32-bit RISC Microcontroller

TXZ+ Family

Reference Manual
Synchronous Serial Interface
(TSSI-A)

Revision 1.0

2020-11

TOSHIBA ELECTRONIC DEVICES & STORAGE CORPORATION

Contents

List of Figures

List of Tables

Preface

Related Document

Document Name
Clock Control and Operation Mode
Exception
Input/Output Ports
Product Information

Conventions

- Numeric formats follow the rules as shown below:
  Hexadecimal: 0xABC
  Decimal: 123 or 0d123 – Only when it needs to be explicitly shown that they are decimal numbers.
  Binary: 0b111 – It is possible to omit the "0b" when the number of bit can be distinctly understood from a sentence.
- "_N" is added to the end of signal names to indicate low active signals.
- It is called "assert" that a signal moves to its active level, "deassert" to its inactive level.
- When two or more signal names are referred, they are described like as [m: n].
  Example: S[3: 0] shows four signal names S3, S2, S1 and S0 together.
- The characters surrounded by ***[ ]*** defines the register.
  Example: ***[ABCD]***
- "n" substitutes suffix number of two or more same kind of registers, fields, and bit names.
  Example: ***[XYZ1], [XYZ2], [XYZ3] → [XYZn]***
- "x" substitutes suffix number or character of units and channels in the Register List.
  In case of unit, "x" means A, B, and C . . .
  Example: ***[ADACR0], [ADBCR0], [ADCCR0] → [ADxCR0]***
  In case of channel, "x" means 0, 1, and 2 . . .
  Example: ***[T32A0RUNA], [T32A1RUNA], [T32A2RUNA] → [T32AxRUNA]***
- The bit range of a register is written like as [m: n].
  Example: Bit[3: 0] expresses the range of bit 3 to 0.
- The configuration value of a register is expressed by either the hexadecimal number or the binary number.
  Example: ***[ABCD]***<EFG> = 0x01 (hexadecimal), ***[XYZn]***<VW> = 1 (binary)
- Word and Byte represent the following bit length.
  Byte: 8 bits
  Half word: 16 bits
  Word: 32 bits
  Double word: 64 bits
- Properties of each bit in a register are expressed as follows:
  R: Read only
  W: Write only
  R/W: Read and Write are possible
- Unless otherwise specified, register access supports only word access.
- The register defined as reserved must not be rewritten. Moreover, do not use the read value.
- The value read from the bit having default value of "-" is unknown.
- When a register containing both of writable bits and read-only bits is written, read-only bits should be written with their default value, In the cases that default is "-", follow the definition of each register.
- Reserved bits of the Write-only register should be written with their default value. In the cases that default is "-", follow the definition of each register.
- Do not use read-modified-write processing to the register of a definition which is different by writing and read out.

Terms and Abbreviation

Some of abbreviations used in this document are as follows:

DMA	Direct Memory Access
DMAC	Direct Memory Access Controller
FIFO	First-In First-Out
MSB	Most Significant Bit
LSB	Least Significant Bit
TSSI	Synchronous Serial Interface

1. Outline

The synchronous serial interface (TSSI) enables synchronous serial communication independently for the transmitter (TSSIxTCK, TSSIxTFS, TSSIxTXD) and the receiver (TSSIxRCK, TSSIxRFS, TSSIxRXD). And transmit and receive (full duplex communication) is also possible by the cooperative operation of the transmitter and receiver.

Function classification	Function	Action description or range
Clock generation	Clock division	Generate serial clock (SCLK) for master device ΦT0 can be divided by 2 to 256
Receiver	Receive serial clock	Master device: Divided clock SCLK Slave device: Select from TSSIxTCK input and TSSIxRCK input
	Data format	-Bit order: Communication from most significant bit -Data frame size: 4 to 32 bits
	Frame synchronization	Frame synchronization signal and receive data are synchronized with the receive serial clock Output updated at serial clock rise Input detected at falling edge of serial clock
		"High" pulse of one cycle of serial clock is used as a trigger to start communication
	Receive FIFO	4 FIFO stages
	Operation mode	-Master receive -Master transmit and receive (receiver cooperative control) -Slave receive / slave transmit and receive
	Interrupt	Receive interrupt: Pulse interrupt occurs upon Receive FIFO not empty, FIFO threshold flag, Receive data comparison match, and completion of receive FIFO transfer Error interrupt (Note 1): Underrun and overrun of receive FIFO generates a level interrupt
	DMA request	Single DMA request with receive FIFO not empty
	Receive data comparison function	During slave operation, receive or transmit and receive of data frames is possible by triggering a match between the receive data and the expected value (Note 2)
	Cooperative control	Master transmit and receive (full duplex communication) is possible in cooperation with the transmitter
Transmitter	Transmit serial clock	Master device: Divided clock SCLK Slave device: Select from TSSIxTCK input and TSSIxRCK input
	Data format	-Bit order: Communication from most significant bit -Data frame size: 4 to 32 bits
	Frame synchronization	Frame synchronization signal and transmit data are synchronized with the transmit serial clock Output updated at serial clock rise Input detected at falling edge of serial clock
		"High" pulse of one cycle of serial clock is used as a trigger to start communication
	Transmit FIFO	4 FIFO stages
	Operation mode	-Master transmit or master transmit and receive -Slave transmit -Slave transmit and receive (transmitter cooperative control)
	Interrupt	Transmit interrupt: Pulse interrupt generated by transmit FIFO not full, transmit FIFO threshold flag Error interrupt (Note 1): Underrun and overrun of transmit FIFO generates a level interrupt
	DMA request	Single DMA request with transmit FIFO not full
	Cooperative control	Slave transmit and receive (full duplex communication) is possible in cooperation with the receiver
Software reset		Overall, receiver, transmitter

Note 1: The error interrupt is shared by the transmitter and receiver.

