M16C/63, 64A, 64C, 65, 65C, 6C, 5LD, 56D, 5L, 56, 5M, and 57 Groups

UART Communication with the PC Terminal Software Using XIN

R01AN0816EJ0100
Rev. 1.00
May. 31, 2012

Abstract

This document describes the setting method and application example for using terminal software on a personal computer (hereinafter referred to as PC terminal software) to perform UART communication with the XIN clock in the M16C/63, 64A, 64C, 65, 65C, 6C, 5LD, 56D, 5L, 56, 5M, and 57 Groups. The M16C/65C MCU is used in this application note.

Products

M16C/63, 64A, 64C, 65, 65C, 6C, 5LD, 56D, 5L, 56, 5M, and 57 Groups

When using this application note with other Renesas MCUs, careful evaluation is recommended after making modifications to comply with the alternate MCU.

Contents

1. Specifications

Data transmission and reception are performed between the PC terminal software and the M16C/65C Group MCU using clock asynchronous serial I/O (UART) mode.
Table 1.1 lists the Peripheral Function and Its Application and Table 1.2 lists PC Terminal Software Settings.

Table 1.1 Peripheral Function and Its Application

Peripheral Function	Application
Serial interface (UART1)	Data transmission and reception with the PC terminal software

Table 1.2 PC Terminal Software Settings

Item	Setting
Bits per second	115200 bps
Data bit	8 bits
Parity	None
Stop bit	1 bit
Flow control	None

Two numerical values are transmitted from the PC terminal software and received by the M16C/65C Group MCU. Then the M16C/65C Group MCU adds the two numerical values and the calculation result is transmitted to the PC terminal software.

Figure 1.1 shows the Operation Outline.

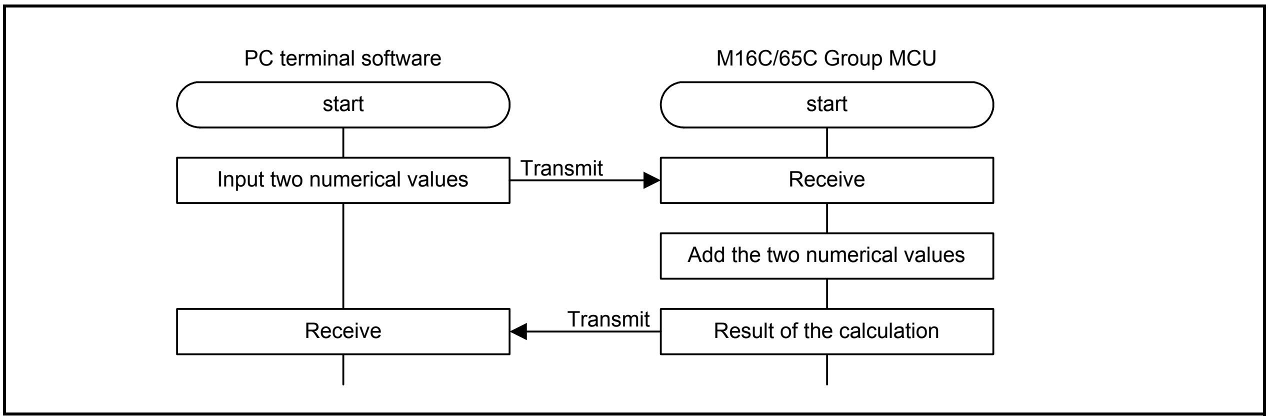


Figure 1.1 Operation Outline

2. Operation Confirmation Conditions

The sample code accompanying this application note has been run and confirmed under the conditions below.

Table 2.1 Operation Confirmation Conditions

Item	Contents
MCU used	M16C/65C Group
Operating frequencies	• XIN clock: 20 MHz • CPU clock: 20 MHz (main clock: no division) • Peripheral clock (UART): f1
Operating voltage	5.0 V (available from 2.7 to 5.0 V)
Integrated development environment	Renesas Electronics Corporation High-performance Embedded Workshop Version 4.09
C compiler	Renesas Electronics Corporation M16C Series, R8C Family C Compiler V.5.45 Release 01
	Compile options -c -finfo -dir "$(CONFIGDIR)" (The default setting is used in the integrated development environment.)
Operating mode	Single-chip mode
Sample code version	Version 1.00
Tool used	Terminal software

3. Hardware

3.1 Hardware Configuration

The PC terminal software is connected to the M16C/65C MCU.

Figure 3.1 shows a Connection Example.

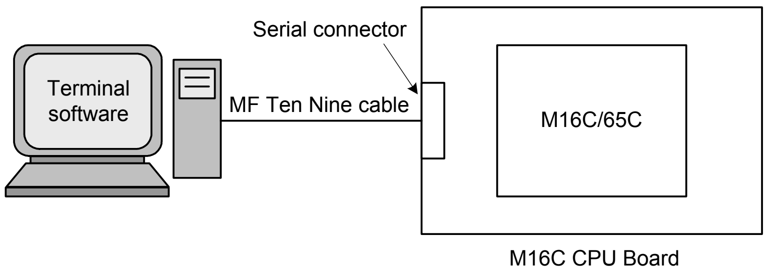


Figure 3.1 Connection Example

3.2 Pins Used

Table 3.1 lists the Pins Used and Their Functions.

