RL78/G23

Using VBAT Pin

Introduction

This application note describes how to use the VBAT pin (battery backup power) of RL78/G23.

While power is supplied from V_{DD}, clock time is displayed on the LCD using fixed-cycle interrupts of the realtime clock. If power is not supplied from V_{DD} and is supplied from the VBAT pin, clock time is not displayed on the LCD though the realtime clock continues operating.

The voltage detector (LVD0) can determine whether power is supplied from V_{DD}.

Target Device

RL78/G23

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. Specification

1.1 Overview of Specification

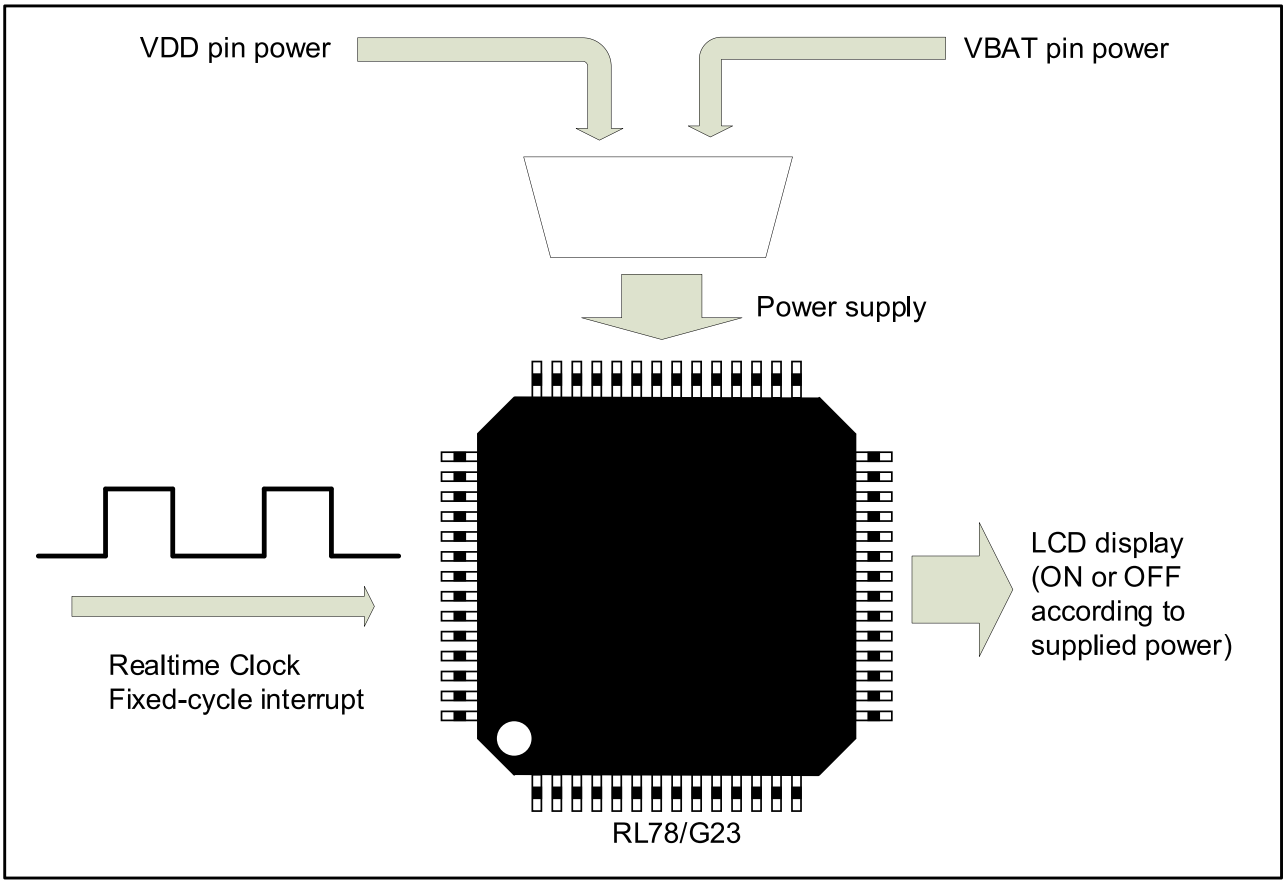
While power is supplied from V_{DD}, clock time is displayed on the LCD using fixed-cycle interrupts of the realtime clock. If power is not supplied from V_{DD} and is supplied from the VBAT pin, clock time is not displayed on the LCD though the realtime clock continues operating.
The voltage detection circuit (LVD0) can determine whether power is supplied from V_{DD}.

Table 1-1 lists peripheral functions to be used and Figure 1-1 Overview of VBAT Operationshows the overview of VBAT operation.

Table 1-1 Peripheral Function and Use

Peripheral Function	Use
Realtime clock	Used to generate realtime clock (RTC) interrupts (INTRTC).
Serial interface IICA0	IIC communication with the LCD module Clock time is displayed in the fixed-cycle interrupt processing. Clock time is not displayed while power is supplied from the VBAT pin.
Voltage detector LVD0	Supply voltage (V_{DD}) monitoring. An interrupt is generated when the power supply source changes to the VBAT pin.

Figure 1-1 Overview of VBAT Operation



1.1.1 Overview of the VBAT Pin

The main purpose of the VBAT pin is to maintain continuous operation of the real-time clock (RTC). If a backup battery is connected to the VBAT pin and power supply to the V_{DD} pin is lost, power can be supplied from the VBAT pin.

The VBAT pin supplies power to the V_{DD} pin via an internal diode. The internal diode of the VBAT pin is always connected to the V_{DD} pin. To prevent backflow to a power circuit or another circuit that is connected to the V_{DD} pin via a diode from the VBAT pin, connect external backflow-preventing diodes to the V_{DD}, EV_{DD0}, and EV_{DD1} pins.

The valid range of input voltage to the VBAT pin is 2.7 to 5.5 V. While power is being supplied from the VBAT pin, if the input voltage to the VBAT pin becomes less than 2.7 V, a POR reset might occur due to voltage drop of the diode.

The maximum current that the VBAT pin can supply is 150 μA. Therefore, while power is being supplied from the VBAT pin, the main system clock and peripheral functions (other than the RTC) must not operate. Make sure that no current flows to ports. Also, make sure that a reset does not occur while power is being supplied from the VBAT pin. This is because the current exceeds the maximum (150 μA) when the main system clock starts operating when reset is canceled.

1.2 Outline of Operation

While power is supplied from V_{DD}, clock time is displayed on the LCD using fixed-cycle interrupts of the realtime clock. If power is not supplied from V_{DD} and is supplied from the VBAT pin, clock time is not displayed on the LCD though the realtime clock continues operating.

The voltage detector (LVD0) is used to determine whether power is supplied from V_{DD}.

Monitoring the clock time on the LCD makes it possible to confirm that the realtime clock is continuously operating even during the V_{DD} power-off period.

In this application note, the interrupt processing is performed as follows:

- Clock time is displayed on the LCD in the fixed-cycle interrupt processing.
- Fixed-cycle interrupts are enabled or disabled in the voltage detection interrupt processing.

Table 1-2 shows LCD display layout (Y: year, M: month, D: day, h: hour, m: minute, s: second). Slash (/) and colon (:) are output as they are. Blank space represents blank (white space).

Table 1-2 LCD Display Layout

Top	Y	Y	/	M	M	/	D	D								
Bottom									h	h	:	m	m	:	s	s

The following describes main settings for peripheral functions.

(1) Initial settings for VBAT

- Set the OSCSEL bit in the CMC register to 1 to enable X1 oscillation mode. Clear the MSTOP bit in the CSC register to 0 to activate the X1 oscillation circuit. These settings prevent penetration current from flowing through the VBAT pin.

(2) Initial settings for the serial interface IICA

- Use IICA0 (P60 set to SCLA0 and P61 set to SDAA0).
- Select $f_{CLK}/2$ at the IICA0 operation clock.
- Set the local address to 0x10.
- Set the standard mode as the operation mode.
- Set the transfer clock to 80000 bps.
- Enable INTIICA0 interrupt.

(4) Initial settings for the LCD module

- Set to 8 bits bus mode, 2-line display, and font type 5x8 dots.
- Make settings to enable display indication, disable cursor display, and disable cursor blinking.
- Set the cursor shift direction to right.

(5) Initial settings for the realtime clock (RTC)

- Select the $f_{RTC128HZ}$ at the RTC operation clock.
- Present the time in 24-hour system.
- Disable the RTC1HZ pin output.
- Initialize the current date and time to 2000/1/1 (Fri) 00 : 00 : 00
- Enable fixed-cycle interrupt and set their cycle time to 1.0 second.
- Enable INTRTC interrupts.
- Set INTRTC interrupts priority level 2.

