RH850/U2B Group

R01AN6342EJ0010
Rev.0.10

MSPI Application Note

Summary

This application notes explains the Multichannel Serial Peripheral Interface (MSPI) function of automotive single-chip microcontroller RH850/U2B series for automobile (hereinafter called U2B).

Aim of this document and software is to provide supplemental information for the function on RH850/U2B. It is not intended to implement in the design for mass production.

There is no guarantee to update in this document and software to reflect the latest manual, errata, technical update and development environment. You are fully responsible for the incorporation or any other use of the information of this document in the design of your product or system, and please refer to latest manual, errata, technical update and development environment.

Target Device

- RH850/U2B Group

Target Integrated Development Environment

CS+(from RENESAS Electronics)

Version :E8.07.00g6

Device filw :DR7F702Z22.DVF

Reference Document

RH850/U2B User's Manual: Hardware

For function details and electrical characteristics, please refer to "User's Manual: Hardware".

This application note is based on the following manual.

- RH850/U2B User's Manual (Rev.0.50): R01UH0923EJ0050

Contents

1. Application 3

2. Overview 4
2.1 Function Overviews 4
2.2 Block Diagram 5

3. Operation Examples 6
3.1 External Loopback Communication in Same Channels 7
3.1.1 Specification Overview 7
3.1.2 System Configuration 8
3.1.1 Software Explanation 8
3.2 External Loopback Communication between Two Channels 12
3.2.1 Specification Overview 12
3.2.2 System Configuration 13
3.2.3 Software Explanation 14
3.3 Interrupt Communication during Frame Communication 19
3.3.1 Specification Overview 19
3.3.2 System Configuration 23
3.3.3 Software Explanation 24
3.4 Frame Communication with CRC between Two Channels 34
3.4.1 Specification Overview 34
3.4.2 System Configuration 37
3.4.3 Software Explanation 38

1. Application

This application note reports the operation examples of RH850/U2B MSPI.

2. Overview

2.1 Function Overviews

There are the following functions of U2B MSPI

- Three-wire serial synchronous data transfer
- Master mode and slave mode selectable
- Multiple slave configurations thanks to up to twelve configurable chip select signals
- Maximum transmission speed of master/slave:
  - LVDS mode: Max 40MHz(MSPI6/MSPI9)
  - Single end mode: Max 20MHz(MSPI0-9)[*1]
- Clock and data phase selectable for each channel.
- Data transfer with MSB or LSB first selectable for each channel
- Transfer control channel: Up to twelve channels for each unit
- Transfer data length of channel is selectable from 2 to 128 bits in 1-bit units.
- Three-transmission modes
  - Transmit-only mode
  - Receive-only mode
  - Transmit/Receive mode
- Error detection
  - Parity error
  - Data consistency error
  - Overwrite error
  - Overread error
  - Overrun error
  - CRC error
- JOB support for AUTOSAR
- JOB effective control for AUTOSAR
- LBM (Loop Back Mode) for self-diagnosis
- Forced chip select idle setting
- Safe-SPI ver1.00 support only master mode
- Four-interrupt requests
  - Communication status
  - Reception status
  - Communication error
  - JOB completion
- Three-DMA requests
  - Communication status
  - Reception status
  - JOB completion
- Receive sampling point
  - Receive sampling point can be shifted to next serial clock edge.

*1 Depending on the selected I/O PORT, the maximum transfer rate of the unit mentioned the above may be lower. Refer to RH850/U2B User's Manual 21.1.10 Combination of Pin and Port and 66.3.11.1 MSPI Communication Speed Overview for the details of the PORTs that can be selected for each unit and the maximum transfer rate of the PORTs

2.2 Block Diagram

The following shows the MSPI block diagram.

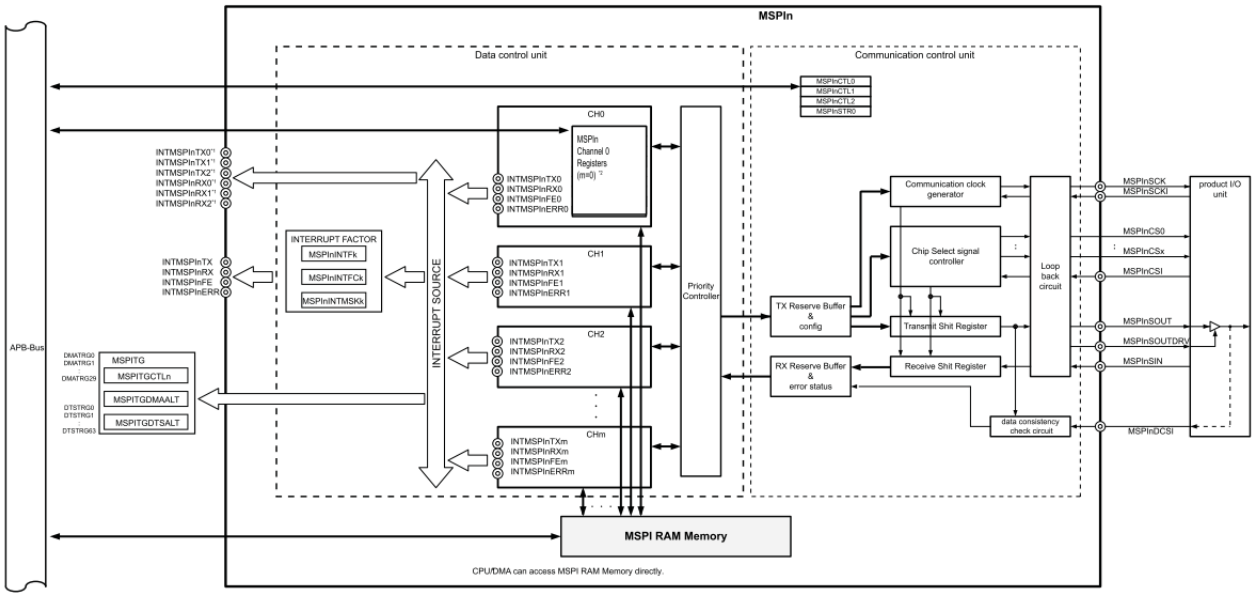

Figure 2-1 MSPI Block Diagram

Note 1. This is the interrupt for MSPI0 and MSPI1. Refer to "RH850/U2B User's Manual Figure 21.1 MSPI0-2 Interrupt Connection Image" for the details.

Note 2. Refer to "RH850/U2A-EVA User's Manual Table 21.19 List of Channel m Register"

3. Operation Examples

This following section describes the MSPI operation example. The MSPI register and bit settings that are not in this operation example are assumed to be set to the value after reset.

Table 3-1 Operation Example List

Operation Example	Unit/Channel	Communication Start Trigger	Operation Mode	Memory Mode
3.1 External Loopback Communication in Same Channels	MSPI0 ch0 (Master)	SW	Transmit/Receive	Direct memory
3.2 エラー! 参照元が見つかりません。	MSPI0 ch0 (Master)	SW	Transmit/Receive	Direct memory
	MSPI1 ch0 (Slave)	-	Transmit/Receive	Direct memory
3.3 Interrupt Communication during Frame Communication	MSPI0 ch0 (Master)	SW	Transmit/Receive	Fixed FIFO memory
	MSPI0 ch1 (Master)	HW	Transmit/Receive	Fixed buffer memory
	MSPI1 ch0 (Slave)	-	Transmit/Receive	Fixed FIFO memory
	MSPI2 ch0 (Slave)	-	Transmit/Receive	Fixed FIFO memory
3.4 Frame Communication with CRC between Two Channels	MSPI0 ch0 (Master)	SW	Transmit/Receive	Direct memory
	MSPI1 ch0 (Slave)	-	Transmit/Receive	Direct memory

3.1 External Loopback Communication in Same Channels

3.1.1 Specification Overview

In this operation example, the external loopback performs in the same MSPI channel. In this time, transmit/receive the 128 bits data in four parts. Also, receives the 128 bits data with transmitting in four parts.

- Connect the MSPI0 data outputs signal “MSPI0SO” and data input signal “MSPI0SI”.
- Uses the MSPI0 channel 0 in master. Set the data transmit mode to the transmit/receive mode (MSPI0CFG00.TXE0 = 1 and MSPI0CFG00.RXE= 1.) Set the baud rate to 10 Mbps.
- Set the frame length to 32 bits (MSPI0CFG02.FLEN0=0x20) and the frame count to four times (MSPI0CFSET0.CFSET0=4).
- Set the memory mode to the direct memory mode.
- Start the communication by writing the data to the transmit data register “MSPI0TXDA00”. The transmit data register writing is started when the interrupt “INTMSPI0TX0” is occurred.
- After received the data, read the received data from the receive data register “MSPI0RXDA00”. The reads are performed when the interrupt “INTMSPI0RX0” is occurred.
- After the last data is received, complete the communication when the interrupt “INTMSPI0FE0” is occurred.
- In this operation example, check the interrupts “INTMSPI0TX0” by MSPI0INTF0 register, the interrupts “INTMSPI0RX0” by MSPI0INTF1 register, and the interrupts “INTMSPI0FE0” by MSPI0INTF2 register.

Refer to “RH850/U2B User’s Manual Figure 21.42 Transmission/Reception in the Master-Direct Memory Mode” for the communication sequence details.

3.1.2 System Configuration

Figure 3-1 shows the system configuration.

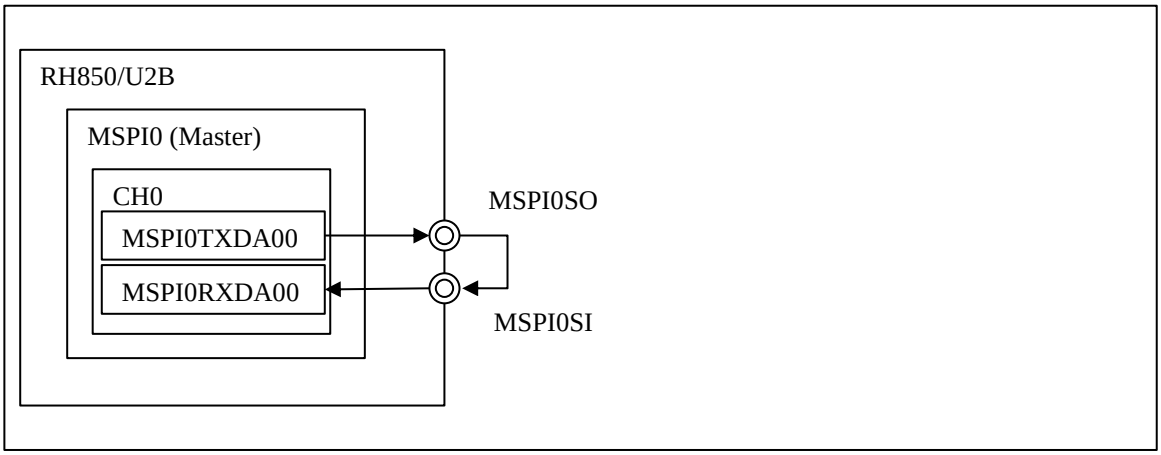

Figure 3-1 System Configuration

3.1.1 Software Explanation

3.1.1.1 Module Explanation

The followings show the module list in this operation example.

Table 3-2 Module

Module Name	Function Name	Function
MSPI initialization routine	mspi_init	Initialize MSPI.
MSPI activation	mspi0_ch0_activate	Enable MSPI0 CH0.
MSPI communication routine	mspi0_ch0_communicate	Transmit/Receive MSPI0 CH0 data.

3.1.1.2 Register Setting

The followings show the register initialization setting of each function in this operation example.

(a) MSPI0 Initial Setting

Table 3-3 MSPI0 Initial Setting

Register Name	Bit Name	Setting Value	Function
MSPI0CTL1	MSSEL	0	Master mode
	CSIE	0	In the master mode (MSPInMSSEL=0), MSPInCSIE must be set to 0.
	SAMP	1	The sampling timing of Master receive is next edge sampling point of SPI protocol.
	CKR	0	The default level of MSPInSCK is low.
	SOLS	0	Set MSPInSOUT to low after macro enable, and holds the level after each transfer.
	CSP	0	The MSPInCS signal is active low.
MSPI0CTL2	DCS	0	Disables the data consistency check.
	LBM	0	Disables the loop-back mode.
MSPI0CTL0	EN	1	Enables the MSPIn function.

(b) MSPI0 CH0 Initial Setting

Table 3-4 MSPI0 CH0 Initial Setting

Register Name	Bit Name	Setting Value	Function
MSPI0CFG00	TXE0	1	Transmission enabled.
	RXE0	1	Reception enabled.
	MD0	0	Direct memory mode
	PRIO0	7	Channel priority level 8(Lowest priority) (default)
	LOCK0	0	Disables the channel m lock operation.
	FCCE0	0	When a last frame ends, MSPInCHENm is cleared and the channel operation ends.
	IERE0	1	Enables the interrupt output.
	IFEE0	1	Enables the interrupt output.
	IRXE0	1	Enables the interrupt output.
	ITXE0	1	Enables the interrupt output.
MSPI0CFG01	CPOL0	0	MSPInSCK is low during idle time.
	CPHA0	1	Shifting bits out for transmission takes place on odd-numbered edges, and sampling for reception takes place on even-numbered edges.
	DIR0	0	Data is transmitted/received with MSB first.
	ICLS0	0	MSPInCS level is inactive for idle time.
	FIDL0	0	The idle time is not inserted each end of a frame.
	CSRI0	1	MSPInCS returns to the inactive level.
	SAFCM0	0	Does not mask the CRC error of the first frame.