Table 3.1 Pins Used and Their Functions

Pin Name	I/O	Function
P6_7/TXD1	Output	ASCII code output
P6_6/RXD1	Input	ASCII code input

4. Software

In the sample code, data transmission and reception are performed between the PC terminal software and the M16C/65C Group MCU using UART1 clock asynchronous serial I/O mode. First, the MCU receives the ‘single-digit number’, ‘+’, and ‘single-digit number’ transmitted by the PC terminal software. Then, the MCU adds the two single-digit numbers received and transmits the result to the PC terminal software.

The operation example is described below.

(1) Use the keyboard to enter a single-digit number, press the '+' key, enter a second single-digit number, and then press the 'Enter' key. For example, press the following keys: 5 + 7 Enter

The following table shows the ASCII code for the example just entered by the keyboard.

Input from a PC	5	+	7	Enter key
ASCII code	35h	2Bh	37h	0Dh

(2) The MCU adds the two single digit numbers entered in step (1) (i.e. 5 + 7 in the example).

(3) The MCU transmits the ‘LF/NL’, ‘=’, ‘calculation result’ (i.e. 12 in the example), ‘CR’ and ‘LF/NL’ to the PC terminal software.

The following table shows the data transmitted from the MCU and the corresponding ASCII code.

Data transmitted from the MCU	LF/NL (line feed)	=	1	2	CR (carriage return)	LF/NL (line feed)
ASCII code	0Ah	3Dh	31h	32h	0Dh	0Ah

Table 4.1 lists the UART1 Setting Conditions.

Table 4.1 UART1 Setting Conditions

Item	Setting
Operation mode	Clock asynchronous serial I/O mode
U1BRG count source	f1SIO
Character bits	8 bits
Bit rate	115200 bps (20 MHz (f1SIO) / (16 × (10 + 1)) ≈ 115200)
Transmit/receive clock	Internal clock
Stop bit	1 bit
Parity	Disabled
TXD, RXD I/O polarity	Not inverted
Error signal	No output
Data output	CMOS output for the TXD1 pin
$\overline{CTS}/\overline{RTS}$ function	Disabled
Bit order	LSB first
Interrupt source for transmission	Transmit buffer is empty (TI = 1)
Transmit interrupt priority level	Level 1
Receive interrupt priority level	Level 2

4.1 Operation Overview

The sample code operations are described below.

(1) The first single-digit number is received by the MCU. The MCU then transmits it back to the PC terminal software. (1)
(2) The addition symbol (2Bh) is received by the MCU. The MCU then transmits it back to the PC terminal software. (1)
(3) The second single-digit number is received by the MCU. The MCU then transmits it back to the PC terminal software. (1)
(4) The 'enter key' (0Dh) is received by the MCU. The MCU then transmits a 'CR' (0Dh) back to the PC terminal software. (1)
(5) The MCU adds the first and second numerical values.
(6) The MCU transmits the calculation result and accompanying data (LF/NL = 'calculation result' CR LF/NL which is 0Ah 3Dh 'calculation result' CR LF/NL in ASCII code).

Note:
1. The characters (ASCII code) received from the PC are transmitted back to the PC as they are for displaying them in the terminal software.

Figure 4.1 shows the Overview of the PC Terminal Software and Sample Code Operations.