Note 2: Receive data compared with expected value is not treated as a data frame. Thus it is not stored in the receive FIFO.

2. Constitution

The block diagram and signal list of the TSSI are shown.

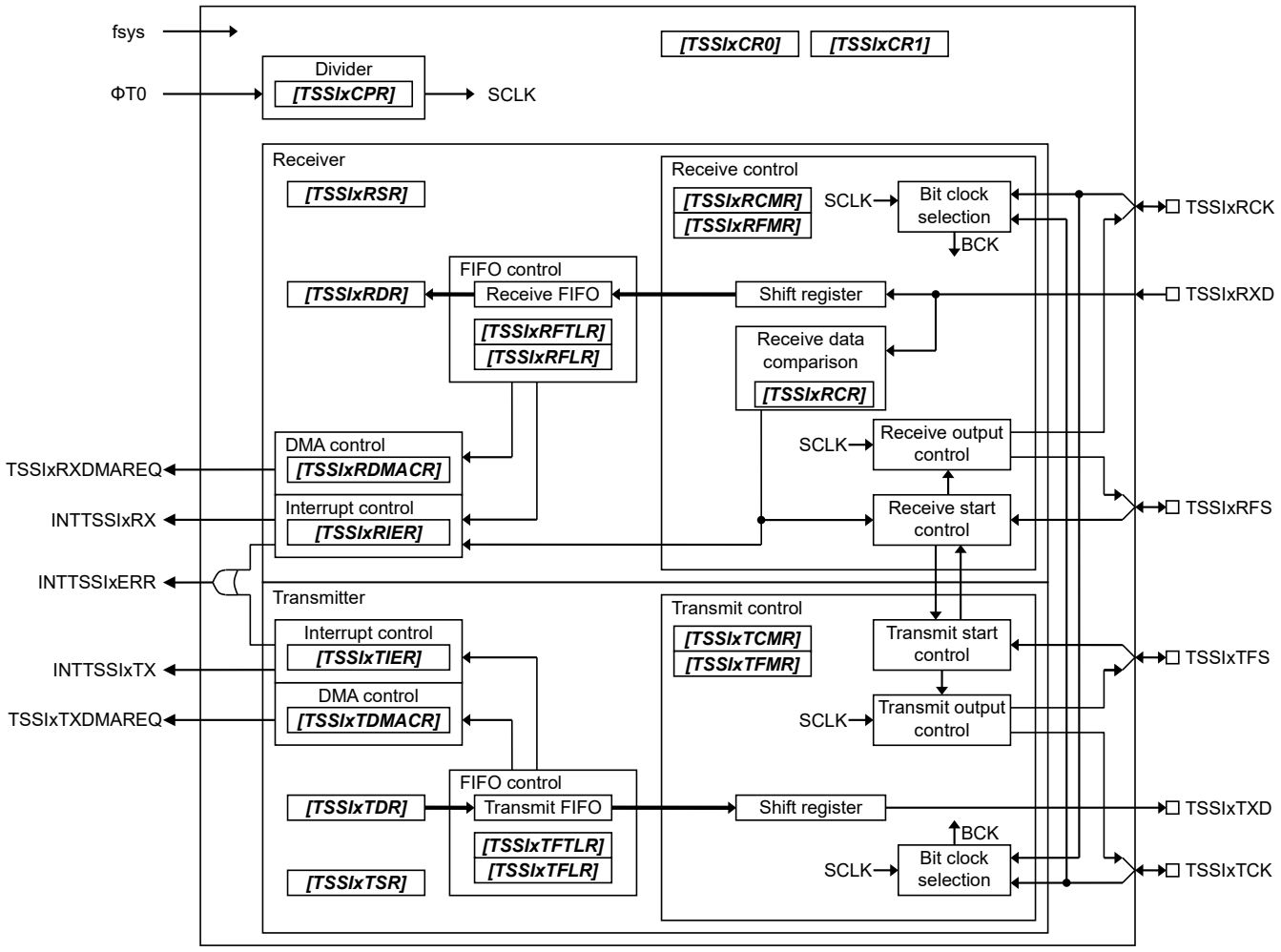


Figure 2.1 TSSI block diagram

Table 2.1 List of Signals

No	Signal symbol	Signal name	I/O	Reference manual
1	fsys	System clock	Input	Clock Control and Operation Mode
2	ΦT0	Clock for prescaler	Input	Clock Control and Operation Mode
3	TSSIxTCK	Transmit clock	Input/ Output	Input/Output Ports, Product Information
4	TSSIxTFS	Transmit frame synchronization signal	Input/ Output	Input/Output Ports, Product Information
5	TSSIxTXD	Transmit data	Output	Input/Output Ports, Product Information
6	TSSIxRCK	Receive clock	Input/ Output	Input/Output Ports, Product Information
7	TSSIxRFS	Receive frame synchronization signal	Input/ Output	Input/Output Ports, Product Information
8	TSSIxRXD	Receive data	Input	Input/Output Ports, Product Information
9	INTTSSIxTX	Transmit interrupt	Output	Exception
10	INTTSSIxRX	Receive interrupt	Output	Exception
11	INTTSSIxERR	Error interrupt	Output	Exception
12	TSSIxTXDMAREQ	Transmit DMA request	Output	Product Information
13	TSSIxRXDMAREQ	Receive DMA request	Output	Product Information