Register Name	Bit Name	Setting Value	Function
	SAFS0	0	In-frame format
	SAFE0	0	Disables Safe SPI protocol function
	PS0	0	Transmission: Adds odd parity, Reception: Odd parity bit is expected.
	DECHK0	0	No parity check
MSPI0CFG02	FLEN0	0x20	32 bits (default)
MSPI0CFG03	PRCS0	0	MSPInSCK baud rate = $\frac{MSPInCLK}{4^{PRCSm} \times CDIVm \times 2}$ = 10MHz
	CDIV0	4	
MSPI0CFG04	HWTS0	0	HW trigger disabled. (Only SW trigger is effective)
	SIZE0	0	Set MSPInSIZEm[1:0] to 00 in the direct memory mode or the fixed buffer memory mode.
MSPI0SEUP0	SEUP0	2	MSPInSCK delay time = MSPInSEUPm [11: 0] × MSPInCLK.
MSPI0HOLD0	HOLD0	1	MSPInCS negation delay time = MSPInHOLDm [11: 0] × MSPInCLK
MSPI0IDLE0	IDLE0	1	MSPInCS[7: 0] next frame time = MSPInIDLEm [11: 0] × MSPInCLK.
MSPI0INDA0	INDA0	8	MSPInCS[7: 0] next frame time = MSPInINDAm [11: 0] × MSPInCLK.
MSPI0CFSET0	CFSET0	4	These bits set the number of frame count.
MSPI0SSEL0	JOBEN0	0	Job ends with this frame. After this frame the channel with a higher priority can transfer data.
	CSR0	1	Activates MSPInCS0 Deactivates MSPInCS[7:1]

3.1.1.3 Operation Flow

The flowchart in this application is shown below.

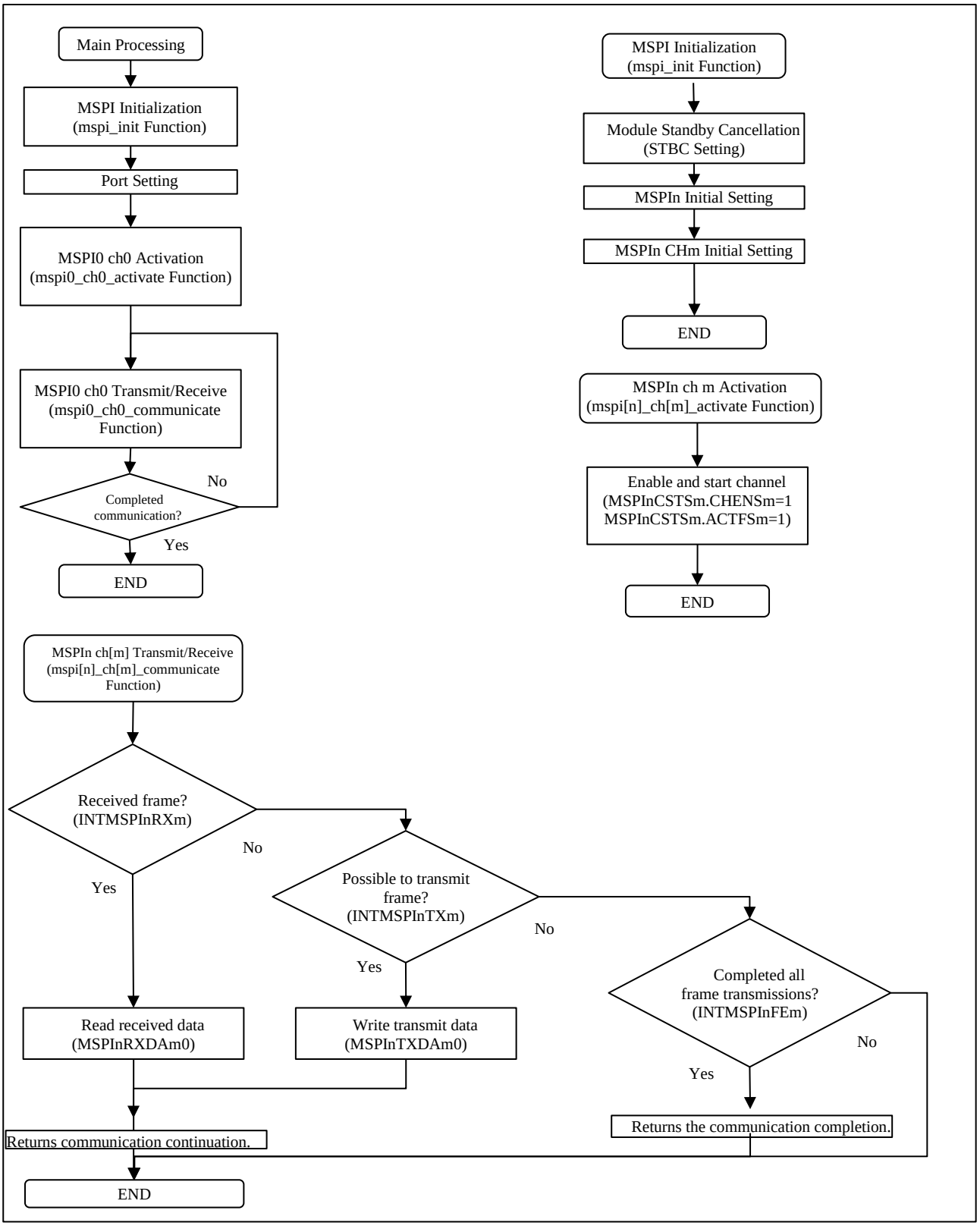

Figure 3-2 Flowchart

Refer to “RH850/U2B User’s Manual Figure 21.46 Master Transmission/Reception Operating Procedure in Direct Memory Mode” for the details.

3.2 External Loopback Communication between Two Channels

3.2.1 Specification Overview

In this operation example, perform the external loopback between two MSPI channels. In this time, transmit/receive the 128 bits data in four parts. Also, receives the 128 bits data with transmitting in four parts.

- Use MSPI0 channel 0 in master, MSPI1 channel 0 in salve.
- Connect each pin of MSPI0 and MSPI1 (Data output signal MSPI0SO and data input signal MSPI1SI, data input signal MSPI0SI and data output signal MSPI1SO, transmit clock MSPI0SC and MSPI1SC, chip select signal MSPI0CSS0 and MSPI1SSI).
- Set the data transmit mode to transmit/receive mode with each channel (MSPInCFGm0.TXEm = 1 and MSPInCFGm0.RXEm = 1). Set the baud rate to 10Mbps for the master channel only.
- Set frame length to 32 bits (MSPInCFGm2.FLENm=0x20) with each channel, and frame count to four times (MSPInCFSETm.CFSETm=4).
- Sets the memory mode to the direct memory mode.
- Starts the communication by writing the data to the transmit data register of master channel “MSPInTXDAm0”. The transmit data register writing is started when the interrupt “INTMSPInTXm” is occurred.
- After received the data, reads the received data of each channel from the receive data register “MSPInRXDAm0”. The reads are performed when the interrupt “INTMSPInRXm” is occurred
- After the last data is received, complete the communication when the interrupt “INTMSPInFEm” is occurred.
- In this operation example, check the interrupts “INTMSPInTXm” by MSPInINTFm register, the interrupts “INTMSPInRXm” by MSPI0INTF1 register, and the interrupts “INTMSPInFEm” by MSPI0INTF2 register.

Refer to “RH850/U2B User’s Manual Figure 21.42 Transmission/Reception in the Master-Direct Memory Mode” for the communication sequence details.

3.2.2 System Configuration

Figure 3-3 shows the system configuration.

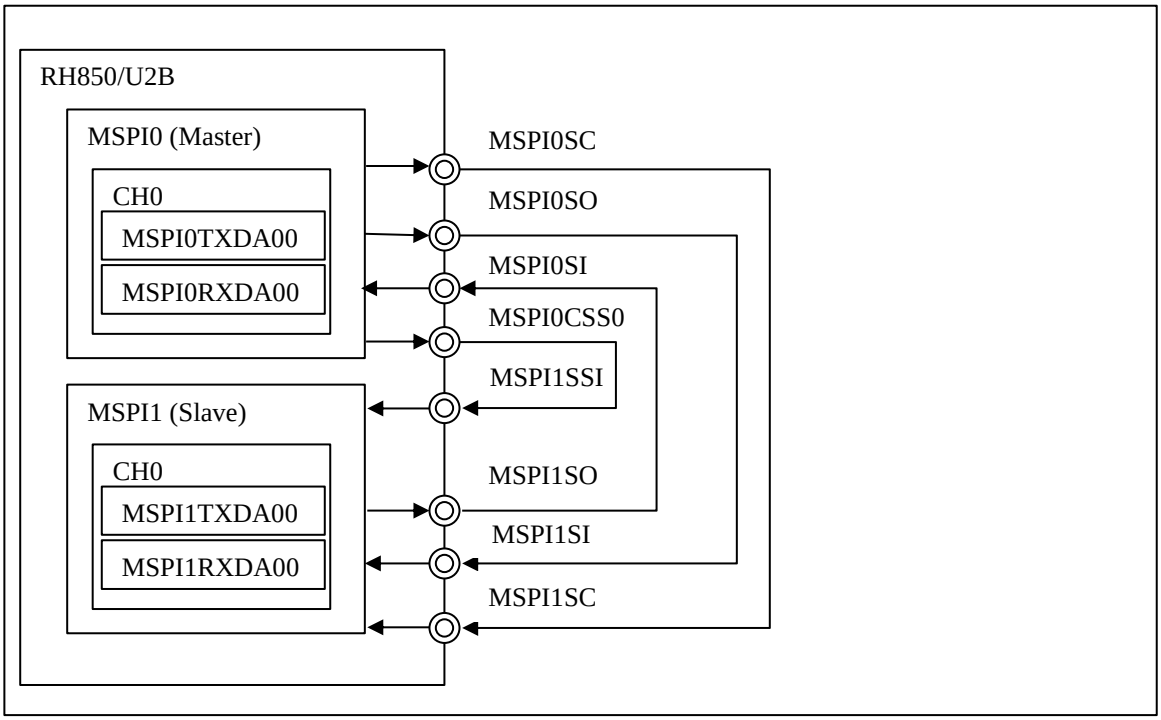

Figure 3-3 System Configuration

3.2.3 Software Explanation

3.2.3.1 Module Explanation

The following shows the module list in this operation example.

Table 3-5 Module

Module Name	Function Name	Function
MSPI initialization routine	mspi_init	Initialize MSPI.
MSPI activation	mspi0_ch0_activate	Enable MSPI0 CH0.
	mspi1_ch0_activate	Enable MSPI1 CH0.
MSPI transmit/receive routine	mspi0_ch0_communicate	Transmit/Receive MSPI0 CH0 data.
	mspi1_ch0_communicate	Transmit/Receive MSPI1 CH0 data.

3.2.3.2 Register Setting

The following shows the register initialization setting of each function.

(a) MSPI0 Initial Setting

Table 3-6 MSPI0 Initial Setting

Register Name	Bit Name	Setting Value	Function
MSPI0CTL1	MSSEL	0	Master mode
	CSIE	0	In the master mode (MSPInMSSEL=0), MSPInCSIE must be set to 0.
	SAMP	1	The sampling timing of Master receive is next edge sampling point of SPI protocol.
	CKR	0	The default level of MSPInSCK is low.
	SOLS	0	Set MSPInSOUT to low after macro enable, and holds the level after each transfer.
	CSP	0	The MSPInCS signal is active low.
MSPI0CTL2	DCS	0	Disables the data consistency check.
	LBM	0	Disables the loop-back mode.
MSPI0CTL0	EN	1	Enables the MSPIn function.

(b) MSPI0 CH0 Initial Setting

Table 3-7 MSPI0 CH0 Initial Setting

Register Name	Bit Name	Setting Value	Function
MSPI0CFG00	TXE0	1	Transmission enabled.
	RXE0	1	Reception enabled.
	MD0	0	Direct memory mode
	PRIO0	7	Channel priority level 8(Lowest priority) (default)
	LOCK0	0	Disables the channel m lock operation.
	FCCE0	0	When a last frame ends, MSPInCHENm is cleared and the channel operation ends.
	IERE0	1	Enables the interrupt output.
	IFEE0	1	Enables the interrupt output.

Register Name	Bit Name	Setting Value	Function
	IRXE0	1	Enables the interrupt output.
	ITXE0	1	Enables the interrupt output.
MSPI0CFG01	CPOL0	0	MSPInSCK is low during idle time.
	CPHA0	1	Shifting bits out for transmission takes place on odd-numbered edges, and sampling for reception takes place on even-numbered edges.
	DIR0	0	Data is transmitted/received with MSB first.
	ICLS0	0	MSPInCS level is inactive for idle time.
	FIDL0	0	The idle time is not inserted each end of a frame.
	CSRI0	1	MSPInCS returns to the inactive level.
	SAFCM0	0	Does not mask the CRC error of the first frame.
	SAFS0	0	In-frame format
	SAFE0	0	Disables Safe SPI protocol function
	PS0	0	Transmission: Adds odd parity, Reception: Odd parity bit is expected.
	DECHK0	0	No parity check
MSPI0CFG02	FLEN0	0x20	32 bits (default)
MSPI0CFG03	PRCS0	0	MSPInSCK baud rate = $\frac{MSPInCLK}{4^{PRCSm} \times CDIVm \times 2}$ = 10MHz
	CDIV0	4	
MSPI0CFG04	HWTS0	0	HW trigger disabled. (Only SW trigger is effective)
	SIZE0	0	Set MSPInSIZEm[1:0] to 00 in the direct memory mode or the fixed buffer memory mode.
MSPI0SEUP0	SEUP0	2	MSPInSCK delay time = MSPInSEUPm [11: 0] × MSPInCLK.
MSPI0HOLD0	HOLD0	1	MSPInCS negation delay time = MSPInHOLDm [11: 0] × MSPInCLK
MSPI0IDLE0	IDLE0	1	MSPInCS[7: 0] next frame time = MSPInIDLEm [11: 0] × MSPInCLK.
MSPI0INDA0	INDA0	8	MSPInCS[7: 0] next frame time = MSPInINDAm [11: 0] × MSPInCLK.
MSPI0CFSET0	CFSET0	4	These bits set the number of frame count.
MSPI0SSEL0	JOBEN0	0	Job ends with this frame. After this frame the channel with a higher priority can transfer data.
	CSR0	1	Activates MSPInCS0 for the communication. Deactivates MSPInCS[7: 1] for the communication.