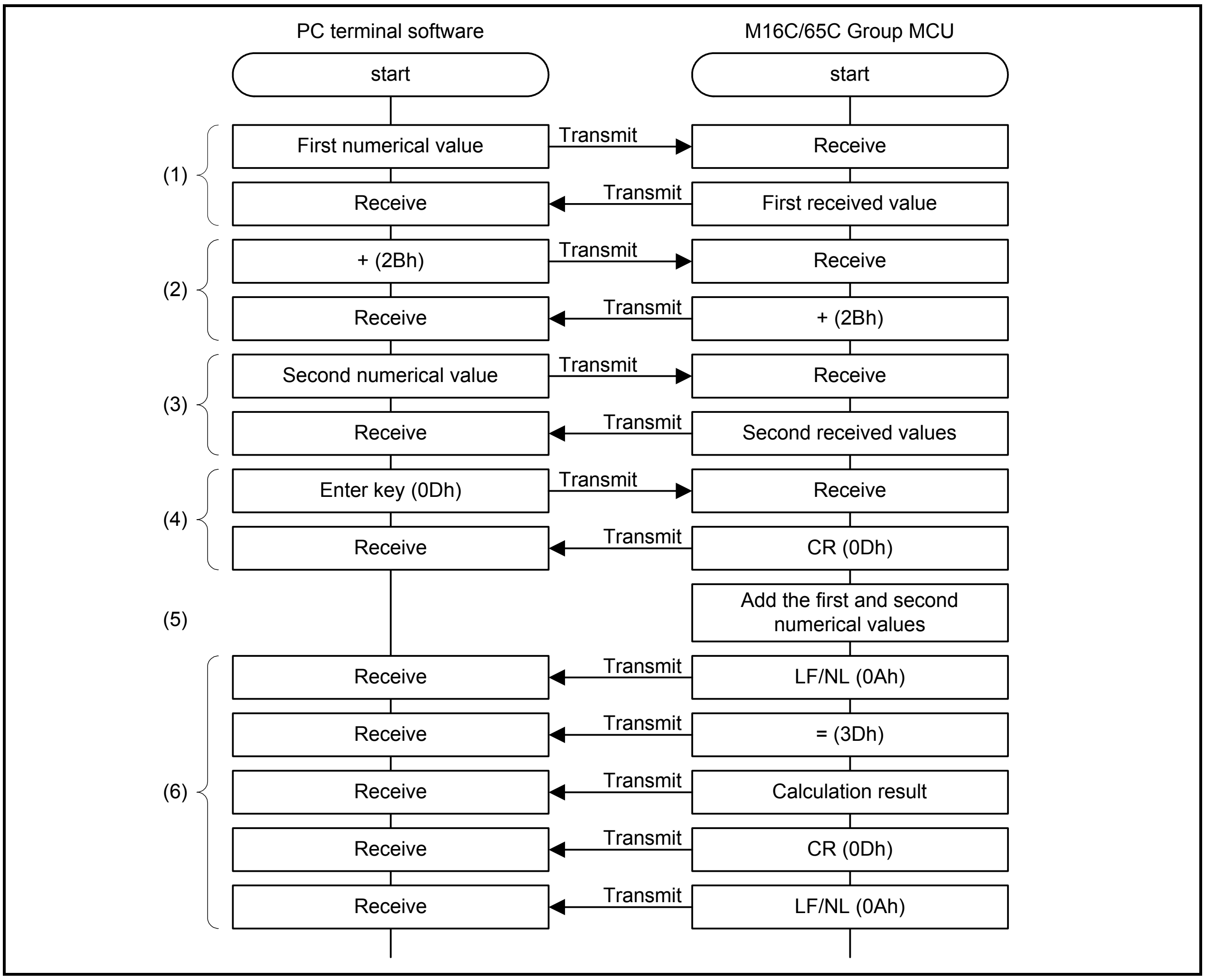


Figure 4.1 Overview of the PC Terminal Software and Sample Code Operations

4.2 Required Memory Size

Table 4.2 lists the Required Memory Size.

Table 4.2 Required Memory Size

Memory Used	Size	Remarks
ROM	436 bytes	In the r01an0816_src.c module
RAM	13 bytes	In the r01an0816_src.c module
Maximum user stack usage	13 bytes	
Maximum interrupt stack usage	23 bytes	

The required memory size varies depending on the C compiler version and compile options.

4.3 Constants

Table 4.3 lists the Constants Used in the Sample Code.

Table 4.3 Constants Used in the Sample Code

Constant Name	Setting Value	Contents
U1BRG_115200	10	Bit rate
INPUT_1ST_NUM	0	First value input mode
INPUT_2ND_NUM	1	Second value input mode
TRN_ST_MODE	2	Transmit start mode
TRN_BSY_MODE	3	Transmitting mode
ASCII_LF_NF	0Ah	LF/NL (line feed)
ASCII_CR	0Dh	CR (carriage return)
ASCII_PLUS	2Bh	+ (plus)
ASCII_EQUAL	3Dh	= (equal)

4.4 Variables

Table 4.4 lists the Global Variables.

Table 4.4 Global Variables

Type	Variable Name	Contents	Function Used
unsigned char	mode	Mode	main, input_data_calc_trn, transmit, _uart1_transmit
unsigned char	rcv_buf	Receive buffer	input_data_calc_trn, _uart1_receive
unsigned char	digit_num	Digit number	input_data_calc_trn
unsigned char	num_1st	First numerical value	input_data_calc_trn, calculation
unsigned char	num_2nd	Second numerical value	input_data_calc_trn, calculation
unsigned char	num_sum	Calculation result (first value plus second value)	calculation
unsigned char	trn_data[ ]	Transmit data storage array	transmit, trn_data_storage, _uart1_transmit
unsigned char	cnt_trn	Transmit counter	transmit, trn_data_storage, _uart1_transmit

4.5 Functions

Table 4.5 lists the Functions.

Table 4.5 Functions

Function Name	Outline
main	Main processing
mcu_init	CPU clock initialization
peripheral_init	Peripheral function initialization
calculation	Data calculation
transmit	Data transmission
trn_data_storage	Processing for transmit data storage
_uart1_transmit	UART1 transmit interrupt handler
_uart1_receive	UART1 receive interrupt handler
input_data_calc_trn	Input data calculation transmission

4.6 Function Specifications

The following tables list the sample code function specifications.

main	
Outline	Main processing
Header	None
Declaration	void main(void)
Description	Call the calculation function and transmit function when in transmit start mode.
Argument	None
Returned value	None

mcu_init	
Outline	CPU clock initialization
Header	None
Declaration	void mcu_init(void)
Description	Set the main clock (no division) as the CPU clock.
Argument	None
Returned value	None

peripheral_init	
Outline	Peripheral function initialization
Header	None
Declaration	void peripheral_init(void)
Description	Initialize SFRs associated with UART1.
Argument	None
Returned value	None

calculation	
Outline	Data calculation
Header	None
Declaration	void calculation(void)
Description	Add the two received values. Call the trn_data_storage function to prepare the transmit data to be sent to the PC.
Argument	None
Returned value	None

transmit	
Outline	Data transmission
Header	None
Declaration	void transmit(void)
Description	Set the first character (LF/NL) of the transmit data storage array to the U1TB register and transmit. Then set the mode to transmitting mode.
Argument	None
Returned value	None

trn_data_storage	
Outline	Processing for transmit data storage
Header	None
Declaration	void trn_data_storage(unsigned char sum_data)
Description	Store the transmit data in the transmit data storage array in order. The values of the calculation result are converted to ASCII code.
Argument	unsigned char sum_data: Calculation result
Returned value	None