3. Operation Description

3.1. Clock Supply

When the TSSI is used, the corresponding clock enable bits should be set to "1" (Clock supply) in fsys supply stop register A (***[CGFSYSENA]*** and ***[CGFSYSMENA]***), fsys supply stop register B (***[CGFSYSENB]*** and ***[CGFSYSMENB]***), fsys supply stop register C (***[CGFSYSMENC]***), and fc supply stop register (***[CGFCEN]***). The corresponding registers and the bit locations depend on a product. Some products do not have all registers. For the details, refer to reference manual "Clock Control and Operation Mode".

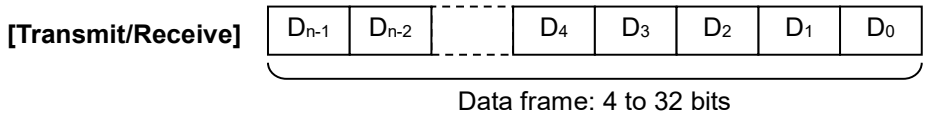
When stopping the supply of a clock, please check that TSSI has stopped (***[TSSIxCR1]***<RXSTS>,<TXSTS> = 0) . Moreover, also when you change the operational mode to STOP1/STOP2, please check that TSSI has stopped.

3.2. Data frame format

The TSSI data frame size can be set from 4 to 32 bits, and communication starts from the most significant bit (MSB).

The data frame size of the transmitter is set by ***[TSSIxTFMR]***<TDFS[4:0]> (transmit data frame size), and the data frame size of the receiver is set by ***[TSSIxRFMR]***<RDFS[4:0]> (receive data frame size).
When performing transmit and receive communication, set <TDFS[4:0]> and <RDFS[4:0]> to the same value to perform cooperative control.

- Normal communication format

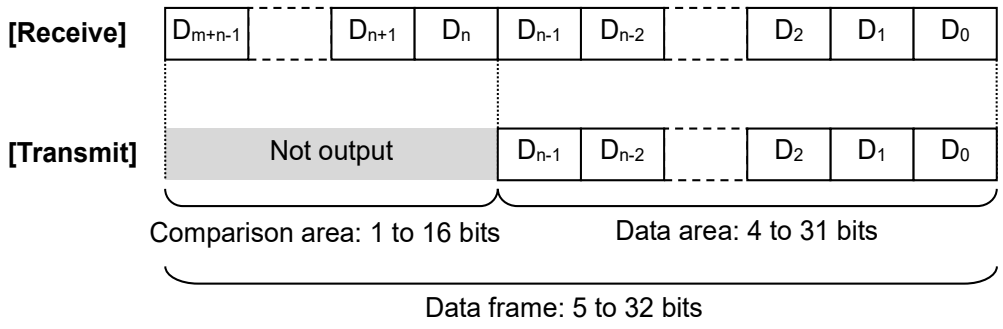


Set ***[TSSIxRFMR]***<RDFS[4:0]>, ***[TSSIxTFMR]***<TDFS[4:0]> to the data frame size.

- Receive data comparison format

The receive data comparison format is used when the slave device uses the receive data comparison function for receive or transmit and receive.

The data frame is divided into a comparison area (m bits on the MSB side) and a data area (n bits on the LSB side).
Compare with ***[TSSIxRCR]***<CMPDP[15:0]> while the slave device receives the comparison area. Receive or transmit and receive of the data area is performed only when the comparison area data matches <CMPDP[15:0]>.
The comparison area data is not saved in the receive FIFO regardless of the match / mismatch.



Set ***[TSSIxRFMR]***<CMPDFS[3:0]> to the comparison area size and ***[TSSIxRFMR]***<RDFS[4:0]> to the data area size in the figure for the receiver. For transmit and receive, set ***[TSSIxTFMR]***<TDFS[4:0]> of the transmitter to the same value as <RDFS[4:0]>.

Note: For the master device, set the data frame size to "comparison area size (m bits) + data area size (n bits)".

3.3. Transfer clock

In the case of a master device, set a common divided clock (SCLK) for the transmitter and receiver.
The bit clock is set in the receiver and transmitter respectively.

