RL78/G12

R01AN2987EJ0100
Rev. 1.00
Oct. 20, 2015

Serial Interface IICA (for Master Transmission/Reception) CC-RL

Introduction

This application note describes master transmission and reception implemented via serial interface IICA. Using IICA, the single master system described here performs master operation (address transmission, data transmission and data reception).

Target Device

RL78/G12

When applying the sample program covered in this application note to another microcomputer, modify the program according to the specifications for the target microcomputer and conduct an extensive evaluation of the modified program.

Contents

1. Specifications

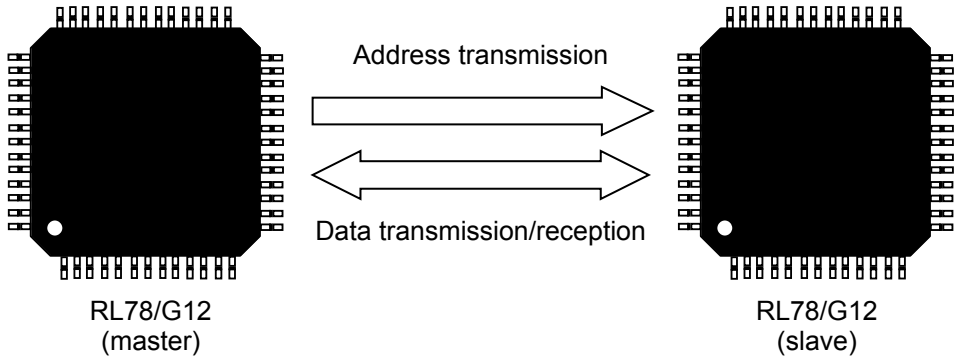
This application note describes how the single master system performs master transmission and reception (address transmission, data transmission and data reception) through serial interface IICA.

Table 1.1 lists the peripheral function to be used and its use. Figure 1.1 presents an overview of IIC communication.

Figures 1.2 to 1.8 show timing charts for explaining the IIC communication.

Table 1.1 Peripheral Function to be Used and Its Use

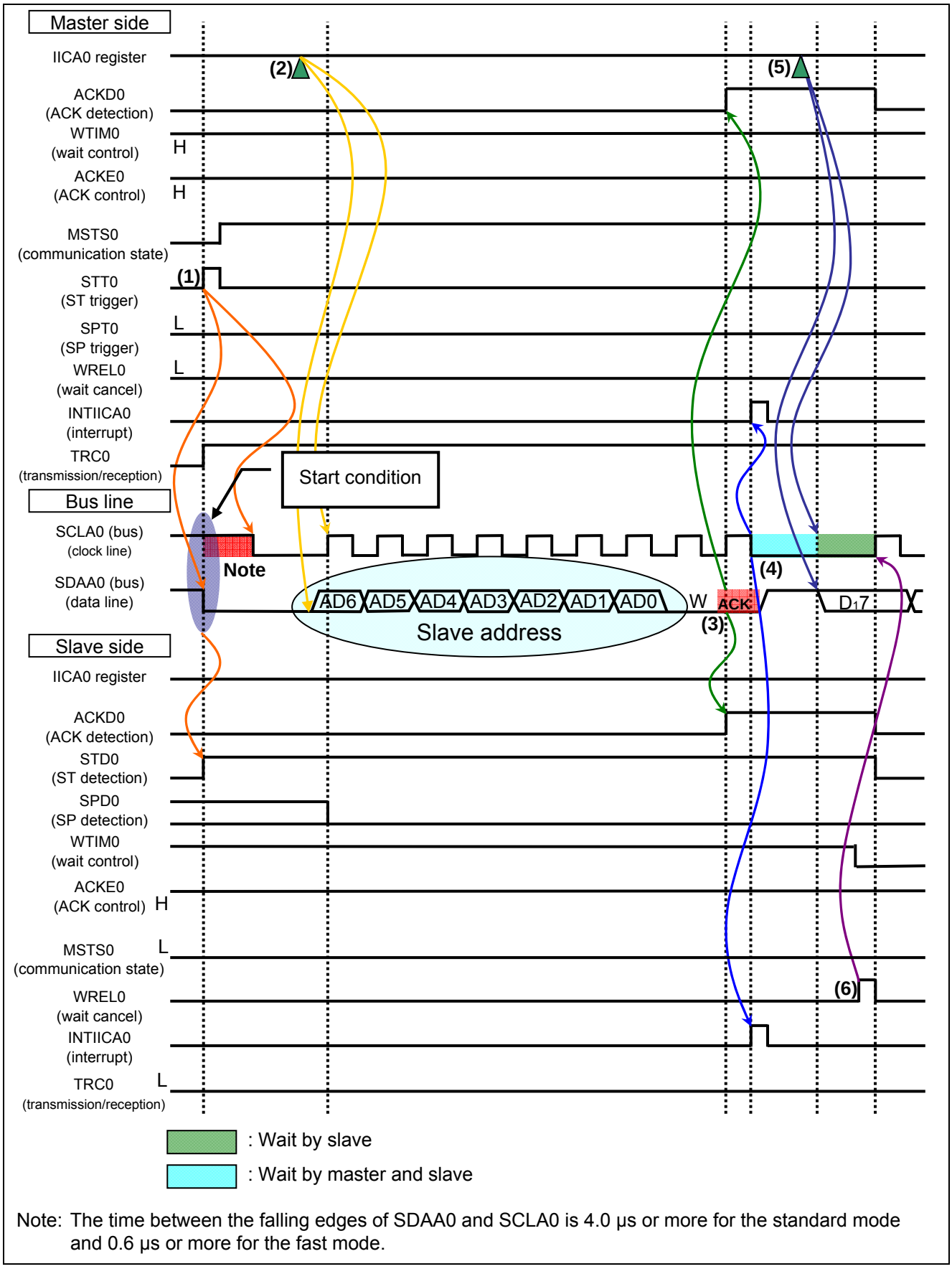
Peripheral Function	Use
Serial interface IICA	IIC master transmission/reception in a single master system (using the SCLA0 and SDAA0 pins)



Address transmission: Initial action of the master in IIC communication. This action is intended to identify which slave to transfer data to. After generating a start condition, the master sends one byte of data consisting of 7 bits (indicating the address) and 1 bit (indicating the transfer direction).
Data transmission/reception: Sending/receiving of data to/from the destination slave after address transmission. After completion of data transmission/reception, the master generates a stop condition and releases the bus.

Figure 1.1 Overview of IIC Communication

(1) Master-to-slave communication 1 (start condition – address – data)



Note: The time between the falling edges of SDAA0 and SCLA0 is 4.0 µs or more for the standard mode and 0.6 µs or more for the fast mode.

Figure 1.2 IIC Communication Timing Chart (Master-to-Slave Communication Example) (1/4)

(1) The start condition trigger is set (STT0 = 1) on the master side. Then, the SDAA0 line falls, thereby generating a start condition. Later, when the start condition is detected (STD0 = 1), the master enters a master device communication state (MSTS0 = 1). The SCLA0 line falls at the end of the hold period. This completes preparations for communication.

(2) The values of the address and data direction bit W (transmission) are written to the IICA0 register on the master side. Then, the slave address is transmitted.

(3) If the received address and slave address match [Note], the slave hardware sends ACK0 to the master. When the ninth clock signal rises, the master detects ACK (ACKD0 = 1).

(4) When the ninth clock signal falls, an address transmission end interrupt (INTIICA0) occurs on the master side. If the addresses match, an address match interrupt (INTIICA0) occurs on the slave side. Both the master and the slave which has the matching address generate a wait (SCLA0 line: Low) [Note].

(5) The master writes transmit data to the IICA0 register and cancels the wait.

(6) The slave selects an 8-clock wait (WTIM0 = 0) because it receives data. When the slave cancels the wait (WREL0 = 1), the master starts transferring data to the slave.

Note: If the sent address and slave address do not match, the slave does not return ACK to the master (NACK). The INTIICA0 interrupt (address match interrupt) does not occur on the slave side and thus the slave does not enter a wait state. However, the INTIICA0 interrupt (address transmission end interrupt) occurs on the master side regardless of whether the master receives ACK or NACK.

(2) Master-to-slave communication 2 (address – data – data)

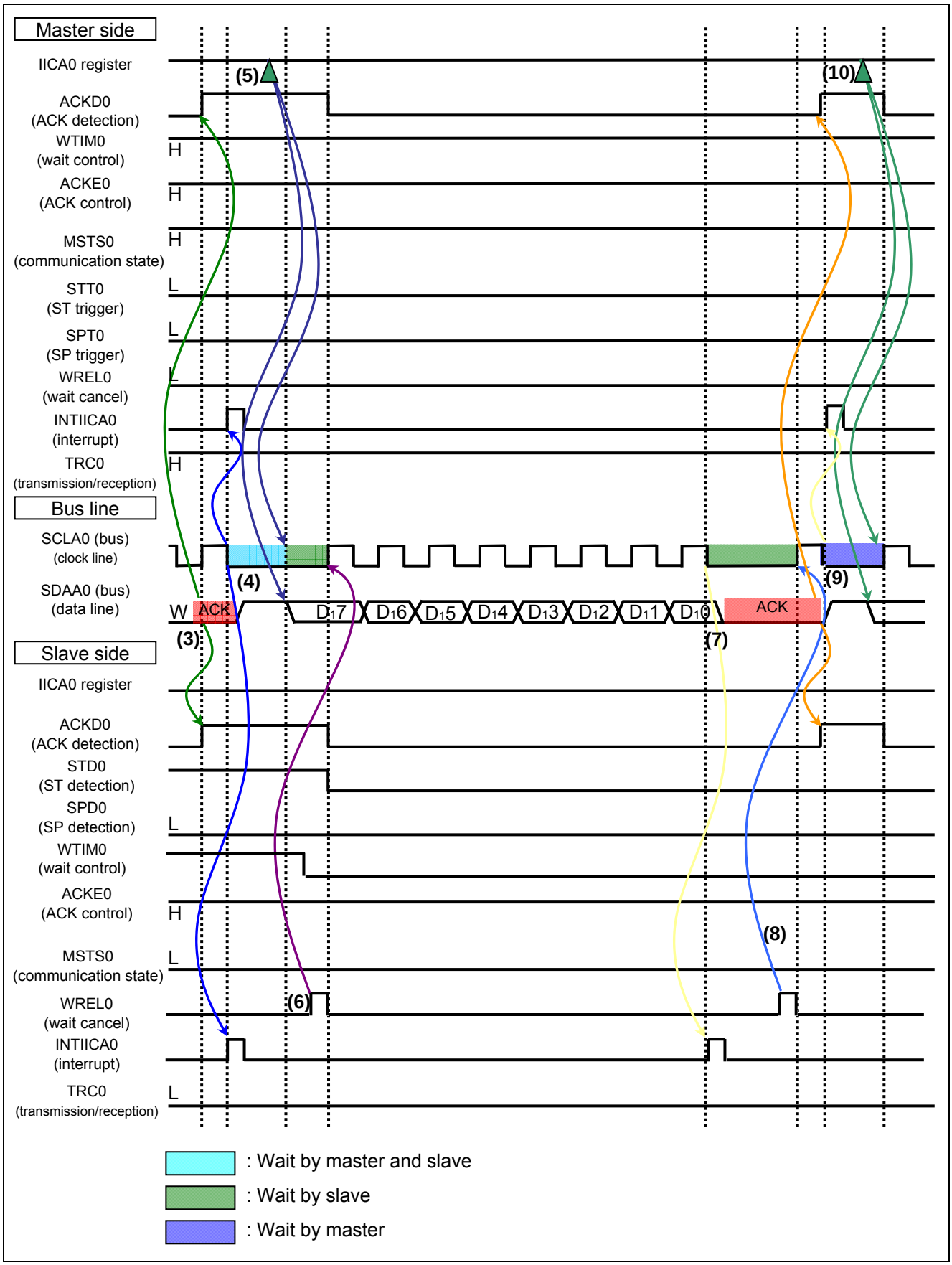
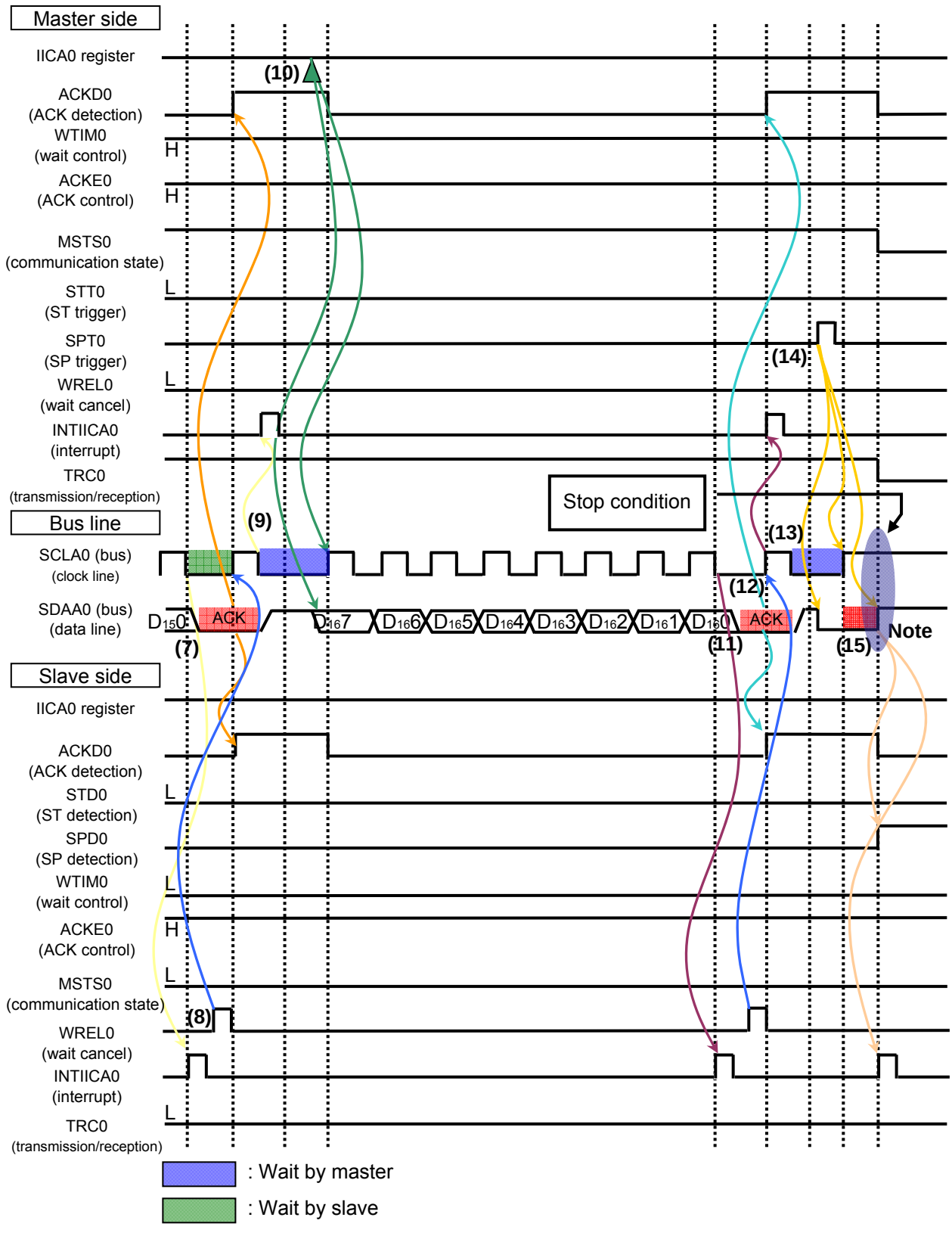


Figure 1.3 IIC Communication Timing Chart (Master-to-Slave Communication Example) (2/4)

(3) If the received address and slave address match [Note], the slave hardware sends ACK to the master. When the ninth clock signal rises, the master detects ACK (ACKD0 = 1).
(4) When the ninth clock signal falls, an address transmission end interrupt (INTIICA0) occurs on the master side. If the addresses match, an address match interrupt (INTIICA0) occurs on the slave side. Both the master and the slave which has the matching address generate a wait (SCLA0 line: Low).
(5) The master writes transmit data to the IICA0 register and cancels the wait.
(6) The slave selects an 8-clock wait (WTIM0 = 0) because it receives data. When the slave cancels the wait (WREL0 = 1), the master starts transferring data to the slave.
(7) When the eighth clock signal falls after the data transfer, the slave hardware generates a wait (SCLA0 line: Low) and a transfer end interrupt (INTIICA0) occurs on the slave side.
(8) When the slave reads the receive data and cancels the wait (WREL0 = 1), the slave sends ACK to the master. When the ninth clock signal rises, the master detects ACK (ACKD0 = 1).
(9) When the ninth clock signal falls, the master generates a wait (SCLA0 line: Low) and a transfer end interrupt (INTIICA0) occurs on the master side.
(10) The master writes transmit data to the IICA0 register and cancels the wait. Then, the master starts transferring data to the slave.