_uart1_transmit	
Outline	UART1 transmit interrupt handler
Header	None
Declaration	void _uart1_transmit(void)
Description	Transmit the data stored in the transmit data storage array in 1-byte units when in transmitting mode. When all data in the transmit data storage array is transmitted, set the mode to first value input mode.
Argument	None
Returned value	None

_uart1_receive	
Outline	UART1 receive interrupt handler
Header	None
Declaration	void _uart1_receive(void)
Description	Read the received data from the U1RB register. If an overrun error does not occur, store the data in the receive buffer, and call the input_data_calc_trn function.
Argument	None
Returned value	None

input_data_calc_trn	
Outline	Input data calculation transmission
Header	None
Declaration	void input_data_calc_trn(void)
Description	Verify that the data received from the PC terminal software is correct data. If the data is verified, keep the data and transmit it to the PC terminal software. After 'CR' is transmitted, set the mode to transmit start mode.
Argument	None
Returned value	None

4.7 Flowcharts

4.7.1 Main Processing

Figure 4.2 shows the Main Processing.

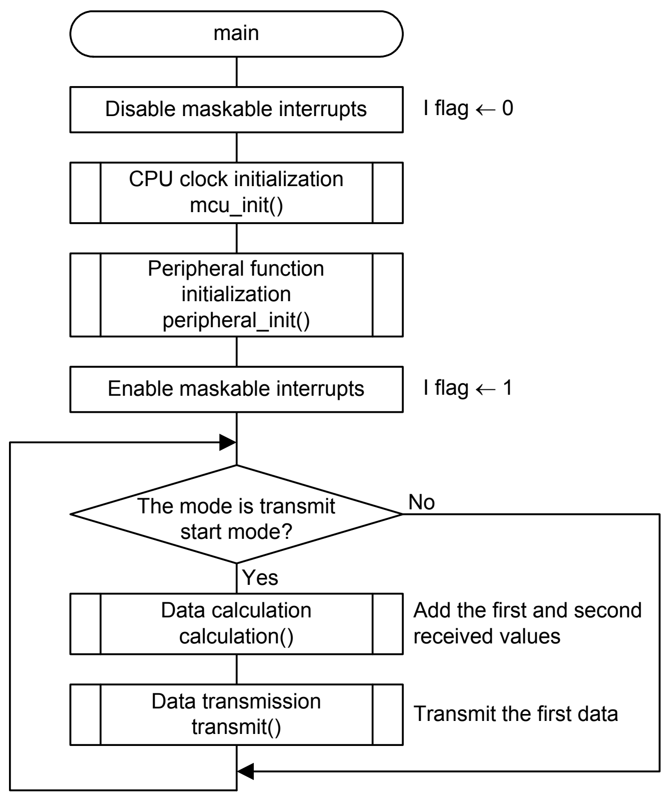


Figure 4.2 Main Processing

4.7.2 Peripheral Function Initialization

Figure 4.3 shows the Peripheral Function Initialization.

