transmitting Wireless Module)

Pin name	I/O	Function
P03/TXD0	Output	UART Data transmission pin (Via XBee ZB S2C)
P04/RXD0	Input	UART Data reception pin (From XBee ZB S2C)
P06/SCLA0	Input/Output	IICA serial clock (with SHT40)
P07/SDDA0	Input/Output	IICA serial data bus (with SHT40)
P20	Output	UART Communication Status Indicator (LED1: ON when connected / OFF when disconnected)
P137 / INTP0	Input	UART Communication Connect/Disconnect Switch (SW)

Table 3-2 Pins to be Used and Their Functions (Receiving Wireless Module)

Pin name	I/O	Function
P03/TXD0	Output	UART Data transmission pin (To XBee ZB S2C)
P04/RXD0	Input	UART Data reception pin (From XBee ZB S2C)
P06/SCLA0	Input/Output	IICA serial clock with the LCD module serial/parallel conversion board (SCL of PCF8574).
P07/SDDA0	Input/Output	IICA serial data bus with the LCD module serial/parallel conversion board (SDA of PCF8574).

4. Module Specifications

This section describes the specifications of the XBee ZB S2C and SHT40.

4.1 Specifications of XBee ZB S2C

Table 4-1 outlines the specifications of the XBee ZB S2C.

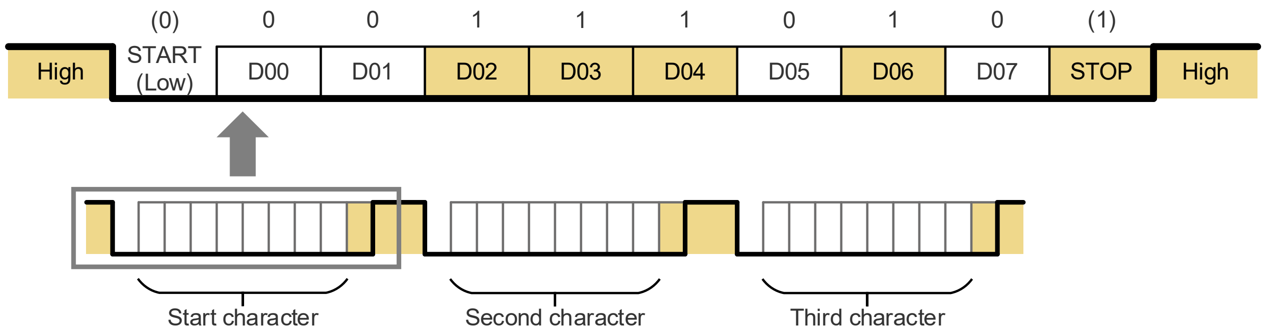
Table 4-1 Specifications of XBee ZB S2C

Item	Description
Data rate	RF 250 Kbps, serial (up to 1 Mbps)
Indoors/urban range	Up to 60 m
Outdoors/expected range	Up to 1,200 m
Transmission power	3.1 mW (+5 dBm) / 6.3 mW (+8 dBm) in boost mode
Receiver sensitivity (1% PER)	-100 dBm / -102 dBm in boost mode
Serial data interface	UART, SPI
Configuration method	API or AT command, local or wireless
Frequency band	ISM 2.4 GHz
Operating voltage	2.7 to 3.6 V
Auxiliary board for connecting MCU	AE-XBee-REG-DIP (2.54-mm pitch conversion board)
Auxiliary board for connecting PC	AE-XBEE-USB (USB interface board)

4.1.1 UART communication interface

The following figure shows the UART format for communicating humidity/temperature data.

Figure 4-1 UART Communication Format



If " : " (start character[0x3A]) is entered as the first character, the command is judged.
The second character is used to judge whether to acquire data (G) or control LED (L).
The third character determines temperature (T) or humidity (H) or ON (1) or OFF (0).

4.2 Specifications of SHT40

Table 4-2 outlines the specifications of the SHT40 humidity and temperature sensor.

Table 4-2 Specifications of SHT40 Sensor

Item	Description
Humidity measurement range	0 to 100%RH
Humidity accuracy	TYP. ±1.8 %RH (10 to 90%RH, 23 °C)
Temperature measurement range	-40 to 12 °C
Temperature accuracy	TYP. ±0.2 °C (0 to 65 °C)
Average current	High accuracy: 2.3 μA Medium accuracy: 1.2 μA Low accuracy: 0.4 μA Current consumption at 1 measurement per second (temperature and humidity)
Sleep current	0.08 μA
Operating voltage	1.08 V to 3.6 V (TYP. 3.3 V)
Operating temperature	-40 to 125°C

Table 4-3 List of Sensor Functions

Item	Description
I²C communication	Sensor data are transferred through I²C communication. The maximum communication speed is 1 MHz, and the I²C slave address is 0x44.
Measurement Accuracy Settings	High, Medium, and Low accuracy settings are available.
Measurement mode	This software operates with the sensor in sleep mode. On completion of measurement triggered by a measurement request, the sensor enters sleep mode.
Heater Settings	Three levels of heating power (High, Medium, Low), two heating durations (0.1 s and 1 s), and an option to disable the heater are available.
CRC Checksum	An 8-bit checksum is appended to verify the integrity of data transmitted via I²C communication.

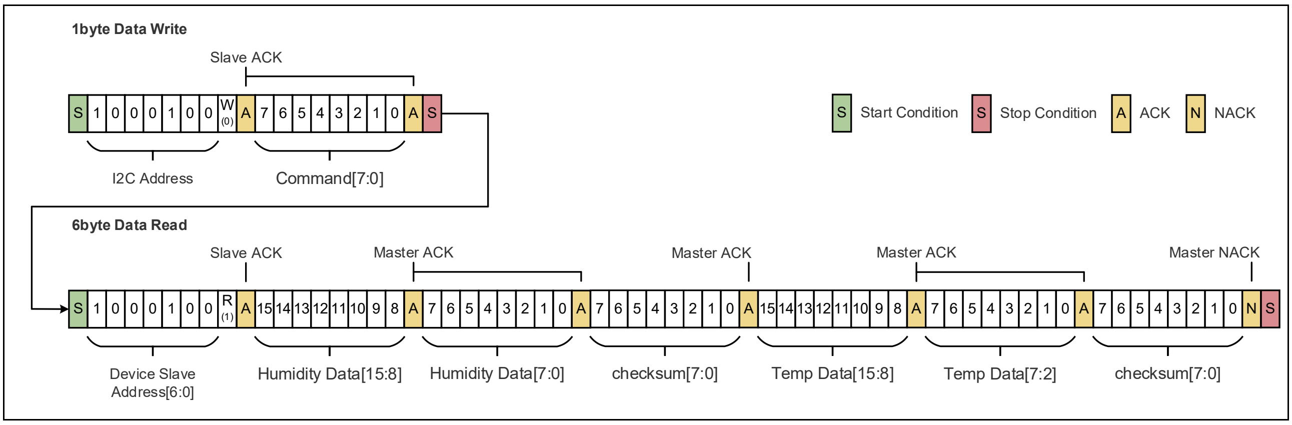
Note: In this sample program, the following SHT40 functions are not used

- Heater settings
- CRC checksum

4.2.1 I2C Communication Interface

The following figure shows the format of measurement data transferred through I[2]C communications.

Figure 4-2 Format of I2C Communication



4.2.2 Expressions for Converting Output Values to Humidity and Temperature

The SHT40 outputs humidity and temperature with a function that converts the acquired data into humidity and temperature.

The humidity conversion expression is as follows.

$$\text{Humidity [\%RH]} = \left(-6 + 125 * \frac{SHR}{2^{16} - 1}\right)$$

The temperature conversion expression is as follows.

$$\text{Temperature [°C]} = \left(-45 + 175 * \frac{ST}{2^{16} - 1}\right)$$

4.3 LCD Control Method

R+78/G15 transmits LCD display data to PCF8574 using the I2C interface. PCF8574 performs an 8- bit parallel conversion from the serial data received from RL78/G15 and outputs it to LCD1602B. In this application, LCD1602B is used in 4-bit mode, so the upper 4 bits are used as commands or data for LCD1602B, and the lower 4 bits are used as data for the mode setting of LCD1602B.

In addition, the I2C slave address of PCF8574 in this application is 0x27.

The data format for the commands sent to PCF8574 is shown below.

Figure 4-3 The data format for the commands

7	6	5	4	3	2	1	0
cmd/DATA				BL	EN	RW	RS

cmd/DATA	Commands or data to LCD1602B
BL	Back Light 1:ON 0:OFF
EN	Enable bit 1:ON 0:OFF
RW	Read/Write 1:Read 0:Write
RS	Register Select 1:Data Transmission 0:Command Transmission

Figure 4-4 Command Codes for LCD1602B Used in This Application

Command	Description
0x28	Function Set 4-bit Mode, LCD Display 2 Lines
0x08	LCD display OFF
0x01	LCD display data clear
0x06	Entry Mode Set
0x0C	LCD display ON

Figure 4-5 LCD Display Example

XBee ZB S2C Address Display

			4	1	F	D	5	1	7	8					

Temperature and Humidity Display

	T	e	m	p	=	+/-	x	x	.	x	x				
	H	u	m		=	x	x	x	.	x	x				

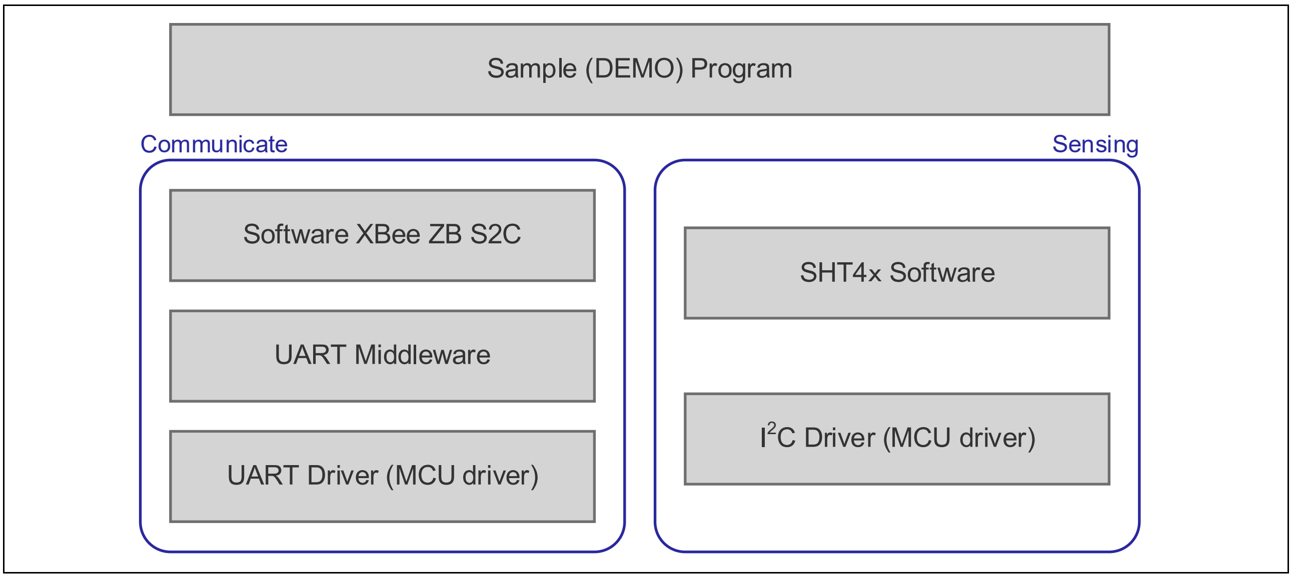
5. Sample Program

5.1 Transmitting Wireless Module

5.1.1 Sample Program Structure

Figure 5-1 is a block diagram of the sample program structure.