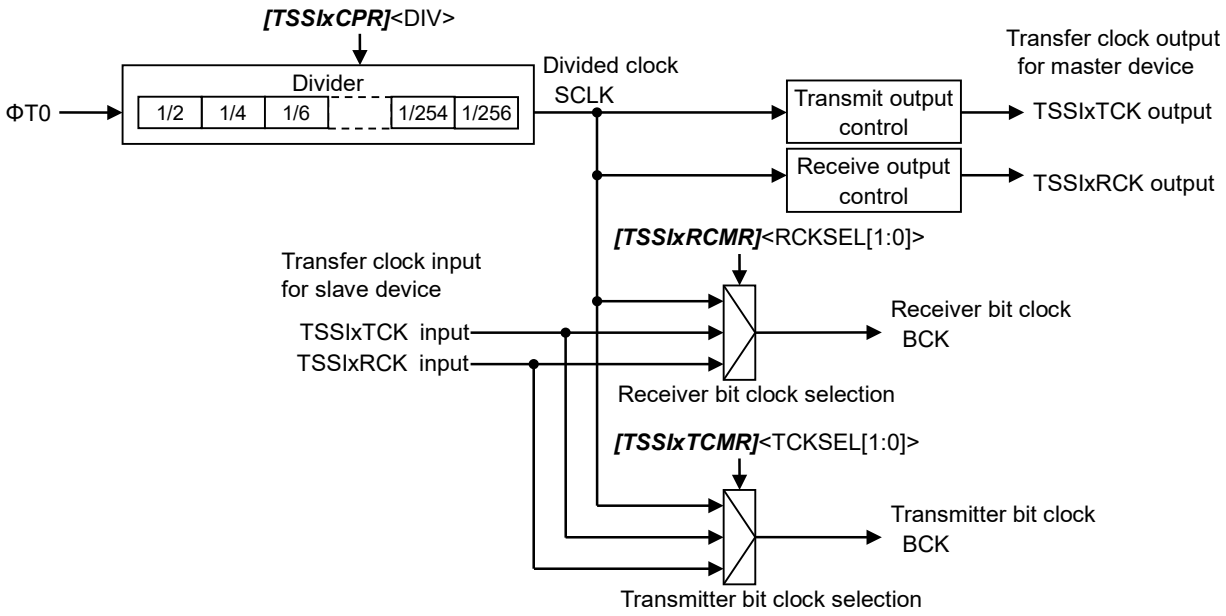


Figure 3.1 Transfer clock

- Master device

(1) Divider setting
Dividing ratio of frequency can be set from 2 to 256 by the clock division register (***[TSSIxCPR]*** <DIV[7:0]>).

SCLK frequency = ΦT0 frequency / (<DIV[7:0]> +1)

Note: SCLK frequency ≤ 1/2 × fsys frequency.

(2) Clock selection
Select the divided clock (SCLK) in clock selection.

Transmitter: ***[TSSIxTCMR]***<TCKSEL[1:0]> = 00
Receiver: ***[TSSIxRCMR]***<RCKSEL[1:0]> = 00

- Slave device

Select the receive clock (TSSIxRCK) or transmit clock (TSSIxTCK) in clock selection. (Note 1)

Transmitter: ***[TSSIxTCMR]***<TCKSEL[1:0]> = 01
Receiver: ***[TSSIxRCMR]***<RCKSEL[1:0]> = 10

Note 1: The serial clock frequency that can be used is 1/2 or less of the fsys frequency.
Note 2: For transmit and receive, set <TCKSEL[1:0]> = <RCKSEL[1:0]> = 10.

3.4. Communication mode

There is a master / slave selection for transmit or receive, and cooperative operation is selected for transmit and receive.

The receiver in the slave device can use the receive data comparison function.

- Master / Slave selection

  The transmitter is set with the transmit clock selection <TCKSEL[1:0]>, transmit clock output mode selection <TCKOUT[1:0]>, and transmit start trigger <TSTART[1:0]> in the transmit clock /mode control register (***[TSSIxTCMR]***).

  The receiver is set with the receive clock selection <RCKSEL[1:0]>, receive clock output mode selection <RCKOUT[1:0]>, and receive start trigger <RSTART[1:0]> in the receive clock /mode control register (***[TSSIxRCMR]***),.

  - Master device

    The communication is performed by outputting the frame synchronization signal (TSSIxRFS, TSSIxTFS) and serial clock (TSSIxRCK, TSSIxTCK).

  - Slave device

    The communication is performed according to the input frame synchronization signal (TSSIxRFS, TSSIxTFS) and serial clock (TSSIxRCK, TSSIxTCK).

- Transmit and receive control

  - Transmit is performed by the transmitter.

    Uses the frame synchronization signal (TSSIxTFS) and serial clock (TSSIxTCK) of the transmitter.

  - Receive is performed by the receiver.

    Uses the frame synchronization signal (TSSIxRFS) and serial clock (TSSIxRCK) of the receiver.

  - For transmit and receive, the transmitter and receiver cooperate.

    In the case of a master device, set the transmitter to master transmit mode (***[TSSIxTCMR***] <TSTART[1:0]> = 00), and the receiver to cooperate with the transmitter (***[TSSIxRCMR]*** <RSTART[1:0]> = 01). Uses the frame synchronization signal (TSSIxTFS) and serial clock (TSSIxTCK) of the transmitter.
    In the case of a slave device, set the receiver to slave receive mode (***[TSSIxRCMR]***<RSTART[1:0]> = 10), and the transmitter to cooperate with the receiver (***[TSSIxTCMR]***<TSTART[1:0]> = 01). Uses the frame synchronization signal (TSSIxRFS) and serial clock (TSSIxRCK) of the receiver.