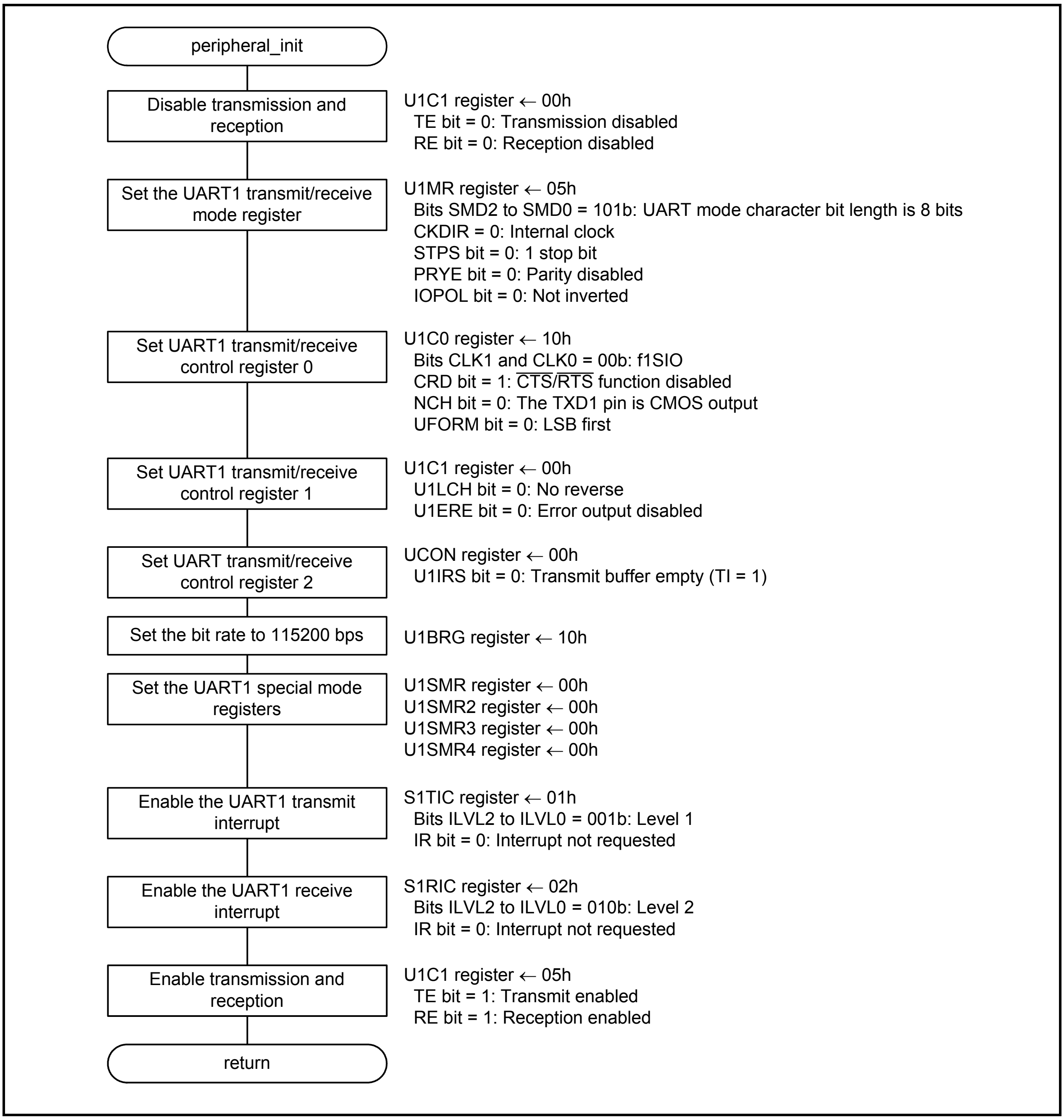


Figure 4.3 Peripheral Function Initialization

4.7.3 Data Calculation

Figure 4.4 shows the Data Calculation.

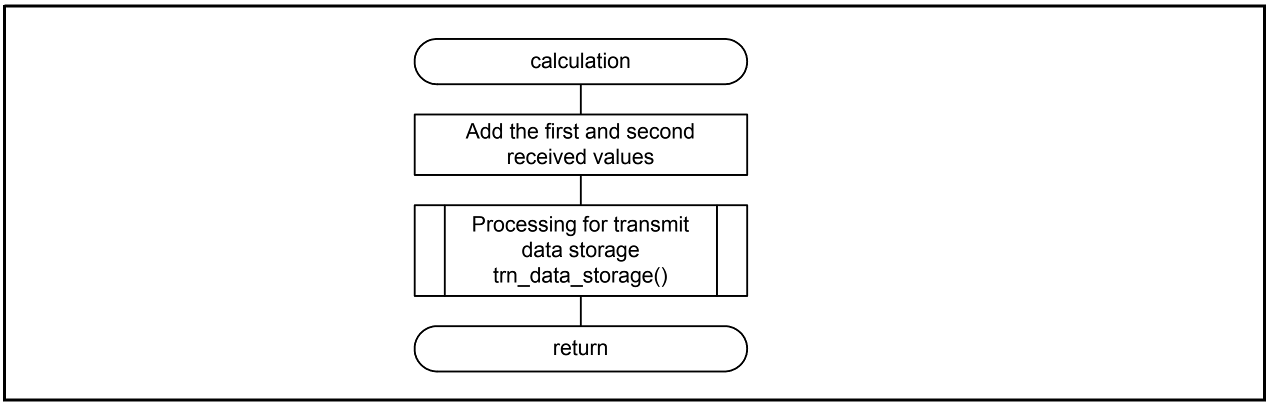


Figure 4.4 Data Calculation

4.7.4 Data Transmission

Figure 4.5 shows the Data Transmission.

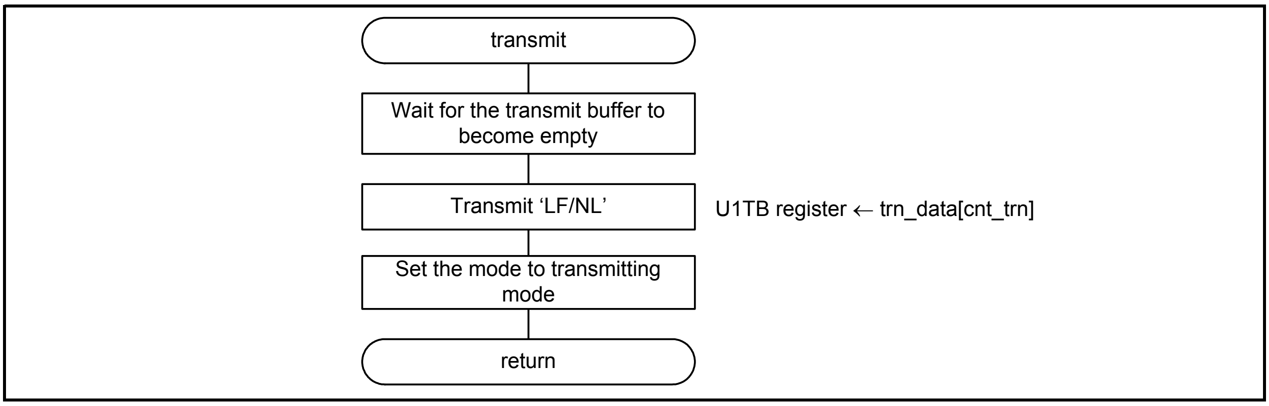


Figure 4.5 Data Transmission

4.7.5 Processing for Transmit Data Storage

Figure 4.6 shows the Processing for Transmit Data Storage.