(c) MSPI1 Initial Setting

Table 3-8 MSPI1 Initial Setting

Register Name	Bit Name	Setting Value	Function
MSPI1CTL1	MSSEL	1	Slave mode
	CSIE	1	Input CS signal (MSPInCSI) is recognized in slave mode.
	SAMP	0	In the slave mode (MSPInMSSEL=1), MSPInSAMP must be set to 0.
	CKR	0	The default level of MSPInSCK is low.
	SOLS	0	In the slave mode (MSPInMSSEL=1), MSPInSOLS[1:0] must be set to 00.
	CSP	0	In the slave mode (MSPInMSSEL=1), MSPInCSP[7:0] must be set to 00H.
MSPI1CTL2	DCS	0	Disables the data consistency check.
	LBM	0	In slave mode (MSPInMSSEL=1), MSPInLBM must be set to 0.
MSPI1CTL0	EN	1	Enables the MSPIn function.

(d) MSPI1 CH0 Initial Setting

Table 3-9 MSPI1 CH0 Initial Setting

Register Name	Bit Name	Setting Value	Function
MSPI1CFG00	TXE0	1	Transmission enabled.
	RXE0	1	Reception enabled.
	MD0	0	Direct memory mode
	PRIO0	7	Set MSPInPRIOm to 111_B in the slave mode.
	LOCK0	0	Set MSPInLOCKm to 0 in the slave mode.
	FCCE0	0	When a last frame ends, MSPInCHENm is cleared and the channel operation ends.
	IERE0	1	Enables the interrupt output.
	IFEE0	1	Enables the interrupt output.
	IRXE0	1	Enables the interrupt output.
	ITXE0	1	Enables the interrupt output.
MSPI1CFG01	CPOL0	0	Set MSPInCPOLm to 0 in the slave mode.
	CPHA0	1	Shifting bits out for transmission takes place on odd-numbered edges, and sampling for reception takes place on even-numbered edges.
	DIR0	0	Data is transmitted/received with MSB first.
	ICLS0	0	Set MSPInICLSm to 0 in the slave mode.
	FIDL0	0	Set MSPInFIDLm to 0 in the slave mode.
	CSRI0	0	Set MSPInCSRIm to 0 in the slave mode.
	SAFCM0	0	Does not mask the CRC error of the first frame.
	SAFS0	0	In-frame format

Register Name	Bit Name	Setting Value	Function
	SAFE0	0	Set MSPInSAFEm to 0 in the slave mode.
	PS0	0	Transmission: Adds odd parity, Reception: Odd parity bit is expected.
	DECHK0	0	No parity check
MSPI1CFG02	FLEN0	0x20	32 bits (default)
MSPI1CFG03	PRCS0	0	In the slave mode (MSPInMSSEL=1), MSPInCFGm3 must be set to 0001_H(default value).
	CDIV0	1	
MSPI1CFG04	HWTS0	0	HW trigger disabled. (Only SW trigger is effective)
	SIZE0	0	Set MSPInSIZEm[1:0] to 00 in the direct memory mode or the fixed buffer memory mode.
MSPI1SEUP0	SEUP0	1	Set MSPInSEUPm to 0001_H in the slave mode.
MSPI1HOLD0	HOLD0	1	Set MSPInHOLDm to 0001_H in slave mode.
MSPI1IDLE0	IDLE0	1	Set MSPInIDLEm to 0001_H in slave mode.
MSPI1INDA0	INDA0	0	Set MSPInINDAm to 0000_H in slave mode.
MSPI1CFSET0	CFSET0	4	These bits set the number of frame count.
MSPI1SSEL0	JOBEN0	0	Set MSPInSSELm to 0000_H in the slave mode.
	CSR0	0	

3.2.3.3 Operation Flow

The flowchart in this application is shown below.

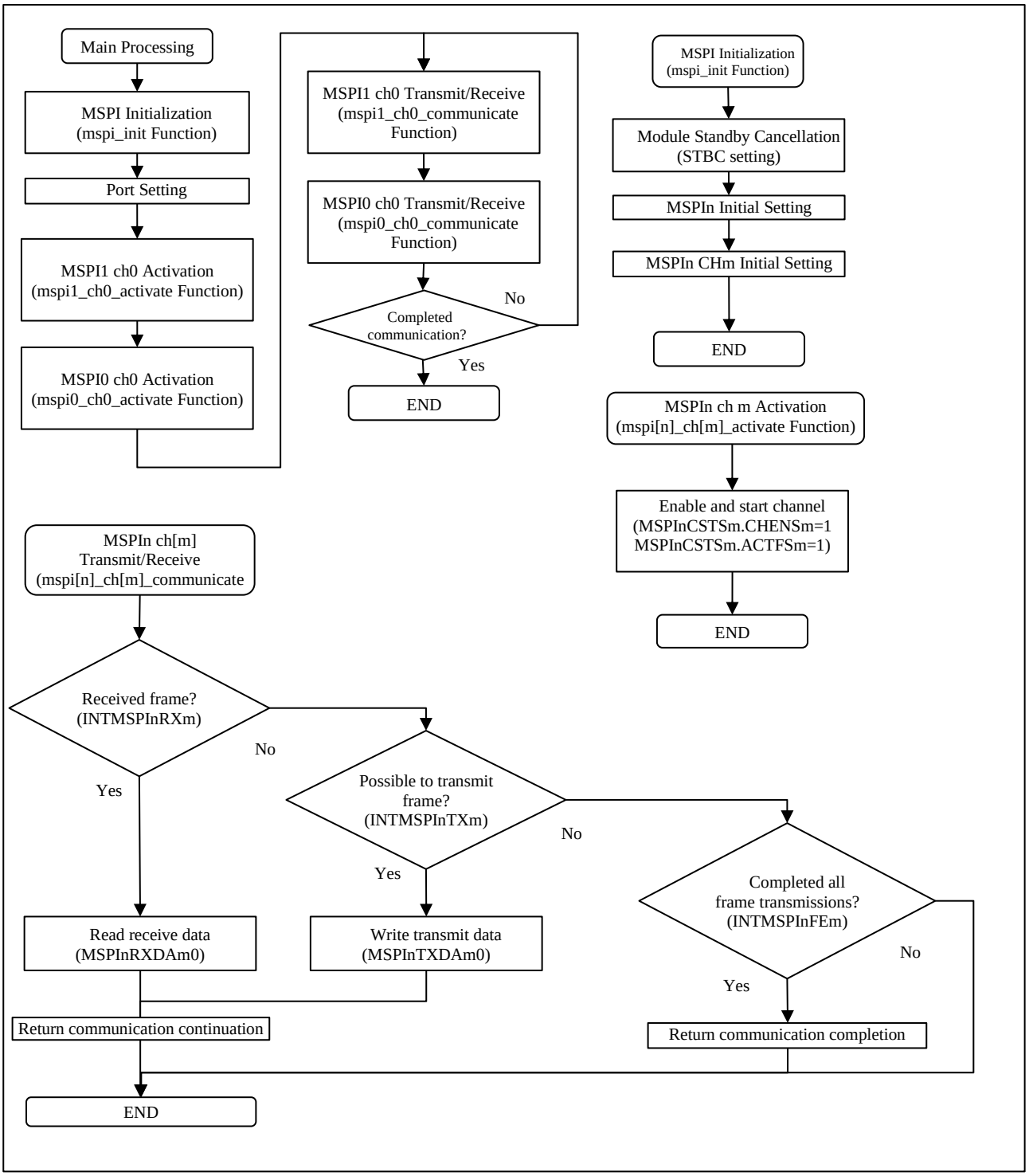

Figure 3-4 Operation Flow

Refer to “RH850/U2B User’s Manual Figure 21.46 Master Transmission/Reception Operating Procedure in Direct Memory Mode, Figure 21.49 Slave Transmission/Reception Operating Procedure in Direct Memory Mode’ for the details of the flowchart.

3.3 Interrupt Communication during Frame Communication

3.3.1 Specification Overview

In this operation example, performs the external loopback communication between four MSPI channels: two master channels and two slave channels. The master channel and slave channel transmit 256 bits data and simultaneously receive 256 bits. When requests the communication start to high-priority master channel by HW trigger or SW trigger during continuous communication of low-priority master channel, the high-priority communication interrupts to the low-priority continuous communication. Resume the left low-priority continuous communication after interrupt completed.

- Use MSPI0 channnel0 and channel1 in master, and MSPI1 channel0 and MSPI2 channel0 in slave.
- Connect the pins of MSPI0, MSPI1 and MSPI2 (Data output signal “MSPInSO” and data input signal “MSPInSI”, transmit clock “MSPInSC” to each other, chip select signal “MSPInCSSm” and “MSPInSSI”) as following.

MSPI0	MSPI1	MSPI2
MSPI0SO	MSPI1SI	MSPI2SI
MSPI0SI	MSPI1SO	MSPI2SO
MSPI0SC	MSPI1SC	MSPI2SC
MSPI0CSS0	MSPI1SSI	—
MSPI0CSS1	—	MSPI2SSI

- Set SW trigger for MSPI0 channel 0 (MSPI0CFG04.HWTS0=0). Set HW trigger for MSPI0 channel 1. Set the external interrupt pin IRQ0 as HW trigger (MSPI0CFG14. HWTS1=1).
- Set the high-priority of MSPI0 channel 1 more than MSPI0 channel 0 (MSPI0CFG00.PRIO0=7, MSPI0CFG01.PRIO1=0).
- Sets the transmit/receive mode (MSPInCFGm0.TXEm = 1 and MSPInCFGm0.RXEm = 1) to the data transmit mode for each channel. Set 10 Mbps to baud rate of master channel.
- Sets 32 bits to frame length (MSPInCFGm2.FLENm=0x20) for each channel and eight times to frame count (MSPInCFSETm.CFSETm=8).
- MSPI0 Channel 0 (Master)
  Transmit/Receive the 256 bits data in eight parts.
  - Set the memory mode to Fixed FIFO Memory Mode and the FIFO stage size to 8.
  - Start the communication by writing the 32 bits transmit data to the transmit data register MSPInTXDAm0. Perform the writing of transmit data register when INTMSPInTXm interrupts is occurred.
  - After 32 bits data reception, read the received data from receive data register of MSPInRXDAm0 for each channel. Perform the read of the receive data register when INTMSPInRXm interrupts is occurred.
  - After the 256 bits data reception, complete the communication when INTMSPInFEm interrupt is occurred.
  - In this operation example, check the INTMSPInTXm interrupt by MSPInINTFm register, INTMSPInRXm interrupt by MSPInINTF1 register, and INTMSPInFEm interrupt by MSPI0INTF2 register.
- MSPI0 Channel 1 (Master)
  Transmit/Receive the 256 bits data in one time.
  - Set the fixed buffer mormory mode to the memory mode.
  - After writing the 256 bits tranmit data to MSPI RAM allocated to MSPI channel 1, the comminiaction is started in SW trigger or HW trigger of MSPI0 channel1.
  - When INTMSPInFEm interrupt is occurred, the communication is ended, and the 256 bits receive data is read from MSPI RAM allocated to MSPI0 channel 1.
  - In this operation example, check the INTMSPInFEm interrupt by MSPI0INTF2 register.

- MSPI1 Channel 0 (Slave), MSPI2 Channel 0 (Slave)
  Transmit/Receive the 256 bits data in eight parts.
  - Set the memory mode to Fixed FIFO Memory Mode and the FIFO stage size to 8.
  - When the master channel srats the communication, the slave channel starts the communication.
  - Write the 32 bits transmit data to the transmit data register MSPInTXDAm0. The transmit data register writing is performed when the INTMSPInTXm interrupt is occurred.
  - After 32 bits data reception, read the received data from receive data register of MSPInRXDAm0 for each channel. Perform the read of the receive data register when INTMSPInRXm interrupts is occurred.
  - After the 256 bits data reception, complete the communication when INTMSPInFEm interrupt is occurred.
  - In this operation example, check the INTMSPInTXm interrupt by MSPInINTFm register, INTMSPInRXm interrupt by MSPInINTF1 register, and INTMSPInFEm interrupt by MSPI0INTF2 register.

- The transmit/receive data in the MSPI RAM area for each channel is stored as the following.

Table 3-10 MSPI RAM Area

Address	Offset	Data	MSPInRASTADm
<MSPI0_base> + 0x1000	0x0000	MSPI0 ch0 Transmit Frame 1	MSPI0RASTAD0=0x0000
:	:	:	
<MSPI0_base> + 0x101C	0x001C	MSPI0 ch0 Transmit Frame 8	
<MSPI0_base> + 0x1020	0x0020	MSPI0 ch0 Receive Frame 1	
:	:	:	
<MSPI0_base> + 0x103C	0x003C	MSPI0 ch0 Receive Frame 8	
<MSPI0_base> + 0x1040	0x0040	MSPI0 ch1 Transmit Frame 1	MSPI0RASTAD1=0x0040
:	:	:	
<MSPI0_base> + 0x105C	0x005C	MSPI0 ch1 Transmit Frame 8	
<MSPI0_base> + 0x1060	0x0060	MSPI0 ch1 Receive Frame 1	
:	:	:	
<MSPI0_base> + 0x107C	0x007C	MSPI0 ch1 Receive Frame 8	
<MSPI0_base> + 0x1080	:	Unused	
:	:		
<MSPI0_base> + 0x11FF	:		
<MSPI1_base> + 0x1000	0x0000	MSPI1 ch0 Transmit Frame 1	MSPI1RASTAD0=0x0000
:	:	:	
<MSPI1_base> + 0x101C	0x001C	MSPI1 ch0 Transmit Frame 8	
<MSPI1_base> + 0x1020	0x0020	MSPI1 ch0 Receive Frame 1	
:	:	:	
<MSPI1_base> + 0x103C	0x003C	MSPI1 ch0 Receive Frame 8	
<MSPI1_base> + 0x1040	:	Unused	
:	:		
<MSPI1_base> + 0x11FF	:		
<MSPI2_base> + 0x1000	0x0000	MSPI2 ch0 Transmit Frame 1	MSPI2RASTAD0=0x0000
:	:	:	
<MSPI2_base> + 0x101C	0x001C	MSPI2 ch0 Transmit Frame 8	
<MSPI2_base> + 0x1020	0x0020	MSPI02 ch0 Receive Frame 1	
:	:	:	
<MSPI2_base> + 0x103C	0x003C	MSPI02 ch0 Receive Frame 8	
<MSPI2_base> + 0x1040	:	Unused	
:	:		
<MSPI2_base> + 0x11FF	:		

The following is the example where the communication between MSPI0 channel 1 and MSPI2 channel 0 interrupts by H/W trigger during continuous communication between MSPI0 channel 0 and MSPI1 channel 0 in this operation example.