Figure 5-1 Block Diagram of Software (Transmitting Wireless Module)



5.1.2 Libraries Used in the Sample Program

This application note partially utilizes libraries provided on GitHub.

These libraries are distributed under the MIT License.

Copyright (c) 2023, Cedric Akilimali

5.1.2.1 List of SHT4x API Functions

The following is the API function of the SHT4x library for the temperature and humidity sensor used in this sample code.

Table 5-1 List of SHT4x API functions

API	Function
sht40x_get_temp_rh	Sensor control, measurement data acquisition and processing

5.1.3 List of Functions

Table 5-2 lists the functions used in the sample code. However, it excludes functions generated by the Smart Configurator that have not been modified.

Table 5-2 List of Functions

Function	Description	Source file
main	Main processing	main.c
key_proc	Temperature and humidity measurement control	main.c
xbee_connect	Communication Connection and Disconnection Processing with XBee ZB S2C	xbee_atcom.c
cmd_send_rcv	AT Command Transmission and Response Reception Processing	xbee_atcom.c
sht40x_get_temp_rh	Temperature and humidity data acquisition	sht40x_driver.c
command_send_temp_and_humi	Temperature and humidity data transmission	command.c
command_send_temperature	Temperature data transmission	command.c
command_send_humidity	Humidity data transmission	command.c

5.1.4 Function Specification

The function specifications of the sample code are shown below.

[Function name] main

Description	Main processing
Header	r_smc_entry.h, sht40x_driver.h, command.h, xbee_atcom.h
Declaration	void main (void);
Details	Perform initial setup of each module and start measuring temperature and humidity. When connected to the XBee ZB S2C, LED1 turns on.
Argument	None
Return value	None
Note	None

[Function name] key_proc

Description	Temperature and humidity measurement control
Header	r_smc_entry.h, sht40x_driver.h, command.h, xbee_atcom.h
Declaration	static void key_proc(void);
Details	Monitor the status of the switch, press and hold to toggle the communication connection/disconnect with the XBee ZB S2C, and send temperature and humidity data to the receiving radio module when the communication is connected. When the communication with the XBee ZB S2C is disconnected, LED1 turns off.
Argument	None
Return value	None
Note	None

[Function name] xbee_connect

Description	Communication Connection and Disconnection Processing with XBee ZB S2C
Header	r_cg_macrodriver.h, Config_UART0.h, r_cg_userdefine.h, command.h, xbee_atcom.h, string.h
Declaration	void xbee_connect(uint8_t conn)
Details	If the argument is 1, it connects to the XBee communication, and if the argument is 0, it reverts to the factory state and disconnects the XBee communication.
Argument	uint8_t conn Communication: 1, Disconnect: 0
Return value	None
Note	None

[Function name] cmd_send_rcv

Description	AT Command Transmission and Response Reception Processing
Header	r_cg_macrodriver.h, Config_UART0.h, r_cg_userdefine.h, command.h, xbee_atcom.h, string.h
Declaration	static void cmd_send_rcv(uint8_t *buf, uint16_t length)
Details	Send an AT command of the arguments to the XBee ZB S2C and receive a response from the XBee ZB S2C.
Argument	uint8_t *buf The address of the buffer that stores the data to be sent uint16_t length Bytes of data to send
Return value	None
Note	None

[Function name] sht40x_get_temp_rh

Description	Temperature and humidity data acquisition
Header	r_smc_entry.h, sht40x_driver.h, command.h, xbee_atcom.h
Declaration	void sht40x_get_temp_rh(uint8_t adr, uint8_t command, sht40x_data_t *pData);
Details	Communicate with temperature and humidity sensors to obtain temperature and humidity data.
Argument	uint8_t adr I^2C slave address uint8_t command Command to send to the SHT40 sensor sht40x_data_t *pData Addresses of structures for storing temperature and humidity data
Return value	None
Note	None

[Function name] command_send_temp_and_humi

Description	Temperature and humidity data transmission
Header	r_cg_macrodriver.h, r_cg_userdefine.h, Config_UART0.h, command.h, sht40x_driver.h
Declaration	void command_send_temp_and_humi (sht40x_data_t *pData);
Details	Transmit temperature and humidity data.
Argument	sht40x_data_t *pData Address of structures for storing temperature and humidity data
Return value	None
Note	None

[Function name] command_send_temperature

Description	Temperature data transmission
Header	r_cg_macrodriver.h, r_cg_userdefine.h, Config_UART0.h, command.h, sht40x_driver.h
Declaration	void command_send_temperature (sht40x_data_t *pData);
Details	The temperature data obtained from the temperature and humidity sensor is corrected and transmitted to the receiving wireless module.
Argument	sht40x_data_t *pData Address of structures for storing temperature and humidity data
Return value	None
Note	None

[Function name] command_send_humidity

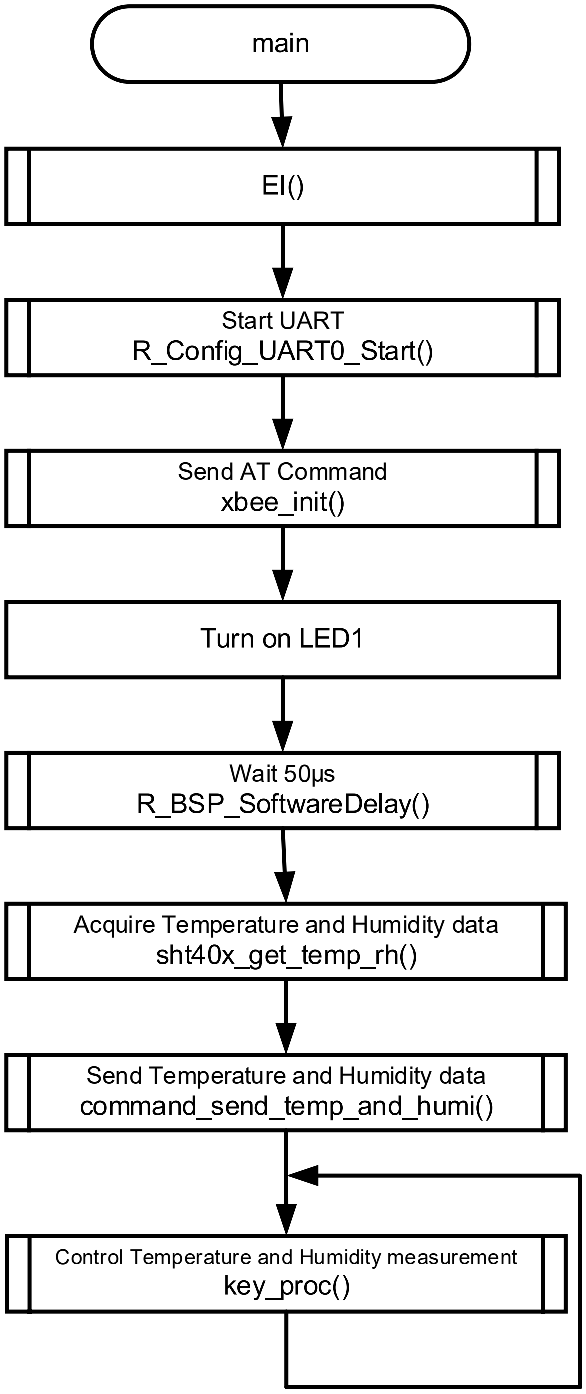
Description	Humidity data transmission
Header	r_cg_macrodriver.h, r_cg_userdefine.h, Config_UART0.h, command.h, sht40x_driver.h
Declaration	void command_send_humidity (sht40x_data_t *pData);
Details	The humidity data obtained from the temperature and humidity sensor is corrected and transmitted to the receiving wireless module.
Argument	sht40x_data_t *pData Address of structures for storing temperature and humidity data
Return value	None
Note	None

5.1.5 Flowchart

5.1.5.1 Main Processing

The flowchart for the main processing is shown below.

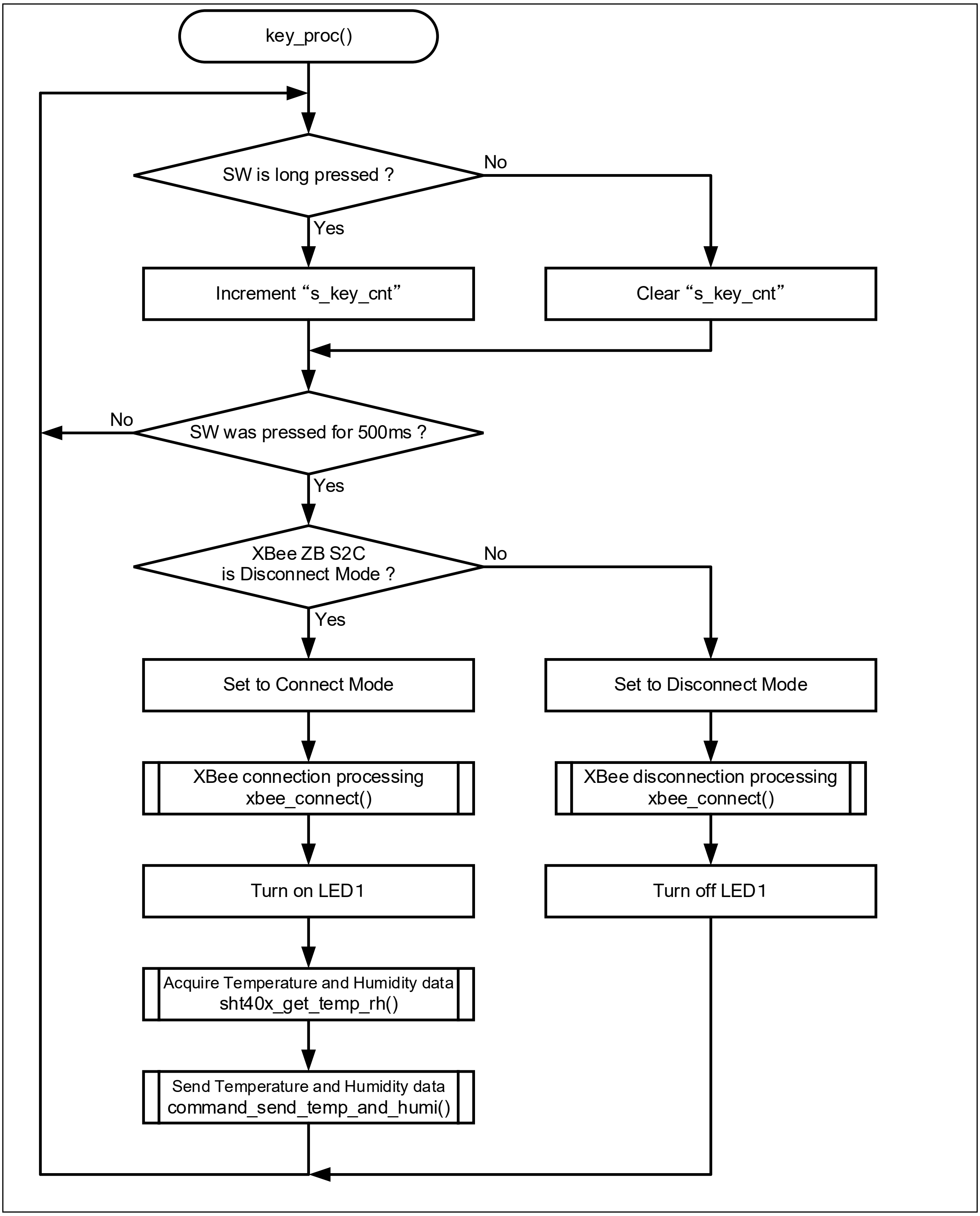
Figure 5-2 Main Processing



5.1.5.2 Temperature and humidity measurement control : key_proc

The flowchart for the function key_proc is shown below.