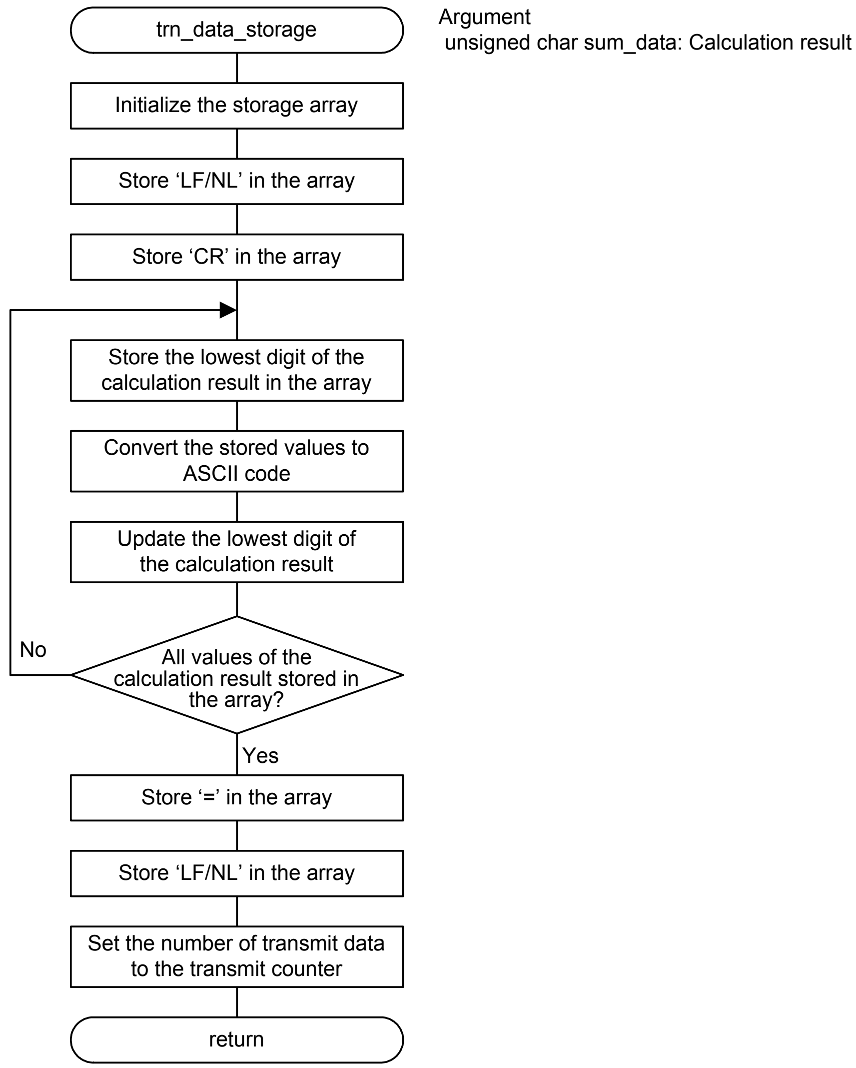


Figure 4.6 Processing for Transmit Data Storage

4.7.6 UART1 Transmit Interrupt Handler

Figure 4.7 shows the UART1 Transmit Interrupt Handler.

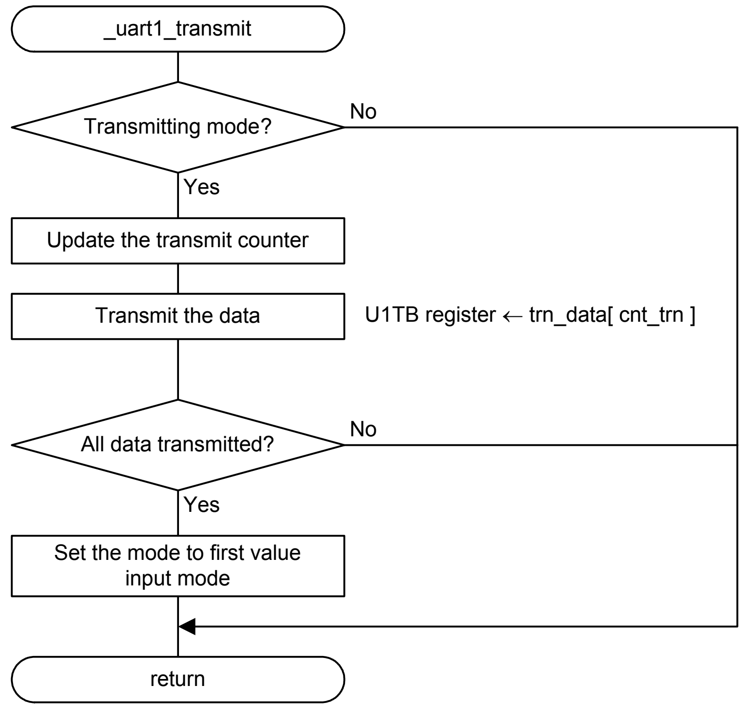
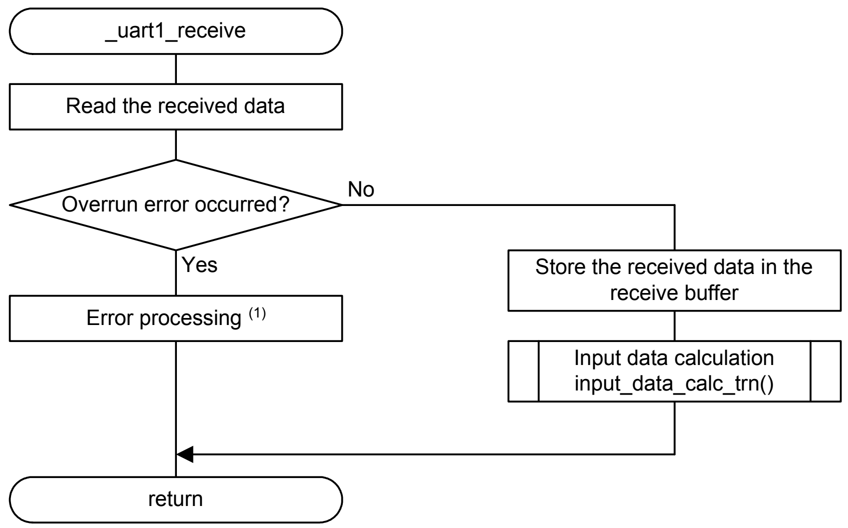