2. Operation Confirmation Conditions

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

Table 2-1 Operation Confirmation Conditions

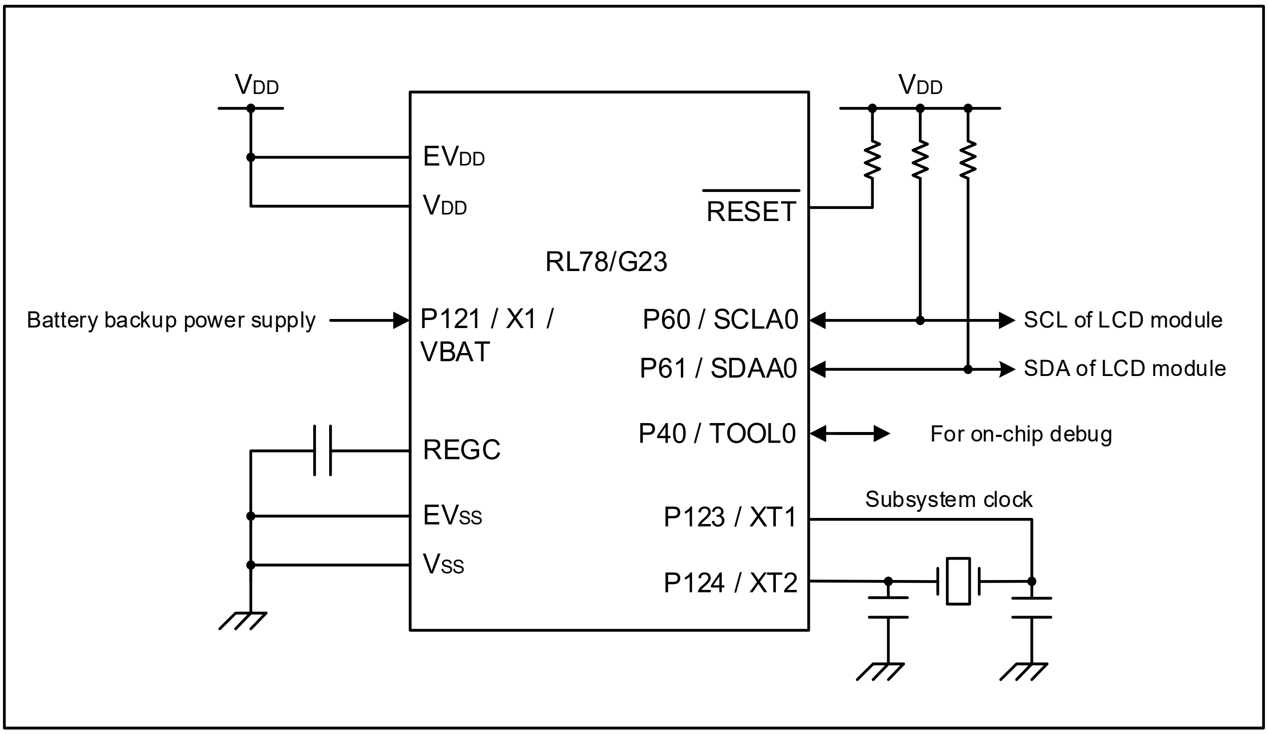
Item	Description
MCU used	RL78/G23 (R7F100GLG)
Board used	RL78/G23-64p Fast Prototyping Board (RTK7RLG230CLG000BJ)
Operating frequency	• High-speed on-chip oscillator clock (f_{IH}): 32 MHz • Subsystem clock (XT1clock (f_{XT})): 32.768 kHz
Operating voltage	During V_{DD} operation: 5.0V During VBAT operation: 3.0 V LVD0 detection voltage: Interrupt mode At rising edge TYP. 3.96 V (3.84 V to 4.08 V) At falling edge TYP. 3.88 V (3.76 V to 4.00 V)
Integrated development environment (CS+)	CS+ for CC V8.07.00 from Renesas Electronics Corp.
C compiler (CS+)	CC-RL V1.11.00 from Renesas Electronics Corp.
Integrated development environment (e2studio)	e2studio V2022-04 (22.4.0) from Renesas Electronics Corp.
C compiler (e2studio)	CC-RL V1.11.00 from Renesas Electronics Corp.
Integrated development environment (IAR)	IAR Embedded Workbench for Renesas RL78 V4.21.2 from IAR Systems Corp.
C compiler (IAR)	IAR C/C++ Compiler for Renesas RL78 V4.21.2.2420 from IAR Systems Corp.
LCD module	ACM1602NI-FLW-FBW-M01

3. Hardware Descriptions

3.1 Example of Hardware Configuration

Figure 3-1 shows an example of the hardware configuration used in the application note.

Figure 3-1 Hardware Configuration



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. Connect pins (with a name beginning with EV_{SS}), if any, to V_{SS}, and connect pins (with a name beginning with EV_{DD}), if any, to V_{DD}.

Note 3. Supply a voltage not less than 2.7 V to the power source for battery backup.

3.2 List of Pins to be Used

Table 3-1 lists the pins to be used and their functions.

Table 3-1 Pins to be Used and Their Functions

Pin name	I/O	Function
P121 / X1 / VBAT	—	Power supply pin for battery backup
P60/ SCLA0, P61 / SDAA0	Input / Output	IIC communication with LCD module

Caution In this application note, only the used pins are processed. When actually designing your circuit, make sure the design includes sufficient pin processing and meets electrical characteristic requirements.

4. Software Explanation

4.1 Changing the Settings of the Custom Board Support Package (BSP)

In the sample program, the P121, X1, VBAT, and EI121 pins are set in X1 oscillation mode. To specify this setting, add an instruction to the mcu_clocks.c file (the instruction indicated in red in the following section):

```
Line 646:   /* Clock operation mode control register(CMC) setting */
Line 647:   cmc_tmp |= 0x40U;
Line 648:   CMC = cmc_tmp;
```

4.2 Setting of Option Byte

Table 4-1 shows the option byte settings.

Table 4-1 Option Byte Settings

Address	Setting Value	Contents
000C0H / 040C0H	11101111B	Disables the watchdog timer. (Counting stopped after reset)
000C1H / 040C1H	10111010B	LVD0 detection voltage: Interrupt mode At rising edge TYP. 3.96 V (3.84 V to 4.08 V) At falling edge TYP. 3.88 V (3.76 V to 4.00 V)
000C2H / 040C2H	11101000B	HS mode, High-speed on-chip oscillator clock (f_{IH}): 32 MHz
000C3H / 040C3H	10000100B	Enables on-chip debugging

4.3 List of Constants

Table 4-2 Constants (1/2)and Table 4-3 Constants (2/2)lists the constants that are used in the sample code.

Table 4-2 Constants (1/2)

Constant Name	Setting Value	Description
_0xA0_LCM_SLAVE_ADDR	0xA0	Slave address for LCM commands
_0x00_LCM_SLAVE_ADDR_RW_LOW	0x00	Data write flag
_0x00_LCM_CONTROL_BYTE	0x00	Control byte for LCM commands
_0x80_LCM_CONTROL_BYTE_RS_HIGH	0x80	Control byte: Data transfer
_0x00_LCM_CONTROL_BYTE_RS_LOW	0x00	Control byte: Command transfer
_0x00_LCM_COMMAND_CLEAR_DISPLAY	0x00	Command: Clear Display
_0x04_LCM_COMMAND_ENTRY_MODE_SET	0x04	Command: Entry Mode Set
_0x02_LCM_COMMAND_ENTRY_MODE_SET_ID_HIGH	0x02	Entry Mode Set: Address increment
_0x00_LCM_COMMAND_ENTRY_MODE_SET_S_LOW	0x00	Entry Mode Set: Display shift OFF
_0x08_LCM_COMMAND_DISPLAY_ONOFF	0x08	Command: Display ON/OFF Control
_0x04_LCM_COMMAND_DISPLAY_ONOFF_D_HIGH	0x04	Display ON/OFF control: Display ON
_0x00_LCM_COMMAND_DISPLAY_ONOFF_C_LOW	0x00	Display ON/OFF control: Cursor display OFF
_0x00_LCM_COMMAND_DISPLAY_ONOFF_B_LOW	0x00	Display ON/OFF control: Cursor blinking OFF
_0x20_LCM_COMMAND_FUNCTION_SET	0x20	Command: Function Set
_0x10_LCM_COMMAND_FUNCTION_SET_DL_HIGH	0x10	Function Set: Mpu 8-bit bus mode
_0x08_LCM_COMMAND_FUNCTION_SET_N_HIGH	0x08	Function Set: 2-line display
_0x00_LCM_COMMAND_FUNCTION_SET_F_LOW	0x00	Function Set: 5x8-dot font
_0x80_LCM_COMMAND_SET_DDRAM_ADDRESS	0x80	Command: Set DDRAM Address

Table 4-3 Constants (2/2)

Constant Name	Setting Value	Description
LCM_COMMAND_EXEC_WAIT	26600	LCD module command execution wait time 5 ms (32 MHz operation)
LCM_CONFIG_FUNCTION_SET_PARAMS	0x18	Parameters for function set _0x10_LCM_COMMAND_FUNCTION_SET_DL_HIGH | _0x08_LCM_COMMAND_FUNCTION_SET_N_HIGH | _0x00_LCM_COMMAND_FUNCTION_SET_F_LOW
LCM_CONFIG_ENTRY_MODE_SET_PARAMS	0x20	Parameters for entry mode set _0x02_LCM_COMMAND_ENTRY_MODE_SET_ID_HIGH | _0x00_LCM_COMMAND_ENTRY_MODE_SET_S_LOW
LCM_CONFIG_DISPLAY_ONOFF_PARAMS	0x40	Display ON/OFF control command parameters _0x04_LCM_COMMAND_DISPLAY_ONOFF_D_HIGH | _0x00_LCM_COMMAND_DISPLAY_ONOFF_C_LOW | _0x00_LCM_COMMAND_DISPLAY_ONOFF_B_LOW
LCM_CONFIG_MAX_CHAR_PER_LINE	16	Maximum number of characters per line
LCM_CONFIG_WAIT_COUNT	13	IIIA0 wait count
LCM_POSITION_TOP	0x00	LCD module display line (top)
LCM_POSITION_BOTTOM	0x40	LCD module display line (bottom)
_01_VDD_DOWN	1	LVD0 (falling) generation flag value LVD0 (falling) generated
_00_VDD_NOT_DOWN	0	LVD0 (falling) generation flag value LVD0 (falling) not generated

4.4 List of Variables

Table 4-4 lists global variables.

Table 4-4 Global Variables

Type	Variable Name	Description	Function Used
int	g_vdd_down	LVD0 (falling) generation flag	r_set_vdd_down_flag r_is_vdd_down
uint8_t	g_LCM_is_sendend	I2C communication (with LCD module) end flag	r_LCM_init r_LCM_turn_sendend_on r_LCM_wait_sendend

4.5 List of Functions

Table 4-5 shows a list of functions.