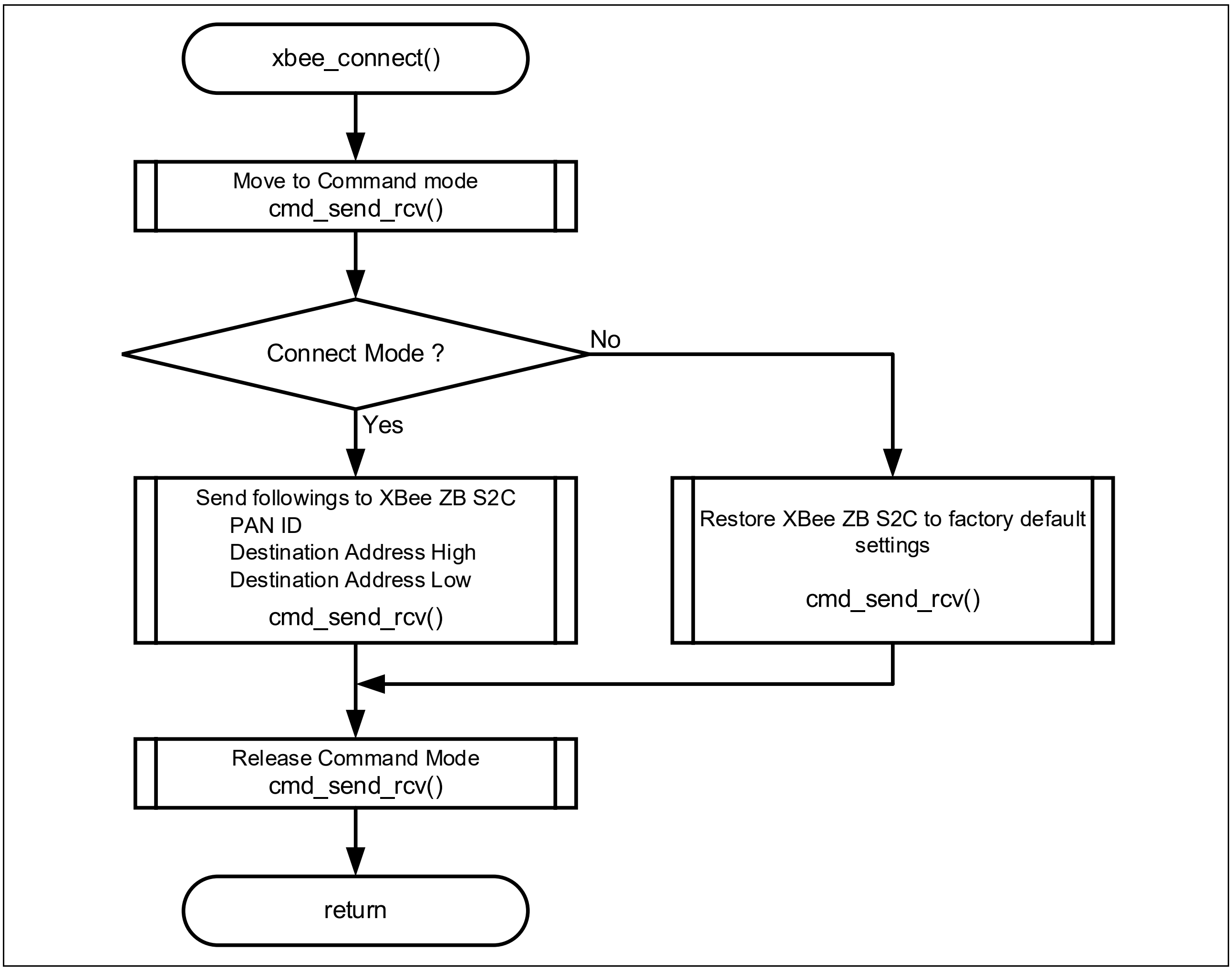
Figure 5-3 key_proc



5.1.5.3 Communication Connection and Disconnection Processing with XBee ZB S2C : xbee_connect

The flowchart for the function xbee_connect is shown below.

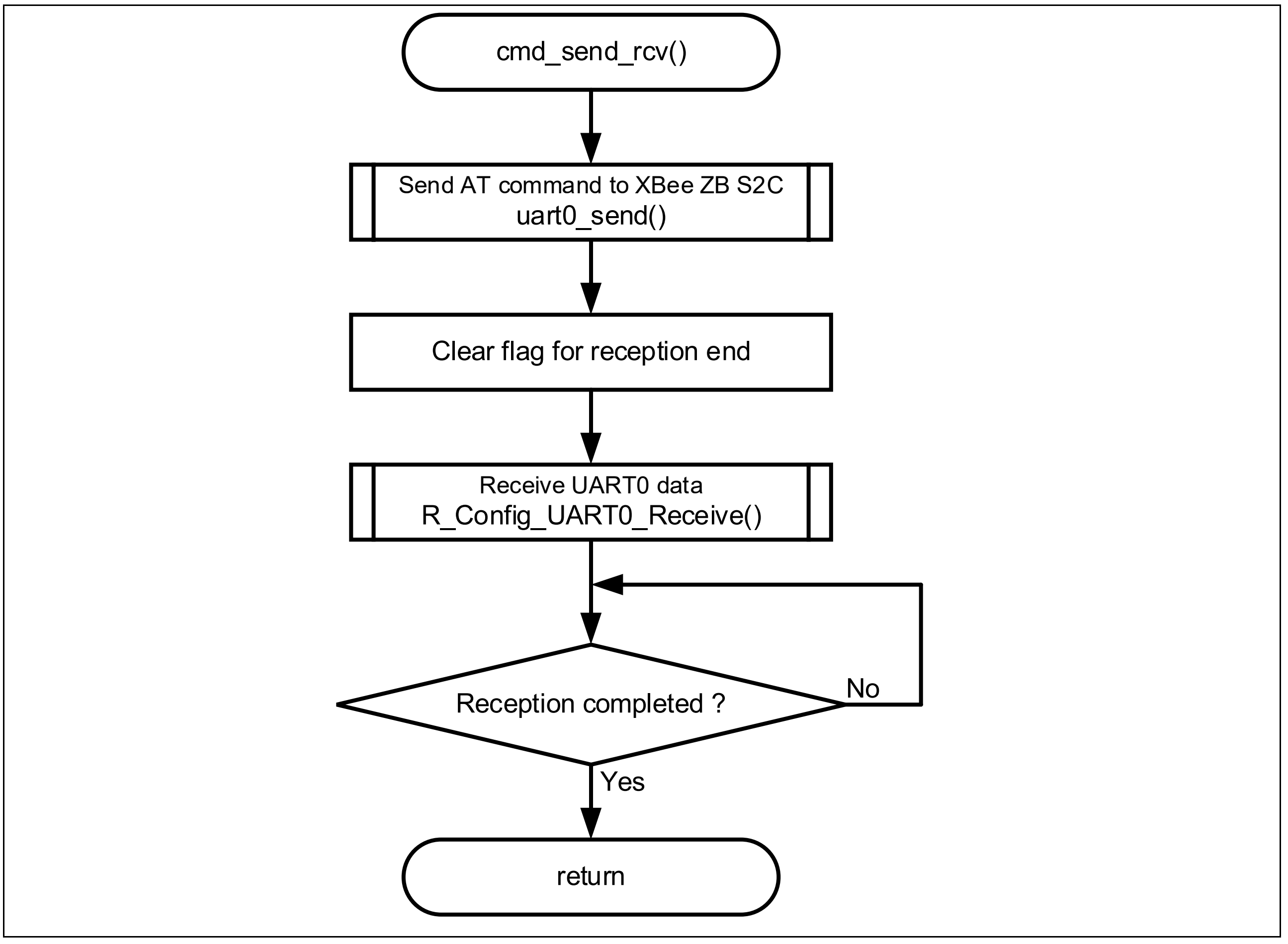
Figure 5-4 xbee_connect



5.1.5.4 AT Command Transmission and Response Reception Processing : cmd_send_rcv

The flowchart for the function cmd_send_rcv is shown below.