- Receive data comparison

  In the case of slave receive or slave transmit and receive, the receiver can use the receive data comparison function (***[TSSIxRCMR]***<RSTART[1:0]> = 11).

Table 3.1 Operation mode settings and combinations of pins used

Operation mode		Receive data comparison	Clock / mode control register setting		Pins used (Note 1)
			Receiver (*[TSSIxRCMR]*)	Transmitter (*[TSSIxTCMR]*)	
Master	Transmit	-	-	<TCKSEL[1:0]>=00 <TCKOUT[1:0]>=01,10 <TSTART[1:0]>=00	TSSIxTCK (output) TSSIxTFS (output) TSSIxTXD
	Receive	-	<RCKSEL[1:0]>=00 <RCKOUT[1:0]>=01, 10 <RSTART[1:0]>=00	-	TSSIxRCK (output) TSSIxRFS (output) TSSIxRXD
	Transmit and receive (Full duplex communication)	-	<RCKSEL[1:0]>=00 <RCKOUT[1:0]>=00 <RSTART[1:0]>=01 (Note2)	<TCKSEL[1:0]>=00 <TCKOUT[1:0]>=01,10 <TSTART[1:0]>=00	TSSIxTCK (output) TSSIxTFS (output) TSSIxTXD TSSIxRXD
Slave	Transmit	-	-	<TCKSEL[1:0]>=01 <TCKOUT[1:0]>=00 <TSTART[1:0]>=10	TSSIxTCK (input) TSSIxTFS (input) TSSIxTXD
	Receive	None	<RCKSEL[1:0]>=10 <RCKOUT[1:0]>=00 <RSTART[1:0]>=10	-	TSSIxRCK (input) TSSIxRFS (input) TSSIxRXD
		Yes	<RCKSEL[1:0]>=10 <RCKOUT[1:0]>=00 <RSTART[1:0]>=11	-	TSSIxRCK (input) TSSIxRFS (input) TSSIxRXD
	Transmit and receive (Full duplex communication)	None	<RCKSEL[1:0]>=10 <RCKOUT[1:0]>=00 <RSTART[1:0]>=10	<TCKSEL[1:0]>=10 <TCKOUT[1:0]>=00 <TSTART[1:0]>=01 (Note2)	TSSIxRCK (input) TSSIxRFS (input) TSSIxRXD TSSIxTXD
		Yes	<RCKSEL[1:0]>=10 <RCKOUT[1:0]>=00 <RSTART[1:0]>=11	<TCKSEL[1:0]>=10 <TCKOUT[1:0]>=00 <TSTART[1:0]>=01 (Note2)	TSSIxRCK (input) TSSIxRFS (input) TSSIxRXD TSSIxTXD

Note 1: Set the input / output port before use the TSSI. For details on the settings, refer to reference manual “Input / Output Ports”.

Note 2: Set the transmitter or receiver for cooperative operation.

3.5. FIFO configuration

The transmit FIFO and receive FIFO each have 4 stages.

3.5.1. FIFO operation

Transmit data is written to the data register ***[TSSIxTDRn]***, and receive data is read from ***[TSSIxRDRn]***. ***[TSSIxTDRn]*** and ***[TSSIxRDRn]*** are 32 bits width and right-justified if the data frame size is less than 32 bits. The number of data entries in the transmit FIFO can be checked in ***[TSSIxTFLR]***, and the number of data entries in the receive FIFO can be checked in ***[TSSIxRFLR]***.

The number of receive FIFO entries is incremented by 1 each time one frame data is transferred from the receive shift register, and decremented by 1 each time ***[TSSIxRDR0]*** is read.

The number of transmit FIFO entries is incremented by 1 for each write to ***[TSSIxTDR0]*** and decremented by 1 for each frame data transfer to the transmit shift register.

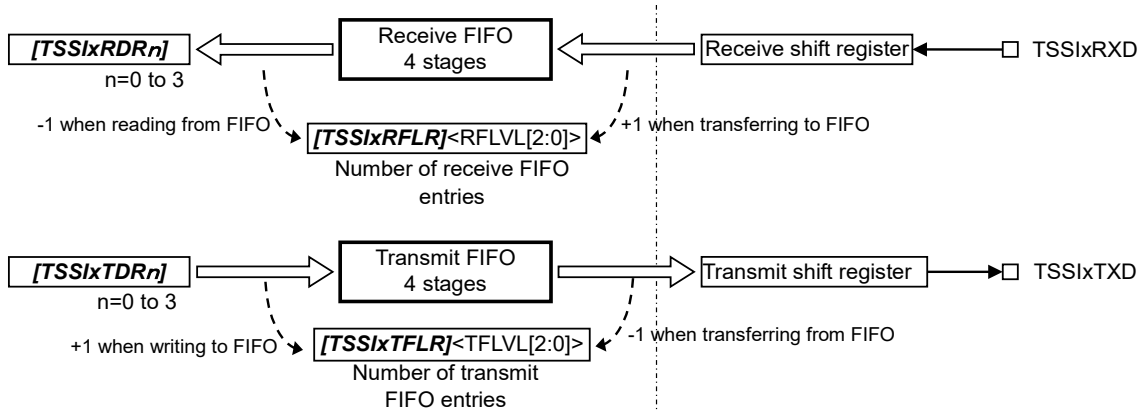


Figure 3.2 FIFO operation

When reading the received data, the first stored data will be read no matter which address of the receive data register(***[TSSIxRDR0]*** to ***[TSSIxRDR3]***) is read. Therefore, for example, a continuous read to ***[TSSIxRDR0]*** and a sequential read from ***[TSSIxRDR0]*** to ***[TSSIxRDR3]*** are treated the same.