Note: If the sent address and slave address do not match, the slave does not return ACK to the master (NACK). The INTIICA0 interrupt (address match interrupt) does not occur on the slave side and thus the slave does not enter a wait state. However, the INTIICA0 interrupt (address transmission end interrupt) occurs on the master side regardless of whether the master receives ACK or NACK.

(3) Master-to-slave communication 3 (data – data – stop condition)

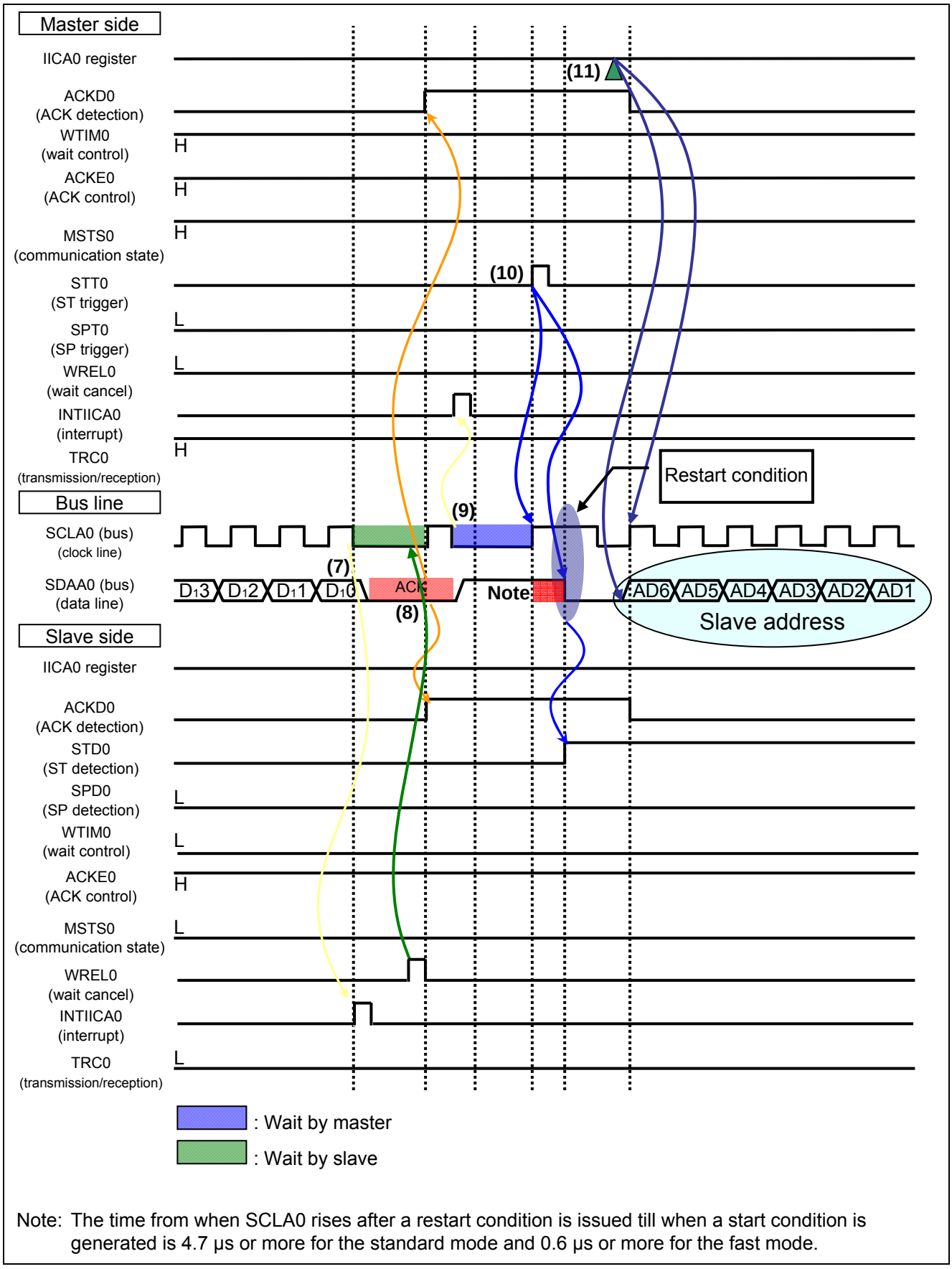


Note: The time from when SCLA0 rises after a stop condition is issued till when a stop condition is generated is 4.0 µs or more for the standard mode and 0.6 µs or more for the fast mode.

Figure 1.4 IIC Communication Timing Chart (Master-to-Slave Communication Example) (3/4)

(7) When the eighth clock signal falls after the data transfer, the slave hardware generates a wait (SCLA0 line: Low) and a transfer end interrupt (INTIICA0) occurs on the slave side.
(8) When the slave reads the receive data and cancels the wait (WREL0 = 1), the slave sends ACK to the master. When the ninth clock signal rises, the master detects ACK (ACKD0 = 1).
(9) When the ninth clock signal falls, the master generates a wait (SCLA0 line: Low) and an address transmission end interrupt (INTIICA0) occurs on the master side.
(10) The master writes transmit data to the IICA0 register and cancels the wait. Then, the master starts transferring the data to the slave.
(11) When the eighth clock signal falls after the data transfer, the slave hardware generates a wait (SCLA0 line: Low) and a transfer end interrupt (INTIICA0) occurs on the slave side.
(12) When the slave reads the receive data and cancels the wait (WREL0 = 1), the slave sends ACK to the master. When the ninth clock signal rises, the master detects ACK (ACKD0 = 1).
(13) When the ninth clock signal falls, the master generates a wait (SCLA0 line: Low) and a transfer end interrupt (INTIICA0) occurs on the master side.
(14) When the stop condition trigger is set (SPT0 = 1), the SDAA0 line falls and the SCLA0 line rises. Upon the elapse of the stop condition setup time, the SDAA0 line rises, thereby generating a stop condition.
(15) When the stop condition is generated, the slave detects it (SPD0 = 1) and a IICA0 interrupt (stop condition interrupt) occurs on the slave side.

(4) Master-to-slave communication 4 (data – restart condition – address)



Note: The time from when SCLA0 rises after a restart condition is issued till when a start condition is generated is 4.7 µs or more for the standard mode and 0.6 µs or more for the fast mode.

Figure 1.5 IIC Communication Timing Chart (Master-to-Slave Communication Example) (4/4)

(7) When the eighth clock signal falls after the data transfer, the slave hardware generates a wait (SCLA0 line: Low) and a transfer end interrupt (INTIICA0) occurs on the slave side.

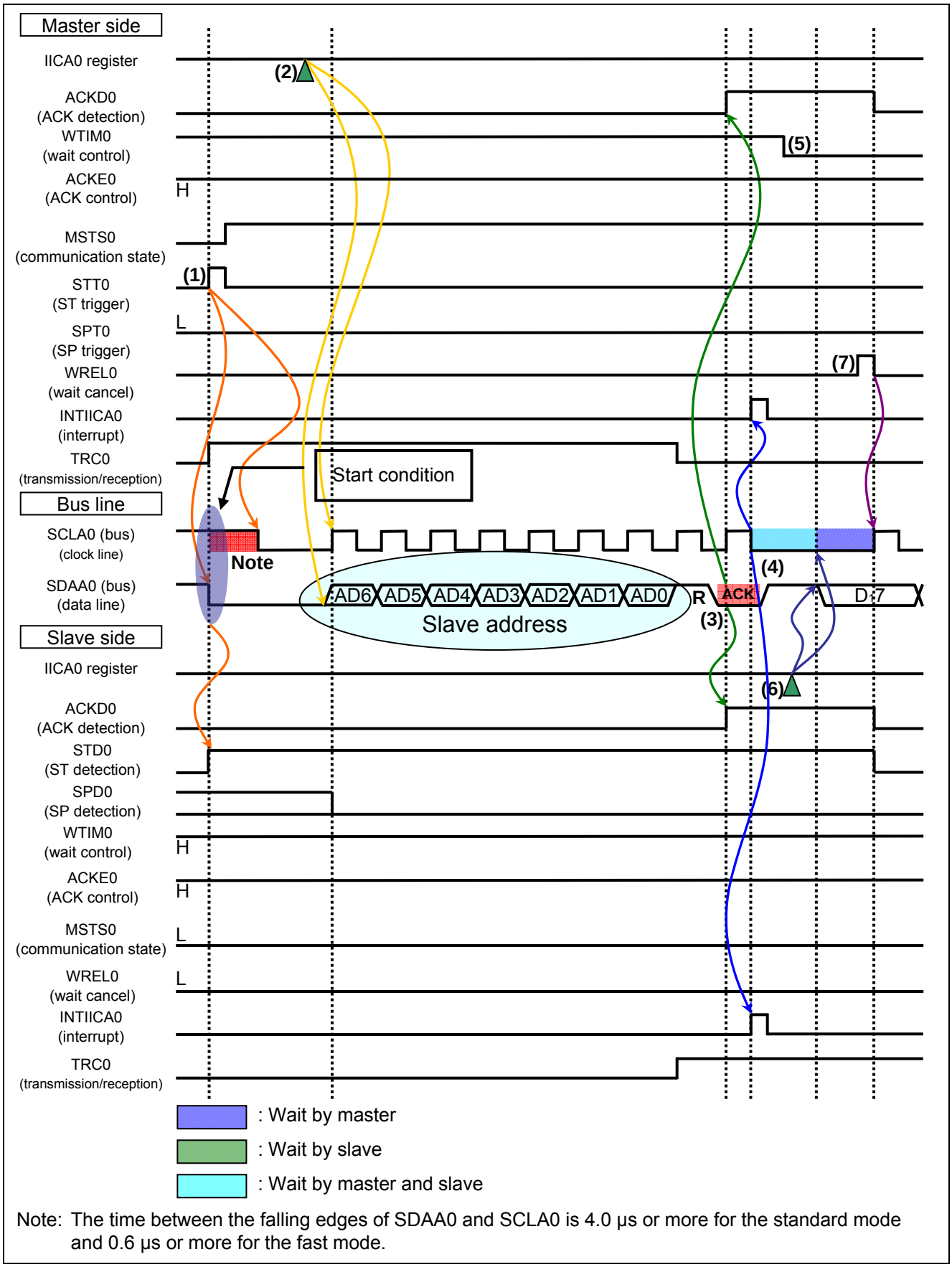
(8) The slave reads the receive data and cancels the wait (WREL0 = 1). Then, the slave sends ACK to the master. When the ninth clock signal rises, the master detects ACK (ACKD0 = 1).

(9) When the ninth clock signal falls, the master generates a wait (SCLA0 line: Low) and a transfer end interrupt (INTIICA0) occurs on the master side.

(10) The start condition trigger is set (STT0 = 1) on the master side again. Then, the SCLA0 line rises. Upon the elapse of the restart condition setup time, the SDAA0 line falls, thereby generating a start condition. Later, at the end of the hold period after the start condition is detected (STD0 = 1), the bus clock line falls, thereby completing preparations for communication.

(11) The master writes the slave address to the IICA0 register and starts transferring the address to the slave.

(5) Slave-to-master communication 1 (start condition – address – data)



Note: The time between the falling edges of SDAA0 and SCLA0 is 4.0 µs or more for the standard mode and 0.6 µs or more for the fast mode.

Figure 1.6 IIC Communication Timing Chart (Slave-to-Master Communication Example) (1/3)

(1) The start condition trigger is set (STT0 = 1) on the master side. Then, the SDAA0 line falls, thereby generating a start condition. Later, when the start condition is detected (STD0 = 1), the master enters a master device communication state (MSTS0 = 1). The SCLA0 line falls at the end of the hold period. This completes preparations for communication.
(2) The values of the address and data direction bit R (reception) are written to the IICA0 register on the master side. Then, the slave address is transmitted.
(3) If the received address and slave address match [Note], the slave hardware sends ACK to the master. When the ninth clock signal rises, the master detects ACK (ACKD0 = 1).
(4) When the ninth clock signal falls, an address transmission end interrupt (INTIICA0) occurs on the master side. If the addresses match, an address match interrupt (INTIICA0) occurs on the slave side. Both the master and the slave which has the matching address generate a wait (SCLA0 line: Low).
(5) The master selects an 8-clock wait (WTIM0 = 0) because it receives data.
(6) The slave writes transmit data to the IICA0 register and cancels the wait.
(7) When the master cancels the wait (WREL0 = 1), the slave starts transferring data to the master.

Note: If the sent address and slave address do not match, the slave does not return ACK to the master (NACK). The INTIICA0 interrupt (address match interrupt) does not occur on the slave side and thus the slave does not enter a wait state. However, the INTIICA0 interrupt (address transmission end interrupt) occurs on the master side regardless of whether the master receives ACK or NACK.

(6) Slave-to-master communication 2 (address – data – data)