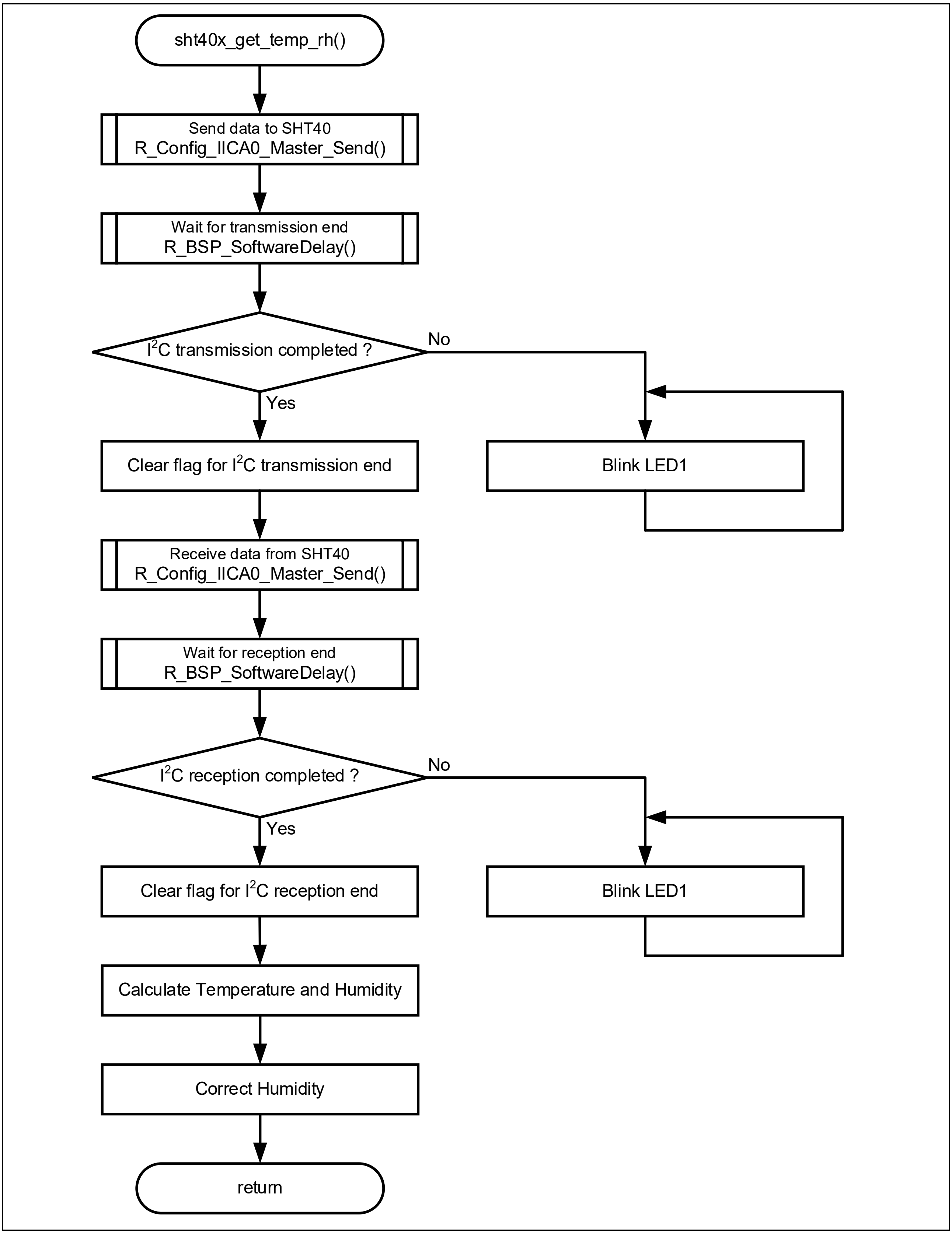
Figure 5-5 cmd_send_rcv



5.1.5.5 Temperature and humidity data acquisition : sht40x_get_temp_rh

The flowchart for the function sht40x_get_temp_rh is shown below.

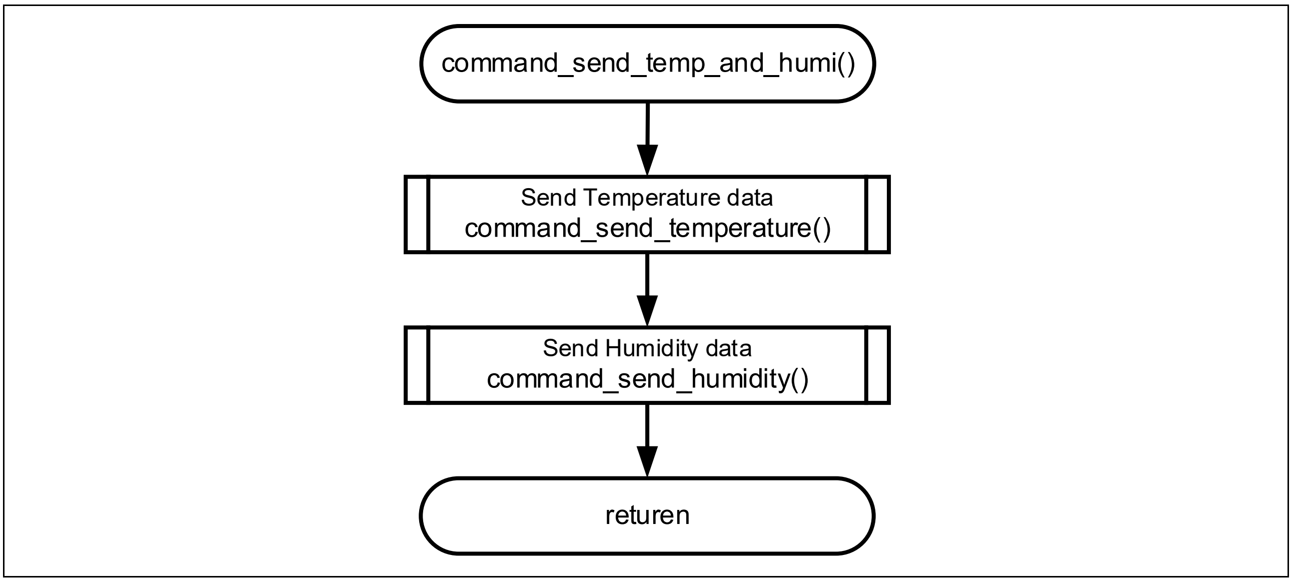
Figure 5-6 sht40x_get_temp_rh



5.1.5.6 Temperature and humidity data transmission : command_send_temp_and_humi

The flowchart for the function command_send_temp_and_humi is shown below.

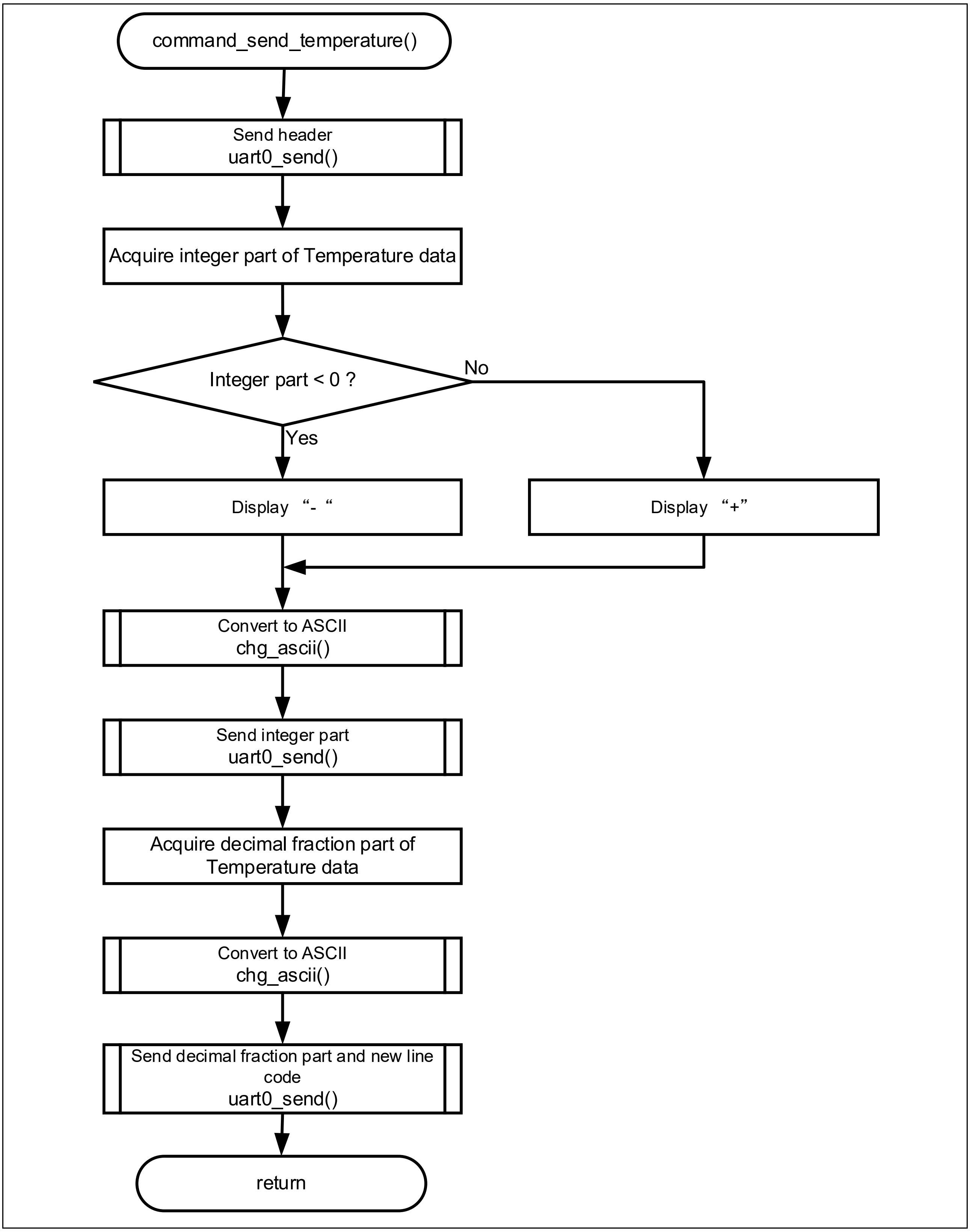
Figure 5-7 command_send_temp_and_humi



5.1.5.7 Temperature data transmission : command_send_temperature

The flowchart for the function command_send_temperature is shown below.