Table 4-5 Functions

Function Name	Outline
r_main_user_init()	Main initial settings
BCD_to_str()	BCD value-to-character string conversion processing
r_is_vdd_over()	LVD0F flag judgment processing
r_set_vdd_down_flag()	LVD0 (falling) generation flag setting
r_is_vdd_down()	LVD0 (falling) generation judgment processing
r_RTC_wait()	HALT/STOP mode transition wait processing at the beginning of RTC operation
r_LCM_init()	LCD module Initialization.
r_LCM_clear()	LCD module display clear processing
r_LCM_send_string()	LCD module character string transmission processing
r_LCM_send_command()	LCD module command transmission processing
r_LCM_send_data()	LCD module data transmission processing
r_LCM_turn_sendend_on()	LCD module communication end flag setting
r_LCM_wait_sendend()	LCD module communication end wait processing
r_lvd_interrupt()	LVD0 interrupt processing.
r_Config_IICA0_callback_master_sendend()	IICA0 send end callback processing.
r_Config_IICA0_callback_master_error()	IICA0 error callback processing.

4.6 Specification of Functions

The function specifications of the sample code are shown below.

r_main_user_init()

Outline	Main initial settings
Header	r_cg_macrodriver.h、Config_LVD0.h、Config_RTC.h、LCM_driver.h
Declaration	static void r_main_user_init(void)
Description	Make initial settings for peripheral functions and peripheral devices used for applications.
Argument	None
Return Value	None

BCD_to_str()

Outline	BCD value-to-character string conversion processing
Header	r_cg_macrodriver.h
Declaration	void BCD_to_str(uint8_t BCD, uint8_t *str)
Description	Convert a 2-digit BCD value to a 2-digit character string.
Argument	uint8_t BCD: BCD value to be converted uint8_t *str: An area of at least two characters is required as a conversion destination
Return Value	None

r_is_vdd_over()	
Outline	LVD0F flag judgment processing
Header	r_cg_macrodriver.h、r_cg_userdefine.h
Declaration	int r_is_vdd_over(void)
Description	Reference the LVD0Fvalue and return the current V_{DD} state.
Argument	None
Return Value	1: VDD is LVD0 detection voltage (rising) or higher. 0: VDD is lower than the LVD0 detection voltage (falling).

r_set_vdd_down_flag()	
Outline	LVD0 (falling) generation flag setting
Header	r_cg_macrodriver.h、r_cg_userdefine.h
Declaration	void r_set_vdd_down_flag(void)
Description	Set _01_VDD_DOWN for g_vdd_down.
Argument	None
Return Value	None

r_is_vdd_down()	
Outline	LVD0 (falling) generation judgment processing
Header	r_cg_macrodriver.h、r_cg_userdefine.h
Declaration	int r_is_vdd_down(void)
Description	Reference g_vdd_down and return the LVD0 (falling) interrupt state. g_vdd_down is cleared by _00_VDD_NOT_DOWN.
Argument	None
Return Value	1: An LVD0 (falling) interrupt was generated. 0: An LVD0 (falling) interrupt is not generated.

r_RTC_wait()	
Outline	HALT/STOP mode transition wait processing at the beginning of RTC operation
Header	r_cg_macrodriver.h、Config_RTC.h
Declaration	void r_RTC_wait(void)
Description	Perform the processing for transition to HALT/STOP mode immediately after RTC operation starts.
Argument	None
Return Value	None

r_LCM_init()	
Outline	LCD module Initialization.
Header	LCM_driver.h、Config_IICA0.h
Declaration	void r_LCM_init(void)
Description	Initializes the LCD module.
Argument	None
Return Value	None

r_LCM_clear()	
Outline	LCD module display clear processing
Header	LCM_driver.h、Config_IICA0.h
Declaration	void r_LCM_clear(void)
Description	Send the clear display command to the LCD module to clear display.
Argument	None
Return Value	None

r_LCM_send_string()	
Outline	LCD module character string transmission processing
Header	LCM_driver.h、Config_IICA0.h
Declaration	void r_LCM_send_string(uint8_t * const str, lcm_position_t pos)
Description	Display the character string transferred with str on the LCD module. The line to display the character string is specified by pos.
Argument	uint8_t * const str: Character string to be displayed lcm_position_t pos: LCM_POSITION_TOP: Displayed at the top LCM_POSITION_BOTTOM: Displayed at the bottom
Return Value	None

r_LCM_send_command()	
Outline	LCD module command transmission processing
Header	LCM_driver.h、Config_IICA0.h
Declaration	void r_LCM_send_command(uint8_t command)
Description	Send the command transferred with the command to the LCD module.
Argument	uint8_t command: Command to be sent to the LCD module
Return Value	None

r_LCM_send_data()	
Outline	LCD module data transmission processing
Header	LCM_driver.h、Config_IICA0.h
Declaration	void r_LCM_send_data(uint8_t data);
Description	Send the data transferred with the data to the LCD module.
Argument	uint8_t data: Data to be sent to the LCD module
Return Value	None

r_LCM_turn_sendend_on()	
Outline	LCD module communication end flag setting
Header	LCM_driver.h、Config_IICA0.h
Declaration	void r_LCM_turn_sendend_on(void)
Description	Set the I2C communication (with the LCD module) end flag for g_LCM_is_sendend.
Argument	None
Return Value	None

r_LCM_wait_sendend()	
Outline	LCD module communication end wait processing
Header	LCM_driver.h、Config_IICA0.h
Declaration	static void r_LCM_wait_sendend(void)
Description	Wait until the I2C communication (with the LCD module) ends, and then perform wait processing during the command execution wait time period (5 ms).
Argument	None
Return Value	None

r_lvd_interrupt()	
Outline	LVD0 interrupt processing.
Header	r_cg_macrodriver.h、Config_LVD0.h、Config_RTC.h
Declaration	static void r_lvd_interrupt(void)
Description	Judge the VDD state and set/reset the INTRTC interrupt mask. If VDD is lower than LVD0 (falling), set the voltage detection interrupt flag.
Argument	None
Return Value	None

r_Config_IICA0_callback_master_sendend()	
Outline	IICA0 send end callback processing.
Header	r_cg_macrodriver.h、Config_IICA0.h、LCM_driver.h
Declaration	static void r_Config_IICA0_callback_master_sendend(void)
Description	This callback function is called upon completion of IICA0 transmission Generate stop conditions, and then call the LCD module communication end flag setting function.
Argument	None
Return Value	None

r_Config_IICA0_callback_master_error()	
Outline	IICA0 error callback processing.
Header	r_cg_macrodriver.h、Config_IICA0.h、LCM_driver.h
Declaration	static void r_Config_IICA0_callback_master_error(MD_STATUS flag)
Description	This callback function is called when an IICA0 error occurs. Call the LCD module communication end flag setting function.
Argument	MD_STATUS flag : error type
Return Value	None

4.7 Flowcharts

4.7.1 Main Processing

Figure 4-1and Figure 4-2 show flowcharts of the main processing.

Figure 4-1 Main Processing (1/2)