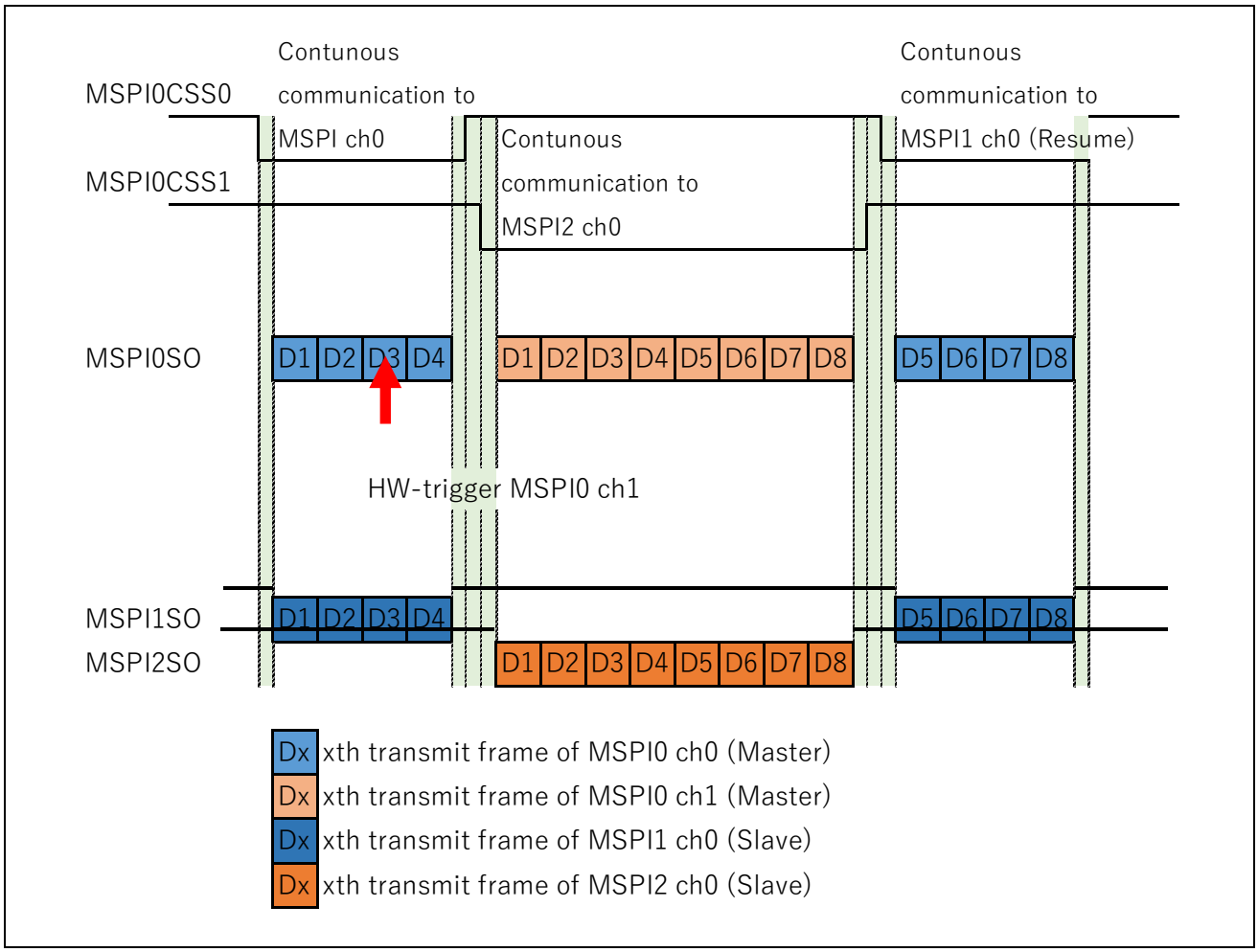

Figure 3-5 Interrupt Communication during Frame Communication

Note In this operation example, set I/O port of data output signal of slave to PIPCn.PIPCn_m bit =1 for Hi-Z control of the data output signal MSPInSO for slaves while the chip select signal MSPInCSSm is inactive. Refer to “RH850/U2B User’s Manua Table 2.29 Alternative functions require “Direct I/O Control” (Must Set PIPCn_m=1)” for the details.

Refer to ”RH850/U2B User’s Manua Figure 21.24 Timing Diagram of the Channel Priority Judgement” for the detail of communication interrupt during frame communication.

3.3.2 System Configuration

Figure 3-6 shows the system configuration.

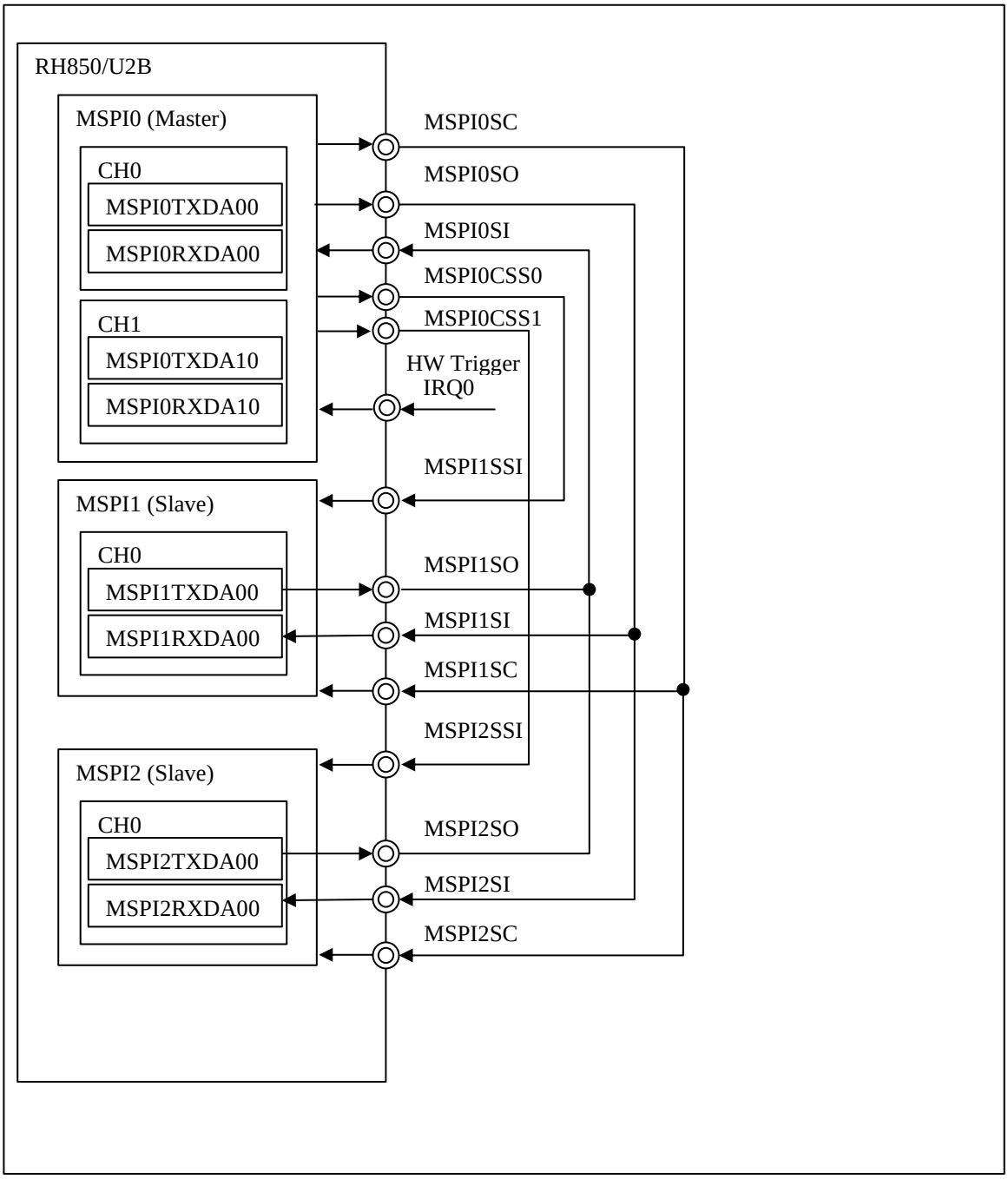

Figure 3-6 System Configuration

3.3.3 Software Explanation

3.3.3.1 Module Explanation

The following shows the mdule list in this operation example.

Table 3-11 Module

Module Name	Function Name	Function
MSPI initialization routine	mspi_init	Initialize MSPI.
MSPI activation	mspi0_ch0_activate	Enable MSPI0 CH0.
	mspi0_ch1_activate	Enable MSPI0 CH1.
	mspi1_ch0_activate	Enable MSPI1 CH0.
	mspi2_ch0_activate	Enable MSPI2 CH0.
MSPI transmit/receive routine	mspi0_ch0_communicate	Transmit/receive MSPI0 CH0 data.
	mspi0_ch1_communicate	Transmit/receive MSPI0 CH1 data.
	mspi1_ch0_communicate	Transmit/receive MSPI1 CH0 data.
	mspi2_ch0_communicate	Transmit/receive MSPI2 CH0 data.

3.3.3.2 Register Setting

Register initial setting of each function in this operation example is shown below.

(a) MSPI0 Initial Setting

Table 3-12 MSPI0 Initial Setting

Register Name	Bit Name	Setting Value	Function
MSPI0CTL1	MSSEL	0	Master mode
	CSIE	0	In the master mode (MSPInMSSEL=0), MSPInCSIE must be set to 0.
	SAMP	1	The sampling timing of Master receive is next edge sampling point of SPI protocol.
	CKR	0	The default level of MSPInSCK is low.
	SOLS	0	Set MSPInSOUT to low after macro enable, and holds the level after each transfer.
	CSP	0	The MSPInCS signal is active low.
MSPI0CTL2	DCS	0	Disables the data consistency check.
	LBM	0	Disables the loop-back mode.
MSPI0CTL0	EN	1	Enables the MSPIn function.

(b) MSPI0 CH0 Initial Setting

Table 3-13 MSPI0 CH0 Initial Setting

Register Name	Bit Name	Setting Value	Function
MSPI0CFG00	TXE0	1	Transmission enabled.
	RXE0	1	Reception enabled.
	MD0	0	Fixed FIFO memory mode
	PRIO0	7	Channel priority level 8(Lowest priority) (default)

Register Name	Bit Name	Setting Value	Function
	LOCK0	0	Disables the channel m lock operation.
	FCCE0	0	Set MSPInFCCEm to 0 in the fixed buffer memory mode and fixed FIFO memory mode.
	IERE0	1	Enables the interrupt output.
	IFEE0	1	Enables the interrupt output.
	IRXE0	1	Enables the interrupt output.
	ITXE0	1	Enables the interrupt output.
MSPI0CFG01	CPOL0	0	MSPInSCK is low during idle time.
	CPHA0	1	Shifting bits out for transmission takes place on odd-numbered edges, and sampling for reception takes place on even-numbered edges.
	DIR0	0	Data is transmitted/received with MSB first.
	ICLS0	0	Set MSPInICLSm to 0 in the fixed buffer memory mode and fixed FIFO memory mode.
	FIDL0	0	The idle time is not inserted each end of a frame.
	CSRI0	1	MSPInCS returns to the inactive level.
	SAFCM0	0	Does not mask the CRC error of the first frame.
	SAFS0	0	In-frame format
	SAFE0	0	Disables Safe SPI protocol function
	PS0	0	Transmission: Adds odd parity, Reception: Odd parity bit is expected.
	DECHK0	0	No parity check
MSPI0CFG02	FLEN0	0x20	32 bits (default)
MSPI0CFG03	PRCS0	0	MSPInSCK baud rate = $\frac{MSPInCLK}{4^{PRCSm} \times CDIVm \times 2}$ = 10MHz
	CDIV0	4	
MSPI0CFG04	HWTS0	0	HW trigger disabled. (Only SW trigger is effective)
	SIZE0	0	The stage size of Buffer is 8 in Fixed FIFO memory mode.
MSPI0SEUP0	SEUP0	2	MSPInSCK delay time = MSPInSEUPm [11: 0] × MSPInCLK.
MSPI0HOLD0	HOLD0	1	MSPInCS negation delay time = MSPInHOLDm [11: 0] × MSPInCLK
MSPI0IDLE0	IDLE0	1	MSPInCS[7: 0] next frame time = MSPInIDLEm [11: 0] × MSPInCLK.
MSPI0INDA0	INDA0	8	MSPInCS[7: 0] next frame time = MSPInINDAm [11: 0] × MSPInCLK.
MSPI0CFSET0	CFSET0	8	These bits set the number of frame count.
MSPI0SSEL0	JOBEN0	0	Set MSPInJOBENm to 0 in the fixed buffer memory mode, fixed FIFO memory mode, or slave mode.
	CSR0	1	Activates MSPInCS0 for the communication. Deactivates MSPInCS[7: 1] for the communication.
MSPI0RASTAD0	—	0x0000	Start address of MSPI RAM in Fixed FIFO memory mode or fixed buffer memory mode.