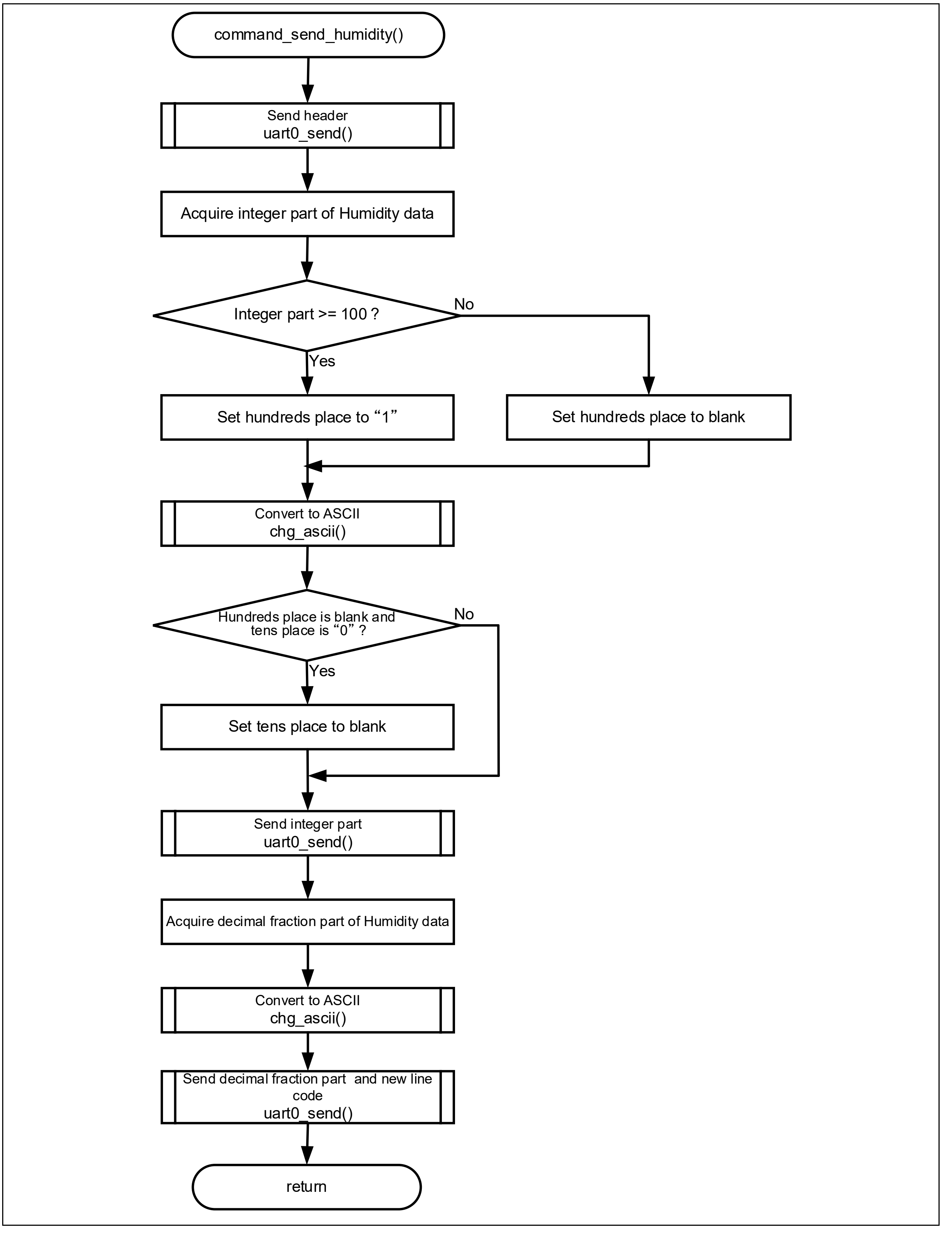
Figure 5-8 command_send_temperature



5.1.5.8 Humidity data transmission : command_send_humidity

The flowchart for the function command_send_humidity is shown below.

Figure 5-9 command_send_humidity



5.1.6 Sample Program Structure

5.1.6.1 Peripheral Functions to Be Used

The following table lists the peripheral functions used in the sample program.

Table 5-3 Peripheral Functions to Be Used (Transmitting Wireless Module)

Peripheral Function	Use
PORT	Used to control LED1 mounted on RL78/G15_FPB.
IICA0	Used for I^2C communication with SHT40 and acquisition of humidity and temperature data from SHT40.
UART0	Used for UART communication with XBee ZB S2C, command transmission to XBee ZB S2C, and confirmation of response results returned from XBee ZB S2C.

5.1.6.2 Settings of Peripheral Functions

The following table lists the settings of Smart Configurator used in the sample program. The items and settings in each table of Smart Configurator Settings are described with the names displayed on the actual setting screen.

Table 5-4 Smart Configurator Settings (1/2)

Category	Item	Settings
Smart Configurator >> Clock		The settings on the [Clock] tab are as follows:
	VDD setting	2.4 V ≦ VDD ≦ 5.5 V
	High-speed on-chip oscillator	Checked Frequency: 16 MHz
	X1 oscillator circuit	Unchecked
	Low-speed on-chip oscillator	15 kHz
	Source selection for main system clock (f_{MAIN})	Set the on-chip oscillator clock (fIH) to (fMAIN) 16 MHz and (fCLK) 16,000 kHz.
	Operating clock supply for 12-bit interval timer	Unchecked (f_{IL}: 15 kHz)
Smart Configurator >> System		The settings on the [System] tab are as follows:
	Pseudo RRM/DMM function setting	Not used
	Start/Stop function setting	Not used
	Security ID setting	Set security ID. Security ID "0x00000000000000000000"
	RESET pin setting	Not used
	Operation mode setting	2.52 V
Smart Configurator >> Component >> r_bsp		Use the default settings except for the following changes:
	API functions disable(R_BSP_StartClock, R_BSP_StopClock)	Disable
	API functions disable(R_BSP_SetClockSource)	Disable
	API functions disable(R_BSP_ChangeClockSetting)	Disable
	API functions disable(R_BSP_SoftwareDelay)	Enable
Smart Configurator >> Component >> PORT		Use the default settings except for the following changes:
	Port selection	PORT2
	PORT2	P20: Select [Output], [Output 1].
Smart Configurator >> Component >> Config_IICA0		The settings on [Config_IICA0] are as follows:
	Local address	16
	Operation mode setting	Set [Standard] and transfer clock (f_{SCL}): 100000 (bps)
	tR and tF settings	Unchecked
	Interrupt setting	Level 3 (lowest priority)
	Callback function setting	Select [Master transmission end], [Master reception end], and [Master error].
	Callback extension setting	Unchecked

Table 5-5 Smart Configurator Settings (2/2)

Category	Item		Settings
Smart Configurator >> Component >> Config_UART			The settings on [Config_UART0] are as follows:
	Transmission	UART0 clock setting	Operating clock: CK00 Clock source: f_{CLK}/2^3
		Transfer mode setting	Continuous transfer mode
		Data bit length setting	8 bits
		Data transfer direction setting	LSB
		Parity setting	No parity bit
		Stop bit length setting	1 bit
		Transmit data level setting	Non-inversion (normal)
		Transfer rate	9600 (bps)
		Interrupt setting	Level 3 (lowest priority)
		Callback function setting	End of transmission
	Reception	UART0 clock setting	Operating clock: CK00 Clock source: f_{CLK}/2^3
		Data bit length setting	8 bits
		Data transfer direction setting	LSB
		Parity setting	No parity bit
		Transmit data level setting	Non-inversion (normal)
		Transfer rate	9600 (bps)
		Interrupt setting	Level 3 (lowest priority)
		Callback function setting	Reception end, reception error
Smart Configurator >> Pin			The settings on the [Pin] tab are as follows:
	Serial array unit (SAU0)		SAU00 RxD0: P04 TxD0: P03
	Serial interface IICA (IICA0)		IICA0 SCLA0: P06 SDAA0: P07

5.1.6.3 File Structure

The following table lists the file structure of the sample program.

Table 5-6 File Structure (Transmitting Wireless Module)

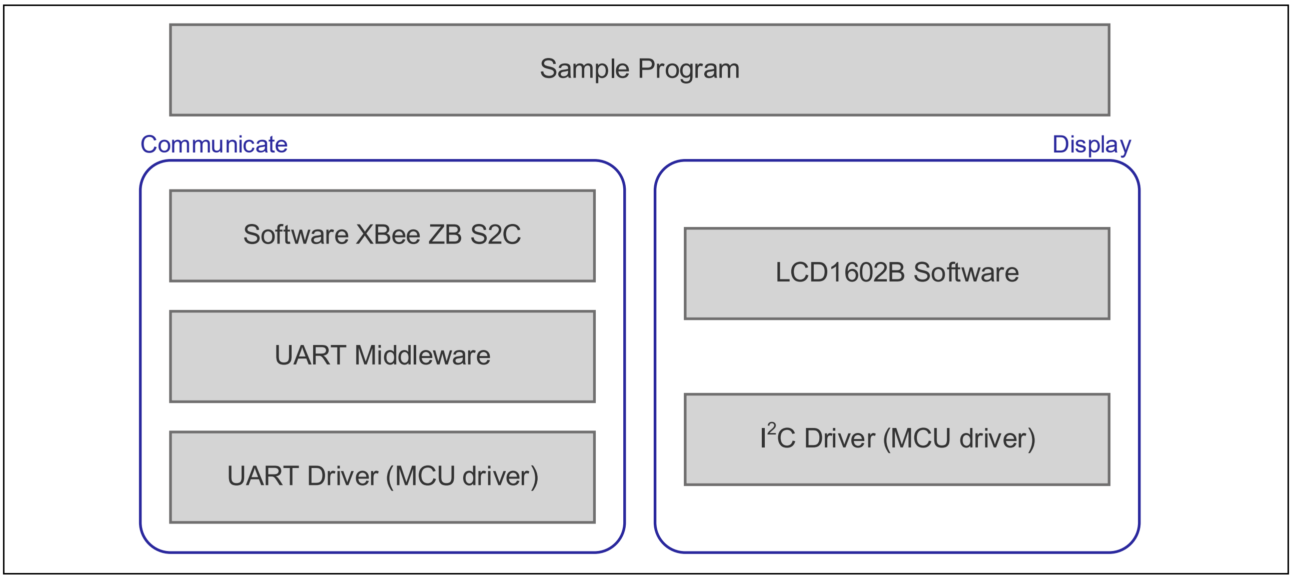
Folder name, File name	Explanation
src	Program storage folder
├ command	Command related program storage folder
│ ├ xbee_atcom.c	AT Command related source file
│ └ xbee_atcom.h	AT Command related header file
│ ├ command.c	Command related source file
│ └ command.h	Command related header file
├ smc_gen	Smart Configurator generation folder
│ ├ Config_IICA0	
│ ├ Config_PORT	
│ ├ Config_UART0	
│ ├ general	
│ ├ r_bsp	
│ └ r_config	
├ main.c	Main processing source file
├ sht40x_driver.c	Temperature and humidity sensor measurement source file
└ sht40x_driver.h	Temperature and humidity sensor measurement header file

5.2 Receiving Wireless Module

5.2.1 Sample program Structure

Table 5-10 is a block diagram of the sample program structure.

Figure 5-10 Block Diagram of Software (Receiving Wireless Module)



5.2.2 List of Functions

Table 5-7 lists the functions used in the sample code. However, it excludes functions generated by the Smart Configurator that have not been modified.

Table 5-7 List of Functions

Function	Description	Source file
main	Main processing	main.c
lcd_disp	LCD display update	lcd_1602b.c
rcv_wait	Temperature / humidity data reception	command.c

5.2.3 Function Specification

The function specifications of the sample code are shown below.