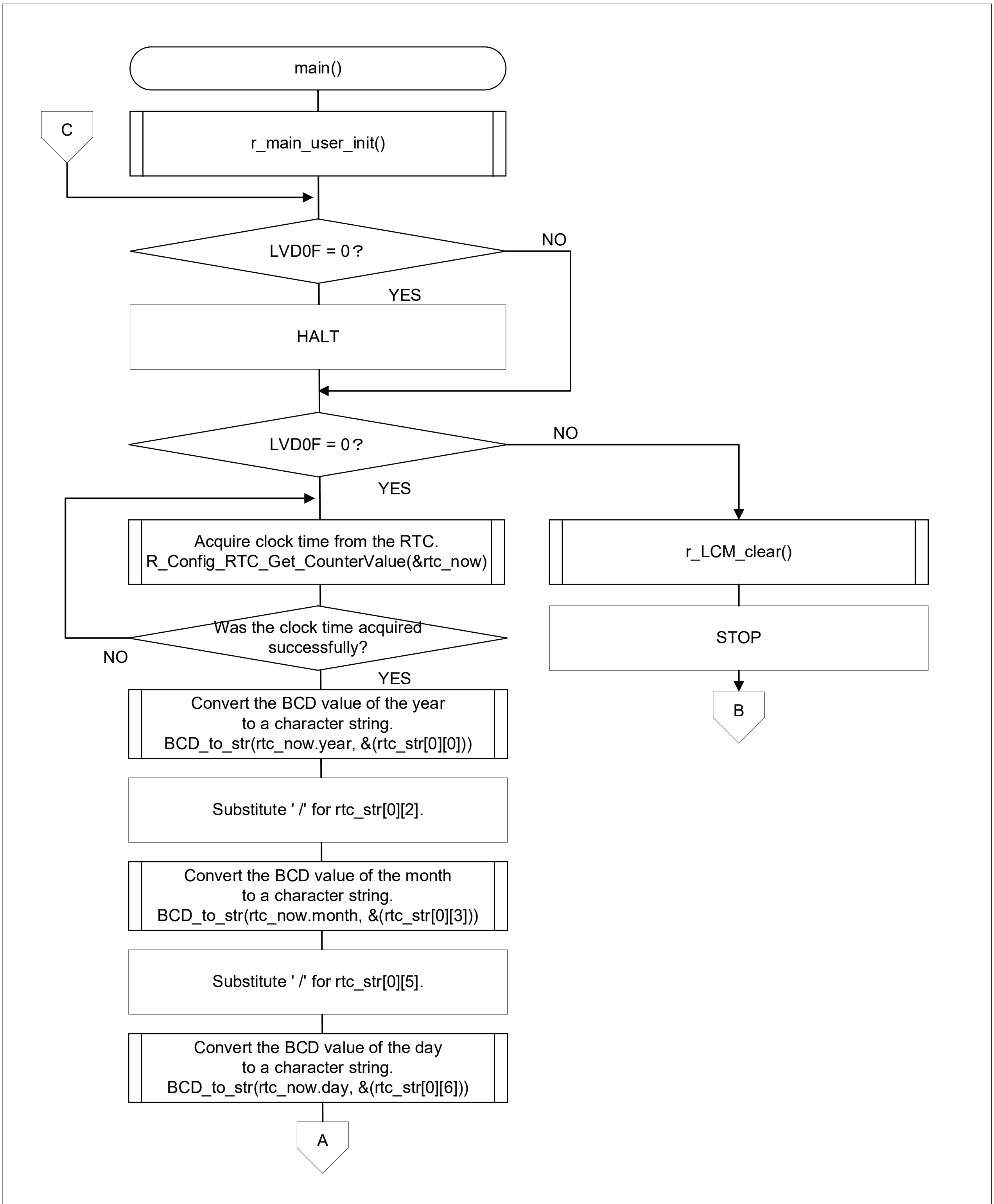
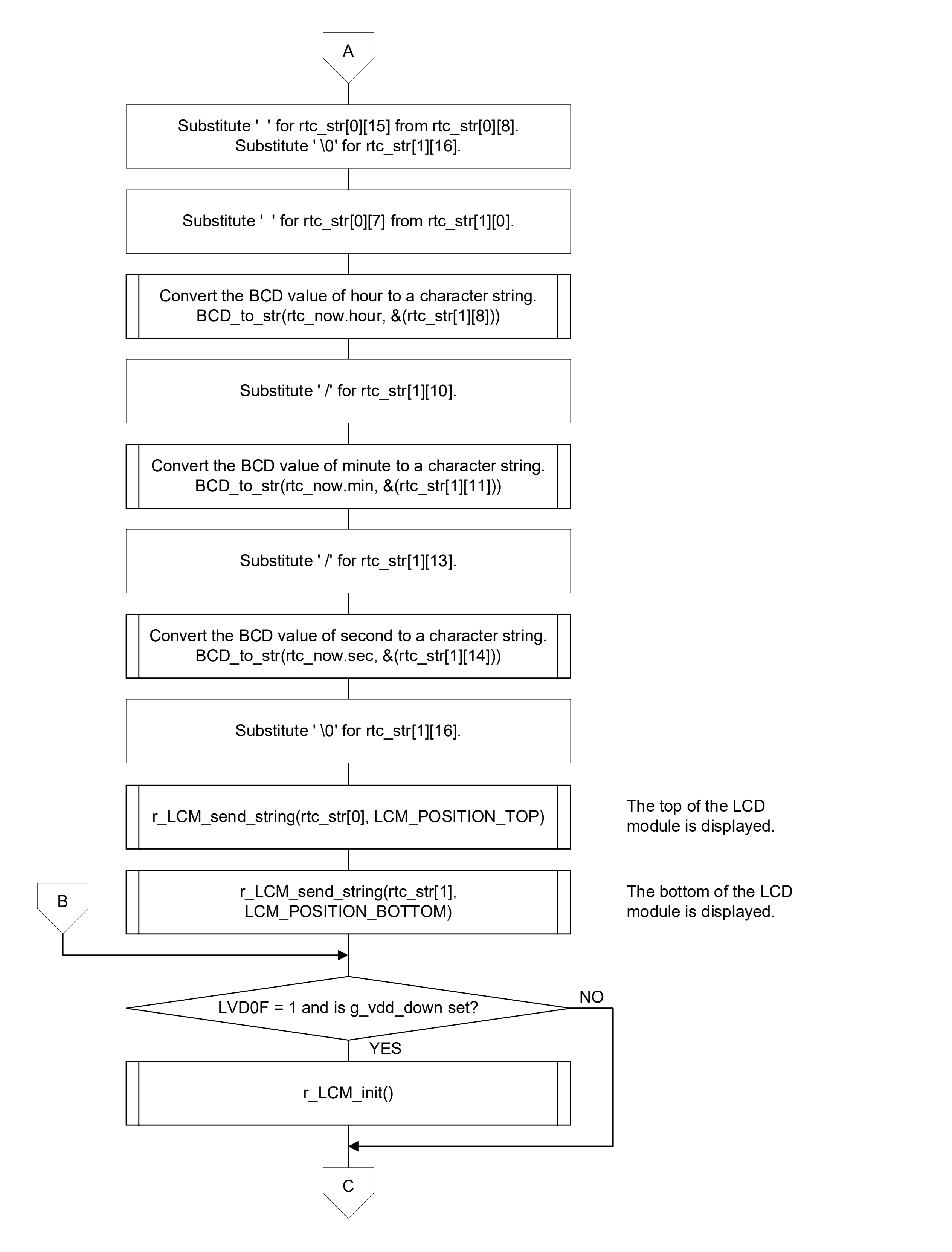


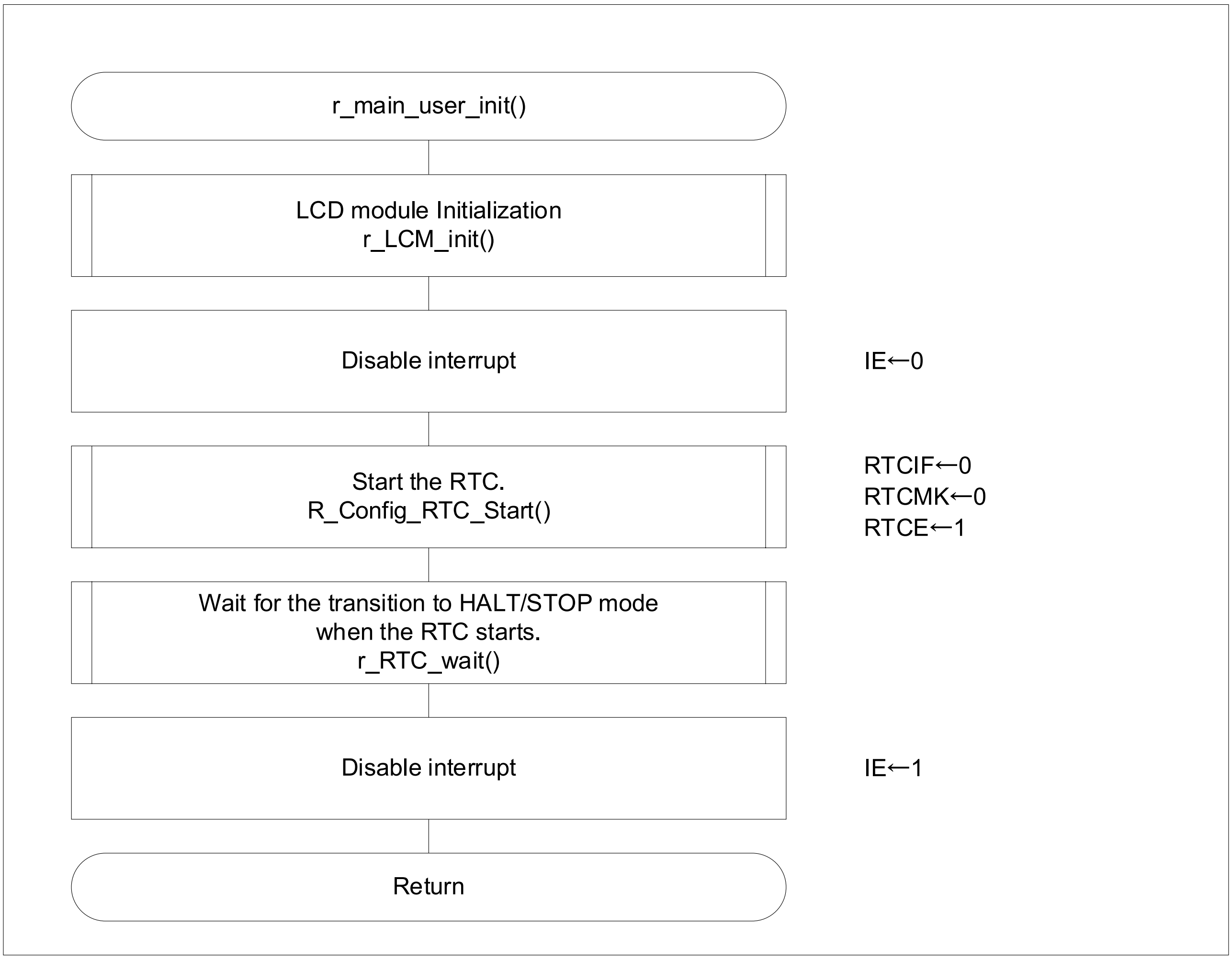
Figure 4-2 Main Processing (2/2)



4.7.2 Main Initial Settings

Figure 4-3 shows the flowchart of the initial settings for main functions.

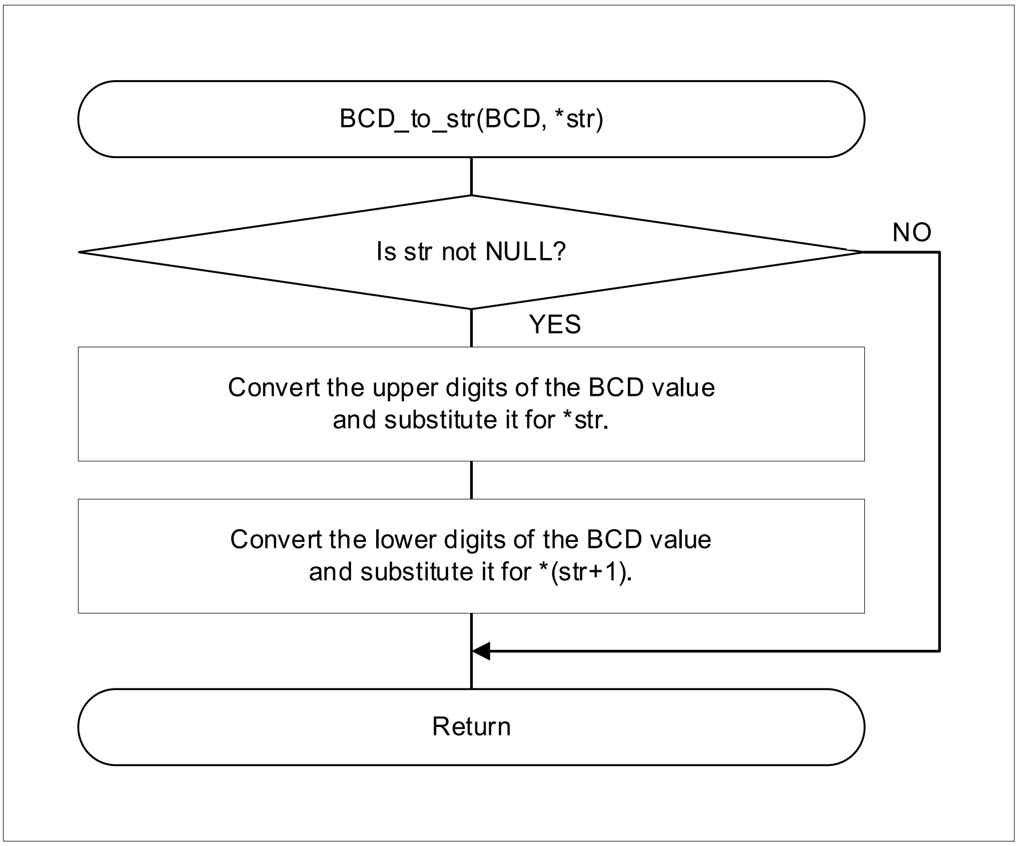
Figure 4-3 Main Initial Settings



4.7.3 BCD Value-to-Character String Conversion Processing

Figure 4-4 shows the flowchart of the processing for converting a BCD value to a character string.