(c) MSPI0 CH1 Initial Setting

Table 3-14 MSPI0 CH1 Initial Setting

Regitser Name	Bit Name	Setting Value	Function
MSPI0CFG10	TXE1	1	Transmission enabled.
	RXE1	1	Reception enabled.
	MD1	1	Fixed buffer memory mode
	PRIO1	0	Channel priority level 0(Highest priority)
	LOCK1	1	Enables the channel m lock operation.
	FCCE1	0	Set MSPInFCCEm to 0 in the fixed buffer memory mode and fixed FIFO memory mode.
	IERE1	1	Enables the interrupt output.
	IFEE1	1	Enables the interrupt output.
	IRXE1	1	Enables the interrupt output.
	ITXE1	1	Enables the interrupt output.
MSPI0CFG11	CPOL1	0	MSPInSCK is low during idle time.
	CPHA1	1	Shifting bits out for transmission takes place on odd-numbered edges, and sampling for reception takes place on even-numbered edges.
	DIR1	0	Data is transmitted/received with MSB first.
	ICLS1	0	Set MSPInICLSm to 0 in the fixed buffer memory mode and fixed FIFO memory mode.
	FIDL1	0	The idle time is not inserted each end of a frame.
	CSRI1	1	MSPInCS returns to the inactive level.
	SAFCM1	0	Does not mask the CRC error of the first frame.
	SAFS1	0	In-frame format
	SAFE1	0	Disables Safe SPI protocol function
	PS1	0	Transmission: Adds odd parity, Reception: Odd parity bit is expected.
	DECHK1	0	No parity check
MSPI0CFG12	FLEN1	0x20	32 bits (default)
MSPI0CFG13	PRCS1	0	MSPInSCK baud rate = $\frac{MSPInCLK}{4^{PRCSm} \times CDIVm \times 2}$ = 10MHz
	CDIV1	4	
MSPI0CFG14	HWTS1	1	HW trigger enabled by External Interrupt 0 (IRQ0). SW trigger is also enabled.
	SIZE1	0	Set MSPInSIZEm[1:0] to 00 in the direct memory mode or the fixed buffer memory mode.
MSPI0SEUP1	SEUP1	2	MSPInSCK delay time = MSPInSEUPm [11: 0] × MSPInCLK.
MSPI0HOLD1	HOLD1	1	MSPInCS negation delay time = MSPInHOLDm [11: 0] × MSPInCLK
MSPI0IDLE1	IDLE1	1	MSPInCS[7: 0] next frame time = MSPInIDLEm [11: 0] × MSPInCLK.

Regitser Name	Bit Name	Setting Value	Function
MSPI0INDA1	INDA1	8	MSPInCS[7: 0] next frame time = MSPInINDAm [11: 0] × MSPInCLK.
MSPI0CFSET1	CFSET1	8	These bits set the number of frame count.
MSPI0SSEL1	JOBEN1	0	Set MSPInJOBENm to 0 in the fixed buffer memory mode, fixed FIFO memory mode, or slave mode.
	CSR1	2	Activates MSPInCS1 for the communication. Deactivates MSPInCS[7: 2] and MSPInCS[0] for the communication.
MSPI0RASTAD1	—	0x0040	Start address of MSPI RAM in Fixed FIFO memory mode or fixed buffer memory mode.

(d) MSPI1 Initial Setting

Table 3-15 MSPI1 Initial Setting

Register Name	Bit Name	Setting Value	Function
MSPI1CTL1	MSSEL	1	Slave mode
	CSIE	1	Input CS signal (MSPInCSI) is recognized in slave mode.
	SAMP	0	In the slave mode (MSPInMSSEL=1), MSPInSAMP must be set to 0.
	CKR	0	The default level of MSPInSCK is low.
	SOLS	0	In the slave mode (MSPInMSSEL=1), MSPInSOLS[1:0] must be set to 00.
	CSP	0	In the slave mode (MSPInMSSEL=1), MSPInCSP[7:0] must be set to 00H.
MSPI1CTL2	DCS	0	Disables the data consistency check.
	LBM	0	In slave mode (MSPInMSSEL=1), MSPInLBM must be set to 0.
MSPI1CTL0	EN	1	Enables the MSPIn function.

(e) MSPI1 CH0 Initial Setting

Table 3-16 MSPI1 CH0 Initial Setting

Register Name	Bit Name	Setting Value	Function
MSPI1CFG00	TXE0	1	Transmission enabled.
	RXE0	1	Reception enabled.
	MD0	0	Fixed FIFO memory mode
	PRIO0	7	Set MSPInPRIOm to 111_B in the slave mode.
	LOCK0	0	Set MSPInLOCKm to 0 in the slave mode.
	FCCE0	0	Set MSPInFCCEm to 0 in the fixed buffer memory mode and fixed FIFO memory mode.
	IERE0	1	Enables the interrupt output.
	IFEE0	1	Enables the interrupt output.
	IRXE0	1	Enables the interrupt output.

Register Name	Bit Name	Setting Value	Function
	ITXE0	1	Enables the interrupt output.
MSPI1CFG01	CPOL0	0	Set MSPInCPOLm to 0 in the slave mode.
	CPHA0	1	Shifting bits out for transmission takes place on odd-numbered edges, and sampling for reception takes place on even-numbered edges.
	DIR0	0	Data is transmitted/received with MSB first.
	ICLS0	0	Set MSPInICLSm to 0 in the slave mode. Set MSPInICLSm to 0 in the fixed buffer memory mode and fixed FIFO memory mode.
	FIDL0	0	Set MSPInFIDLm to 0 in the slave mode.
	CSRI0	0	Set MSPInCSRIm to 0 in the slave mode.
	SAFCM0	0	Does not mask the CRC error of the first frame.
	SAFS0	0	In-frame format
	SAFE0	0	Set MSPInSAFEm to 0 in the slave mode.
	PS0	0	Transmission: Adds odd parity, Reception: Odd parity bit is expected.
	DECHK0	0	No parity check
MSPI1CFG02	FLEN0	0x20	32 bits (default)
MSPI1CFG03	PRCS0	0	In the slave mode (MSPInMSSEL=1), MSPInCFGm3 must be set to 0001_H(default value).
	CDIV0	1	
MSPI1CFG04	HWTS0	0	HW trigger disabled. (Only SW trigger is effective)
	SIZE0	0	The stage size of Buffer is 8 in Fixed FIFO memory mode.
MSPI1SEUP0	SEUP0	1	Set MSPInSEUPm to 0001_H in the slave mode.
MSPI1HOLD0	HOLD0	1	Set MSPInHOLDm to 0001_H in slave mode.
MSPI1IDLE0	IDLE0	1	Set MSPInIDLEm to 0001_H in slave mode.
MSPI1INDA0	INDA0	0	Set MSPInINDAm to 0000_H in slave mode.
MSPI1CFSET0	CFSET0	8	These bits set the number of frame count.
MSPI1SSEL0	JOBEN0	0	Set MSPInSSELm to 0000_H in the slave mode.
	CSR0	0	
MSPI1RASTAD0	—	0x0000	Start address of MSPI RAM in Fixed FIFO memory mode or fixed buffer memory mode.

(f) MSPI2 Initial Setting

Table 3-17 MSPI2 Initial Setting

Register Name	Bit Name	Setting Value	Function
MSPI2CTL1	MSSEL	1	Slave mode
	CSIE	1	Input CS signal (MSPInCSI) is recognized in slave mode.
	SAMP	0	In the slave mode (MSPInMSSEL=1), MSPInSAMP must be set to 0.
	CKR	0	The default level of MSPInSCK is low.

	SOLS	0	In the slave mode (MSPInMSSEL=1), MSPInSOLS[1:0] must be set to 00.
	CSP	0	In the slave mode (MSPInMSSEL=1), MSPInCSP[7:0] must be set to 00H.
MSPI2CTL2	DCS	0	Disables the data consistency check.
	LBM	0	In slave mode (MSPInMSSEL=1), MSPInLBM must be set to 0.
MSPI2CTL0	EN	1	Enables the MSPIn function.

(g) MSPI2 CH0 Initial Setting

Table 3-18 MSPI2 CH0 Initial Setting

Register Name	Bit Name	Setting Value	Function
MSPI2CFG00	TXE0	1	Transmission enabled.
	RXE0	1	Reception enabled.
	MD0	0	Fixed FIFO memory mode
	PRIO0	7	Set MSPInPRIOm to 111_B in the slave mode.
	LOCK0	0	Set MSPInLOCKm to 0 in the slave mode.
	FCCE0	0	Set MSPInFCCEm to 0 in the fixed buffer memory mode and fixed FIFO memory mode.
	IERE0	1	Enables the interrupt output.
	IFEE0	1	Enables the interrupt output.
	IRXE0	1	Enables the interrupt output.
	ITXE0	1	Enables the interrupt output.
MSPI2CFG01	CPOL0	0	Set MSPInCPOLm to 0 in the slave mode.
	CPHA0	1	Shifting bits out for transmission takes place on odd-numbered edges, and sampling for reception takes place on even-numbered edges.
	DIR0	0	Data is transmitted/received with MSB first.
	ICLS0	0	Set MSPInICLSm to 0 in the slave mode. Set MSPInICLSm to 0 in the fixed buffer memory mode and fixed FIFO memory mode.
	FIDL0	0	Set MSPInFIDLm to 0 in the slave mode.
	CSRI0	0	Set MSPInCSRIm to 0 in the slave mode.
	SAFCM0	0	Does not mask the CRC error of the first frame.
	SAFS0	0	In-frame format
	SAFE0	0	Set MSPInSAFEm to 0 in the slave mode.
	PS0	0	Transmission: Adds odd parity, Reception: Odd parity bit is expected.
	DECHK0	0	No parity check
MSPI2CFG02	FLEN0	0x20	32 bits (default)
MSPI2CFG03	PRCS0	0	In the slave mode (MSPInMSSEL=1), MSPInCFGm3 must be set to 0001_H(default value).
	CDIV0	1	

Register Name	Bit Name	Setting Value	Function
MSPI2CFG04	HWTS0	0	HW trigger disabled. (Only SW trigger is effective)
	SIZE0	0	The stage size of Buffer is 8 in Fixed FIFO memory mode.
MSPI2SEUP0	SEUP0	1	Set MSPInSEUPm to 0001_H in the slave mode.
MSPI2HOLD0	HOLD0	1	Set MSPInHOLDm to 0001_H in slave mode.
MSPI2IDLE0	IDLE0	1	Set MSPInIDLEm to 0001_H in slave mode.
MSPI2INDA0	INDA0	0	Set MSPInINDAm to 0000_H in slave mode.
MSPI2CFSET0	CFSET0	8	These bits set the number of frame count.
MSPI2SSEL0	JOBEN0	0	Set MSPInSSELm to 0000_H in the slave mode.
	CSR0	0	
MSPI2RASTAD0	—	0x0000	Start address of MSPI RAM in Fixed FIFO memory mode or fixed buffer memory mode.

3.3.3.3 Operation Flow

The flowchart in this operation example is shown below.