[Function name] main

Description	Main processing
Header	r_smc_entry.h, command.h, xbee_atcom.h, lcd_1602b.h
Declaration	void main (void);
Details	Perform initial setup of each module and receive data from the wireless transmission module and display it on the LCD.
Argument	None
Return value	None
Note	None

[Function name] lcd_disp

Description	LCD display update
Header	r_cg_macrodriver.h, Config_IICA0.h, lcd_1602b.h, command.h, xbee_atcom.h, string.h
Declaration	void lcd_disp (void);
Details	If the LCD is in its initial state, display the XBee ZB S2C address on the LCD; if it is in normal mode, display the temperature and humidity data received from the wireless transmission module on the LCD.
Argument	None
Return value	None
Note	None

[Function name] rcv_wait

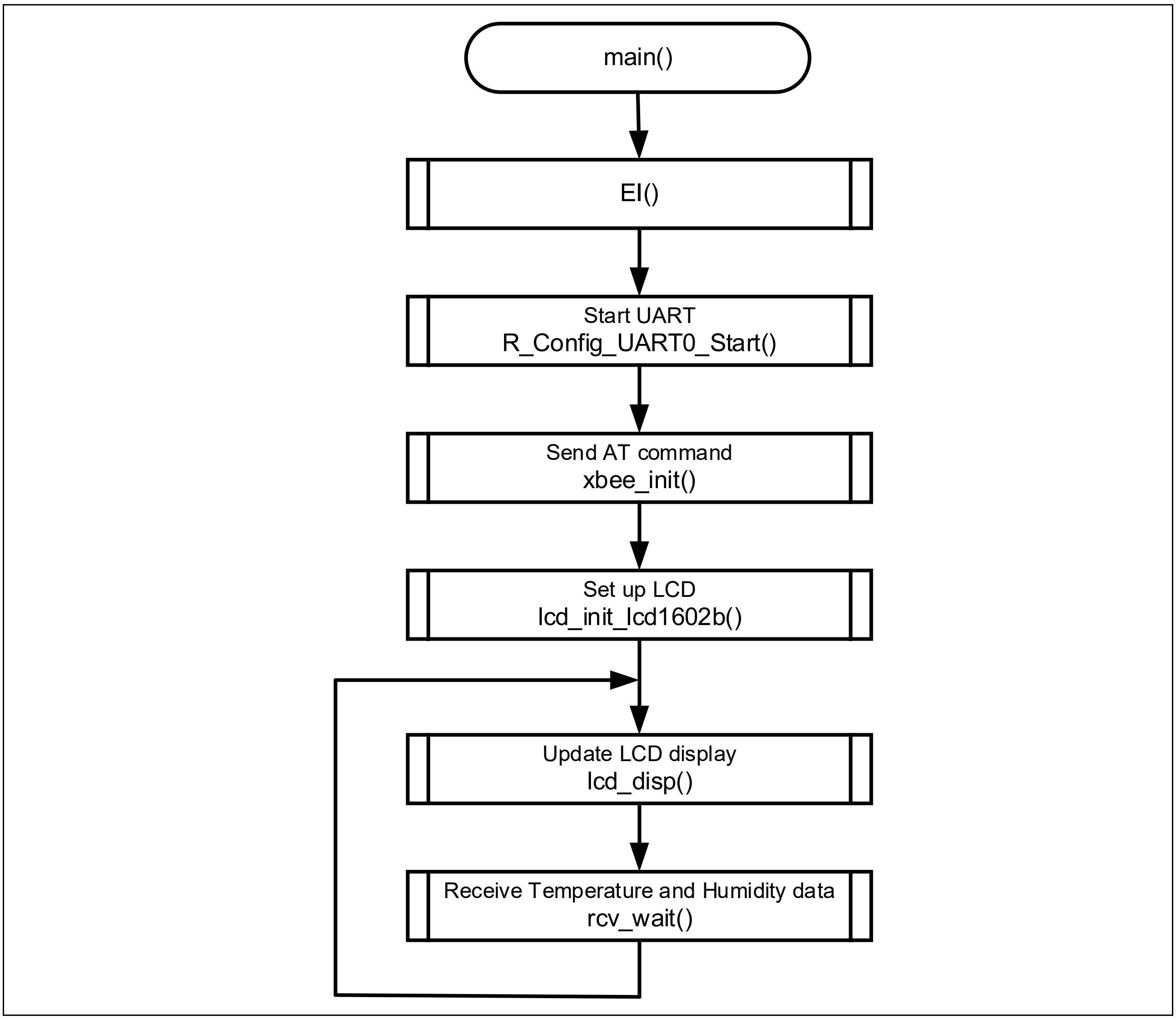
Description	Temperature / humidity data reception
Header	r_cg_macrodriver.h, r_cg_userdefine.h, Config_UART0.h, command.h, string.h
Declaration	void rcv_wait (void);
Details	Wait for data reception from the wireless transmission module, and after receiving it, extract the temperature and humidity data.
Argument	None
Return value	None
Note	None

5.2.4 Flowchart

5.2.4.1 Main Processing

The flowchart for the main processing is shown below.

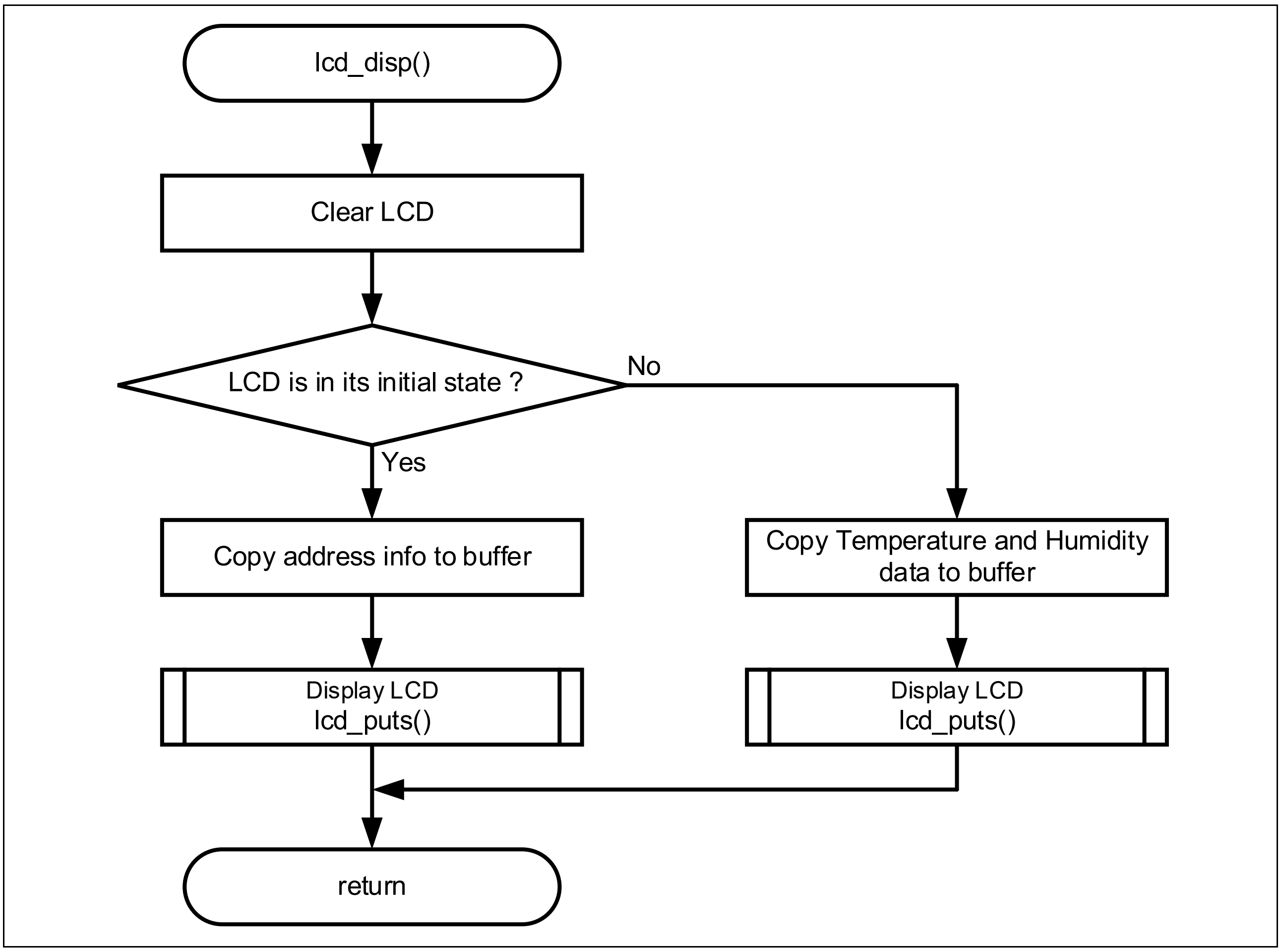
Figure 5-11 Main Processing



5.2.4.2 LCD display Update : lcd_disp

The flowchart for the function lcd_disp is shown below.

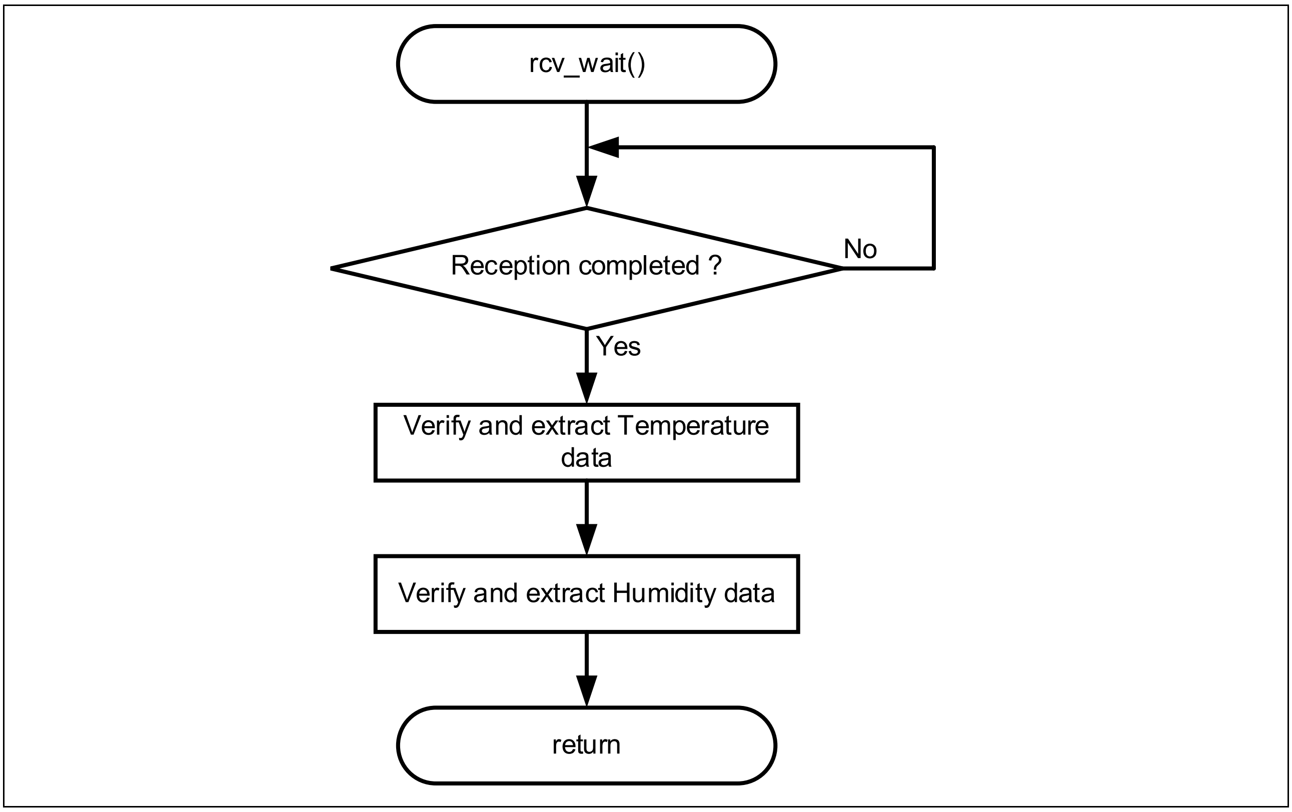
Figure 5-12 lcd_disp



5.2.4.3 Temperature / humidity data reception : rcv_wait

The flowchart for the function rcv_wait is shown below.