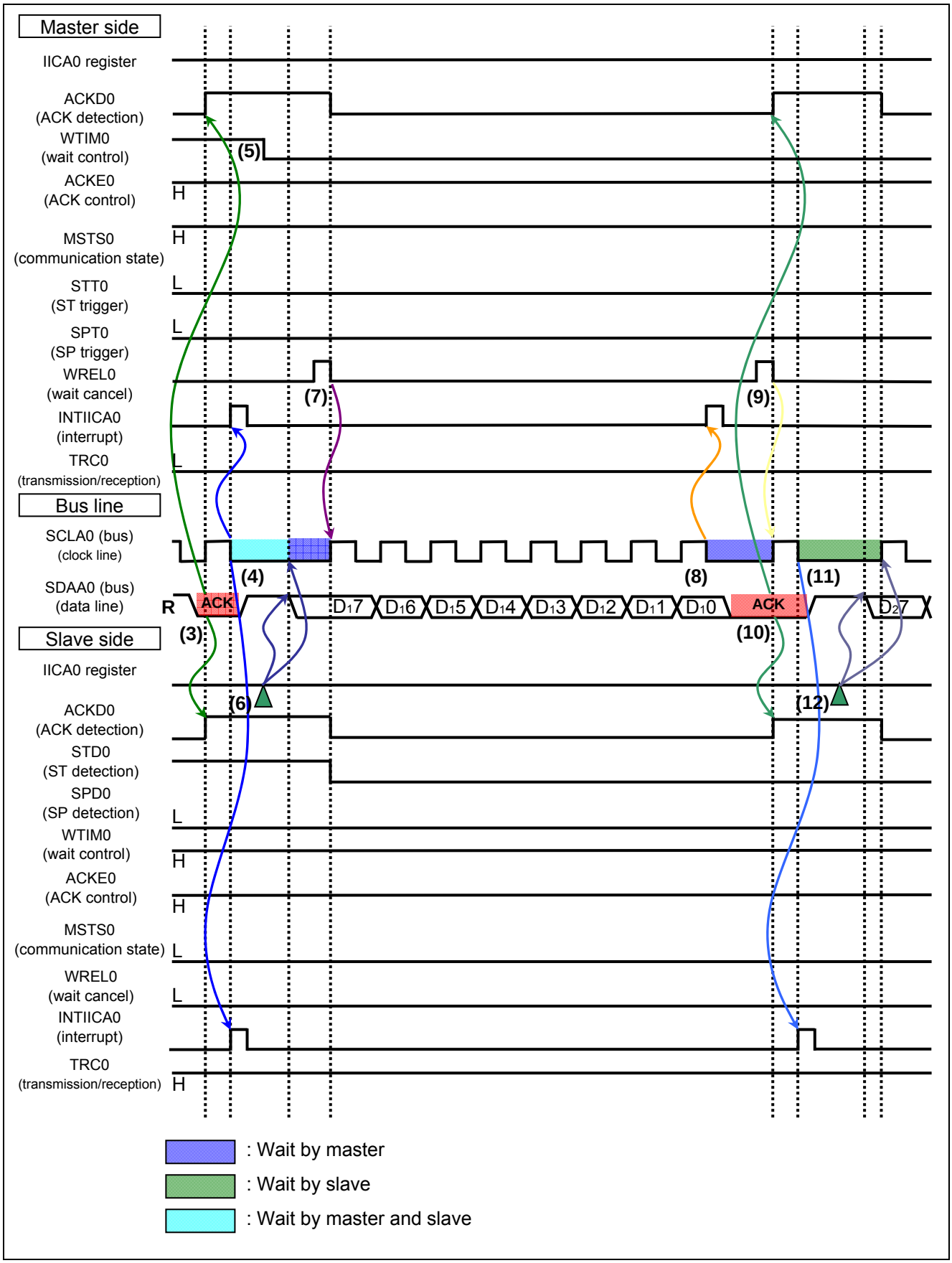
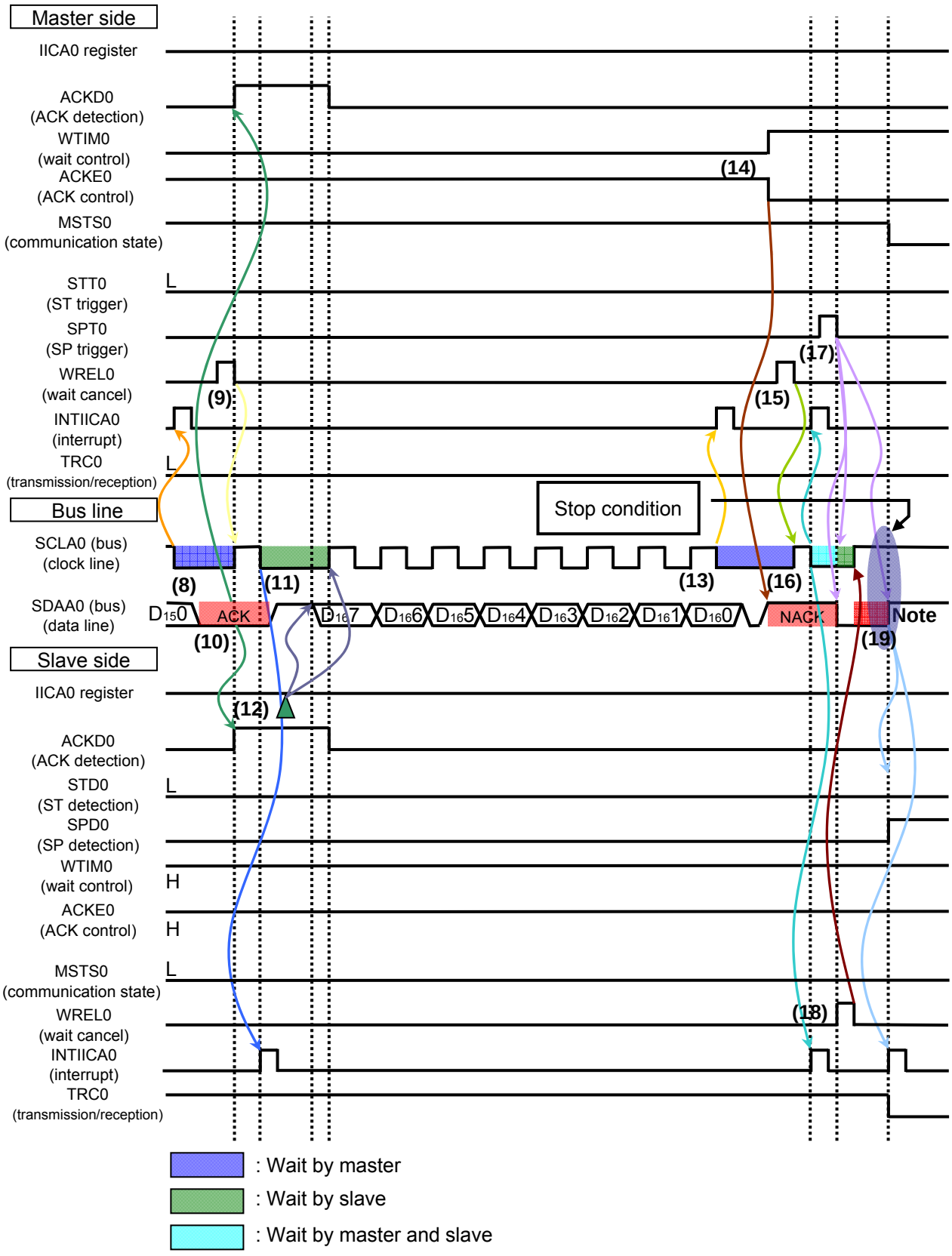


Figure 1.7 IIC Communication Timing Chart (Slave-to-Master Communication Example) (2/3)

(3) If the received address and slave address match [Note], the slave hardware sends ACK to the master. When the ninth clock signal rises, the master detects ACK (ACKD0 = 1).
(4) When the ninth clock signal falls, an address transmission end interrupt (INTIICA0) occurs on the master side. If the addresses match, an address match interrupt (INTIICA0) occurs on the slave side. Both the master and the slave which has the matching address generate a wait (SCLA0 line: Low).
(5) The master selects an 8-clock wait (WTIM0 = 0) because it receives data.
(6) The slave writes transmit data to the IICA0 register and cancels the wait.
(7) When the master cancels the wait (WREL0 = 1), the slave starts transferring data to the master.
(8) When the eighth clock signal falls, the master generates a wait (SCLA0 line: Low) and a transfer end interrupt (INTIICA0) occurs on the master side. The master hardware sends ACK to the slave.
(9) The master reads the receive data and cancels the wait (WREL0 = 1).
(10) When the ninth clock signal rises, the slave detects ACK (ACKD0 = 1).
(11) When the ninth clock signal falls, the slave generates a wait (SCLA0 line: Low) and a transfer end interrupt (INTIICA0) occurs on the slave side.
(12) The slave writes transmit data to the IICA0 register and cancels the wait. Then, the slave starts transferring data to the master.

Note: If the sent address and slave address do not match, the slave does not return ACK to the master (NACK). The INTIICA0 interrupt (address match interrupt) does not occur on the slave side and thus the slave does not enter a wait state. However, the INTIICA0 interrupt (address transmission end interrupt) occurs on the master side regardless of whether the master receives ACK or NACK.

(7) Slave-to-master communication 3 (data – data – stop condition)



Note: The time from when SCLA0 rises after a stop condition is issued till when a stop condition is generated is 4.0 µs or more for the standard mode and 0.6 µs or more for the fast mode.

Figure 1.8 IIC Communication Timing Chart (Slave-to-Master Communication Example) (3/3)

(8) When the eighth clock signal falls, the master generates a wait (SCLA0 line: Low) and a transfer end interrupt (INTIICA0) occurs on the master side. The master hardware sends ACK to the slave.
(9) The master reads the receive data and cancels the wait (WREL0 = 1).
(10) When the ninth clock signal rises, the slave detects ACK (ACKD0 = 1).
(11) When the ninth clock signal falls, the slave generates a wait (SCLA0 line: Low) and a transfer end interrupt (INTIICA0) occurs on the slave side.
(12) The slave writes transmit data to the IICA0 register and cancels the wait. Then, the slave starts transferring data to the master.
(13) When the eighth clock signal falls, a transfer end interrupt (INTIICA0) occurs on the master side and the master generates a wait (SCLA0 line: Low). The master hardware sends ACK to the slave.
(14) The master sets a NACK response (ACKE0 = 0) to inform the slave that the master has sent the last data (at the end of communication). Then, the master changes the wait time to 9 clock periods (WTIM0 = 1).
(15) After the master cancels the wait (WREL0 = 1), the slave detects NACK (ACKD0 = 0) at the rising edge of the ninth clock signal.
(16) When the ninth clock signal falls, the master and slave generate a wait (SCLA0 line: Low) and a transfer end interrupt (INTIICA0) occurs on the master and slave sides.
(17) When the master issues a stop condition (SPT0 = 1), the SDAA0 line falls, thereby canceling the wait on the master side. Later, the master waits until the SCLA0 line rises.
(18) The slave cancels the wait (WREL0 = 1) to terminate communication. Then, the SCLA0 line rises.
(19) The master confirms that the SCLA0 line has risen. Upon the elapse of the stop condition setup time after this confirmation, the master makes the SDAA0 line rise and issues a stop condition. When the stop condition is generated, the slave detects the stop condition (SPD0 = 1) and a stop condition interrupt (INTIICA0) occurs on the master and slave sides.

2. Operation Check Conditions

The sample code contained in this application note has been checked under the conditions listed in the table below.

Table 2.1 Operation Check Conditions

Item	Description
Microcontroller used	RL78/G12 (R5F1026A)
Operating frequency	• High-speed on-chip oscillator (HOCO) clock: 24 MHz • CPU/peripheral hardware clock: 24 MHz
Operating voltage	5.0 V (Operation is possible over a voltage range of 2.9 V to 5.5 V.) LVD operation (V_{LVD}): Reset mode which uses 2.81 V (2.76 V to 2.87 V)
Integrated development environment (CS+)	CS+ for CC V3.01.00 from Renesas Electronics Corp.
Assembler (CS+)	CC-RL V1.01.00 from Renesas Electronics Corp.
Integrated development environment (e^2 studio)	e^2 studio V4.0.2.008 from Renesas Electronics Corp.
Assembler (e^2 studio)	CC-RL V1.01.00 from Renesas Electronics Corp.
Board to be used	RL78/G12 target board (QB-R5F1026A-TB)

3. Related Application Notes

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

RL78/G12 Initialization (R01AN2582E) Application Note
RL78/G12 Serial Interface IICA (for Slave Transmission/Reception) (R01AN2988E) Application Note

4. Description of the Hardware

4.1 Hardware Configuration Example

Figure 4.1 shows an example of hardware configuration that is used for this application note.

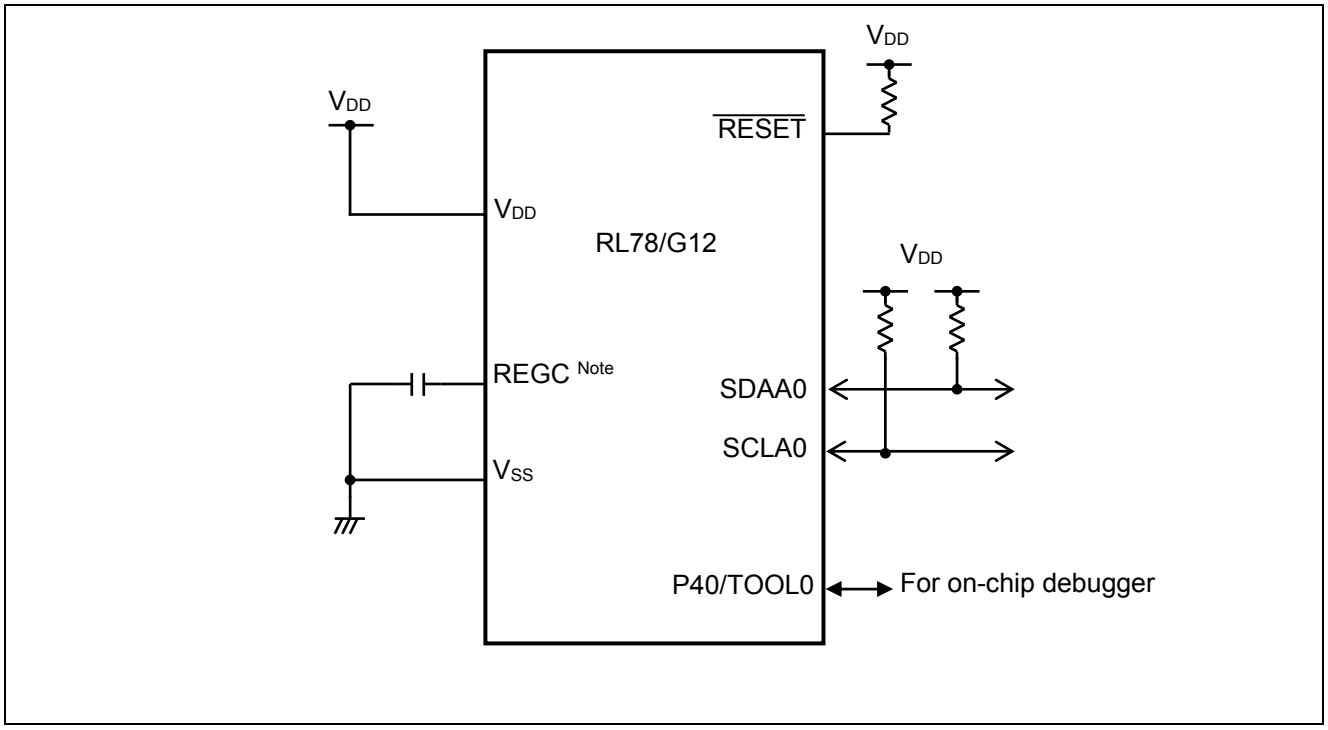


Figure 4.1 Hardware Configuration

Note: Only for 30-pin products.

Cautions: 1. The purpose of this circuit is only to provide the connection outline and the circuit is simplified accordingly. When designing and implementing an actual circuit, provide proper pin treatment and make sure that the hardware's electrical specifications are met (connect the input-only ports separately to V_{DD} or V_{SS} via a resistor).

2. V_{DD} must be held at not lower than the reset release voltage (V_{LVD}) that is specified as LVD.

4.2 List of Pins to be Used

Table 4.1 lists the pins to be used and their functions.

Table 4.1 Pins to be Used and their Functions

Pin Name	I/O	Description
P60/KR4/SCLA0	I/O	Serial clock input/output pin
P61/KR5/SDAA0	I/O	Serial data transmission/reception pin

5. Description of the Software

5.1 Operation Outline

The sample program covered in this application note provides IICA master transmission and reception (address field transmission, data transmission and data reception) through serial interface IICA.

(1) Initialize serial interface IICA.
<Conditions for setting>

- Select the fast mode as the operation mode.
- Set the transfer clock frequency to 400 kHz.
- Set the local address to 0x50.
- Turn the digital filter on.
- Enable acknowledgements.
- Generate an interrupt in response to the ninth clock signal.
- Disable stop condition interrupts.
- Use the P60/KR4/SCLA0 pin for transfer clock I/O and the P61/KR5/SDAA0 pin for data transmission/reception.

(2) Get the communication buffer (16 bytes for each of transmission and reception) ready for use. Set an initial value. Activate the timer which provides a time base (10 ms) for communication. Set a parameter for communication in the FCB structure.

(3) After a timer interrupt occurs, transmit data (16 bytes) to the slave according to the contents of the FCB structure. After completion of communication, set a parameter for next communication in the FCB structure. Then, wait for a timer interrupt.

(4) After a timer interrupt occurs, receive data from the slave. This data (16 bytes) is stored in the communication buffer sequentially. After completion of communication, set a parameter for next communication in the FCB structure. Then, wait for a timer interrupt.

(5) Repeat steps (3) and (4) above.

Caution: This sample code is related to RL78/G12 Serial Interface IICA (for Slave Transmission/Reception) (R01AN2988E) Application Note.
When the master sends a slave address or data to the slave, the master might receive negative acknowledgements (NACKs) from the slave. In this case, the master resends the slave address or data to the slave in response to the first, second or third NACK. Upon receipt of the fourth or subsequent NACK, it terminates data communication (by issuing a stop condition) and waits for a timer interrupt. It reattempts data communication after the timer interrupt occurs.

5.2 List of Option Byte Settings

Table 5.1 summarizes the settings of the option bytes.

Table 5.1 Option Byte Settings

Address	Value	Description
000C0H	01101110B	Disables the watchdog timer. (Stops counting after the release from the reset state.)
000C1H	01111111B	LVD reset mode, 2.81 V (2.76 to 2.87 V)
000C2H	11100000B	HS mode, HOCO: 24 MHz
000C3H	10000101B	Enables the on-chip debugger.

5.3 List of Constants

Table 5.2 lists the constants that are used in this sample program.