When writing transmit data, no matter which address of the transmit data register (***[TSSIxTDR0]*** to ***[TSSIxTDR3]***) is used to write the data, the data will be stored at the end of the FIFO. Therefore, for example, continuous writing to ***[TSSIxTDR0]*** and sequential writing from ***[TSSIxTDR0]*** to ***[TSSIxTDR3]*** are treated the same.

3.6. Communication operation

Table 3.2 Communication start condition

Operation mode		Description
Master	Transmit	-When ***[TSSIxCR1]***<TXSTS> = 1 and there is no data in the transmit FIFO, write to ***[TSSIxTDRn]***. -When ***[TSSIxCR1]***<TXSTS> = 0 and there is data in the transmit FIFO, write "01" to ***[TSSIxCR1]***<TXEN[1:0]>. (Resume)
	Receive	Write "01" to ***[TSSIxCR1]***<RXEN[1:0]>
	Transmit and receive	-When ***[TSSIxCR1]***<TXSTS> and <RXSTS> are “1” and there is no data in the transmit FIFO,write to ***[TSSIxTDRn]***. -When ***[TSSIxCR1]***<TXSTS> and <RXSTS> are “0” and there is data in the transmit FIFO, Write “01” to ***[TSSIxCR1]***<TXEN[1:0]> and <RXEN[1:0]> (Note 3). (Resume)
Slave	Transmit	TSSIxTFS input detection
	Receive	TSSIxRFS input detection (Note 1)
	Transmit and receive	TSSIxRFS input detection (Note 1) (Note 2)

Note 1: When using the receive data comparison function, the data area is not received unless the receive data comparison area matches.

Note 2: When using the receive data comparison function, the transmit data is not output unless the receive data comparison area matches.

Note 3: The transmitter and receiver must be enabled at the same time.

Table 3.3 Communication completion condition

Operation mode		Description
Master	Transmit	- After transmit for each frame, transmit FIFO is empty -End of frame transmit (Suspend) during disabled operation (***[TSSIxCR1]*** <TXEN[1:0]> = 10) -Disabled operation (***[TSSIxCR1]***<TXEN[1:0]> = 11) (Forced termination)
	Receive	- Receive of the set number of receive data frames (***[TSSIxRFMR]***<RDFC[3:0]>) completed -End of frame receive (Suspend) during disabled operation (***[TSSIxCR1]*** <RXEN[1:0]> = 10) -Disabled operation (***[TSSIxCR1]***<RXEN[1:0]> = 11) (Forced termination)
	Transmit and receive	-After transmit and receive for each frame, transmit FIFO is empty -End of frame transmit and receive (Suspend) during disabled operation (***[TSSIxCR1]***<RXEN[1:0]> = <TXEN[1:0]> = 10) -Disabled operation (***[TSSIxCR1]***<RXEN[1:0]> = <TXEN[1:0]> = 11) (Forced termination)
Slave	Transmit	-Serial clock input for frame (***[TSSIxTFMR]***<TDFS[4:0]> +1 bits) -***[TSSIxCR1]***<TXSTS> = 0 after disable operation
	Receive	-Serial clock input for frame (***[TSSIxRFMR]***<RDFS[4:0]> +1 bits) -***[TSSIxCR1]***<RXSTS> = 0 after disable operation (Note 2)
	Transmit and receive	-Serial clock input for frame (***[TSSIxRFMR]***<RDFS[4:0]> +1 bits) -***[TSSIxCR1]***<TXSTS> = <RXSTS> = 0 after disable operation (Note 1) (Note 2)

Note 1: The transmitter and receiver must be disabled at the same time.

Note 2: It is necessary to stop processing after stopping communication on the master device.