Figure 5-13 rcv_wait



5.2.5 Sample Program Structure

5.2.5.1 Peripheral Functions to Be Used

The following table lists the peripheral functions used in the sample program.

Table 5-8 Peripheral Functions to Be Used (Receiving Wireless Module)

Peripheral Function	Use
IICA0	Used for I2C communication with PCF8574 to control the display of the LCD module.
UART0	Used for UART communication with XBee ZB S2C, command transmission to XBee ZB S2C, and confirmation of response results returned from XBee ZB S2C.

5.2.5.2 Settings of Peripheral Functions

The following table lists the settings of Smart Configurator used in the sample program. The items and settings in each table of Smart Configurator Settings are described with the names displayed on the actual setting screen.

Table 5-9 Smart Configurator Settings (1/2)

Category	Item	Settings
Smart Configurator >> Clock		The settings on the [Clock] tab are as follows:
	VDD setting	2.4 V ≦ VDD ≦ 5.5 V
	High-speed on-chip oscillator	Checked Frequency: 16 MHz
	X1 oscillator circuit	Unchecked
	Low-speed on-chip oscillator	15 kHz
	Source selection for main system clock (f_{MAIN})	Set the on-chip oscillator clock (fIH) to (fMAIN) 16 MHz and (fCLK) 16,000 kHz.
	Operating clock supply for 12-bit interval timer	Unchecked (f_{IL}: 15 kHz)
Smart Configurator >> System		The settings on the [System] tab are as follows:
	Pseudo RRM/DMM function setting	Not used
	Start/Stop function setting	Not used
	Security ID setting	Set security ID. Security ID "0x00000000000000000000"
	RESET pin setting	Not used
	Operation mode setting	2.52 V
Smart Configurator >> Component >> r_bsp		Use the default settings except for the following changes:
	API functions disable(R_BSP_StartClock, R_BSP_StopClock)	Disable
	API functions disable(R_BSP_SetClockSource)	Disable
	API functions disable(R_BSP_ChangeClockSetting)	Disable
	API functions disable(R_BSP_SoftwareDelay)	Disable
Smart Configurator >> Component >> Config_IICA0		The settings on [Config_IICA0] are as follows:
	Local address	16
	Operation mode setting	Set [Standard] and transfer clock (f_{SCL}): 100000 (bps)
	tR and tF settings	Unchecked
	Interrupt setting	Level 3 (lowest priority)
	Callback function setting	Select [Master transmission end], [Master reception end], and [Master error].
	Callback extension setting	Unchecked

Table 5-10 Smart Configurator Settings (2/2)

Category	Item		Settings
Smart Configurator >> Component >> Config_UART			The settings on [Config_UART0] are as follows:
	Transmission	UART0 clock setting	Operating clock: CK00 Clock source: f_{CLK}/2^3
		Transfer mode setting	Continuous transfer mode
		Data bit length setting	8 bits
		Data transfer direction setting	LSB
		Parity setting	No parity bit
		Stop bit length setting	1 bit
		Transmit data level setting	Non-inversion (normal)
		Transfer rate	9600 (bps)
		Interrupt setting	Level 3 (lowest priority)
		Callback function setting	End of transmission
	Reception	UART0 clock setting	Operating clock: CK00 Clock source: f_{CLK}/2^3
		Data bit length setting	8 bits
		Data transfer direction setting	LSB
		Parity setting	No parity bit
		Transmit data level setting	Non-inversion (normal)
		Transfer rate	9600 (bps)
		Interrupt setting	Level 3 (lowest priority)
		Callback function setting	Reception end, reception error
Smart Configurator >> Pin			The settings on the [Pin] tab are as follows:
	Serial array unit (SAU0)		SAU0 RxD0: P04 TxD0: P03
	Serial interface IICA (IICA0)		IICA0 SCLA0: P06 SDAA0: P07

5.2.5.3 File Structure

The following table lists the file structure of the sample program.

Table 5-11 File Structure (Receiving Wireless Module)

Folder name, file name	Explanation
src	Program storage folder
├ command	Command related program storage folder
│ ├ xbee_atcom.c	AT Command related source file
│ └ xbee_atcom.h	AT Command related header file
│ ├ command.c	Command related source file
│ └ command.h	Command related header file
├ smc_gen	Smart Configurator generation folder
│ ├ Config_IICA0	
│ ├ Config_UART0	
│ ├ general	
│ ├ r_bsp	
│ └ r_config	
├ main.c	Main processing source file
├ lcd_1602b.c	LCD display related source file
└ lcd_1602b.h	LCD display related header file

6. Description of Software Operation

6.1 Initialization Using the AT Commands

To enable communication between two XBee ZB S2C devices, it is necessary to configure them, as they do not communicate with each other in their default settings. Here, I will explain the method for initializing the XBee ZB S2C devices to enable communication using AT commands.

For the initial configuration of the XBee ZB S2C, you can use the UART interface to send AT commands from a microcontroller and set the parameters related to communication. The settings involve writing parameter values to the XBee ZB S2C using AT commands.

Table 6-1 List of AT Commands Used for Configuring XBee ZB S2C

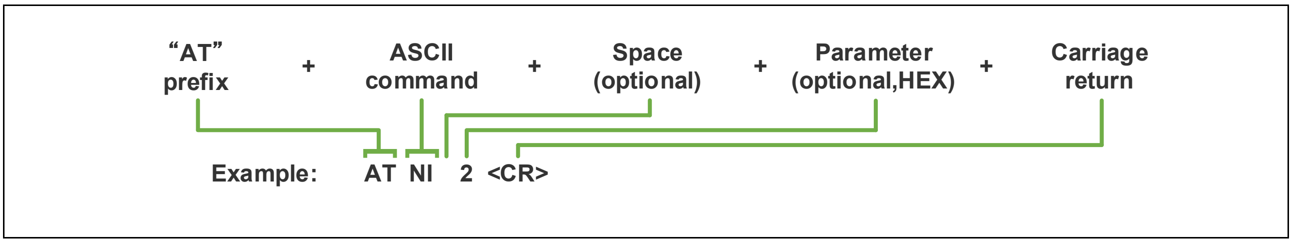
Configuration Items	Configuration Values (Commands)	Description
ID PAN ID	Any value (in the sample program, '5555' is used). *Ensure it is the same for both transmission and reception.	XBee ZB S2C can communicate only with devices having the same ID parameter.
DH Destination Address High	13A200	Specify the upper address of the communication destination. XBee ZB S2C is fixed at '13A200'.
DL Destination Address Low	The values of DL for each other. Please refer to the XBee ZB S2C device itself for this information.	Specify the lower address of the destination. For XBee ZB S2C, you can check the value by connecting to XCTU and confirming the DL value, or by inspecting the device (MAC).
Entering command mode	+++	Switching XBee ZB S2C to command mode.
Exiting command mode	ATCN	Exiting the command mode of XBee ZB S2C.
RESTORE defaults	ATRE<CR>	Restoring XBee ZB S2C to factory default settings.
Reading the address	ATSL	Reading the configured address of XBee ZB S2C.

To enable communication between two XBee ZB S2C modules, you need to configure three settings: ID, DH, and DL. Please configure these values in the following order: ID → DH → DL. Make sure to set each of these values on both XBee ZB S2C modules.

If you enter command mode, make sure to reconfigure ID, DH, and DL.

The format for sending AT commands is as follows:

Figure 6-1 Transmission Format of AT Commands



To control the XBee ZB S2C using the AT commands, it is necessary to enter command mode before sending the AT commands. Refer to the following online manual for details about how to enter command mode and about parameters to be sent for each AT command.

https://www.digi.com/resources/documentation/Digidocs/90001500/Default.htm

6.2 Preparation of Hardware

Connect the transmitting wireless module RL78/G15, XBee ZB S2C, and SHT40 as follows. Also, connect the receiving wireless module RL78/G15, XBee ZB S2C, PCF8574, and LCD1602B as follows.

Figure 6-2 Overall View of Connection

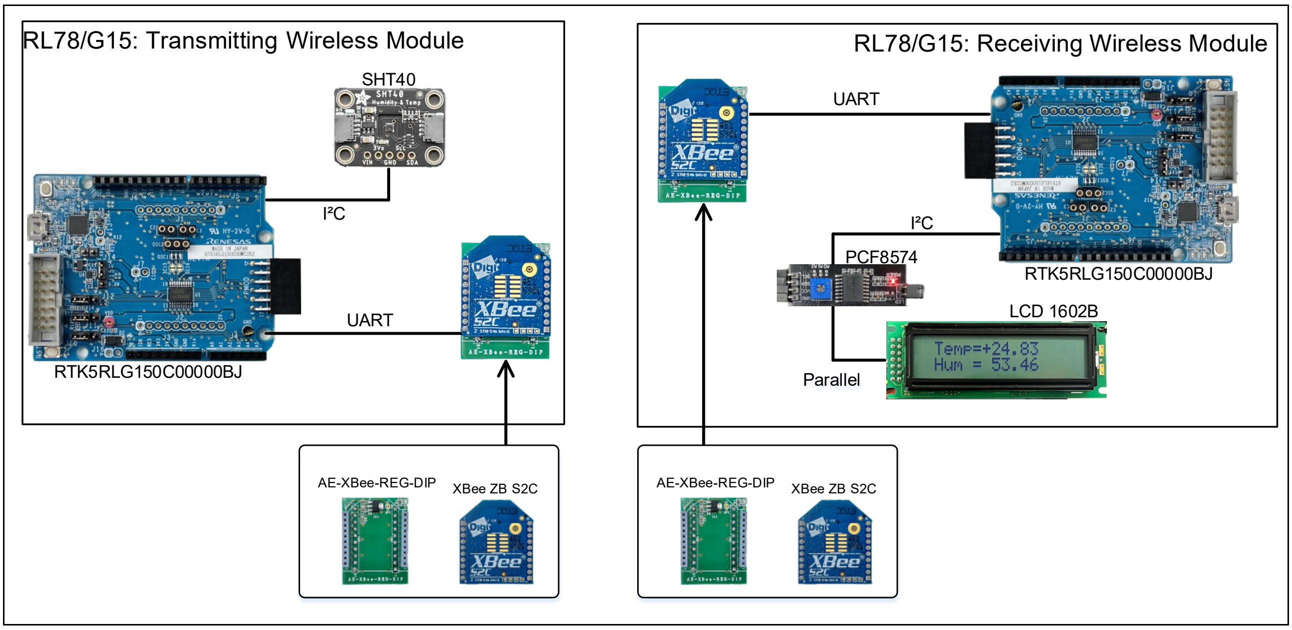


Table 6-2 Pins to which RL78/G15 and XBee ZB S2C Connect (common to Transmitting and Receiving)

RL78/G15		XBee ZB S2C		Description
Pin number	Name	Pin number	Name	
-	VDD	Pin1	VCC	Power supply from RL78/G15 to XBee ZB S2C
-	GND	Pin10	GND	GND of RL78/G15 and XBee ZB S2C
P03	TXD0	Pin3	DIN	UART transmission from RL78/G15 to XBee ZB S2C
P04	RXD0	Pin2	DOUT	UART Reception from XBee ZB S2C to RL78/G15

Table 6-3 Pins to which RL78/G15 and SHT40 Connect (Transmitting Wireless Module)

RL78/G15		SHT40		Description
Pin number	Name	Pin number	Name	
P06	SCLA0	Pin2	SCL	I²C clock transmission from RL78/G15 to SHT40
P07	SDAA0	Pin1	SDA	I²C data transmission from RL78/G15 to SHT40
-	VDD	Pin3	VDD	Power supply from RL78/G15 to SHT40
-	GND	Pin4	VSS	GND of RL78/G15 and SHT40

Table 6-4 Pins to which RL78/G15 and PCF8574 Connect (Receiving Wireless Module)

RL78/G15		PCF8574		Description
Pin number	Name	Pin number	Name	
-	VDD	16	VDD	Power supply from RL78/G15 to PCF8574
-	GND	8	VSS	GND of RL78/G15 and PCF8574
P06	SCLA0	14	SCL	I^2C clock transmission from RL78/G15 to PCF8574
P07	SDDA0	15	SDA	I^2C data transmission from RL78/G15 to PCF8574
-	-	1	A0	I2C address input pin0
-	-	2	A1	I2C address input pin1
-	-	3	A2	I2C address input pin2

Note. The address of PCF8574 used in this application is 0x27 (A0, A1, A2 = High).