Table 5.2 Constants for the Sample Program

Constant	Setting	Description
BAUDRATE	400	Transfer speed in units of kbps (400 kbps (fast mode))
FHOCO	24	HOCO oscillation frequency in units of MHz (24 MHz)
OPCLK	FHOCO/2	IICA0 operation clock frequency in units of MHz (12 MHz)
RATE	OPCLK * 1000/400	Clock division ratio for transfer speed
DIICWL	RATE * 52/100 + 1	Value set in IICWH0 register (16)
RISETIME	100	Signal rise time (100 ns)
FALLTIME	100	Signal fall time (100 ns)
TERMHIGH	48000 / (BAUDRATE/10)	Time other than SCLA0 low-level time (ns)
WIDTHHIGH	TERMHIGH – (RISETIME + FALLTIME)	SSLA0 high-level width
DIICWH	(WIDTHHIGH * OPCLK) / 1000 + 1	Value set in IICWH0 register (13)
CSLFADDR	50H	Master address
CRETRYCNT	4	IIC transmission error count
SLAVEADDR	A0H	Slave address
BUFSIZE	16	Byte count of data to be transmitted/received
CWRITEMODE	40H	Master transmission mode command
CINWRITE	80H	During master transmission
CREADMODE	41H	Master reception mode command
CINREAD	81H	During master reception
CSUCCESSW	00H	Normal end of master transmission
CSUCCESSR	01H	Normal end of master reception
CBUSBUSY	11H	IIC bus error
CSLAVEBUSY	12H	Slave busy
CNOACK	13H	Data transmission error
CCOMERROR	1FH	Command error
RB2C	FEEAH	C register address of the register bank2
INTERVAL	10	Set value for interval timer
PERIOD	24000/4*INTERVAL	Value set in TDR00 register

5.4 List of Variables

Table 5.3 lists the global variables that are used in this sample program.

Table 5.3 Global Variables for the Sample Program

Type	Variable Name	Contents	Function Used
8 bits	RCOMSTAT	Member of structure FCB and flag indicating communication direction command/communication state. bit 7 : Communication state 0: Idle state 1: During communication bit 4 : Error flag 0: Normal end 1: Error bit 0 : Communication direction 0: Master transmission 1: Master reception <Command> 40H: Master transmission command 41H: Master reception command <Status> 80H: During transmission 81H: During reception 00H: Normal end of transmission 01H: Normal end of reception 11H: IIC bus error 12H: Slave busy (NACK of address) 13H: Data transmission error (NACK of data) 1FH: Command error	main, SSTTRX
8 bits	RSLVADDR	Member of structure FCB and slave address	main, SSTTRX
16 bits	RBUFFADDR	Member of structure FCB and start address of area for storing transfer data	main, SSTTRX
8 bits	RDTNUM	Member of structure FCB and byte count of transfer data	main, SSTTRX
8 bits × 16 arrays	RRCVBUF	Receive data buffer area	main, SINTIICA0
8 bits × 16 arrays	RTRSBUF	Transmit data buffer area	main, SINTIICA0

5.5 List of Functions (Subroutines)

Table 5.4 summarizes the functions (subroutines) that are used in this sample program.

Table 5.4 Functions (Subroutines)

Function Name	Outline
STARTIICA0	Make initial setting of IICA0 and put it in communication standby state
SINITAU0	Set TAU0 channel 0 to 10-ms interval timer
SSTARTINTV	Start timer count of TAU0 channel 0
SSTTRX	Master communication start processing
IINTIICA0	IICA0 interrupt processing
SINTIICA0	IICA0 master communication processing
SDATATRNS	SINTIICA0 data communication processor (reception)
SIICATXDT	SINTIICA0 data communication processor (transmission)
STARTCOND	Issue of start condition
STOPCOND	Issue of stop condition

5.6 Function Specifications

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

[Function Name] STARTIICA0

Synopsis	Initial setup of IICA0 hardware
Explanation	This function initializes the local address of IICA0 to 50H for fast mode.
Arguments	None
Return value	None
Remarks	None

[Function Name] SSTARTINTV

Synopsis	Startup of 10-ms interval timer
Explanation	This function starts the 10-ms interval timer for communication start timing.
Arguments	None
Return value	None
Remarks	None

[Function Name] SSTTRX

Synopsis	Master communication start processing
Explanation	This function issues a start condition and transmits an address according to the parameter indicated by structure FCB. If the bus is not free or communication is performed, this function ends the address transmission by setting error. If transfer is possible, two FCB parameters are copied into the register bank. Each register in register bank 2 is used for communication processing.
Arguments	HL register Start of structure in which a parameter is stored
Return value	None
Remarks	This function sets error status in structure FCB upon detecting error.

[Function Name] IINTIICA0

Synopsis	INTIICA0 interrupt processing
Explanation	This function performs a routine which accepts and processes INTIICA0. Subroutine SINTIICA0 processes communication.
Arguments	None
Return value	None
Remarks	None

[Function Name] SINTIICA0

Synopsis	IICA0 master communication processing
Explanation	This function performs main communication processing which uses INTIICA0.
Arguments	None
Return value	None
Remarks	This function sets status in structure FCB.

[Function Name] SDATATRNS

Synopsis	Data reception processing of IICA0 interrupt handler
Explanation	This function performs data reception processing which uses INTIICA0 interrupt.
Arguments	None
Return value	None
Remarks	None

[Function Name] SIICATXDT

Synopsis	Data transmission processing of IICA0 interrupt handler	
Explanation	This function performs data transmission processing which uses INTIICA0 interrupt.	
Arguments	flag	MD_SPT: Bus abnormal MD_NACK: NACK detected
Return value	None	
Remarks	None	

[Function Name] STARTCOND

Synopsis	Processing of start condition issue to IIC bus
Explanation	This function issues a start condition to IIC bus and waits until it detects the condition.
Arguments	None
Return value	None
Remarks	None

[Function Name] STOPCOND

Synopsis	Processing of stop condition issue to IIC bus
Explanation	This function issues a stop condition to IIC bus and waits until it detects the condition.
Arguments	None
Return value	None
Remarks	None

5.7 Flowcharts

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

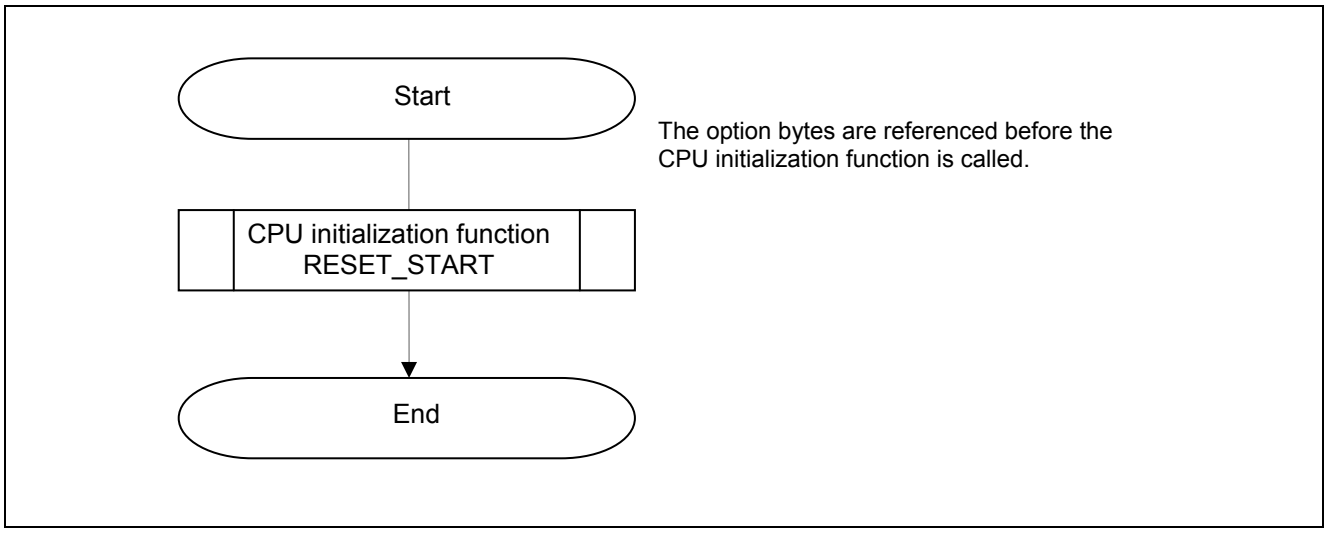


Figure 5.1 Overall Flow

5.7.1 CPU Initialization Function

Figure 5.2 shows the flowchart for the CPU initialization function.

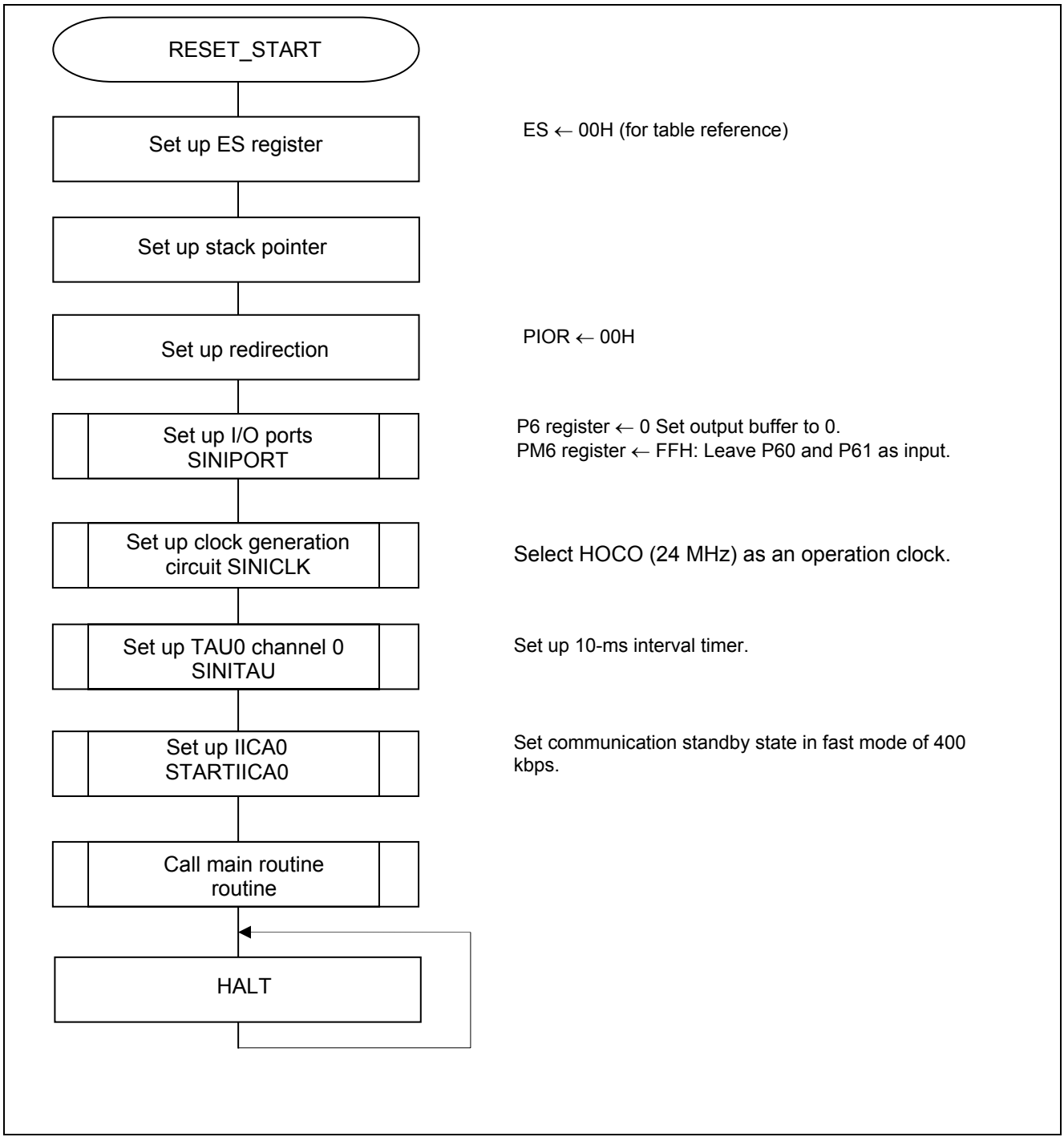


Figure 5.2 CPU Initialization Function

5.7.2 I/O Port Setup

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

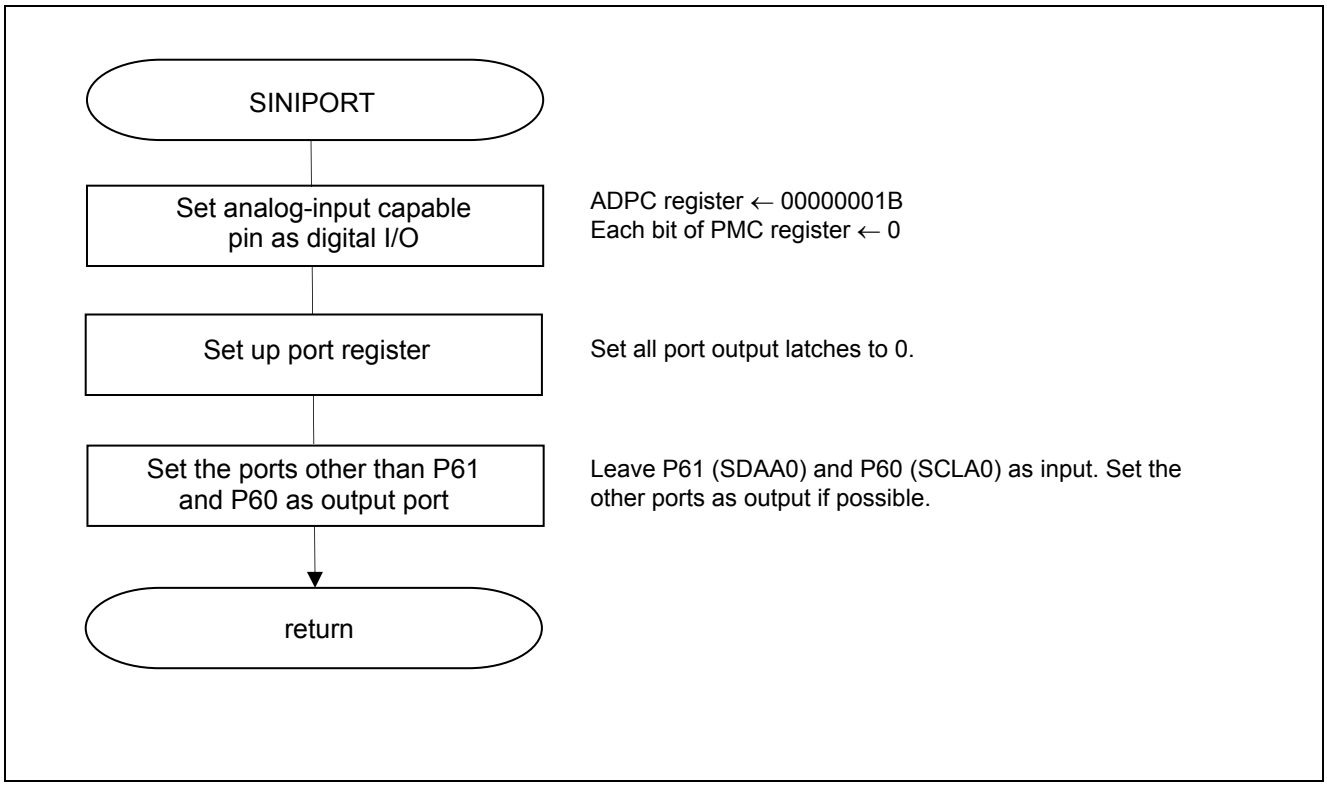


Figure 5.3 I/O Port Setup

Note: Refer to the section entitled "Flowcharts" in RL78/G12 Initialization Application Note (R01AN2582E) for the configuration of the unused ports.

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

5.7.3 Clock Generation Circuit Setup

Figure 5.4 shows the flowchart for clock generation circuit setup.