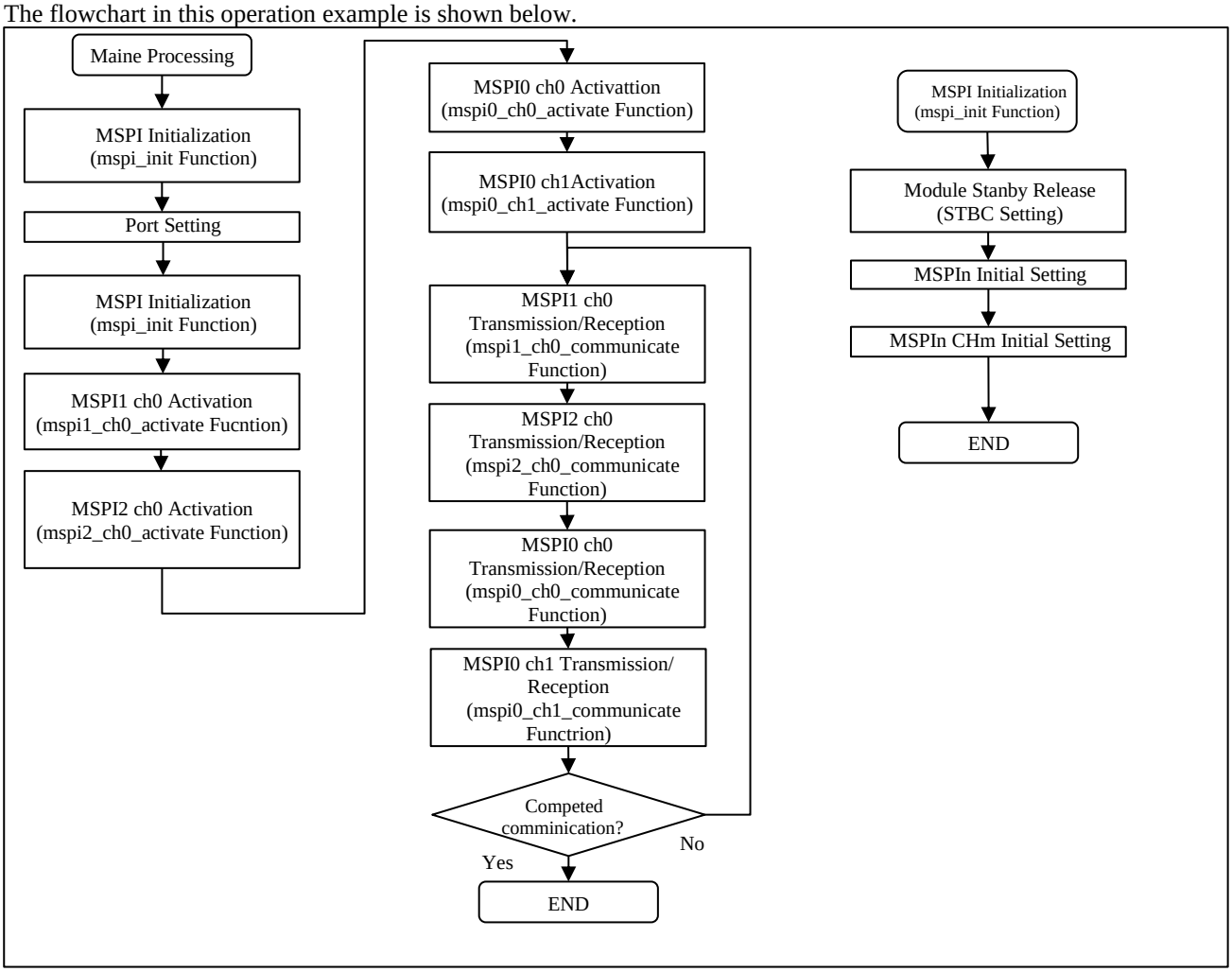

Figure 3-7 Operation Flow

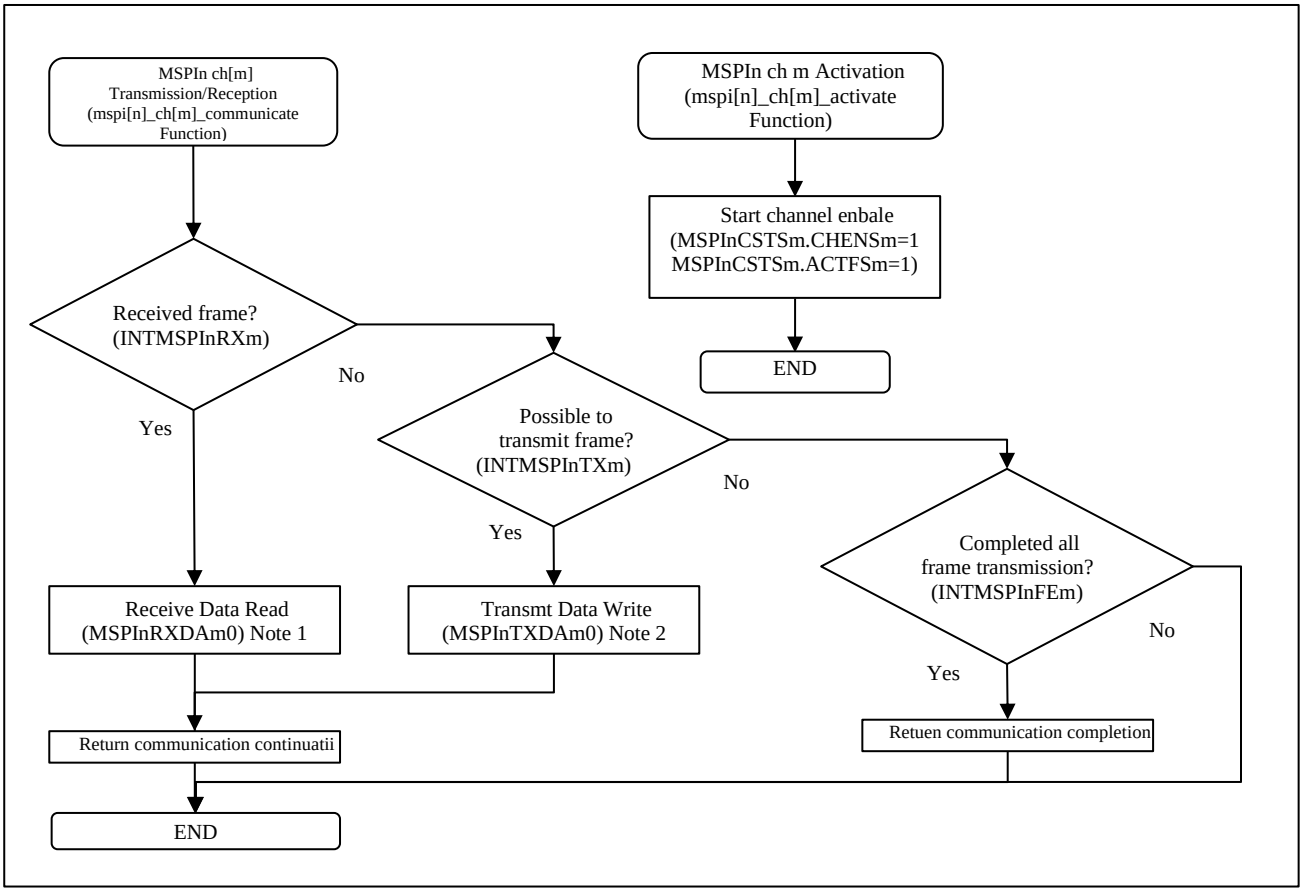

Figure 3-8 Operation Flow (n=0,m=0 or n=1,m=0 or n=2,m=0)

Note 1. Read half of FIFO stage size specified by MSPInCFGm4.MSPInSIZEm [1:0].
Note2. Write half of FIFO stage size specified by MSPInCFGm4.MSPInSIZEm [1:0].

Refer to “RH850/U2B User’s Manual Figure Figure 21.60 Master Transmission/Reception Operating Procedure in the Fixed FIFO Memory Mode, Figure 21.63 Slave Transmission/Reception Operating Procedure in the Fixed FIFO Memory Mode” for the flowchart details.

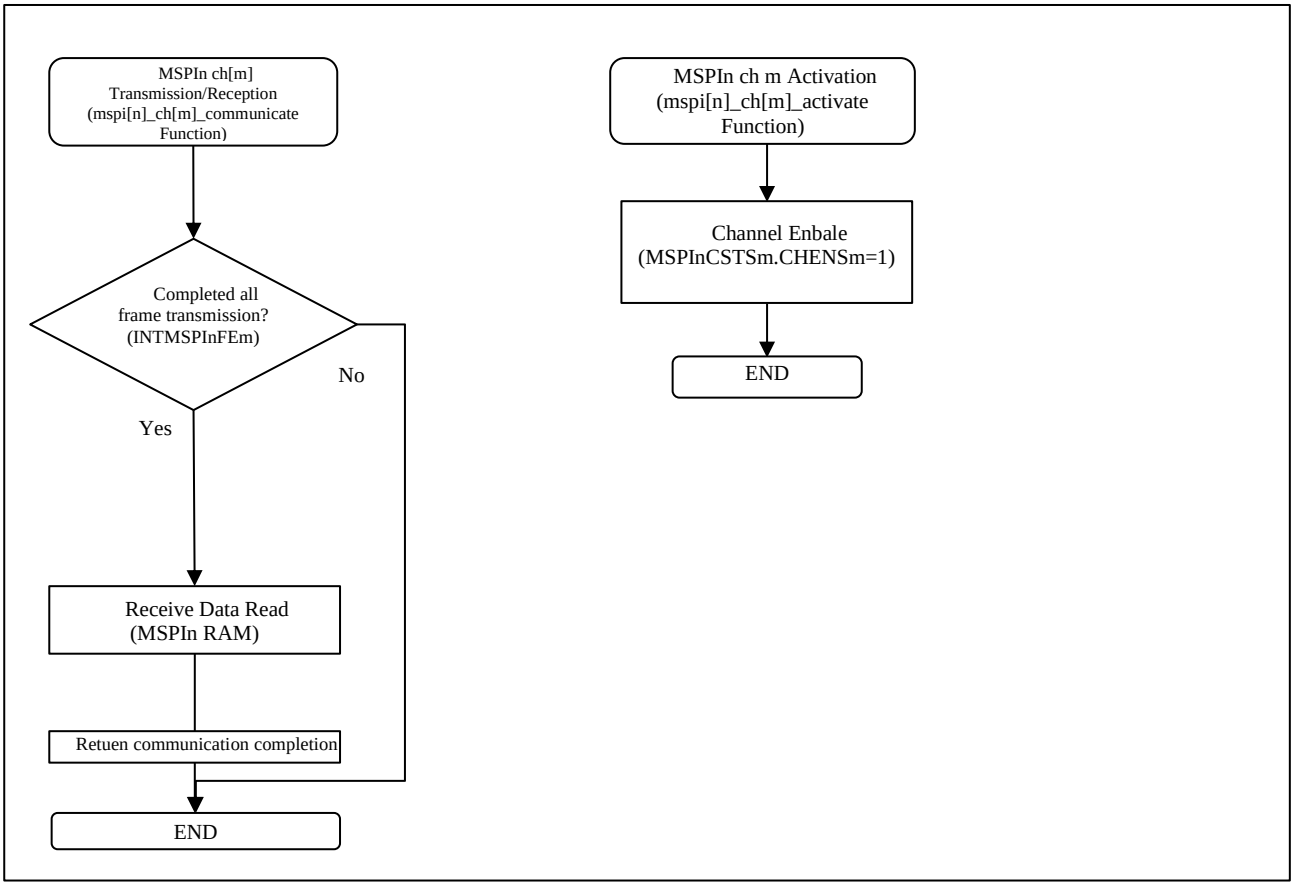

Figure 3-9 Operation Flow (n=0,m=1)

mspi0_ch1_activate function does not set MSPI0CSTS1.ACTFS1 bit since MSPI0 chanbnel 1 in this operation example starts the commnunication by HW trigger.

Refere to “RH850/U2B User’s Manual Figure Figure 21.55 Master Transmission/Reception or Transmission or Reception Operating Procedure in the Fixed Buffer Memory Mode” for the operation flow details.

3.4 Frame Communication with CRC between Two Channels

3.4.1 Specification Overview

In this operation example, performs the external loopback communication between 2 channels of MSPI with CRC. Transmit and receive two sets of 128 bits data as one set. Transmit/Receive 128 bits data in four parts.

- Use MSPI0 channel 0 in master, and MSPI1 channel 0 in slave.
- Connect each pin of MSPI0 and MSPI1 (Data output signal “MSPI0SO” and data input signal “MSPI1SI”, Data input signal “MSPI0SI” and data output signal “MSPI1SO”, Transmit clock “MSPI0SC” and “MSPI1SC”, chip select signal “MSPI0CSS0” and “MSPI1SSI”).
- Set transmit/receive mode to data transmit mode of each channel (MSPInCFGm0.TXEm = 1 and MSPInCFGm0.RXEm = 1). Set 10 Mbps to baud rate of master channel.
- Set 32 bits to the frame length of each channel (MSPInCFGm2.FLENm=0x20), and 4 times (MSPInCFSETm.CFSETm=4) to frame count.
- Set Direct Memory Mode as the memory mode for each channel.
- Enable CRC on master channel only (MSPI0CFG01.SAFE0=1) and set out-of-frame format (MSPI0CFG01.SAFS0=1).
- Start the communication by writing the transmit data to the transmit data register MSPInTXDAm0 of master channel. Write transmit data register when “INTMSPInTXm” interrupt is occurred.
- After data reception, read the received data from the receive data register MSPInRXDAm0 of each channel. Read received data register when INTMSPInRXm is occurred.
- In this operation example, after received the last data since starting 1st set of communication, complete the 1st set of communication when the INTMSPInFEm interrupt occurs. Continuously start the communication of 2nd set, after received the last data since starting 2nd set of communication, complete the communication of 2nd set when INTMSPInFEm is occurred.
- In this operation example, check INTMSPInTXm interrupt by MSPInINTFm rgister, INTMSPInRXm interrupt by MSPInINTF1 register, INTMSPInFEm interrupt by MSPI0INTF2 register.

Refer to “RH850/U2B User’s Manual Figure 21.46 Transmission/Reception in the Master-Direct Memory Mode and 21.6.11.2 Safe-SPI protocol function in out-of-frame format” for the communication sequence detais.

- In the communication of Out-of-frame format of Safe SPI protocol, MSPI calculates CRC code using bit 31-3 of setting value of transmit data and transmit the CRC code as bit 2-0 of transmit data. MSPI also calculates CRC code using bit 31-3 of received data and compares the calculated CRC code with the CRC embedded in bits 2-0 of the received data. If these two CRCs are different, MSPI issues INTMSPInERR and sets MSPInCE flag.
- Out-of-frame format is used the following.

CRC	3 bit CRC
Polynomial	0x5 (x^3+x^1+1)
Start value	101_B
Target value	000_B

- The transmit data example of master channel in this operation example are shown below. CRC code is calculated by MSPI, therefore write the transmit data setting value to MSPI0TXDA00 register.

Master	Transmit data (Hexadecimal)	Transmit data (Binary number)	
		Setting value of transmit data (Bit 31-3)	CRC code (Bit 2-0)
Transmit frame 1	0x 00 00 00 03	0000 0000 0000 0000 0000 0000 0000 0_B	011_B
Transmit frame 2	0x FF FF FF F8	1111 1111 1111 1111 1111 1111 1111 1_B	000_B
Transmit frame 3	0x 0F 0F 0F 0A	0000 1111 0000 1111 0000 1111 0000 1_B	010_B
Transmit frame 4	0x 0F F2 C8 FE	0000 1111 1111 0010 1100 1000 1111 1_B	110_B

- The transmit data example of slave channel in this operation example are shown below. Write the transmit data setting value and CRC code to MSPI1TXDA00 register.

Slave	Transmit data (Hexadecimal)	Transmit data (Binary number)	
		Setting value of transmit data (Bit 31-3)	CRC code (Bit 2-0)
Transmit frame 1	0x 00 00 00 03	0000 0000 0000 0000 0000 0000 0000 0_B	011_B
Transmit frame 2	0x FF FF FF F8	1111 1111 1111 1111 1111 1111 1111 1_B	000_B
Transmit frame 3	0x 0F 0F 0F 0A	0000 1111 0000 1111 0000 1111 0000 1_B	010_B
Transmit frame 4	0x 0F F2 C8 FE	0000 1111 1111 0010 1100 1000 1111 1_B	110_B

- When setting value of transmit data (bit31-3) is 0000 0000 0000 0000 0000 0000 0000 0_B, CRC is calculated like below.