Figure 4.7 UART1 Transmit Interrupt Handler

4.7.7 UART1 Receive Interrupt Handler

Figure 4.8 shows the UART1 Receive Interrupt Handler.



Note:
1. The processing when an overrun error occurs is not performed in this application note. Add the processing as required.

Figure 4.8 UART1 Receive Interrupt Handler

4.7.8 Input Data Calculation Transmission

Figure 4.9 shows the Input Data Calculation Transmission.

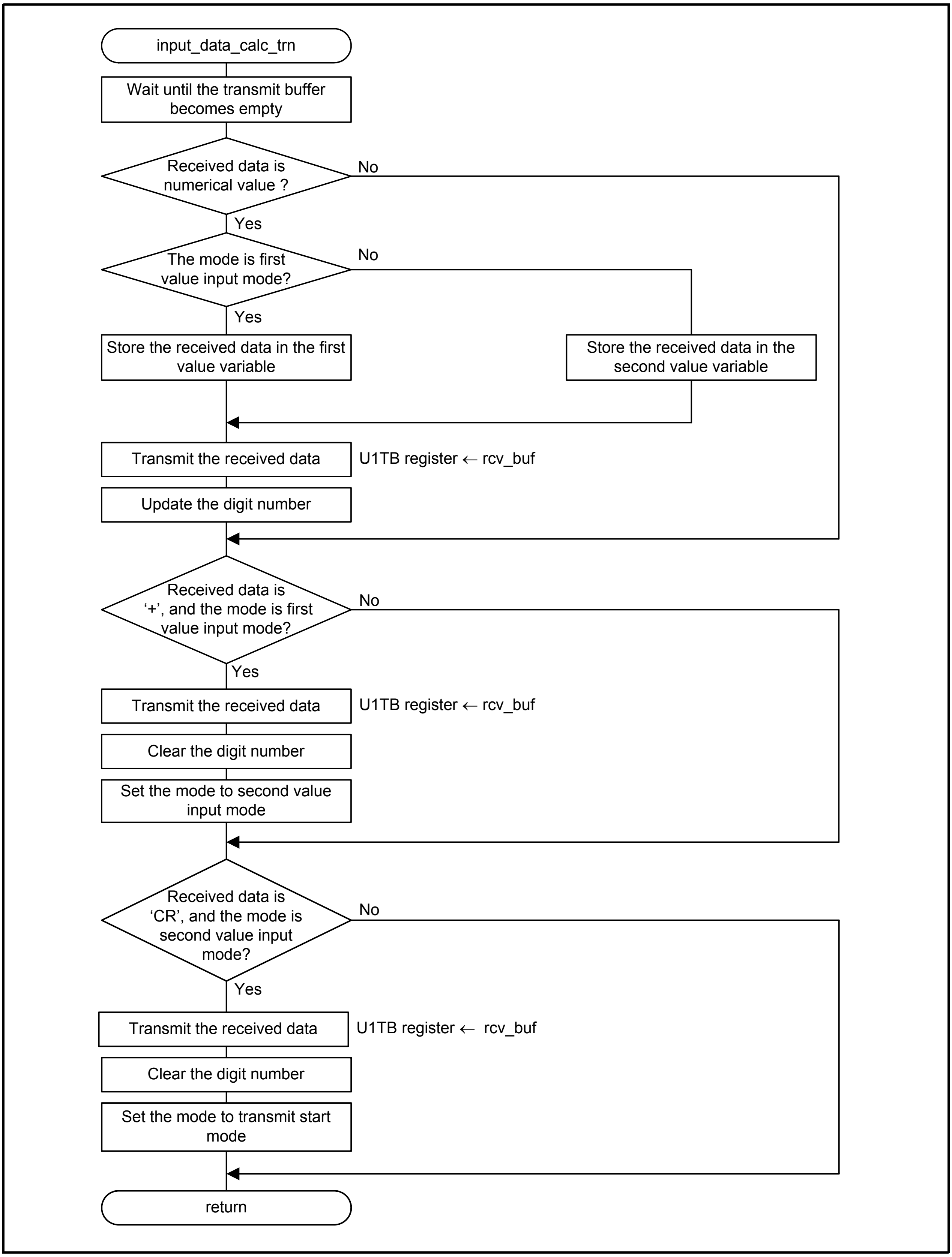


Figure 4.9 Input Data Calculation Transmission

5. Sample Code

Sample code can be downloaded from the Renesas Electronics website.