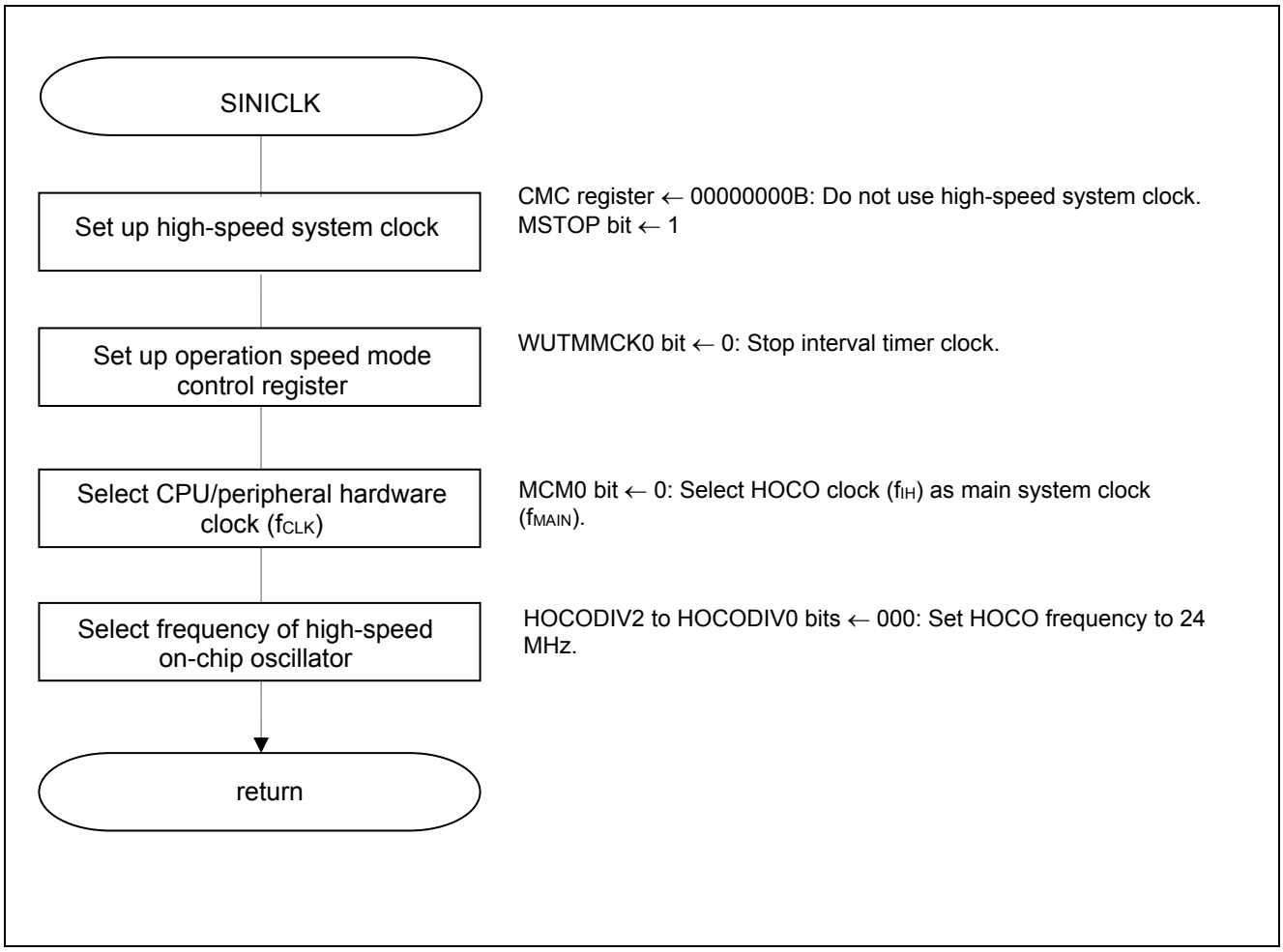


Figure 5.4 Clock Generation Circuit Setup

5.7.4 TAU0 Initial Setup

Figure 5.5 shows the flowchart for TAU0 initial setup.

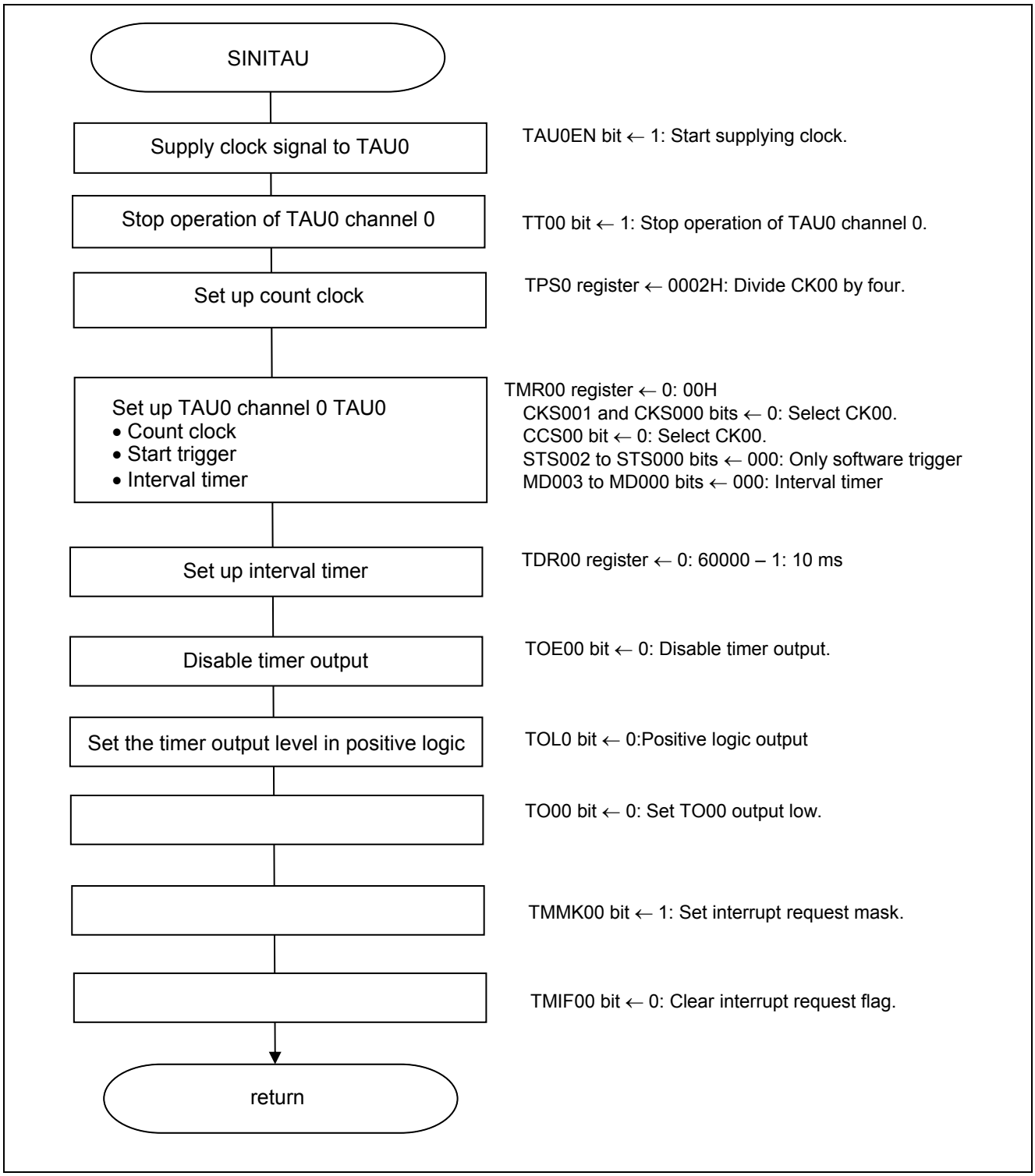


Figure 5.5 TAU0 Initial Setup

5.7.5 Serial Interface IICA Setup

Figure 5.6 shows the flowchart for serial interface IICA setup.

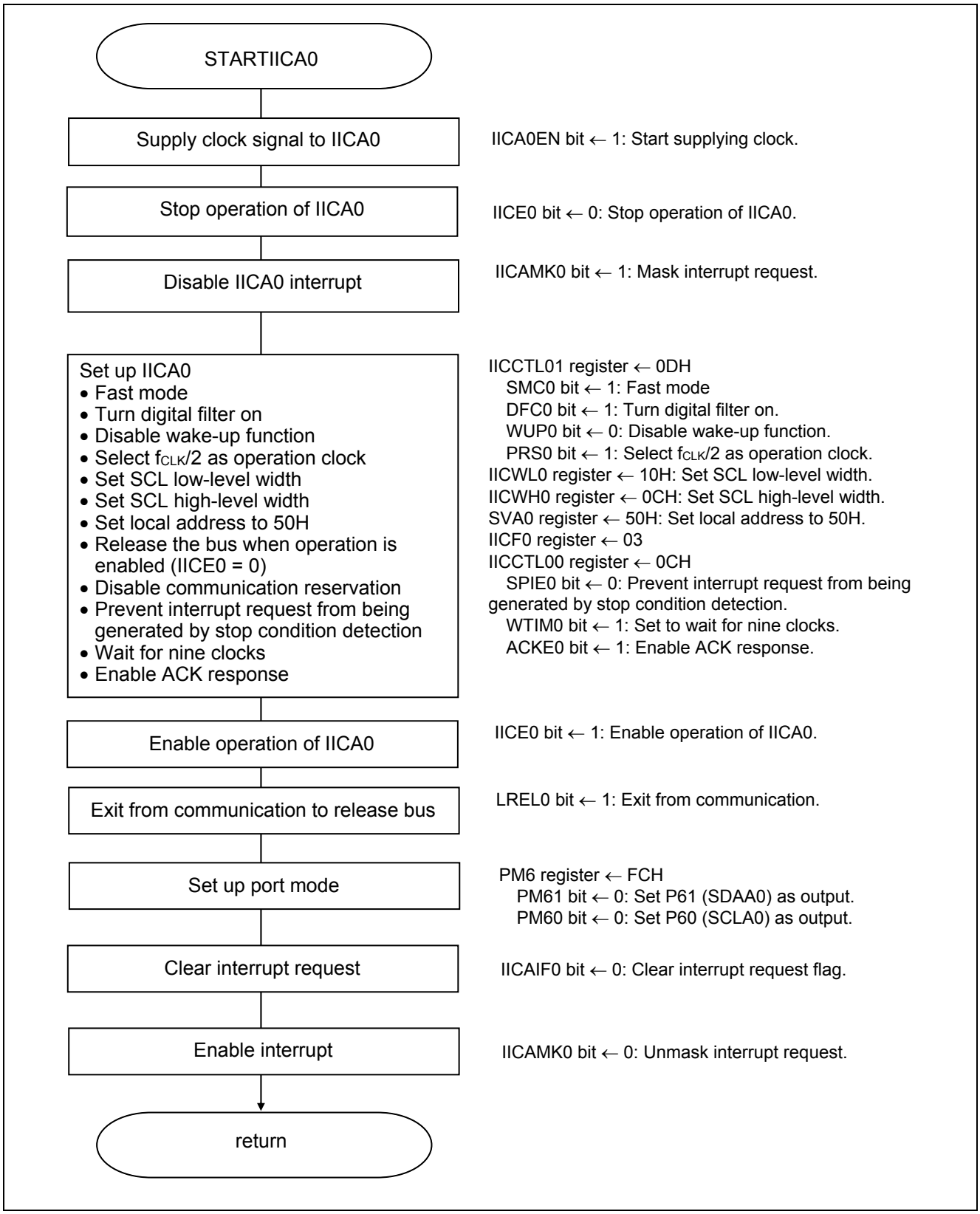


Figure 5.6 Serial Interface IICA Setup

Starting clock signal supply to serial interface IICA0

- Peripheral enable register 0 (PER0)
  Start supplying clock signals to IICA0 by using IICA0EN.

Symbol: PER0

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

Bit 4

IICA0EN	Control of serial interface IICA0 input clock supply
0	Stops supply of input clock.
1	**Enables supply of input clock.**

Caution: For details on the register setup procedures, refer to RL78/G12 User's Manual: Hardware.

Setting up the IICA0 operation mode

- IICA control register 01 (IICCTL01)
  Select an operation clock frequency.
  Turn the digital filter on.
  Select the fast mode.
  Disable the wakeup function.

Symbol: IICCTL01

7	6	5	4	3	2	1	0
WUP0	0	CLD0	DAD0	SMC0	DFC0	0	PRS0
0	0	x	x	**1**	**1**	0	**1**

Bit 7

WUP0	Address match wakeup control
0	**Disables the address match wakeup function in STOP mode.**
1	Enables the address match wakeup function in STOP mode.

Bit 3

SMC0	Operation mode selection
0	Standard mode
1	**Fast mode**

Bit 2

DFC0	Digital filter operation control
0	Turns the digital filter off.
1	**Turns the digital filter on.**

Bit 0

PRS0	Operation clock frequency selection
0	Selects f_{CLK} as the operation clock frequency.
1	**Selects f_{CLK}/2 as the operation clock frequency.**

Caution: For details on the register setup procedures, refer to RL78/G12 User's Manual: Hardware.

Configuring the transfer clock

- IICA low-level width setting register 0 (IICWL0)
- IICA high-level width setting register 0 (IICWH0)
  Set the low-level width and high-level width of the SCLA0 pin signal.

Symbol: IICWL0

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

Symbol: IICWH0

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

Setting the local address

- Slave address register 0 (SVA0)
  Set the local address.

Symbol: SVA0

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

Caution: For details on the register setup procedures, refer to RL78/G12 User's Manual: Hardware.

Setting up the conditions for starting communication

- IICA flag register 0 (IICF0)
  Set up the conditions for generating a start condition.
  Enable communication reservation.

Symbol: IICF0

7	6	5	4	3	2	1	0
STCF0	IICBSY0	0	0	0	0	STCEN0	IICRSV0
x	x	0	0	0	0	**1**	**1**

Bit 1

STCEN0	Initial start enable trigger
0	Enables generation of a start condition by detecting a stop condition after enabling operation (IICE0 = 1).
1	**Enables generation of a start condition without detecting a stop condition after enabling operation (IICE0 = 1).**

Bit 0

IICRSV0	Communication reservation function disable bit
0	Enables communication reservation.
1	**Disables communication reservation.**

Caution: For details on the register setup procedures, refer to RL78/G12 User's Manual: Hardware.

Setting up the IICA operation

- IICA control register 00 (IICCTL00)
  Enable I^2C operation.
  Disable stop condition interrupts.
  Set the wait and interrupt request generation timing.
  Enable acknowledgement output.

Symbol: IICCTL00

7	6	5	4	3	2	1	0
IICE0	LREL0	WREL0	SPIE0	WTIM0	ACKE0	STT0	SPT0
1	**1**	**0**	**0**	**0/1**	**0/1**	x	x

Bit 7

IICE0	I^2C operation enable
0	Stops the operation.
1	**Enables the operation.**

Bit 6

LREL0	Transition from the communication state
0	Normal operation
1	**Makes a transition from the current communication state to the standby state. Automatically cleared to 0 after the transition.**

Bit 4

SPIE0	Enabling/disabling generation of interrupt requests due to stop condition detection
0	**Disabled**
1	Enabled

Bit 3

WTIM0	Control of wait/interrupt request generation
0	Interrupt request is generated at the falling edge of the eighth clock signal.
1	**Interrupt request is generated at the falling edge of the ninth clock signal.**

Bit 2

ACKE0	Acknowledgement control
0	Disables acknowledgements.
1	**Enables acknowledgements. Sets the SDAA0 line to a low level during the duration of the ninth clock signal.**

Caution: For details on the register setup procedures, refer to RL78/G12 User's Manual: Hardware.

Setting up the IICA pins

- Port register 6 (P6)
- Port mode register 6 (PM6)
  Use P60 for SCLA0 and P61 for SDAA0 in output mode.

Symbol: P6

7	6	5	4	3	2	1	0
P67	P66	P65	P64	P63	P62	P61	P60
x	x	x	x	x	x	**0**	**0**

Bit 1

P61	Output data control
0	**Output 0**
1	Output 1

Bit 0

P60	Output data control
0	**Output 0**
1	Output 1

Symbol: PM6

7	6	5	4	3	2	1	0
PM67	PM66	PM65	PM64	PM63	PM62	PM61	PM60
x	x	x	x	x	x	**0**	**0**

Bit 1

PM61	P61 I/O mode selection
0	**Output mode (output buffer on)**
1	Input mode (output buffer off)

Bit 0

PM60	P60 I/O mode selection
0	**Output mode (output buffer on)**
1	Input mode (output buffer off)

Caution: For details on the register setup procedures, refer to RL78/G12 User's Manual: Hardware.

Configuring interrupt settings

- Interrupt request flag register (IF0H: 20-pin and 24-pin products, IF1L: 30-pin products)
  Clear the interrupt request flag.
- Interrupt mask flag register (MK0H: 20-pin and 24-pin products, MK1L: 30-pin products)
  Unmask interrupts.