	Start value			Transmit Data Setting Value [31:3]																													Target value			
	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
XOR	1	0	1	1																																
				1	0	0	0																													
		XOR		1	0	1	1																													
						1	1	0	0																											
			XOR			1	0	1	1																											
							1	1	1	0																										
				XOR			1	0	1	1																										
								1	0	1	0																									
					XOR			1	0	1	1																									
										1	0	0	0																							
								XOR		1	0	1	1																							
												1	1	0	0																					
									XOR			1	0	1	1																					
													1	1	1	0																				
										XOR			1	0	1	1																				
														1	0	1	0																			
											XOR			1	0	1	1																			
																	1	0	0	0																
												XOR					1	0	1	1																
																			1	1	0	0														
													XOR					1	0	1	1															
																				1	1	1	0													
														XOR					1	0	1	1														
																					1	0	0	0												
															XOR					1	0	1	1													
																						1	0	1	0											
																XOR					1	0	1	1												
																							1	0	1	1										
																											1	0	0	0						
																												XOR								
																													1	0	1	1				
																																	1	0	1	0
																																		XOR		
																																	1	0	1	1
																																				0	1	1 ←CRC Code

Figure 3-10 Safe SPI protocol out-of-frame Format CRC Calculation Example

3.4.2 System Configuration

Figure 3-11 shows the system configuration.

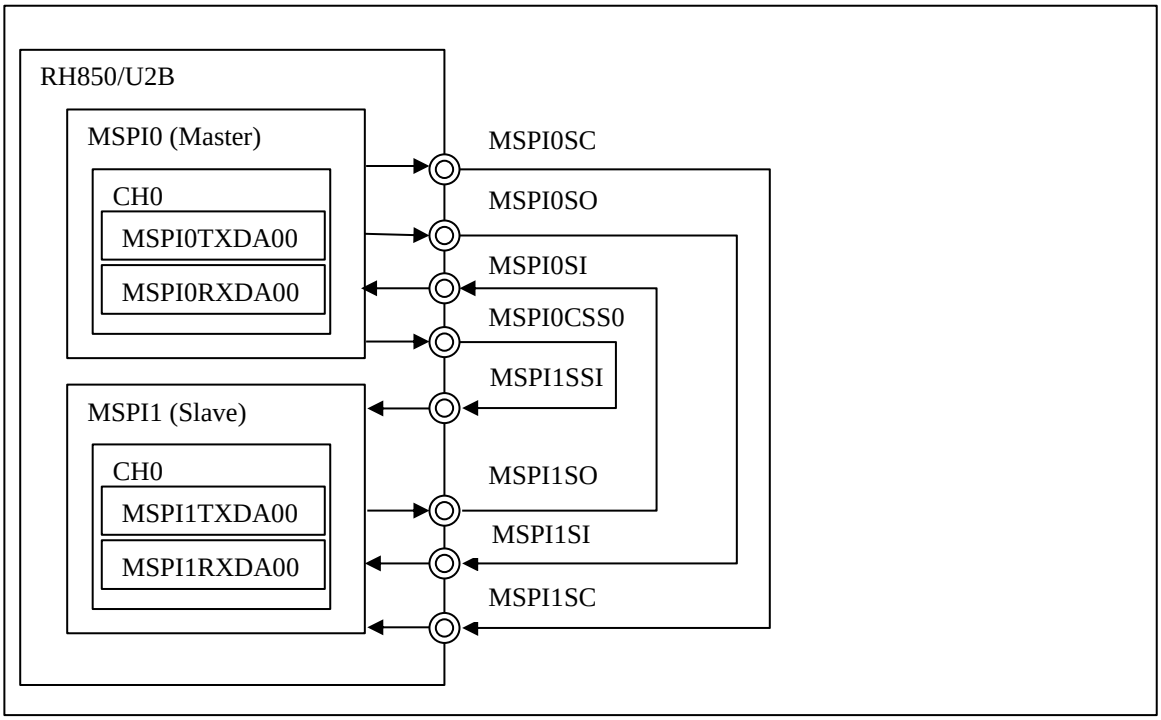

Figure 3-11 System Configuration

3.4.3 Software Explanation

3.4.3.1 Module Explanation

Module list in this operation example is shown below.

Table 3-19 Module

Module Name	Function Name	Function
MSPI initialization routine	mspi_init	Initialize MSPI.
MSPI activation	mspi0_ch0_activate	Enable MSPI0 CH0.
	mspi1_ch0_activate	Enable MSPI1 CH0.
MSPI transmit/receive routine	mspi0_ch0_communicate	Transmit/Receive data of MSPI0 CH0.
	mspi1_ch0_communicate	Transmit/Receive data of MSPI1 CH0.

3.4.3.2 Register Setting

Register setting of each function in this operation example are shown below.

(a) MSPI0 Initial Setting

Table 3-20 MSPI0 Initial Setting

Register Name	Bit Name	Setting Value	Function
MSPI0CTL1	MSSEL	0	Master mode
	CSIE	0	In the master mode (MSPInMSSEL=0), MSPInCSIE must be set to 0.
	SAMP	1	The sampling timing of Master receive is next edge sampling point of SPI protocol.
	CKR	0	The default level of MSPInSCK is low.
	SOLS	0	Set MSPInSOUT to low after macro enable, and holds the level after each transfer.
	CSP	0	The MSPInCS signal is active low.
MSPI0CTL2	DCS	0	Disables the data consistency check.
	LBM	0	Disables the loop-back mode.
MSPI0CTL0	EN	1	Enables the MSPIn function.

(b) MSPI0 CH0 Initial Setting

Table 3-21 MSPI0 CH0 Initial Setting

Register Name	Bit Name	Setting Value	Function
MSPI0CFG00	TXE0	1	Transmission enabled.
	RXE0	1	Reception enabled.
	MD0	0	Direct memory mode
	PRIO0	7	Channel priority level 8(Lowest priority) (default)
	LOCK0	0	Disables the channel m lock operation.
	FCCE0	0	When a last frame ends, MSPInCHENm is cleared and the channel operation ends.
	IERE0	1	Enables the interrupt output.
	IFEE0	1	Enables the interrupt output.

Register Name	Bit Name	Setting Value	Function
	IRXE0	1	Enables the interrupt output.
	ITXE0	1	Enables the interrupt output.
MSPI0CFG01	CPOL0	0	MSPInSCK is low during idle time.
	CPHA0	0	Shifting bits out for transmission takes place on even-numbered edges, and sampling for reception takes place on odd-numbered edges.
	DIR0	0	Data is transmitted/received with MSB first.
	ICLS0	0	MSPInCS level is inactive for idle time.
	FIDL0	1	The idle time is forcibly inserted after each end of a frame.
	CSRI0	1	MSPInCS returns to the inactive level.
	SAFCM0	0	Does not mask the CRC error of the first frame.
	SAFS0	1	Out-of-frame format
	SAFE0	1	Enables Safe SPI protocol function
	PS0	0	Transmission: Adds odd parity, Reception: Odd parity bit is expected.
	DECHK0	0	No parity check
MSPI0CFG02	FLEN0	0x20	32 bits (default)
MSPI0CFG03	PRCS0	0	MSPInSCK baud rate = $\frac{MSPInCLK}{4^{PRCSm} \times CDIVm \times 2}$ = 10MHz
	CDIV0	4	
MSPI0CFG04	HWTS0	0	HW trigger disabled. (Only SW trigger is effective)
	SIZE0	0	Set MSPInSIZEm[1:0] to 00 in the direct memory mode or the fixed buffer memory mode.
MSPI0SEUP0	SEUP0	4	MSPInSCK delay time = MSPInSEUPm [11: 0] × MSPInCLK.
MSPI0HOLD0	HOLD0	1	MSPInCS negation delay time = MSPInHOLDm [11: 0] × MSPInCLK
MSPI0IDLE0	IDLE0	8	MSPInCS[7: 0] next frame time = MSPInIDLEm [11: 0] × MSPInCLK.
MSPI0INDA0	INDA0	8	MSPInCS[7: 0] next frame time = MSPInINDAm [11: 0] × MSPInCLK.
MSPI0CFSET0	CFSET0	4	These bits set the number of frame count.
MSPI0SSEL0	JOBEN0	0	Job ends with this frame. After this frame the channel with a higher priority can transfer data.
	CSR0	1	Activates MSPInCS0 for the communication. Deactivates MSPInCS[7: 1] for the communication.

(c) MSPI1 Initial Setting

Table 3-22 MSPI1 Initial Setting

Register Name	Bit Name	Setting Value	Function
MSPI1CTL1	MSSEL	1	Slave mode
	CSIE	1	Input CS signal (MSPInCSI) is recognized in slave mode.
	SAMP	0	In the slave mode (MSPInMSSEL=1), MSPInSAMP must be set to 0.
	CKR	0	The default level of MSPInSCK is low.
	SOLS	0	In the slave mode (MSPInMSSEL=1), MSPInSOLS[1:0] must be set to 00.
	CSP	0	In the slave mode (MSPInMSSEL=1), MSPInCSP[7:0] must be set to 00H.
MSPI1CTL2	DCS	0	Disables the data consistency check.
	LBM	0	In slave mode (MSPInMSSEL=1), MSPInLBM must be set to 0.
MSPI1CTL0	EN	1	Enables the MSPIn function.

(d) MSPI1 CH0 Initial Setting

Table 3-23 MSPI1 CH0 Initial Setting

Regsiter Name	Bit Name	Settng Value	Function
MSPI1CFG00	TXE0	1	Transmission enabled.
	RXE0	1	Reception enabled.
	MD0	0	Direct memory mode
	PRIO0	7	Set MSPInPRIOm to 111_B in the slave mode.
	LOCK0	0	Set MSPInLOCKm to 0 in the slave mode.
	FCCE0	0	When a last frame ends, MSPInCHENm is cleared and the channel operation ends.
	IERE0	1	Enables the interrupt output.
	IFEE0	1	Enables the interrupt output.
	IRXE0	1	Enables the interrupt output.
	ITXE0	1	Enables the interrupt output.
MSPI1CFG01	CPOL0	0	Set MSPInCPOLm to 0 in the slave mode.
	CPHA0	0	Shifting bits out for transmission takes place on even-numbered edges, and sampling for reception takes place on odd-numbered edges.
	DIR0	0	Data is transmitted/received with MSB first.
	ICLS0	0	Set MSPInICLSm to 0 in the slave mode.
	FIDL0	0	Set MSPInFIDLm to 0 in the slave mode.
	CSRI0	0	Set MSPInCSRIm to 0 in the slave mode.
	SAFCM0	0	Does not mask the CRC error of the first frame.
	SAFS0	0	In-frame format

Regsiter Name	Bit Name	Settng Value	Function
	SAFE0	0	Set MSPInSAFEm to 0 in the slave mode.
	PS0	0	Transmission: Adds odd parity, Reception: Odd parity bit is expected.
	DECHK0	0	No parity check
MSPI1CFG02	FLEN0	0x20	32 bits (default)
MSPI1CFG03	PRCS0	0	In the slave mode (MSPInMSSEL=1), MSPInCFGm3 must be set to 0001_H(default value).
	CDIV0	1	
MSPI1CFG04	HWTS0	0	HW trigger disabled. (Only SW trigger is effective)
	SIZE0	0	Set MSPInSIZEm[1:0] to 00 in the direct memory mode or the fixed buffer memory mode.
MSPI1SEUP0	SEUP0	1	Set MSPInSEUPm to 0001_H in the slave mode.
MSPI1HOLD0	HOLD0	1	Set MSPInHOLDm to 0001_H in slave mode.
MSPI1IDLE0	IDLE0	1	Set MSPInIDLEm to 0001_H in slave mode.
MSPI1INDA0	INDA0	0	Set MSPInINDAm to 0000_H in slave mode.
MSPI1CFSET0	CFSET0	4	These bits set the number of frame count.
MSPI1SSEL0	JOBEN0	0	Set MSPInSSELm to 0000_H in the slave mode.
	CSR0	0	

3.4.3.3 Operation Flow

The flowchart in this operation example is shown below.

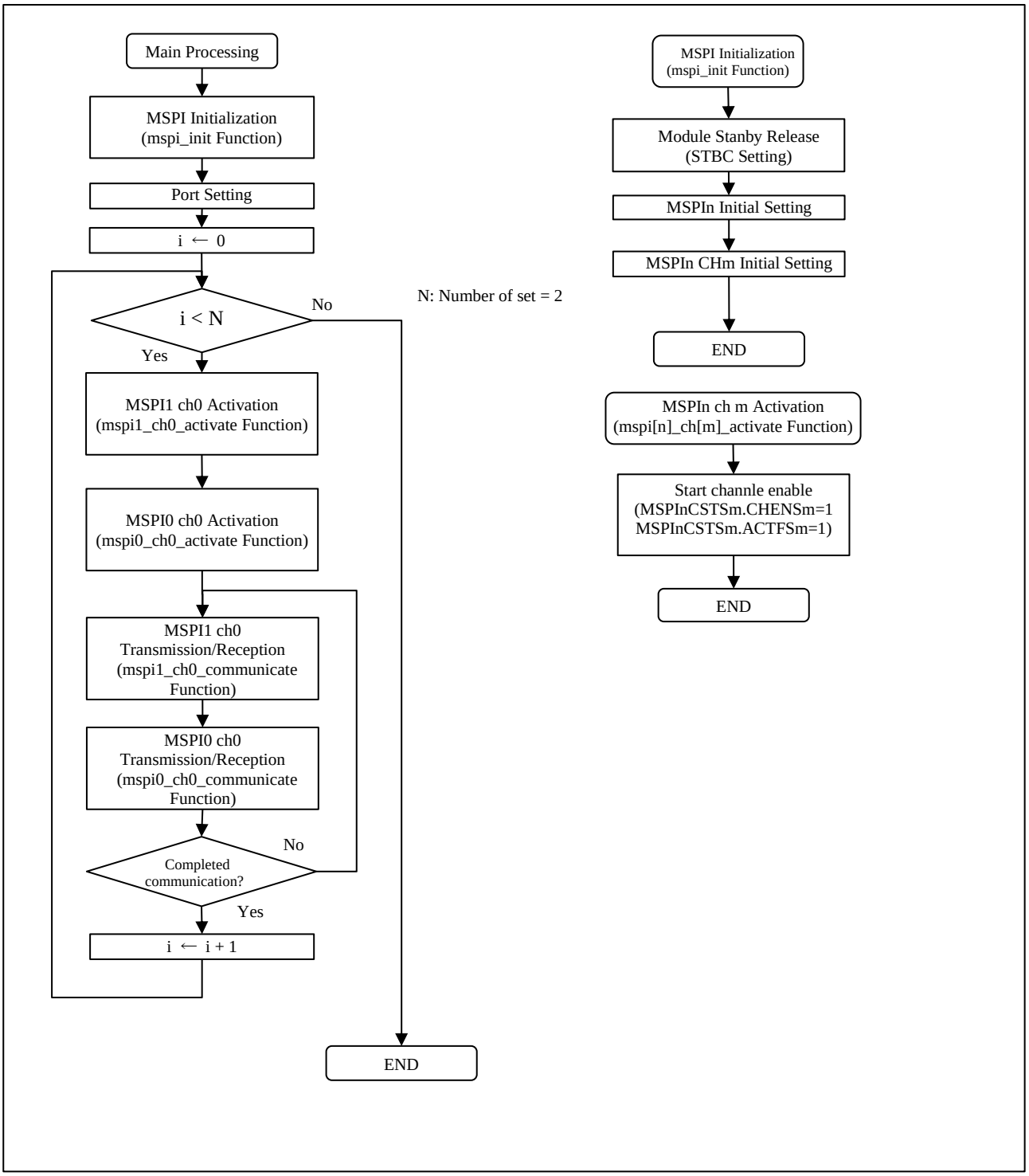

Figure 3-12 Operation Flow (1/2)