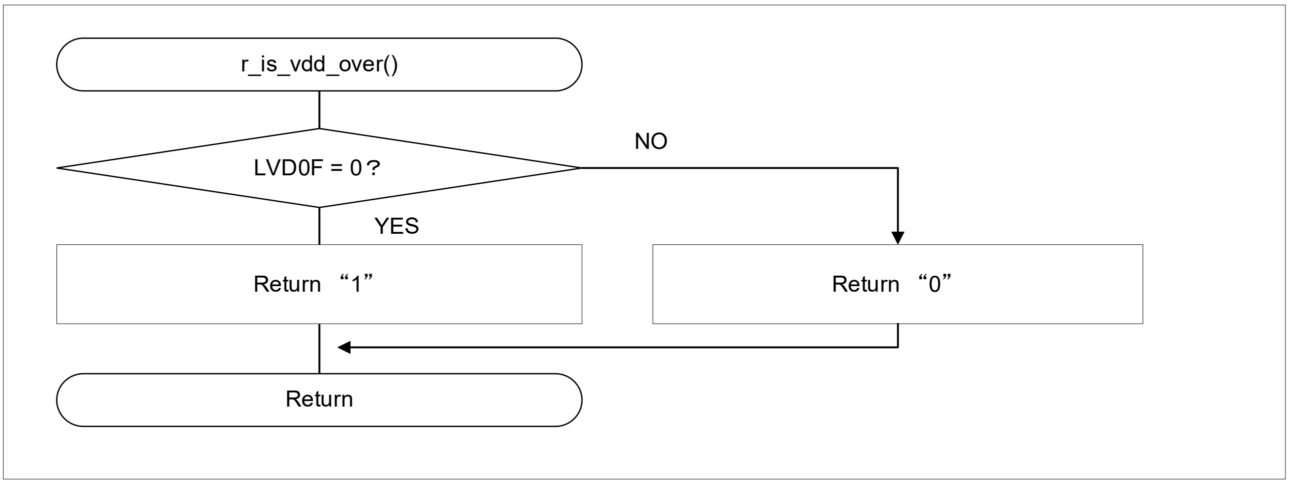
Figure 4-4 BCD Value-to-Character String Conversion Processing



4.7.4 LVD0F Flag Judgment Processing

Figure 4-5 shows the flowchart of the LVD0F flag judgment processing.

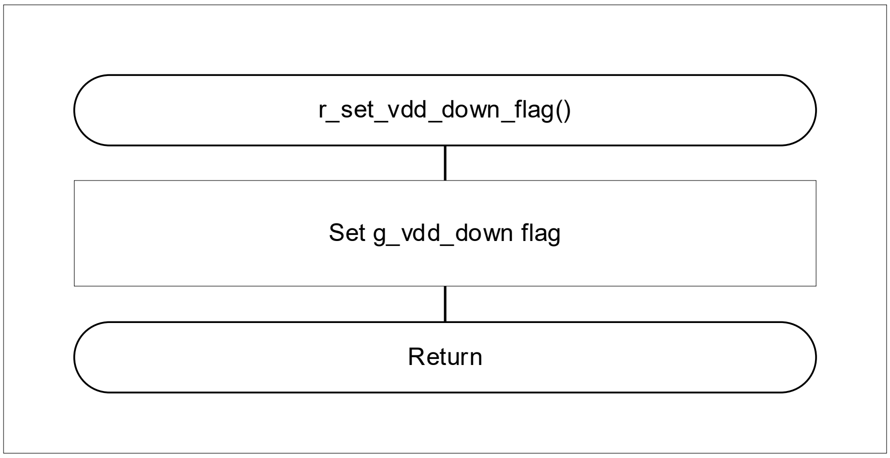
Figure 4-5 LVD0F Flag Judgment Processing



4.7.5 LVD0 (Falling) Generation Flag Setting

Figure 4-6 shows the flowchart for setting the LVD0 (falling) generation flag.

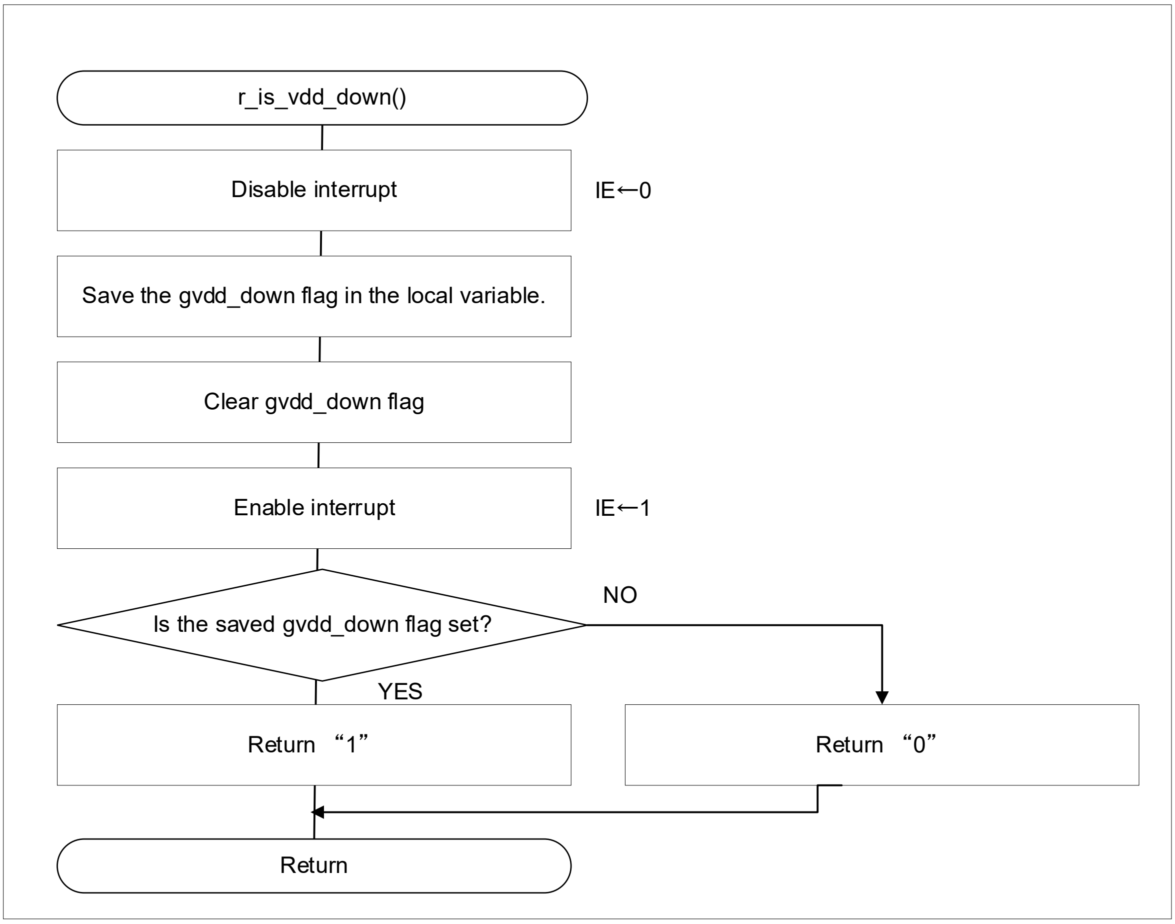
Figure 4-6 LVD0 (Falling) Generation Flag Setting



4.7.6 LVD0 (Falling) Generation Judgment Processing

Figure 4-7 shows the flowchart of the LVD0 (Falling) generation judgment processing.

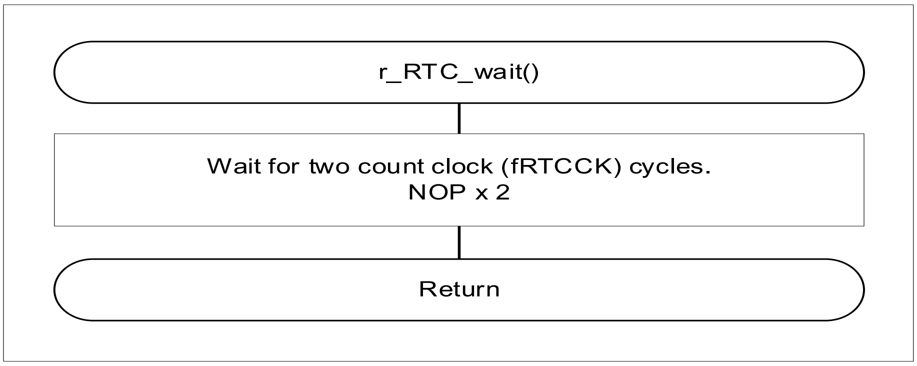
Figure 4-7 LVD0 (Falling) Generation Judgment Processing



4.7.7 HALT/STOP Mode Transition Wait Processing at the Beginning of RTC Operation

Figure 4-8 shows the flowchart of the HALT/STOP mode transition wait processing at the beginning of RTC operation.