Symbol: IF0H (24-pin products)

7	6	5	4	3	2	1	0
TMIF01	TMIF00	IICAIF0	TMIF03H	TMIF01H	SREIF0	SRIF0 CSIIF01 IICIF01	STIF0 CSIIF00 IICIF00
x	x	**0**	x	x	x	x	x

Bit 5

IICAIF0	Interrupt request flag
0	**No interrupt request signal is generated.**
1	Interrupt request is generated, interrupt request status

Symbol: MK0H (24-pin products)

7	6	5	4	3	2	1	0
TMMK01	TMMK00	IICAMK0	TMMK03H	TMMK01H	SREMK0	SRMK0 CSIMK01 IICMK01	STMK0 CSIMK00 IICMK00
x	x	x	x	**1/0**	x	x	x

Bit 5

IICAMK0	Interrupt processing control
0	**Enables interrupt processing.**
1	**Disables interrupt processing.**

Symbol: IF1L (30-pin products)

7	6	5	4	3	2	1	0
TMIF03	TMIF02	TMIF01	TMIF00	IICAIF0	SREIF1 TMIF03H	SRIF1 CSIIF11 IICIF11	STIF1 CSIIF10 IICIF10
x	x	x	x	**0**	x	x	x

Bit 3

IICAIF0	Interrupt request flag
0	**No interrupt request signal is generated.**
1	Interrupt request is generated, interrupt request status

Symbol: MK1L (30-pin products)

7	6	5	4	3	2	1	0
TMMK03	TMMK02	TMMK01	TMMK00	IICAMK0	SREMK1 TMMK03 H	SRMK1 CSIMK11 IICMK11	STMK1 CSIMK10 IICMK10
x	x	x	x	**1/0**	x	x	x

Bit 3

IICAMK0	Interrupt processing control
0	**Enables interrupt processing.**
1	**Disables interrupt processing.**

Caution: For details on the register setup procedures, refer to RL78/G12 User's Manual: Hardware.

5.7.6 Main Processing

Figures 5.7 and 5.8 show the flowcharts for the main processing.

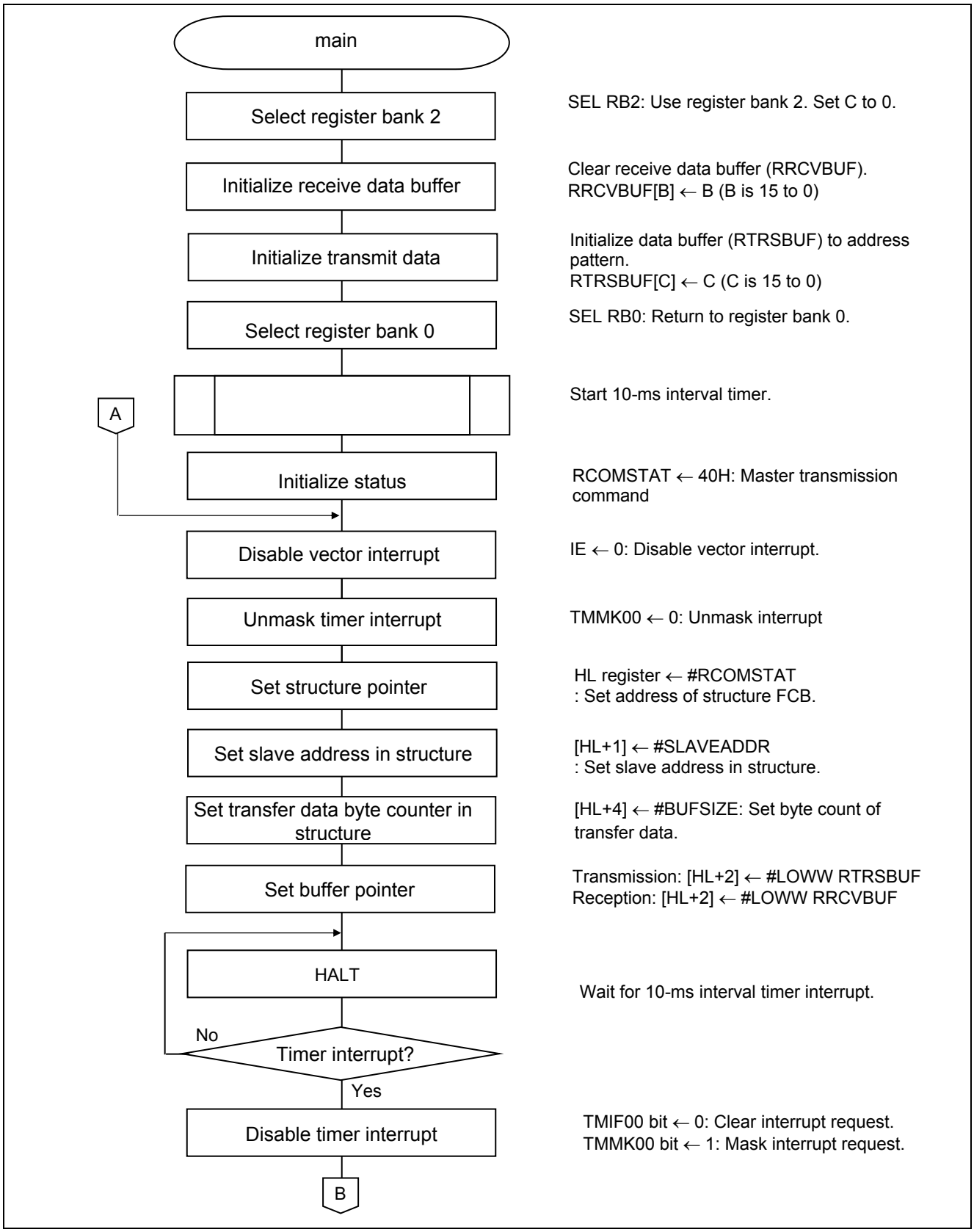


Figure 5.7 Main Processing (1/2)

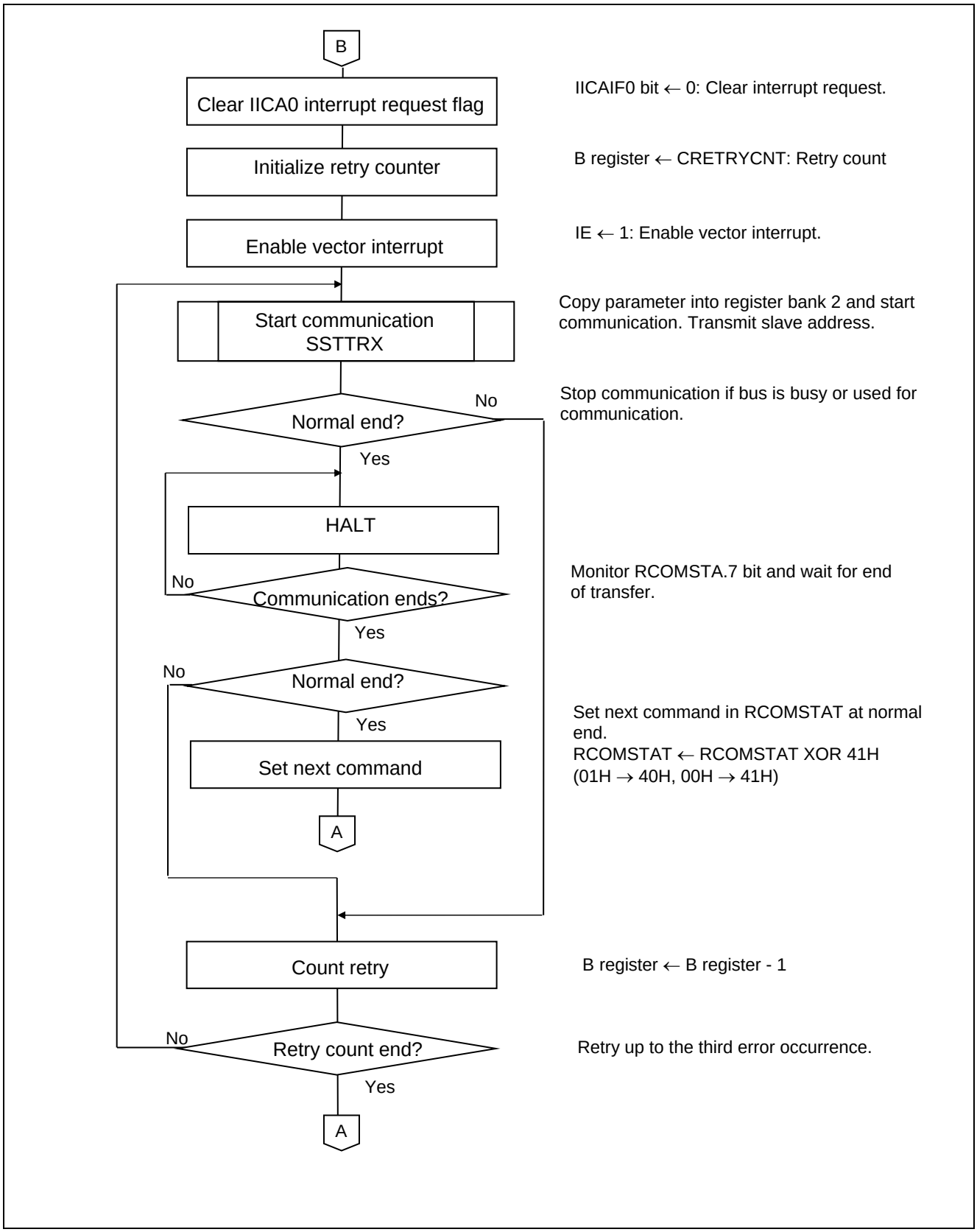


Figure 5.8 Main Processing (2/2)

Configuring interrupt settings

- Interrupt request flag register (IF0H: 20-pin and 24-pin products, IF1L: 30-pin products)
  Clear the interrupt request flag.

Symbol: IF0H (24-pin products)

7	6	5	4	3	2	1	0
TMIF01	TMIF00	IICAIF0	TMIF03H	TMIF01H	SREIF0	SRIF0 CSIIF01 IICIF01	STIF0 CSIIF00 IICIF00
x	x	**0**	x	x	x	x	x

Bit 5

IICAIF0	Interrupt request flag
0	**No interrupt request signal is generated.**
1	Interrupt request is generated, interrupt request status

Symbol: IF1L (30-pin products)

7	6	5	4	3	2	1	0
TMIF03	TMIF02	TMIF01	TMIF00	IICAIF0	SREIF1 TMIF03H	SRIF1 CSIIF11 IICIF11	STIF1 CSIIF10 IICIF10
x	x	x	x	**0**	x	x	x

Bit 3

IICAIF0	Interrupt request flag
0	**No interrupt request signal is generated.**
1	Interrupt request is generated, interrupt request status

Caution: For details on the register setup procedures, refer to RL78/G12 User's Manual: Hardware.

5.7.7 Startup of 10-ms interval timer

Figures 5.9 show the flowcharts for Startup of 10-ms interval timer.

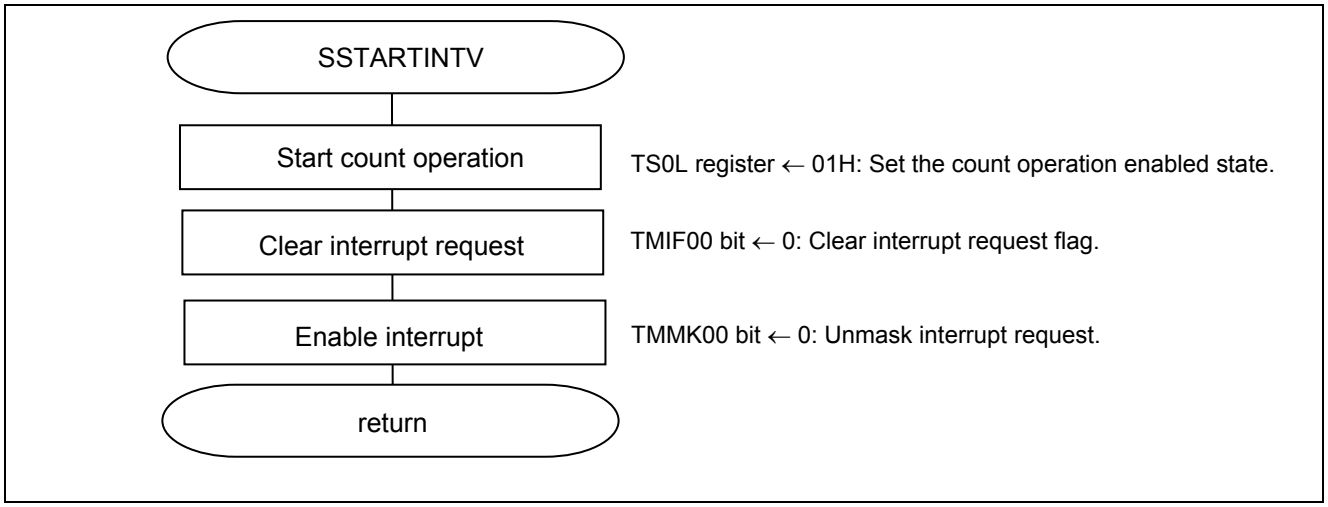


Figure 5.9 Startup of 10-ms interval timer

5.7.8 Master Communication Start Processing

Figures 5.10 and 5.11 show the flowcharts for starting master transmission.

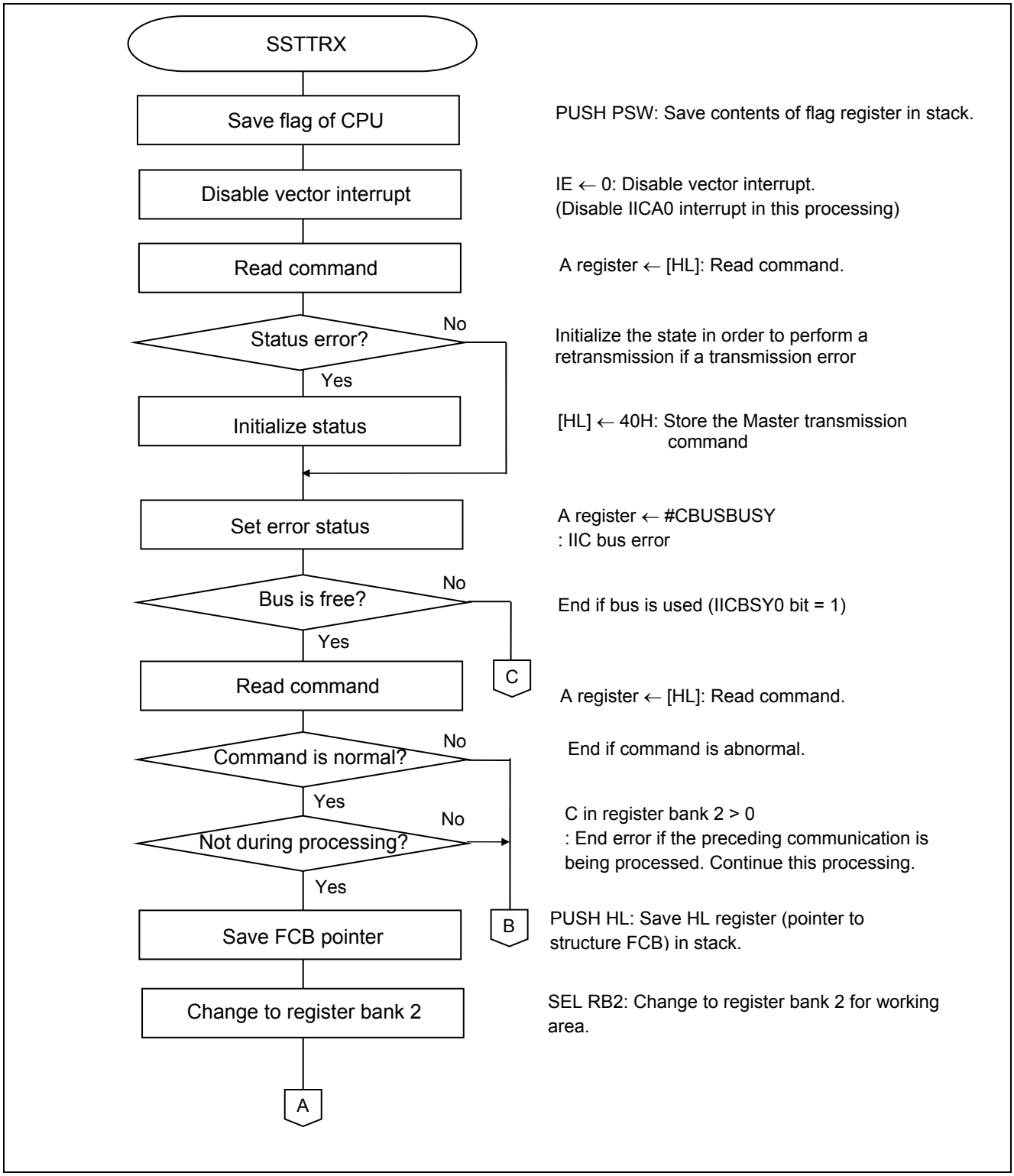


Figure 5.10 Master Transmission Start Processing (1/2)