3.6.1. Transmit

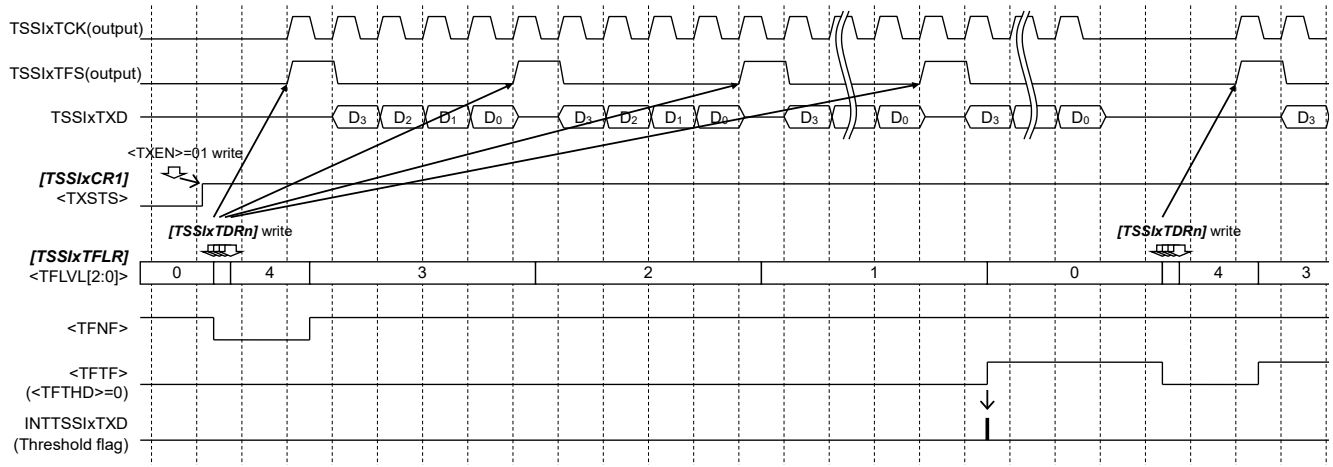
Check ***[TSSIxCR1]***<TXSTS> = 0 before set the transmitter.
Set ***[TSSIxTCMR]***, ***[TSSIxTFMR]*** and ***[TSSIxTFTLR]***. Set ***[TSSIxTIER]*** and ***[TSSIxTDMACR]*** for interrupts and DMA requests.

- Master transmit

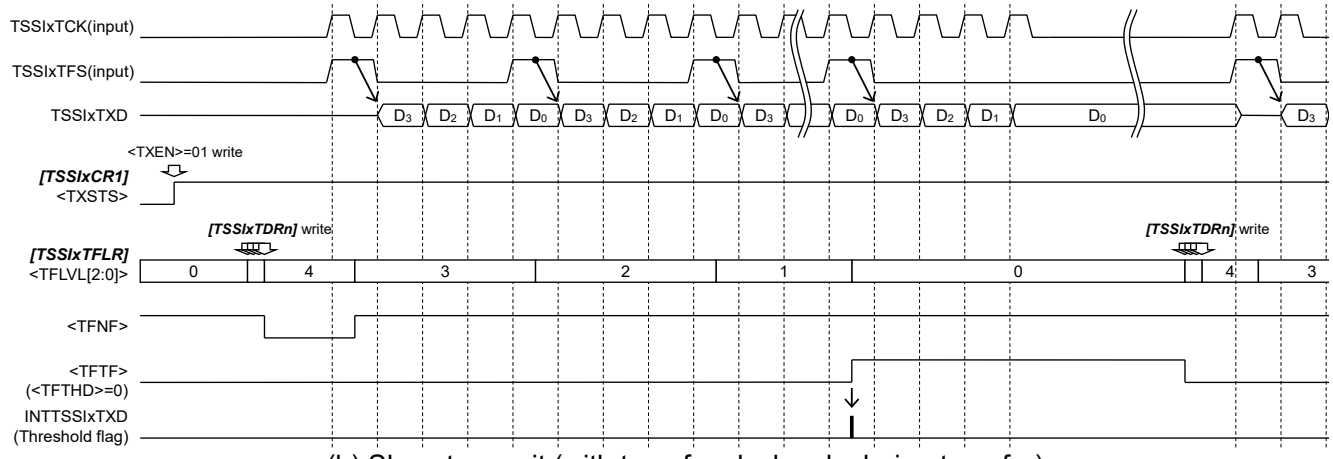
  Set ***[TSSIxCR1]***<TXEN[1:0]> to "01". After checking that ***[TSSIxCR1]***<TXSTS> = 1, write transmit data to ***[TSSIxTDR**n**]*** to start transmit. Starts the output of the transmit clock (TSSIxTCK) and outputs the transmit data to TSSIxTXD in synchronization with the rising edge of TSSIxTCK after outputting the transmit frame synchronization signal (TSSIxTFS).

- Slave transmit

  Set ***[TSSIxCR1]***<TXEN[1:0]> to "01". After checking that ***[TSSIxCR1]***<TXSTS> = 1, write transmit data to ***[TSSIxTDRn]***. After that, transmit starts when the transmit frame synchronization signal (TSSIxTFS) is detected. After detecting the frame synchronization signal, the transmit data is output to TSSIxTXD in synchronization with the rising edge of TSSIxTCK.



(a) Master transmit (***[TSSIxTCMR]***<TCKOUT[1:0]> = 10)



(b) Slave transmit (with transfer clock only during transfer)

Figure 3.3 Operation example of transmit mode

3.6.2. Receive

Check ***[TSSIxCR1]***<RXSTS> = 0 before set the receiver.

Set ***[TSSIxRCMR]***, ***[TSSIxRFMR]*** and ***[TSSIxRFTLR]***. Set ***[TSSIxRIER]*** and ***[TSSIxRDMACR]*** for interrupts and DMA requests. Set ***[TSSIxRCR]*** to the expected value data in as necessary.