Table 6-5 Pins to which PCF8574 and LCD1602B Connect (Receiving Wireless Module)

PCF8574		LCD1602B		Description
Pin number	Name	Pin number	Name	
1	VDD	1	VDD	GND of PCF8574 and LCD1602B
2	VSS	2	VSS	Power supply from PCF8574 to LCD1602B
3	VO	3	VO	Contrast control from PCF8574 to LCD1602B
4	P0	4	RS	Register Select from PCF8574 to LCD1602B: Specify whether it's a command or data.
5	P1	5	R/W	Read/Write specification from PCF8574 to LCD1602B.
6	P2	6	E	Enable signal from PCF8574 to LCD1602B.
-	-	7	DB0	Data0 from PCF8574 to LCD1602B
-	-	8	DB1	Data1 from PCF8574 to LCD1602B
-	-	9	DB2	Data2 from PCF8574 to LCD1602B
-	-	10	DB3	Data3 from PCF8574 to LCD1602B
11	P4	11	DB4	Data4 from PCF8574 to LCD1602B
12	P5	12	DB5	Data5 from PCF8574 to LCD1602B
13	P6	13	DB6	Data6 from PCF8574 to LCD1602B
14	P7	14	DB7	Data7 from PCF8574 to LCD1602B

Note. This application does not use LCD1602B DB0~DB3 pins because the LCD1602B is controlled in 4-bit mode.

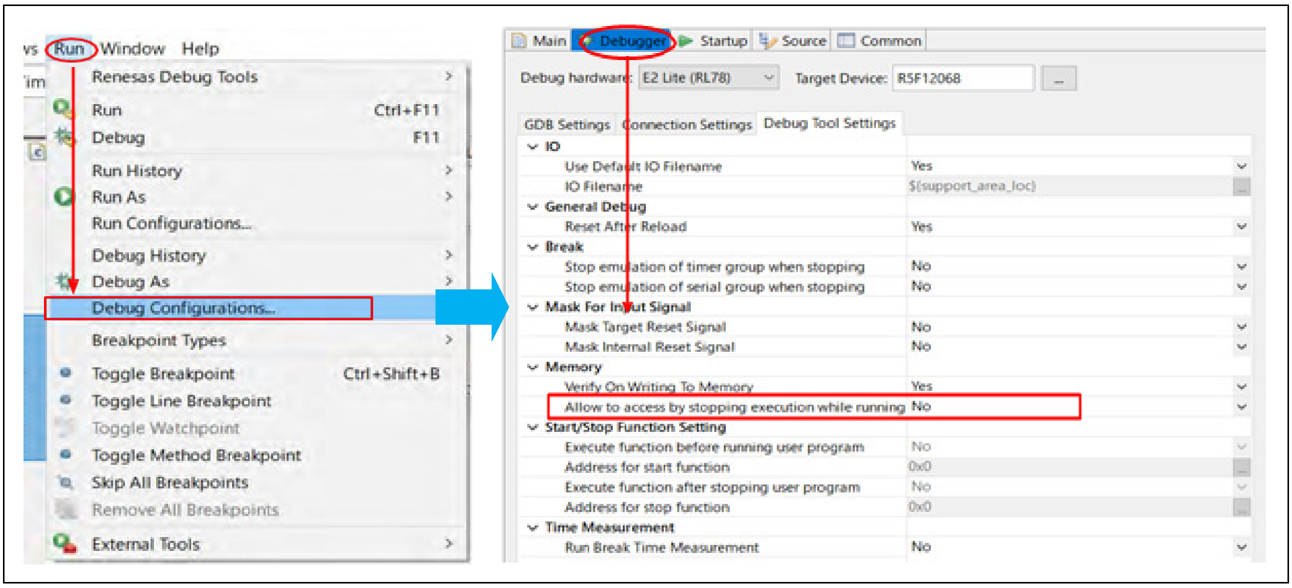
6.3 Debugger settings

Before performing the operation check, please verify the following in the debugger settings. When using the debugger's RRM function, the debugger occupies the first 4 bytes of RAM, so it is necessary to disable the RRM function.

(1) e^2 studio

As shown in Figure 6-3, select [Debug configuration] of [Run] menu and change setting of [Allow to access by stopping execution while running] to "No" in [Debugger] dialog.

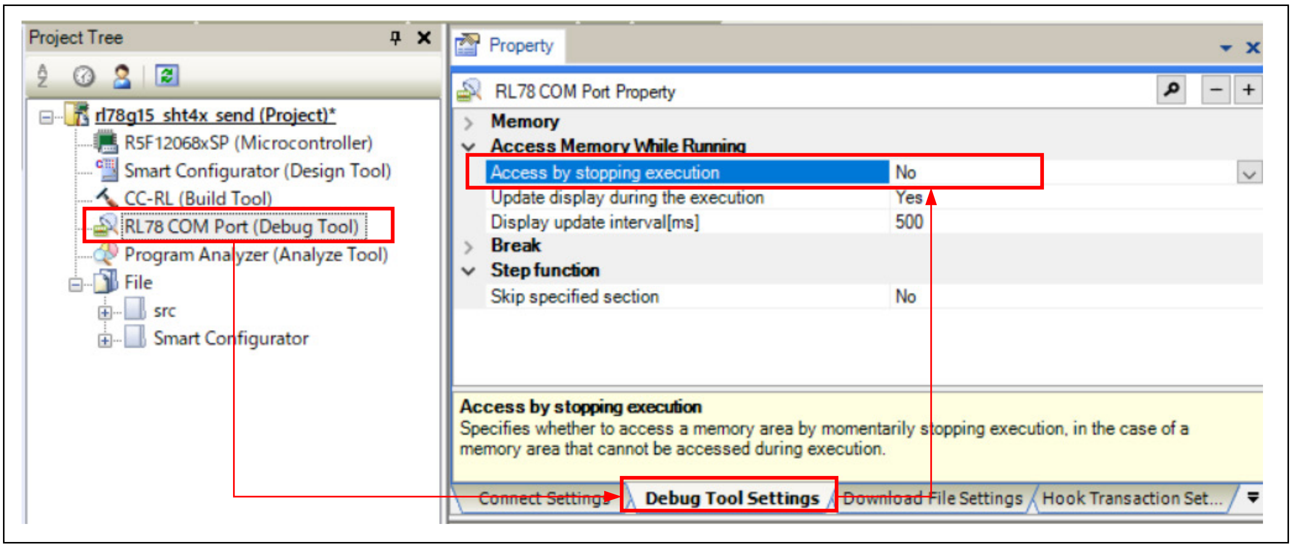
Figure 6-3 Change Debugger Settings (e^2 studio)



(2) CS+ for CC

As shown in Figure 6-4, select [RL78 COM Port (Debug Tool)] and change setting of [Access by stopping execution] to “No” in [Debug Tool Settings].

Figure 6-4 Change Debugger Settings (CS+ for CC)



(3) IAR

In IAR Embedded Workbench for Renesas RL78, the debugger options do not provide settings equivalent to functions related to memory access. Therefore, after starting the debugger, if you register variables in ‘Live Watch,’ execution will temporarily halt and the RRM function will operate.

In the IAR environment for this application note, unlike e2 studio or CS+, no configuration changes are required. However, to avoid unexpected execution halts caused by the RRM function, please debug without using the Live Watch feature.

6.4 Operation Confirmation Methods

This sample program allows the control of LED1 on the RL78/G15 Fast Prototyping Board and the SHT40 sensor connected to the same board using the transmitting wireless module. It receives temperature and humidity data sent from the transmitting wireless module and displays the temperature and humidity on the LCD1602B.

(1) Set the power selection header of the receiving wireless module's RL78/G15 Fast Prototyping Board to 3.3V power (J15: short 2-3), and supply power to the RL78/G15. At this point, the XBee ZB S2C also receives power.

(2) Set the power selection header of the transmitting wireless module's RL78/G15 Fast Prototyping Board to 3.3V power (J15: short 2-3), and supply power to the RL78/G15. At this point, the XBee ZB S2C also receives power.

(3) Press the RST (reset) switch on the receiving wireless module. The XBee ZB S2C address is displayed on the LCD1602B.

(4) Press the RST (reset) switch on the transmitting wireless module. The transmitting wireless module connects communication with the XBee ZB S2C, and LED1 turns on. When communication is established, temperature and humidity data are transmitted to the receiving wireless module.

(5) The receiving wireless module receives temperature and humidity data from the transmitting wireless module and data is displayed on LCD1602B.

(6) Press and hold the SW on the transmitting wireless module. The transmitting wireless module disconnects communication with the XBee ZB S2C, and LED1 turns off.

(7) Press and hold the SW on the transmitting wireless module. The transmitting wireless module connects communication with the XBee ZB S2C, and LED1 turns on. When communication is established, temperature and humidity data are transmitted to the receiving wireless module.

(8) Repeat steps from (5) to (7).

7. Reference Documents

- RL78/G15 User’s Manual: Hardware (R01UH0959)
- RL78/G15 Fast Prototyping Board User’s Manual (R12UM0042)
- RL78 Smart Configurator User's Guide: CS+ (R20AN0580)
- RL78 Smart Configurator User's Guide: e² studio (R20AN0579)
- RL78 Smart Configurator User's Guide: IAREW (R20AN0581)

(The latest versions can be downloaded from the Renesas Electronics website.)

Technical Update

(The latest information can be downloaded from the Renesas Electronics website.)

Revision History

Rev.	Date	Description	
		Page	Summary
1.00	Feb 18, 2026	-	First edition

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/.