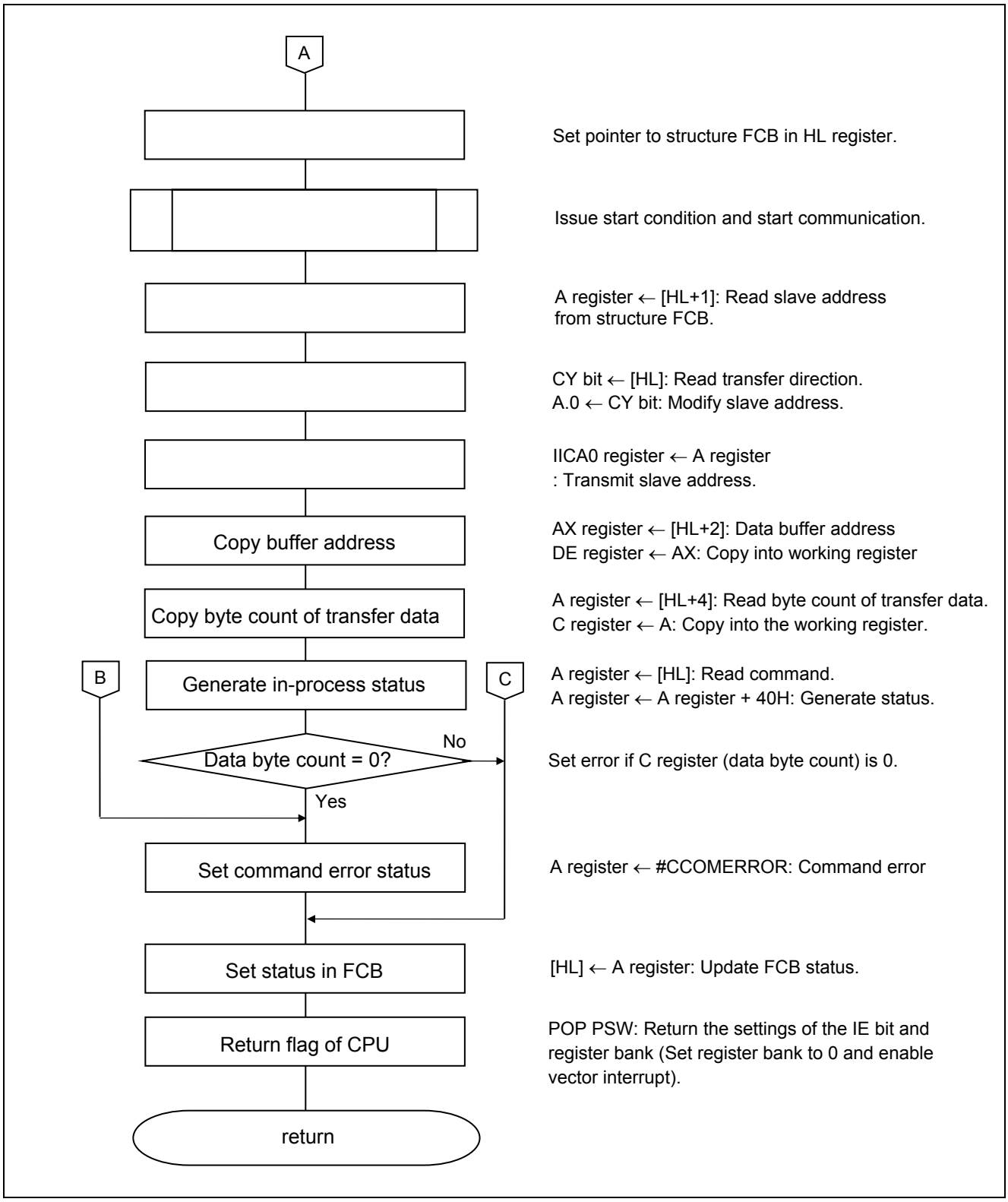


Figure 5.11 Master Transmission Start Processing (2/2)

5.7.9 Start Condition Issue

Figure 5.12 shows the flowchart for starting operation of the timer array unit.

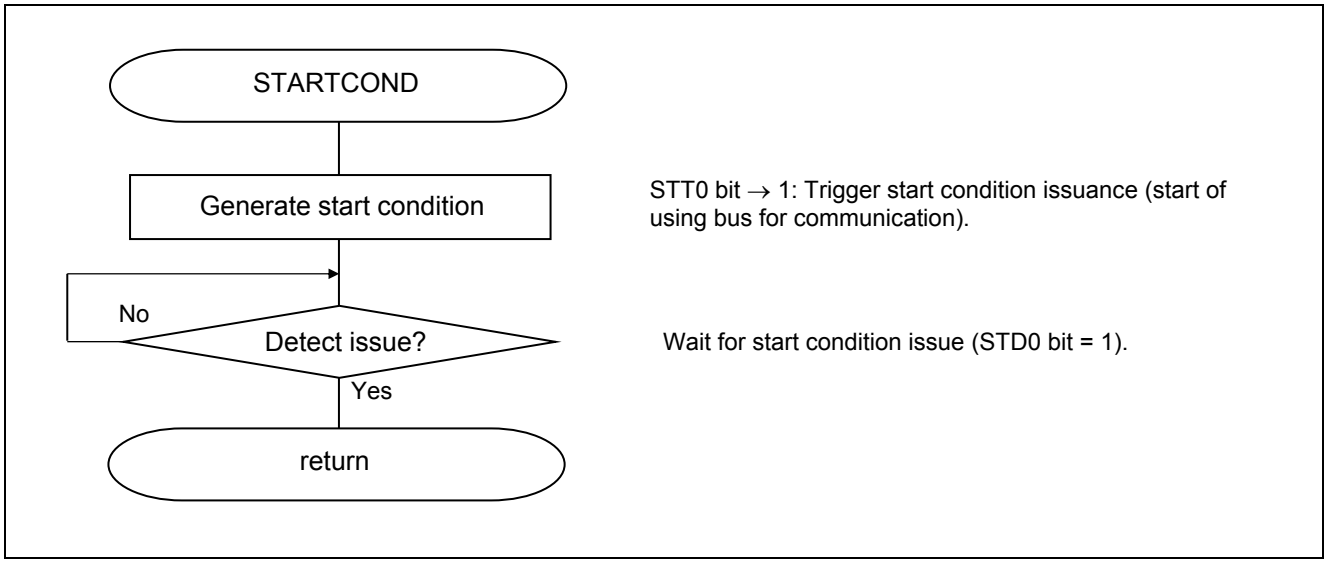


Figure 5.12 Start Condition Issue

Generating a start condition

- IICA control register 00 (IICCTL00)
  Configure the start condition generation settings.

Symbol: IICCTL00

7	6	5	4	3	2	1	0
IICE0	LREL0	WREL0	SPIE0	WTIM0	ACKE0	STT0	SPT0
x	x	x	x	x	x	**1**	**0**

Bit 1

STT0	Start condition trigger
0	Does not generate a start condition.
1	**Generates a start condition (for starting master device's transfer).**

Caution: For details on the register setup procedures, refer to RL78/G12 User's Manual: Hardware.

5.7.10 Stop Condition Issue

Figure 5.13 shows the flowchart for issuing a stop condition.

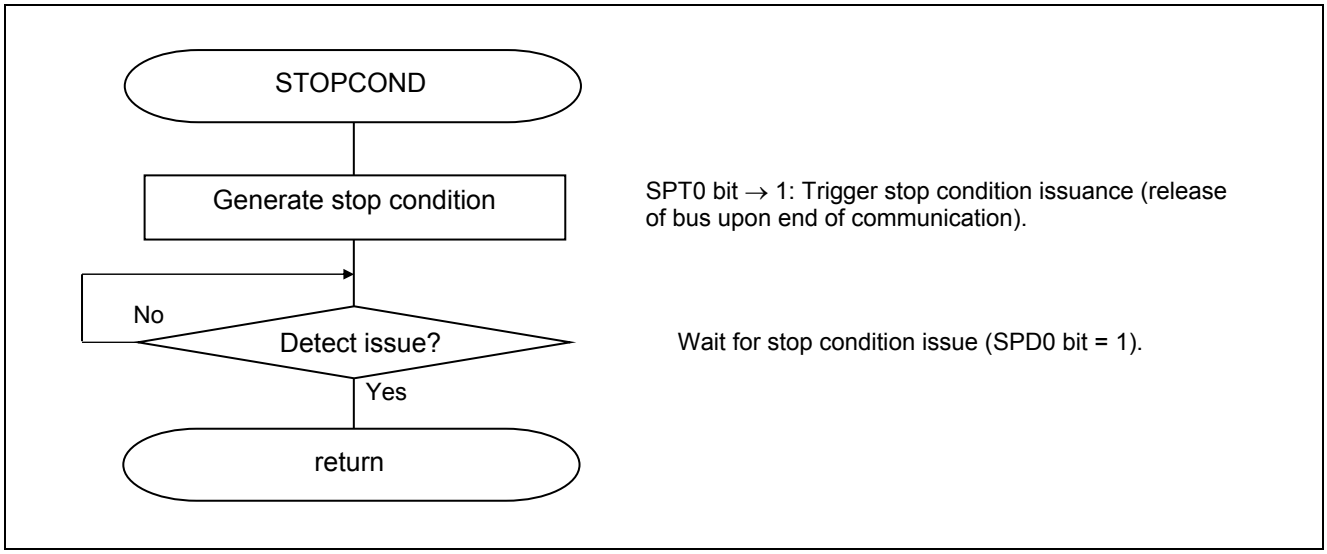


Figure 5.13 Stop Condition Issue

Generating a stop condition

- IICA control register 00 (IICCTL00)
  Configure the stop condition generation settings.

Symbol: IICCTL00

7	6	5	4	3	2	1	0
IICE0	LREL0	WREL0	SPIE0	WTIM0	ACKE0	STT0	SPT0
x	x	x	x	x	x	**0**	**1**

Bit 0

SPT0	Stop condition trigger
0	Does not generate a stop condition.
1	**Generates a stop condition (for terminating master device's transfer).**

Caution: For details on the register setup procedures, refer to RL78/G12 User's Manual: Hardware.

5.7.11 IICA0 Interrupt Processing

Figure 5.14 shows the flowchart for IICA0 interrupt processing.

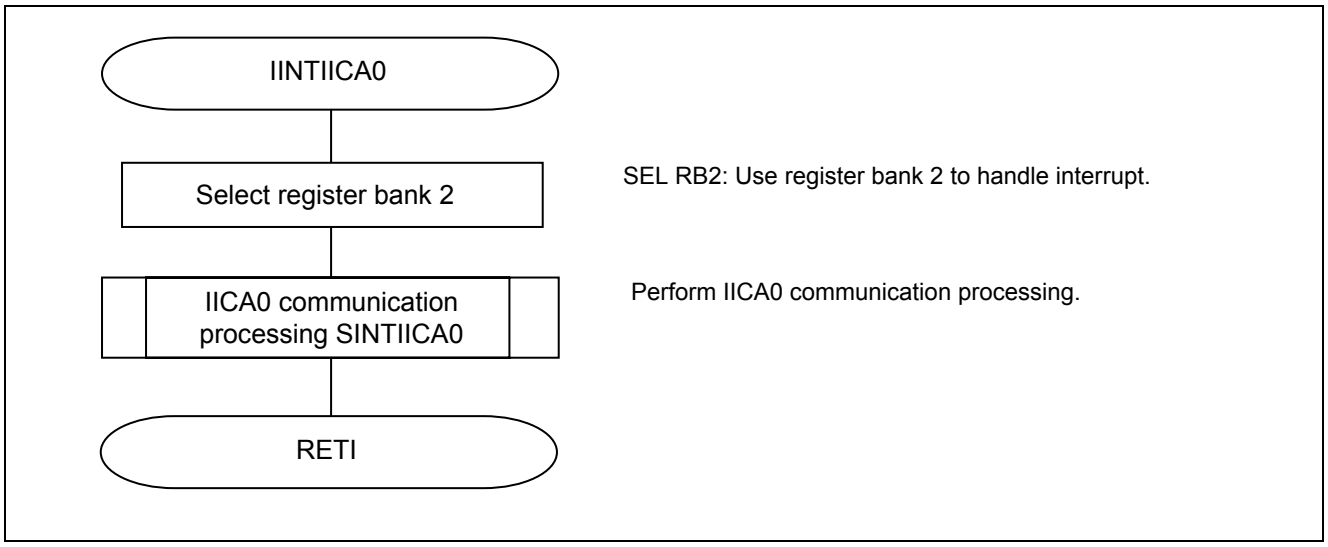


Figure 5.14 IICA0 Interrupt Processing

5.7.12 IICA0 Master Communication Processing

Figures 5.15 to 5.17 show the flowcharts for IICA0 master communication processing.

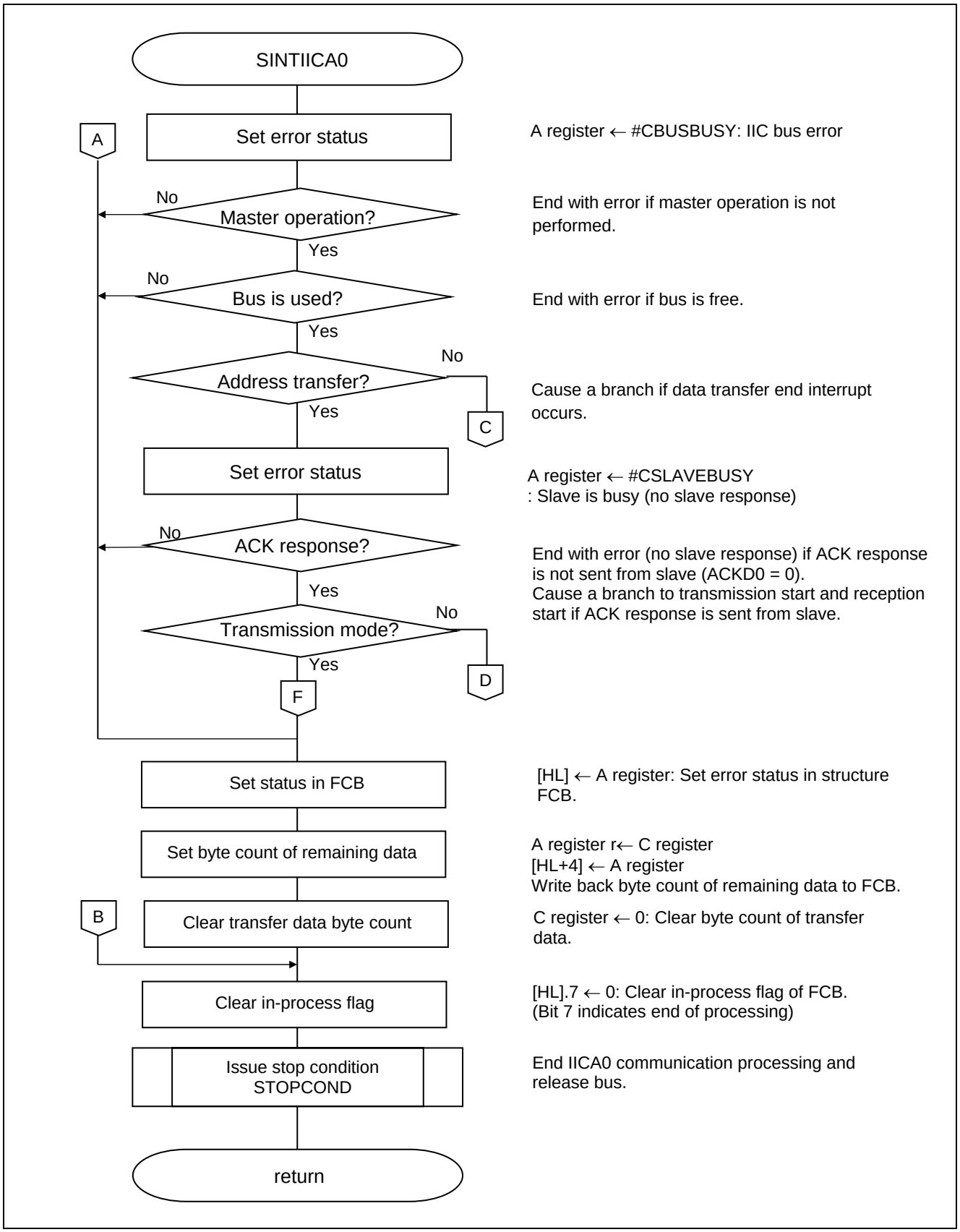


Figure 5.15 IICA0 Master Communication Processing (1/3)