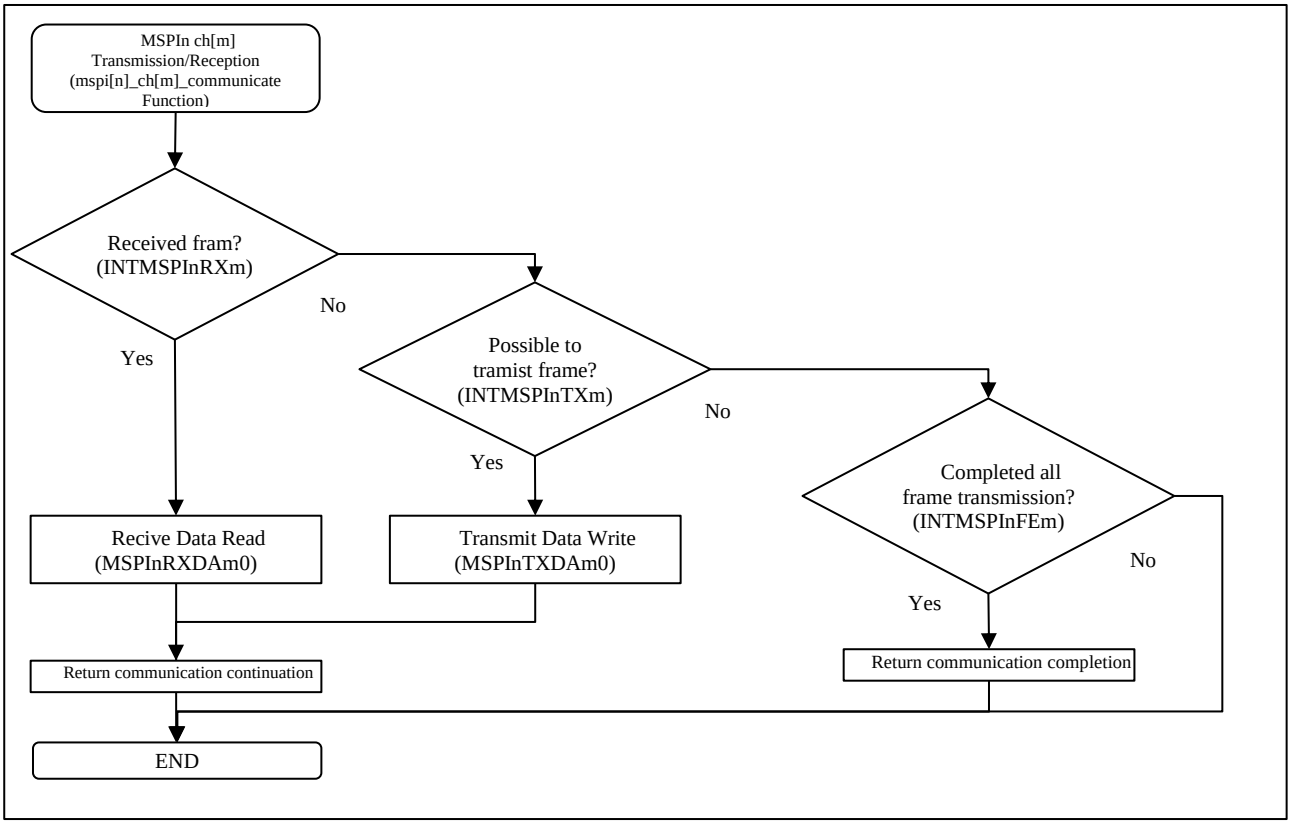

Figure 3-12 Operation Flow (2/2)

Refer to “RH850/U2B User’s Manual Figure 21.46 Master Transmission/Reception Operating Procedure in Direct Memory Mode, Figure 21.49 Slave Transmission/Reception Operating Procedure in Direct Memory Mode” for the flowchart datils.

Revision History

Rev.	Date	Description	
		Page	Point
0.10	2022.03.31	All items	Initial edition

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

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

1. Precaution against Electrostatic Discharge (ESD)

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

2. Processing at power-on

   The state of the product is undefined at the time when power is supplied. The states of internal circuits in the LSI are indeterminate and the states of register settings and pins are undefined at the time when power is supplied. In a finished product where the reset signal is applied to the external reset pin, the states of pins are not guaranteed from the time when power is supplied until the reset process is completed. In a similar way, the states of pins in a product that is reset by an on-chip power-on reset function are not guaranteed from the time when power is supplied until the power reaches the level at which resetting is specified.

3. Input of signal during power-off state

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

4. Handling of unused pins

   Handle unused pins in accordance with the directions given under handling of unused pins in the manual. The input pins of CMOS products are generally in the high-impedance state. In operation with an unused pin in the open-circuit state, extra electromagnetic noise is induced in the vicinity of the LSI, an associated shoot-through current flows internally, and malfunctions occur due to the false recognition of the pin state as an input signal become possible.

5. Clock signals

   After applying a reset, only release the reset line after the operating clock signal becomes stable. When switching the clock signal during program execution, wait until the target clock signal is stabilized. When the clock signal is generated with an external resonator or from an external oscillator during a reset, ensure that the reset line is only released after full stabilization of the clock signal. Additionally, when switching to a clock signal produced with an external resonator or by an external oscillator while program execution is in progress, wait until the target clock signal is stable.

6. Voltage application waveform at input pin

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

7. Prohibition of access to reserved addresses

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

8. Differences between products

   Before changing from one product to another, for example to a product with a different part number, confirm that the change will not lead to problems. The characteristics of a microprocessing unit or microcontroller unit products in the same group but having a different part number might differ in terms of internal memory capacity, layout pattern, and other factors, which can affect the ranges of electrical characteristics, such as characteristic values, operating margins, immunity to noise, and amount of radiated noise. When changing to a product with a different part number, implement a system-evaluation test for the given product.

Notice

1. Descriptions of circuits, software and other related information in this document are provided only to illustrate the operation of semiconductor products and application examples. You are fully responsible for the incorporation or any other use of the circuits, software, and information in the design of your product or system. Renesas Electronics disclaims any and all liability for any losses and damages incurred by you or third parties arising from the use of these circuits, software, or information.
2. Renesas Electronics hereby expressly disclaims any warranties against and liability for infringement or any other claims involving patents, copyrights, or other intellectual property rights of third parties, by or arising from the use of Renesas Electronics products or technical information described in this document, including but not limited to, the product data, drawings, charts, programs, algorithms, and application examples.
3. No license, express, implied or otherwise, is granted hereby under any patents, copyrights or other intellectual property rights of Renesas Electronics or others.
4. You shall be responsible for determining what licenses are required from any third parties, and obtaining such licenses for the lawful import, export, manufacture, sales, utilization, distribution or other disposal of any products incorporating Renesas Electronics products, if required.
5. You shall not alter, modify, copy, or reverse engineer any Renesas Electronics product, whether in whole or in part. Renesas Electronics disclaims any and all liability for any losses or damages incurred by you or third parties arising from such alteration, modification, copying or reverse engineering.
6. Renesas Electronics products are classified according to the following two quality grades: “Standard” and “High Quality”. The intended applications for each Renesas Electronics product depends on the product’s quality grade, as indicated below.
   "Standard": Computers; office equipment; communications equipment; test and measurement equipment; audio and visual equipment; home electronic appliances; machine tools; personal electronic equipment; industrial robots; etc.
   "High Quality": Transportation equipment (automobiles, trains, ships, etc.); traffic control (traffic lights); large-scale communication equipment; key financial terminal systems; safety control equipment; etc.
   Unless expressly designated as a high reliability product or a product for harsh environments in a Renesas Electronics data sheet or other Renesas Electronics document, Renesas Electronics products are not intended or authorized for use in products or systems that may pose a direct threat to human life or bodily injury (artificial life support devices or systems; surgical implantations; etc.), or may cause serious property damage (space system; undersea repeaters; nuclear power control systems; aircraft control systems; key plant systems; military equipment; etc.). Renesas Electronics disclaims any and all liability for any damages or losses incurred by you or any third parties arising from the use of any Renesas Electronics product that is inconsistent with any Renesas Electronics data sheet, user’s manual or other Renesas Electronics document.
7. No semiconductor product is absolutely secure. Notwithstanding any security measures or features that may be implemented in Renesas Electronics hardware or software products, Renesas Electronics shall have absolutely no liability arising out of any vulnerability or security breach, including but not limited to any unauthorized access to or use of a Renesas Electronics product or a system that uses a Renesas Electronics product. RENESAS ELECTRONICS DOES NOT WARRANT OR GUARANTEE THAT RENESAS ELECTRONICS PRODUCTS, OR ANY SYSTEMS CREATED USING RENESAS ELECTRONICS PRODUCTS WILL BE INVULNERABLE OR FREE FROM CORRUPTION, ATTACK, VIRUSES, INTERFERENCE, HACKING, DATA LOSS OR THEFT, OR OTHER SECURITY INTRUSION (“Vulnerability Issues”). RENESAS ELECTRONICS DISCLAIMS ANY AND ALL RESPONSIBILITY OR LIABILITY ARISING FROM OR RELATED TO ANY VULNERABILITY ISSUES. FURTHERMORE, TO THE EXTENT PERMITTED BY APPLICABLE LAW, RENESAS ELECTRONICS DISCLAIMS ANY AND ALL WARRANTIES, EXPRESS OR IMPLIED, WITH RESPECT TO THIS DOCUMENT AND ANY RELATED OR ACCOMPANYING SOFTWARE OR HARDWARE, INCLUDING BUT NOT LIMITED TO THE IMPLIED WARRANTIES OF MERCHANTABILITY, OR FITNESS FOR A PARTICULAR PURPOSE.
8. When using Renesas Electronics products, refer to the latest product information (data sheets, user’s manuals, application notes, “General Notes for Handling and Using Semiconductor Devices” in the reliability handbook, etc.), and ensure that usage conditions are within the ranges specified by Renesas Electronics with respect to maximum ratings, operating power supply voltage range, heat dissipation characteristics, installation, etc. Renesas Electronics disclaims any and all liability for any malfunctions, failure or accident arising out of the use of Renesas Electronics products outside of such specified ranges.
9. Although Renesas Electronics endeavors to improve the quality and reliability of Renesas Electronics products, semiconductor products have specific characteristics, such as the occurrence of failure at a certain rate and malfunctions under certain use conditions. Unless designated as a high reliability product or a product for harsh environments in a Renesas Electronics data sheet or other Renesas Electronics document, Renesas Electronics products are not subject to radiation resistance design. You are responsible for implementing safety measures to guard against the possibility of bodily injury, injury or damage caused by fire, and/or danger to the public in the event of a failure or malfunction of Renesas Electronics products, such as safety design for hardware and software, including but not limited to redundancy, fire control and malfunction prevention, appropriate treatment for aging degradation or any other appropriate measures. Because the evaluation of microcomputer software alone is very difficult and impractical, you are responsible for evaluating the safety of the final products or systems manufactured by you.
10. Please contact a Renesas Electronics sales office for details as to environmental matters such as the environmental compatibility of each Renesas Electronics product. You are responsible for carefully and sufficiently investigating applicable laws and regulations that regulate the inclusion or use of controlled substances, including without limitation, the EU RoHS Directive, and using Renesas Electronics products in compliance with all these applicable laws and regulations. Renesas Electronics disclaims any and all liability for damages or losses occurring as a result of your noncompliance with applicable laws and regulations.
11. Renesas Electronics products and technologies shall not be used for or incorporated into any products or systems whose manufacture, use, or sale is prohibited under any applicable domestic or foreign laws or regulations. You shall comply with any applicable export control laws and regulations promulgated and administered by the governments of any countries asserting jurisdiction over the parties or transactions.
12. It is the responsibility of the buyer or distributor of Renesas Electronics products, or any other party who distributes, disposes of, or otherwise sells or transfers the product to a third party, to notify such third party in advance of the contents and conditions set forth in this document.
13. This document shall not be reprinted, reproduced or duplicated in any form, in whole or in part, without prior written consent of Renesas Electronics.
14. Please contact a Renesas Electronics sales office if you have any questions regarding the information contained in this document or Renesas Electronics products.

(Note1) “Renesas Electronics” as used in this document means Renesas Electronics Corporation and also includes its directly or indirectly controlled subsidiaries.
(Note2) “Renesas Electronics product(s)” means any product developed or manufactured by or for Renesas Electronics.

(Rev.5.0-1 October 2020)

Corporate Headquarters

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

Trademarks

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

Contact information

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

© 2023 Renesas Electronics Corporation. All rights reserved.