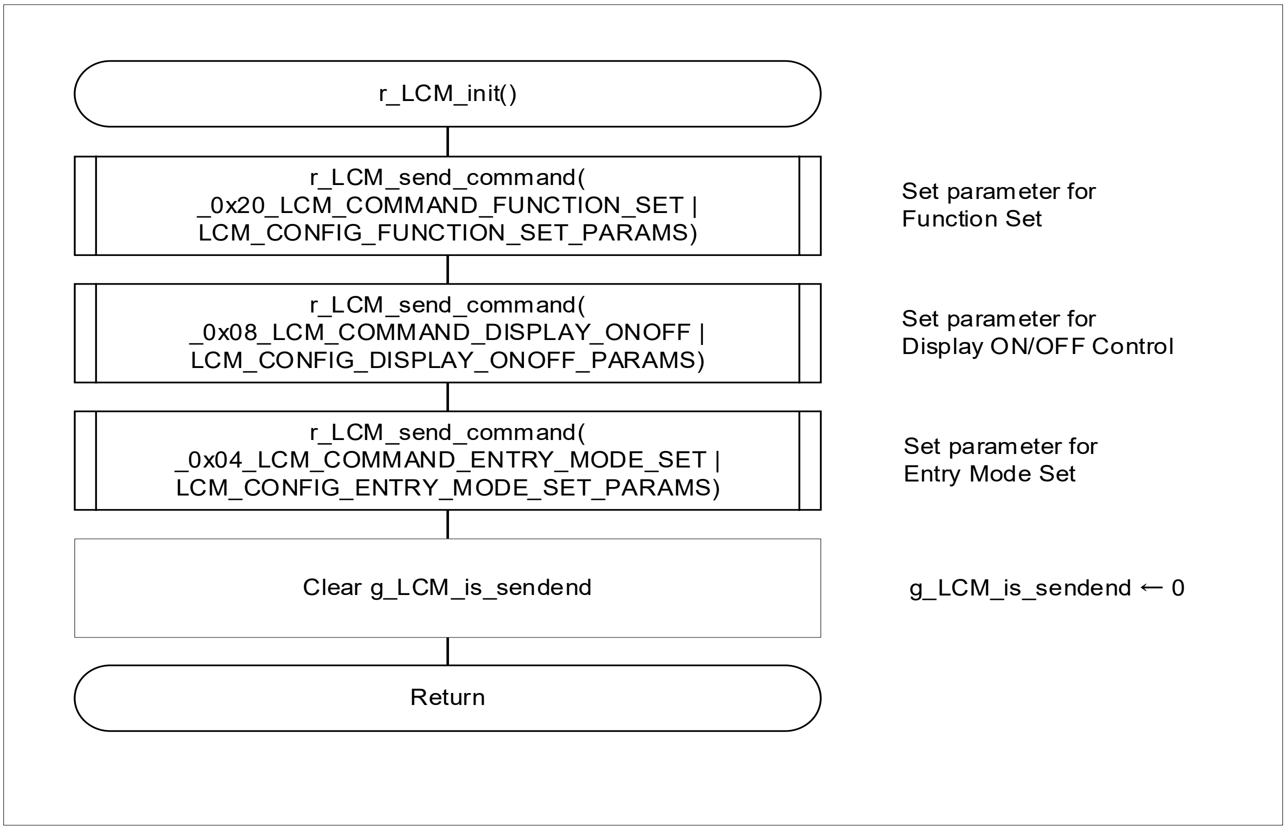
Figure 4-8 HALT/STOP Mode Transition Wait Processing at the Beginning of RTC Operation



4.7.8 LCD Module Initialization

Figure 4-9 shows the flowchart of the LCD module initialization.

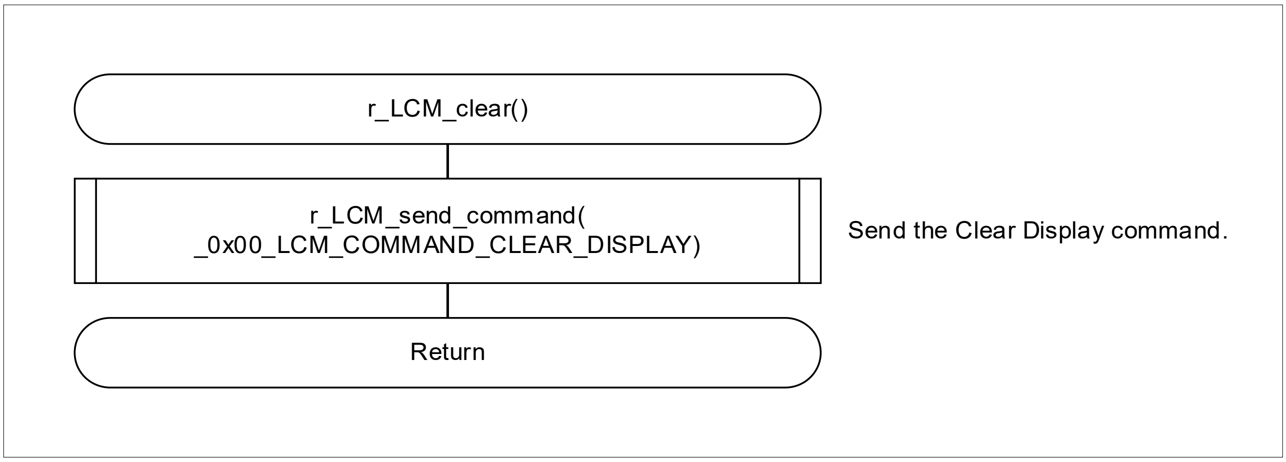
Figure 4-9 LCD Module Initialization



4.7.9 LCD Module Display Clear Processing

Figure 4-10 shows the flowchart of the display clear processing for the LCD module.

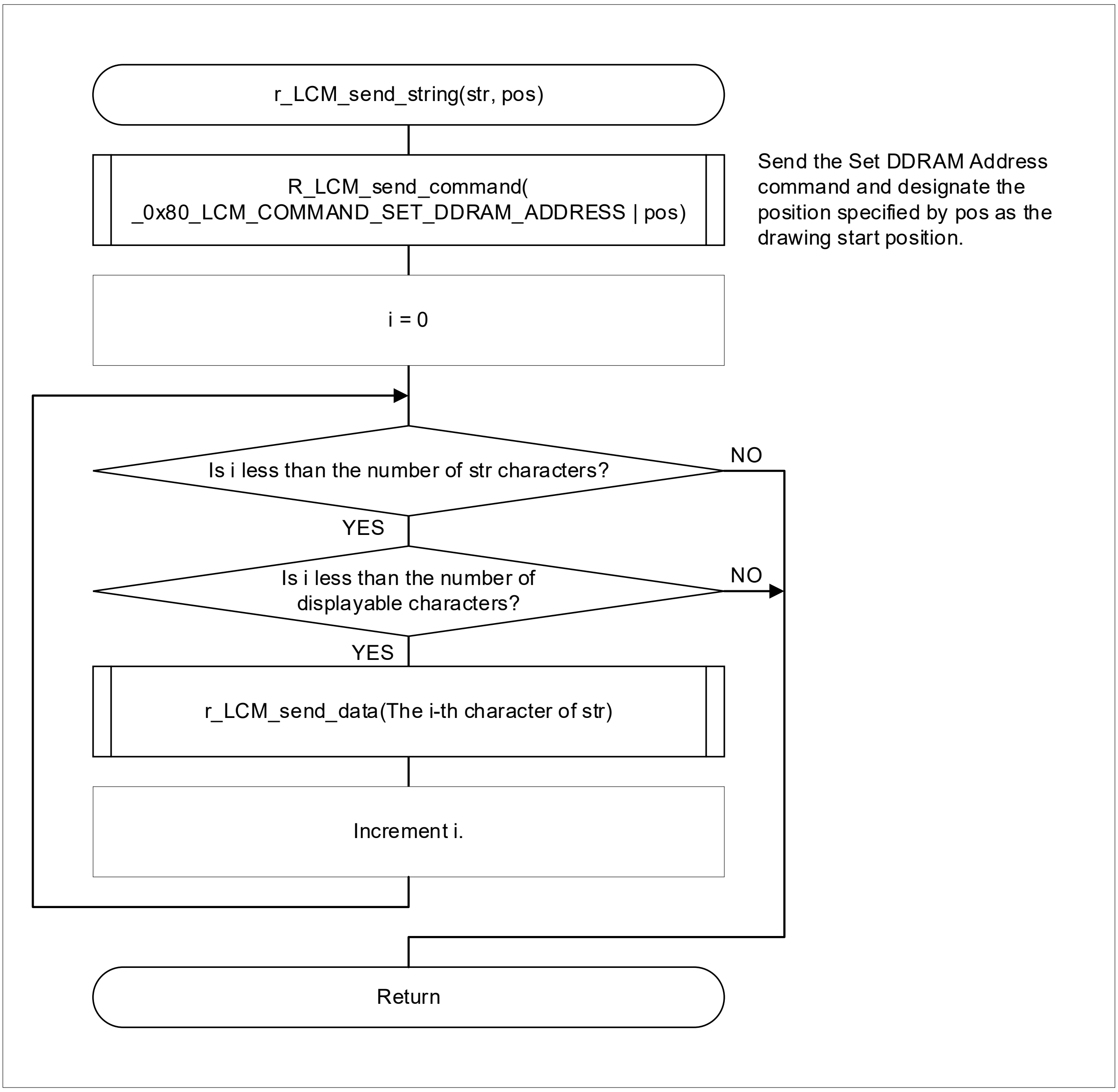
Figure 4-10 LCD Module Display Clear Processing



4.7.10 LCD Module Character String Transmission Processing

Figure 4-11 shows the flowchart of the character string transmission processing for the LCD module.