6. Reference Documents

M16C/63 Group User's Manual: Hardware Rev. 2.00
M16C/64A Group User's Manual: Hardware Rev. 2.00
M16C/64C Group User's Manual: Hardware Rev. 1.00
M16C/65 Group User's Manual: Hardware Rev. 2.00
M16C/65C Group User's Manual: Hardware Rev. 1.00
M16C/6C Group User's Manual: Hardware Rev. 2.00
M16C/5LD Group, M16C/56D Group User's Manual: Hardware Rev. 1.20
M16C/5L Group, M16C/56 Group User's Manual: Hardware Rev. 1.10
M16C/5M Group, M16C/57 Group User's Manual: Hardware Rev. 1.10
The latest versions can be downloaded from the Renesas Electronics website.

Technical Update/Technical News
The latest information can be downloaded from the Renesas Electronics website.

C Compiler Manual
M16C Series/R8C Series C Compiler Package V.5.45
C Compiler User's Manual Rev. 2.00
The latest version can be downloaded from the Renesas Electronics website.

Website and Support

Renesas Electronics website
http://www.renesas.com/

Inquiries
http://www.renesas.com/contact

Revision History	M16C/63, 64A, 64C, 65, 65C, 6C, 5LD, 56D, 5L, 56, 5M, and 57 Groups UART Communication with the PC Terminal Software Using XIN

Rev.	Date	Description	
		Page	Summary
1.00	May. 31, 2012	—	First edition issued

General Precautions in the Handling of MPU/MCU Products

The following usage notes are applicable to all MPU/MCU products from Renesas. For detailed usage notes on the products covered by this manual, refer to the relevant sections of the manual. If the descriptions under General Precautions in the Handling of MPU/MCU Products and in the body of the manual differ from each other, the description in the body of the manual takes precedence.

1. Handling of Unused Pins

   Handle unused pins in accord with the directions given under Handling of Unused Pins in the manual.

   — The input pins of CMOS products are generally in the high-impedance state. In operation with an unused pin in the open-circuit state, extra electromagnetic noise is induced in the vicinity of LSI, an associated shoot-through current flows internally, and malfunctions occur due to the false recognition of the pin state as an input signal become possible. Unused pins should be handled as described under Handling of Unused Pins in the manual.

2. Processing at Power-on

   The state of the product is undefined at the moment when power is supplied.

   — The states of internal circuits in the LSI are indeterminate and the states of register settings and pins are undefined at the moment when power is supplied.

     In a finished product where the reset signal is applied to the external reset pin, the states of pins are not guaranteed from the moment when power is supplied until the reset process is completed.

     In a similar way, the states of pins in a product that is reset by an on-chip power-on reset function are not guaranteed from the moment when power is supplied until the power reaches the level at which resetting has been specified.

3. Prohibition of Access to Reserved Addresses

   Access to reserved addresses is prohibited.

   — The reserved addresses are provided for the possible future expansion of functions. Do not access these addresses; the correct operation of LSI is not guaranteed if they are accessed.

4. Clock Signals

   After applying a reset, only release the reset line after the operating clock signal has become stable. When switching the clock signal during program execution, wait until the target clock signal has stabilized.

   — When the clock signal is generated with an external resonator (or from an external oscillator) during a reset, ensure that the reset line is only released after full stabilization of the clock signal. Moreover, when switching to a clock signal produced with an external resonator (or by an external oscillator) while program execution is in progress, wait until the target clock signal is stable.

5. Differences between Products

   Before changing from one product to another, i.e. to one with a different part number, confirm that the change will not lead to problems.

   — The characteristics of MPU/MCU in the same group but having different part numbers may differ because of the differences in internal memory capacity and layout pattern. When changing to products of different part numbers, implement a system-evaluation test for each of the products.

SALES OFFICES Renesas Electronics Corporation http://www.renesas.com

Refer to "http://www.renesas.com/" for the latest and detailed information.

Renesas Electronics America Inc.
2880 Scott Boulevard Santa Clara, CA 95050-2554, U.S.A.
Tel: +1-408-588-6000, Fax: +1-408-588-6130

Renesas Electronics Canada Limited
1101 Nicholson Road, Newmarket, Ontario L3Y 9C3, Canada
Tel: +1-905-898-5441, Fax: +1-905-898-3220

Renesas Electronics Europe Limited
Dukes Meadow, Millboard Road, Bourne End, Buckinghamshire, SL8 5FH, U.K
Tel: +44-1628-585-100, Fax: +44-1628-585-900

Renesas Electronics Europe GmbH
Arcadiastrasse 10, 40472 Düsseldorf, Germany
Tel: +49-211-65030, Fax: +49-211-6503-1327

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

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

Renesas Electronics Hong Kong Limited
Unit 1601-1613, 16/F., Tower 2, Grand Century Place, 193 Prince Edward Road West, Mongkok, Kowloon, Hong Kong
Tel: +852-2886-9318, Fax: +852 2886-9022/9044

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

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

Renesas Electronics Malaysia Sdn.Bhd.
Unit 906, Block B, Menara Amcorp, Amcorp Trade Centre, No. 18, Jln Persiaran Barat, 46050 Petaling Jaya, Selangor Darul Ehsan, Malaysia
Tel: +60-3-7955-9390, Fax: +60-3-7955-9510

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

Colophon 1.1