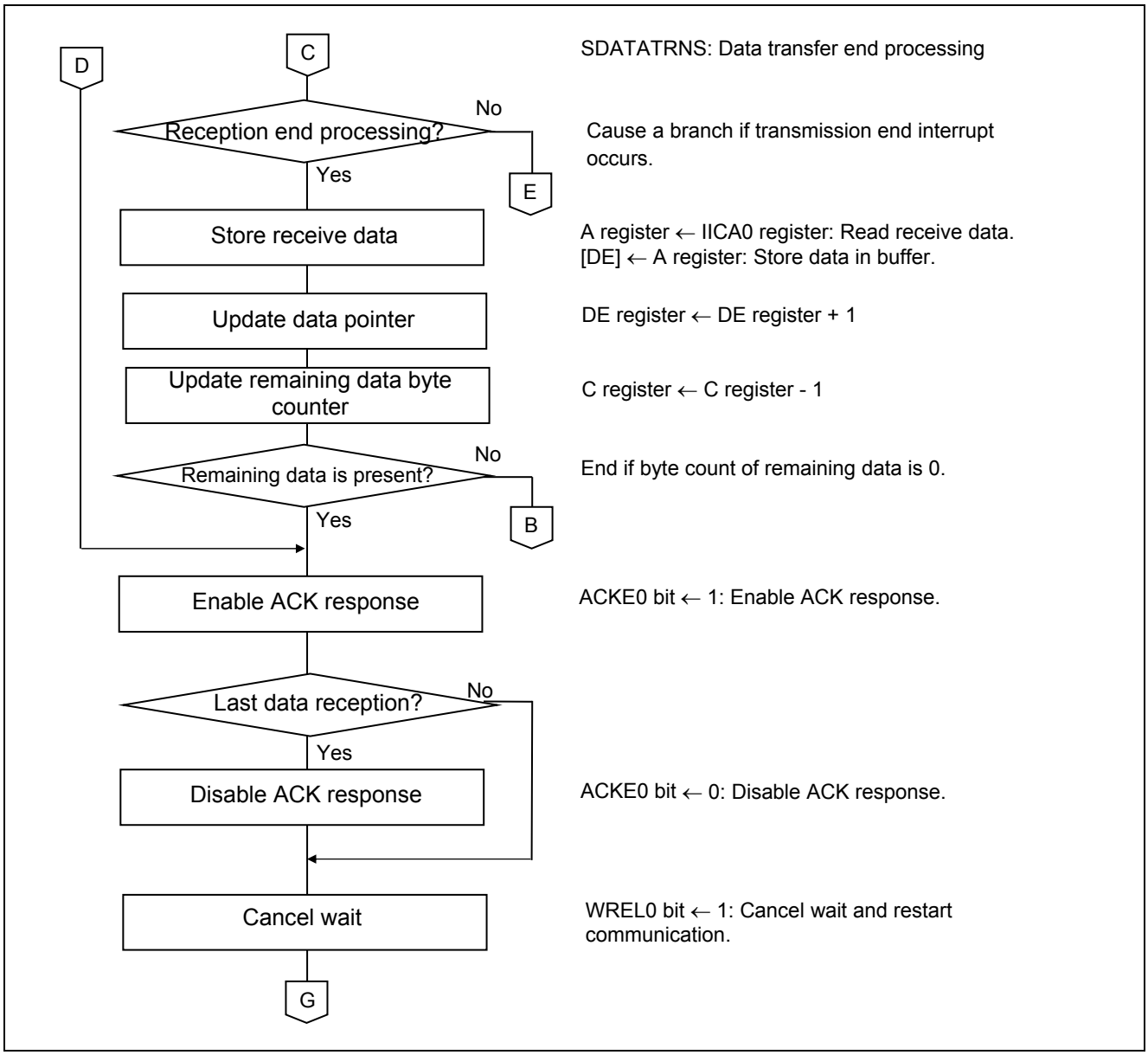


Figure 5.16 IICA0 Master Communication Processing (2/3)

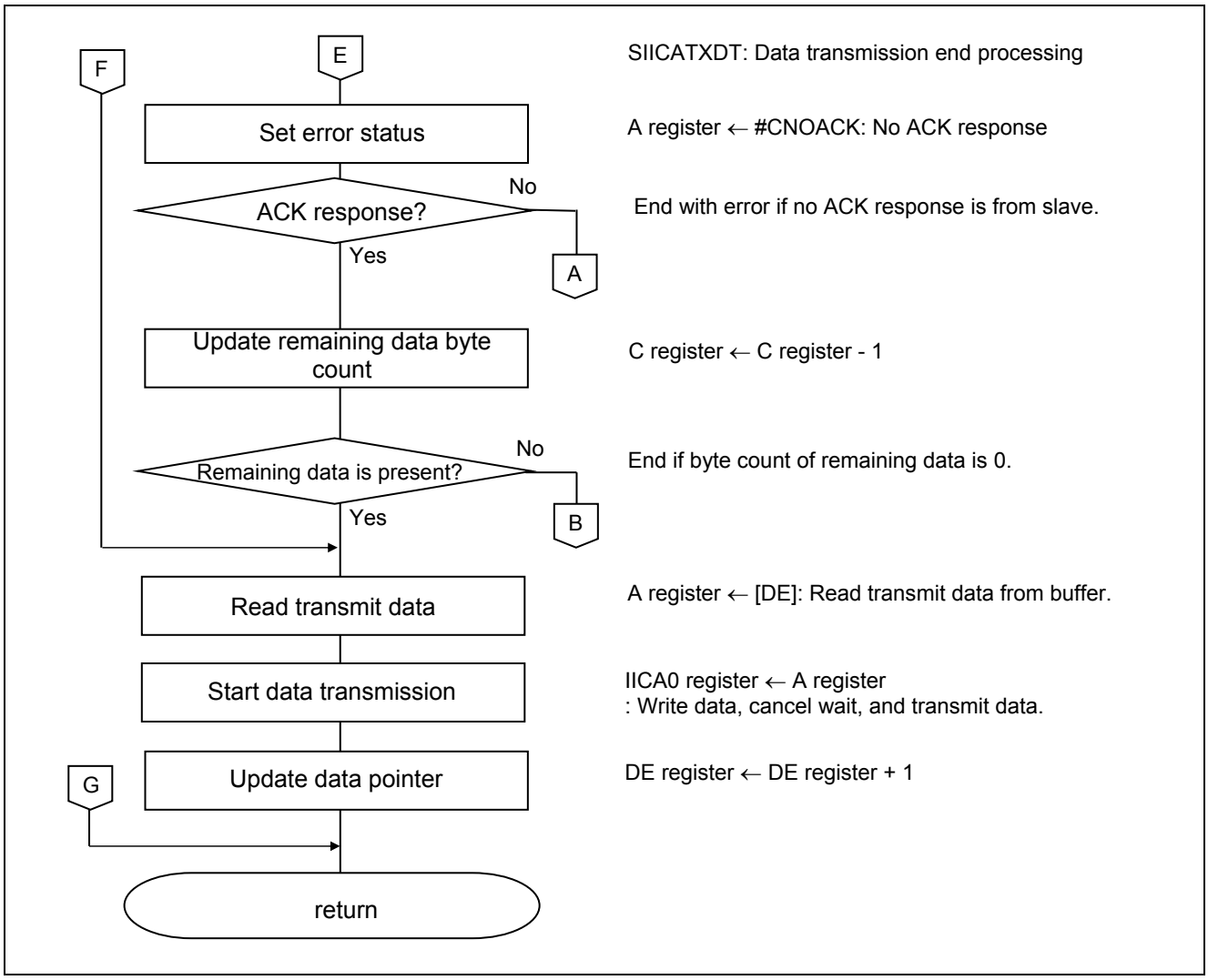


Figure 5.17 IICA0 Master Communication Processing (3/3)

Setting transmit data

- IICA shift register 0 (IICA0)
  When starting master communication: Write the communication destination address (Set bit 0 to 0 at transmission and 1 at reception)
  When sending data: Write the transmit data.
  When receiving data: Read the receive data.

Symbol: IICA0

7	6	5	4	3	2	1	0

Confirming the communication status

- IICA status register 0 (IICS0)
  Check the master communication status.
  Check the transfer direction.
  Detect an acknowledgement.

Symbol: IICS0

7	6	5	4	3	2	1	0
MSTS0	ALD0	EXC0	COI0	TRC0	ACKD0	STD0	SPD0
0/1	x	x	x	**0/1**	**0/1**	**0/1**	**0/1**

Bit 7

MSTS0	Master state
0	**Slave device state or communication standby state**
1	**Master device communication state**

Bit 3

TRC0	Transmission/reception state detection
0	**Reception state (other than a transmission state)**
1	**Transmission state**

Bit 2

ACKD0	Acknowledgement detection
0	**Acknowledgement is not detected.**
1	**Acknowledgement is detected.**

Bit 1

STD0	Start condition detection
0	**Start condition is not detected.**
1	**Start condition is detected.**

Bit 0

SPD0	Stop condition detection
0	**Stop condition is not detected.**
1	**Stop condition is detected.**

Caution: For details on the register setup procedures, refer to RL78/G12 User's Manual: Hardware.

6. Sample Code

The sample code is available on the Renesas Electronics Website.

7. Documents for Reference

RL78/G12 User's Manual: Hardware (R01UH0200E)

RL78 Family User's Manual: Software (R01US0015E)

(The latest versions of the documents are available on the Renesas Electronics Website.)

Technical Updates/Technical Brochures

(The latest versions of the documents are available on the Renesas Electronics Website.)

Website and Support

Renesas Electronics Website

- http://www.renesas.com/index.jsp

Inquiries

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

Revision Record	RL78/G12 Serial Interface IICA (for Master Transmission/Reception) CC-RL

Rev.	Date	Description	
		Page	Summary
1.00	Oct. 20, 2015	—	First edition issued

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. Handling of Unused Pins

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

   ¾ 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.

   ¾ 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.

   ¾ 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.

   ¾ 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 a product with a different part number, confirm that the change will not lead to problems.

   ¾ The characteristics of Microprocessing unit or Microcontroller unit products in the same group but having a different part number may differ in terms of the 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 of these circuits, software, and information in the design of your equipment. Renesas Electronics assumes no responsibility for any losses incurred by you or third parties arising from the use of these circuits, software, or information.
2. Renesas Electronics has used reasonable care in preparing the information included in this document, but Renesas Electronics does not warrant that such information is error free. Renesas Electronics assumes no liability whatsoever for any damages incurred by you resulting from errors in or omissions from the information included herein.
3. Renesas Electronics does not assume any liability for infringement of patents, copyrights, or other intellectual property rights of third parties by or arising from the use of Renesas Electronics products or technical information described in this document. No license, express, implied or otherwise, is granted hereby under any patents, copyrights or other intellectual property rights of Renesas Electronics or others.
4. You should not alter, modify, copy, or otherwise misappropriate any Renesas Electronics product, whether in whole or in part. Renesas Electronics assumes no responsibility for any losses incurred by you or third parties arising from such alteration, modification, copy or otherwise misappropriation of Renesas Electronics product.
5. Renesas Electronics products are classified according to the following two quality grades: "Standard" and "High Quality". The recommended applications for each Renesas Electronics product depends on the product's quality grade, as indicated below.
   "Standard": Computers; office equipment; communications equipment; test and measurement equipment; audio and visual equipment; home electronic appliances; machine tools; personal electronic equipment; and industrial robots etc.
   "High Quality": Transportation equipment (automobiles, trains, ships, etc.); traffic control systems; anti-disaster systems; anti-crime systems; and safety equipment etc.
   Renesas Electronics products are neither intended nor authorized for use in products or systems that may pose a direct threat to human life or bodily injury (artificial life support devices or systems, surgical implantations etc.), or may cause serious property damages (nuclear reactor control systems, military equipment etc.). You must check the quality grade of each Renesas Electronics product before using it in a particular application. You may not use any Renesas Electronics product for any application for which it is not intended. Renesas Electronics shall not be in any way liable for any damages or losses incurred by you or third parties arising from the use of any Renesas Electronics product for which the product is not intended by Renesas Electronics.
6. You should use the Renesas Electronics products described in this document within the range specified by Renesas Electronics, especially with respect to the maximum rating, operating supply voltage range, movement power voltage range, heat radiation characteristics, installation and other product characteristics. Renesas Electronics shall have no liability for malfunctions or damages arising out of the use of Renesas Electronics products beyond such specified ranges.
7. Although Renesas Electronics endeavors to improve the quality and reliability of its products, semiconductor products have specific characteristics such as the occurrence of failure at a certain rate and malfunctions under certain use conditions. Further, Renesas Electronics products are not subject to radiation resistance design. Please be sure to implement safety measures to guard them against the possibility of physical injury, and injury or damage caused by fire in the event of the failure of a Renesas Electronics product, such as safety design for hardware and software including but not limited to redundancy, fire control and malfunction prevention, appropriate treatment for aging degradation or any other appropriate measures. Because the evaluation of microcomputer software alone is very difficult, please evaluate the safety of the final products or systems manufactured by you.
8. Please contact a Renesas Electronics sales office for details as to environmental matters such as the environmental compatibility of each Renesas Electronics product. Please use Renesas Electronics products in compliance with all applicable laws and regulations that regulate the inclusion or use of controlled substances, including without limitation, the EU RoHS Directive. Renesas Electronics assumes no liability for damages or losses occurring as a result of your noncompliance with applicable laws and regulations.
9. Renesas Electronics products and technology may not be used for or incorporated into any products or systems whose manufacture, use, or sale is prohibited under any applicable domestic or foreign laws or regulations. You should not use Renesas Electronics products or technology described in this document for any purpose relating to military applications or use by the military, including but not limited to the development of weapons of mass destruction. When exporting the Renesas Electronics products or technology described in this document, you should comply with the applicable export control laws and regulations and follow the procedures required by such laws and regulations.
10. It is the responsibility of the buyer or distributor of Renesas Electronics products, who distributes, disposes of, or otherwise places the product with a third party, to notify such third party in advance of the contents and conditions set forth in this document, Renesas Electronics assumes no responsibility for any losses incurred by you or third parties as a result of unauthorized use of Renesas Electronics products.
11. This document may not be reproduced or duplicated in any form, in whole or in part, without prior written consent of Renesas Electronics.
12. Please contact a Renesas Electronics sales office if you have any questions regarding the information contained in this document or Renesas Electronics products, or if you have any other inquiries.

(Note 1) "Renesas Electronics" as used in this document means Renesas Electronics Corporation and also includes its majority-owned subsidiaries.

(Note 2) "Renesas Electronics product(s)" means any product developed or manufactured by or for Renesas Electronics.

RENESAS

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.
2801 Scott Boulevard Santa Clara, CA 95050-2549, U.S.A.
Tel: +1-408-588-6000, Fax: +1-408-588-6130

Renesas Electronics Canada Limited
9251 Yonge Street, Suite 8309 Richmond Hill, Ontario Canada L4C 9T3
Tel: +1-905-237-2004

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: +49-211-6503-0, Fax: +49-211-6503-1327

Renesas Electronics (China) Co., Ltd.
Room 1709, Quantum Plaza, No.27 ZhiChunLu Haidian District, Beijing 100191, P.R.China
Tel: +86-10-8235-1155, Fax: +86-10-8235-7679

Renesas Electronics (Shanghai) Co., Ltd.
Unit 301, Tower A, Central Towers, 555 Langao Road, Putuo District, Shanghai, P. R. China 200333
Tel: +86-21-2226-0888, Fax: +86-21-2226-0999

Renesas Electronics Hong Kong Limited
Unit 1601-1611, 16/F., Tower 2, Grand Century Place, 193 Prince Edward Road West, Mongkok, Kowloon, Hong Kong
Tel: +852-2265-6688, Fax: +852 2886-9022

Renesas Electronics Taiwan Co., Ltd.
13F, No. 363, Fu Shing North Road, Taipei 10543, Taiwan
Tel: +886-2-8175-9600, Fax: +886 2-8175-9670

Renesas Electronics Singapore Pte. Ltd.
80 Bendemeer Road, Unit #06-02 Hyflux Innovation Centre, Singapore 339949
Tel: +65-6213-0200, Fax: +65-6213-0300

Renesas Electronics Malaysia Sdn.Bhd.
Unit 1207, 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 India Pvt. Ltd.
No.777C, 100 Feet Road, HALII Stage, Indiranagar, Bangalore, India
Tel: +91-80-67208700, Fax: +91-80-67208777

Renesas Electronics Korea Co., Ltd.
12F., 234 Teheran-ro, Gangnam-Gu, Seoul, 135-080, Korea
Tel: +82-2-558-3737, Fax: +82-2-558-5141

Colophon 5.0