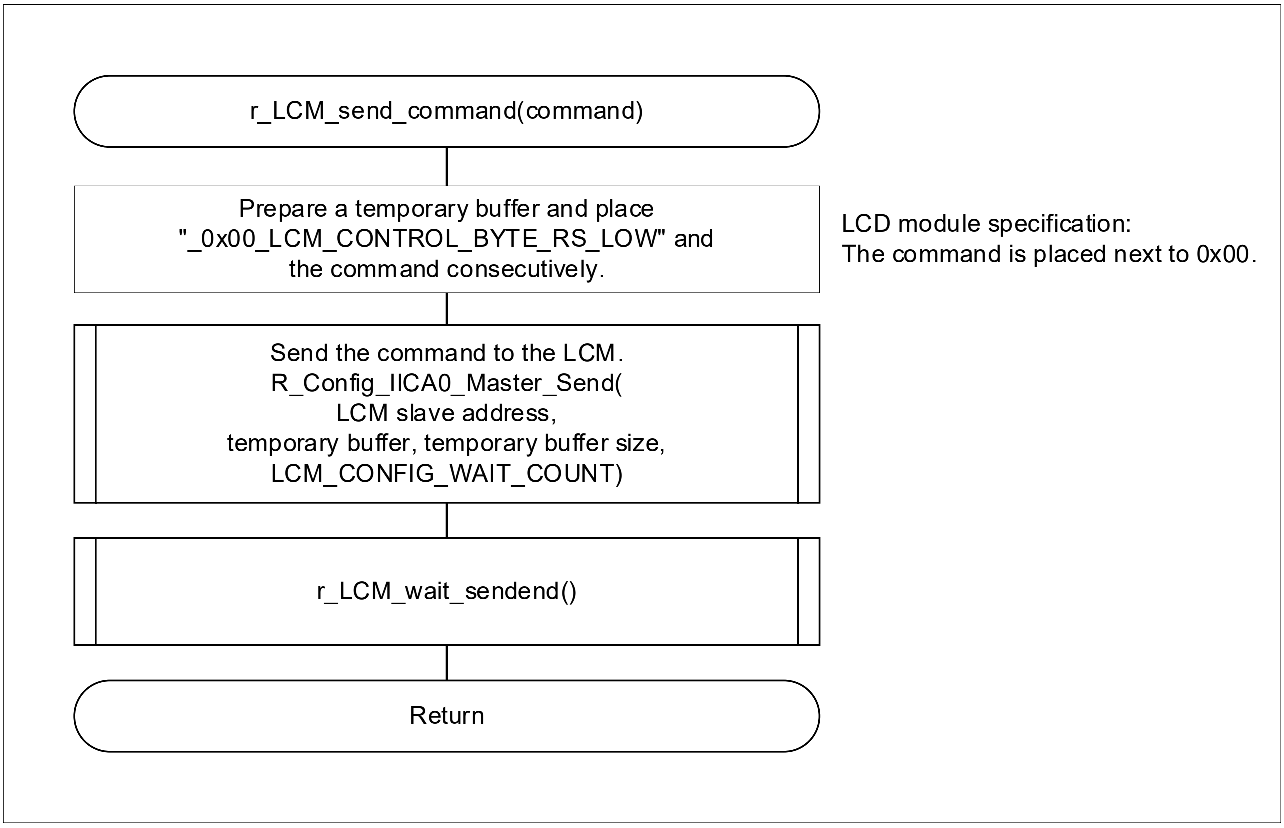
Figure 4-11 LCD Module Character String Transmission Processing



4.7.11 LCD Module Command Transmission Processing

Figure 4-12 shows the flowchart of the command transmission processing for the LCD module.

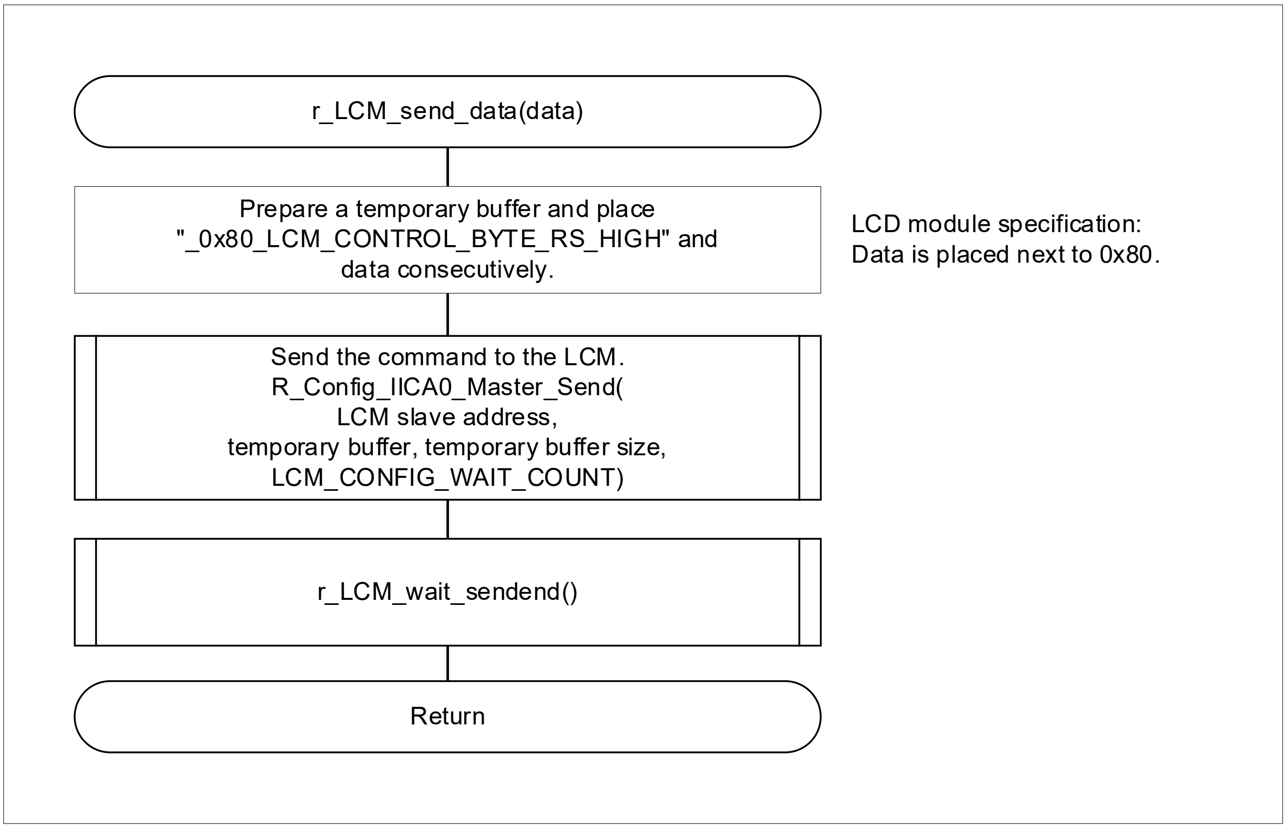
Figure 4-12 LCD Module Command Transmission Processing



4.7.12 LCD Module Data Transmission Processing

Figure 4-13 shows the flowchart of the data transmission processing for the LCD module.

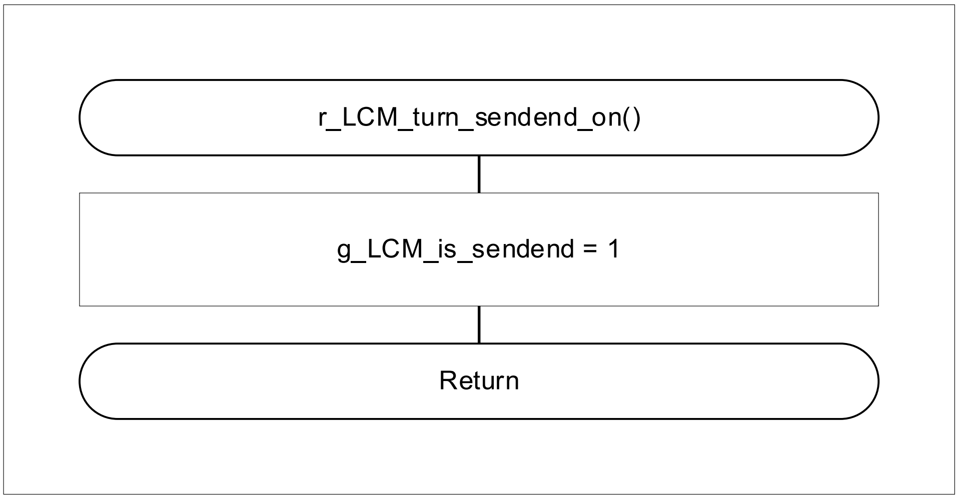
Figure 4-13 LCD Module Data Transmission Processing



4.7.13 LCD Module Communication End Flag Setting

Figure 4-14 shows the flowchart for setting the communication end flag for the LCD module.

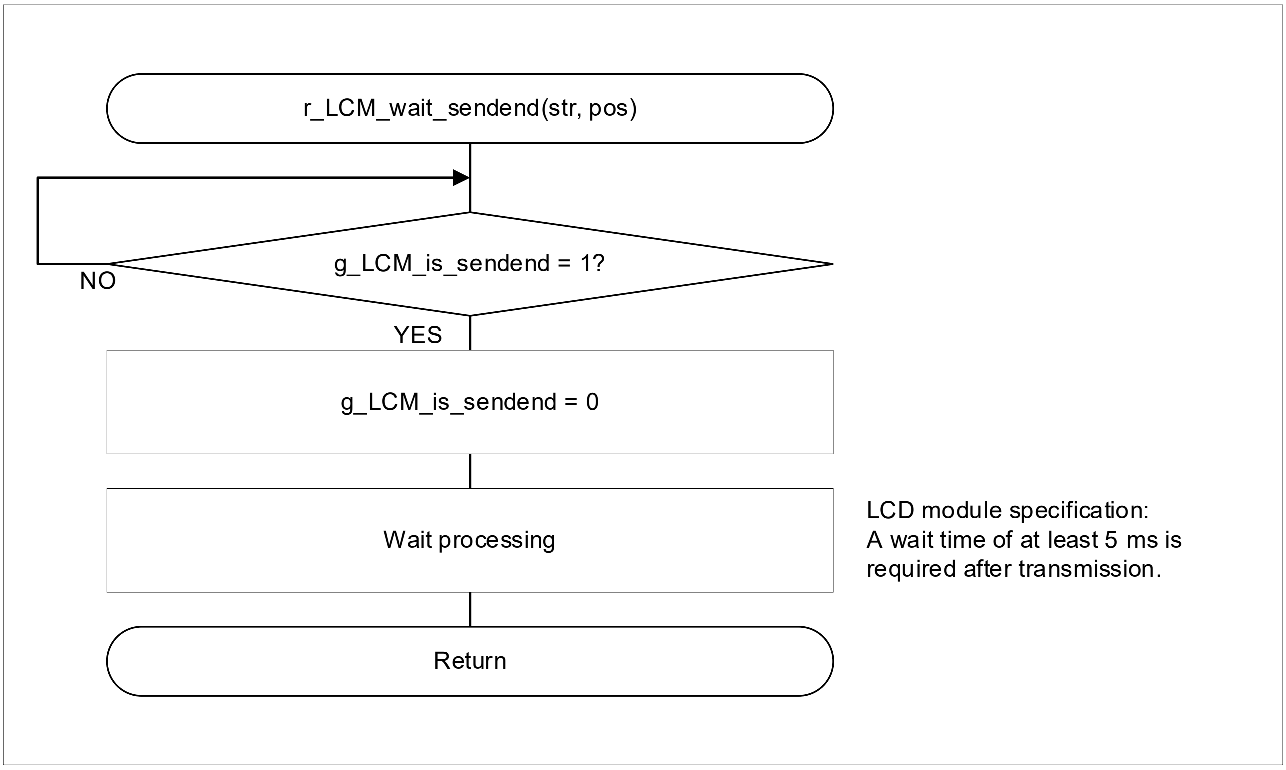
Figure 4-14 LCD Module Communication End Flag Setting



4.7.14 LCD Module Communication End Wait Processing

Figure 4-15 shows the flowchart of the communication end wait processing for the LCD module.

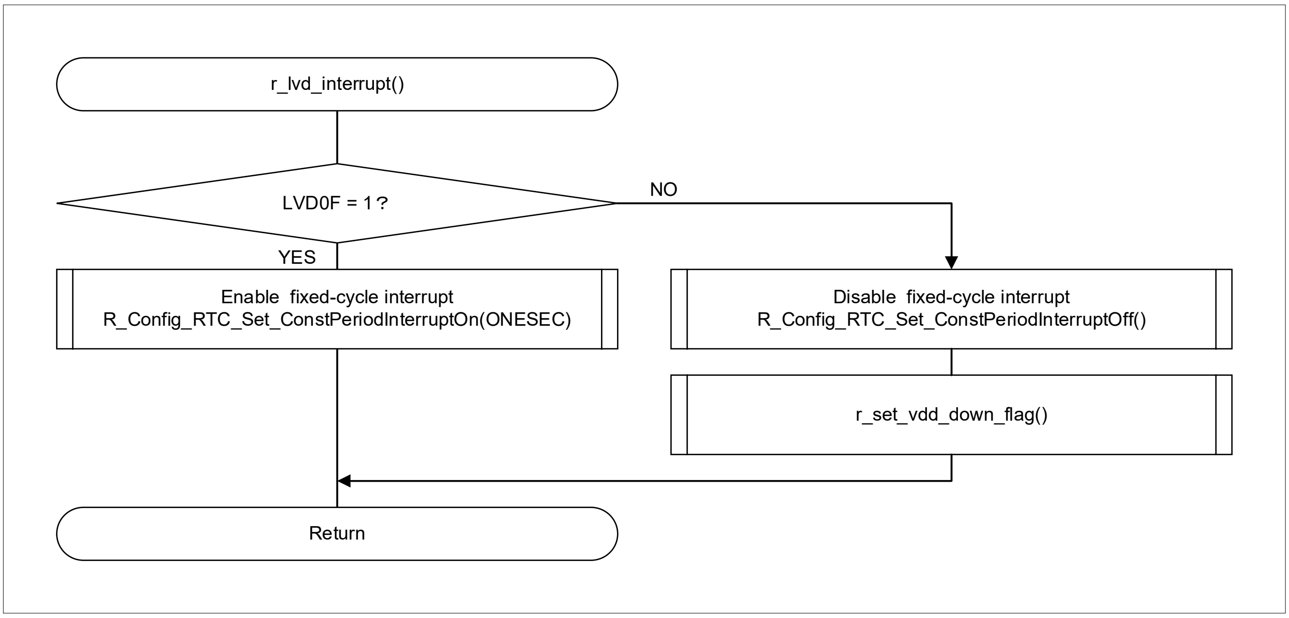
Figure 4-15 LCD Module Communication End Wait Processing



4.7.15 LVD0 Interrupt Processing

Figure 4-16 shows the flowchart of the LVD0 interrupt processing.

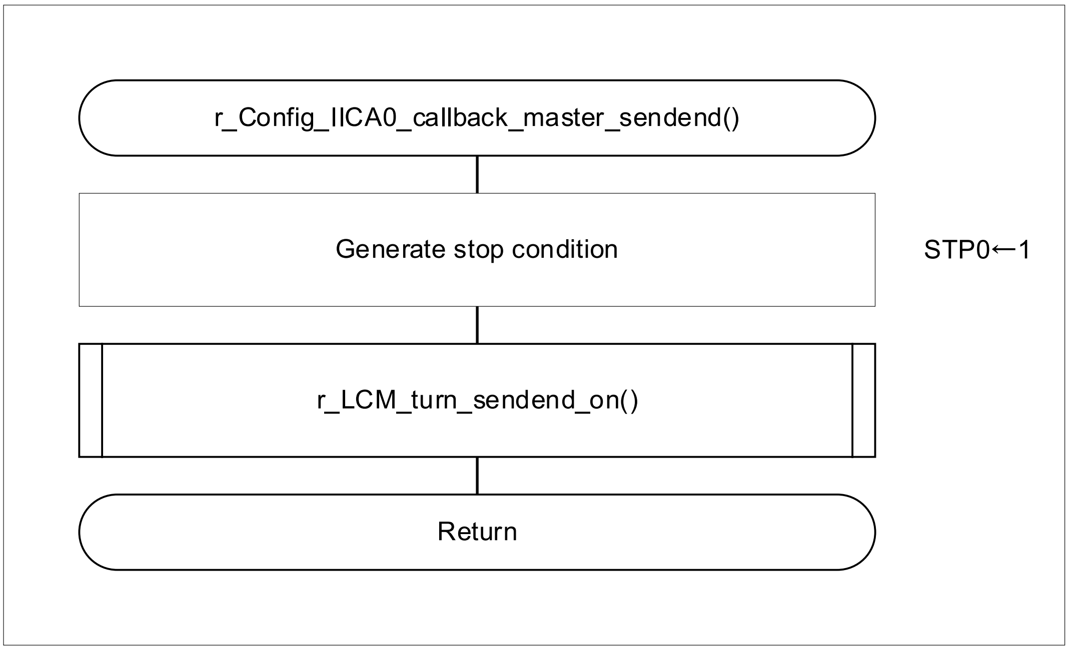
Figure 4-16 LVD0 Interrupt Processing



4.7.16 IICA0 Send End Callback Processing

Figure 4-17 shows the flowchart of the IICA0 send end callback processing.

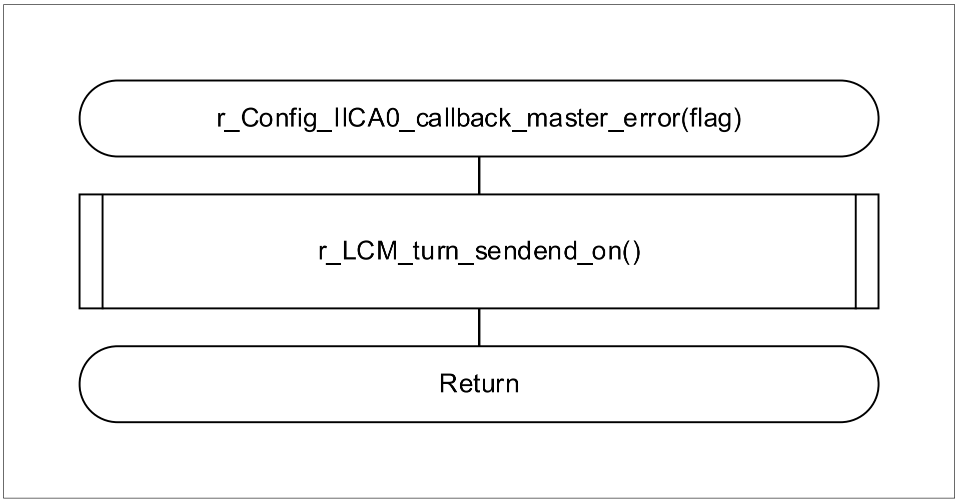
Figure 4-17 IICA0 Send End Callback Processing



4.7.17 IICA0 Error Callback Processing

Figure 4-18 shows the flowchart of the IICA0 error callback processing.

Figure 4-18 IICA0 Error Callback Processing



5. Sample Code

Sample code can be downloaded from the Renesas Electronics website.

6. Reference Documents

RL78/G23 User’s Manual: Hardware (R01UH0896)
RL78 family user's manual software (R01US0015)
The latest versions can be downloaded from the Renesas Electronics website.

Technical update
The latest versions can be downloaded from the Renesas Electronics website.
LCD module datasheet

(ACM1602NI-FLW-FBW-M01 (ZETTLER DISPLAYS) CHARACTER MODULE VER1.4)

Revision History

Rev.	Date	Description	
		Page	Summary
1.00	2021.04.13	-	First Edition
1.01	2021.07.12	6	Updated the Operation Confirmation Conditions
2.00	2022.06.17	1, 3, 5, 11-13, 15, 19, 20, 27	Changed LVD1 to LVD0
		4, 9	Add a description
		7	Updated operation check conditions

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.

Corporate Headquarters

TOYOSU FORESIA, 3-2-24 Toyosu,
Koto-ku, Tokyo 135-0061, Japan
www.renesas.com

Trademarks

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/.