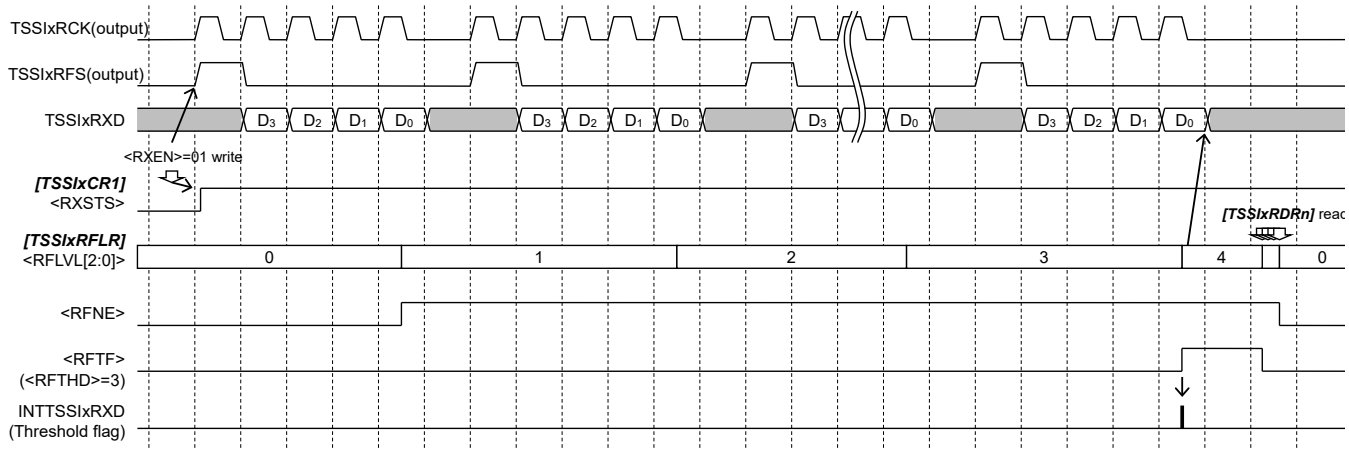
- Master receive

  When ***[TSSIxCR1]***<RXEN[1:0]> is set to "01", receive starts.

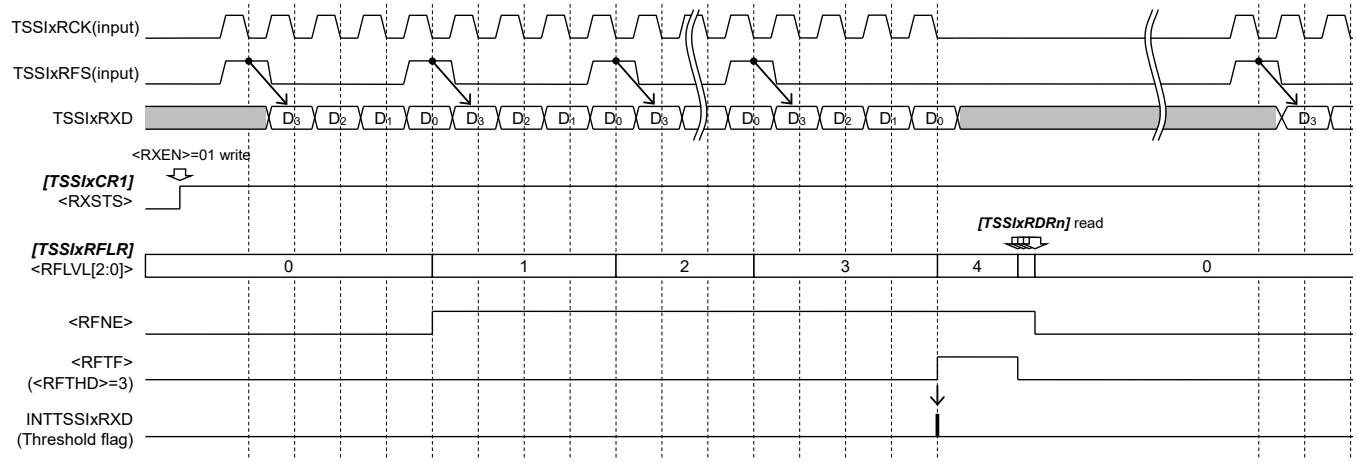
  Starts output of the transfer clock (TSSIxRCK) and detects receive data from the TSSIxRXD input in synchronization with the falling edge of TSSIxRCK after outputting the receive frame synchronization signal (TSSIxRFS).

- Slave receive

  After setting ***[TSSIxCR1]***<RXEN[1:0]> to "01", receive starts when a receive frame synchronization signal (TSSIxRFS) is detected. After detecting the frame synchronization signal, the receive data is detected from the TSSIxRXD input in synchronization with the falling edge of TSSIxRCK.



(a) Master receive (***[TSSIxRCMR]***<RCKOUT[1:0]> = 10, ***[TSSIxRFMR]***<RDFC[3:0]> = 0011)



(b) Slave receive (with transfer clock only during transfer)

Figure 3.4 Operation example of receive mode

3.6.3. Transmit and receive (full duplex communication)

Check that ***[TSSIxCR1]***<TXSTS> and ***[TSSIxCR1]***<RXSTS> are “0” before setting the transmitter and receiver.

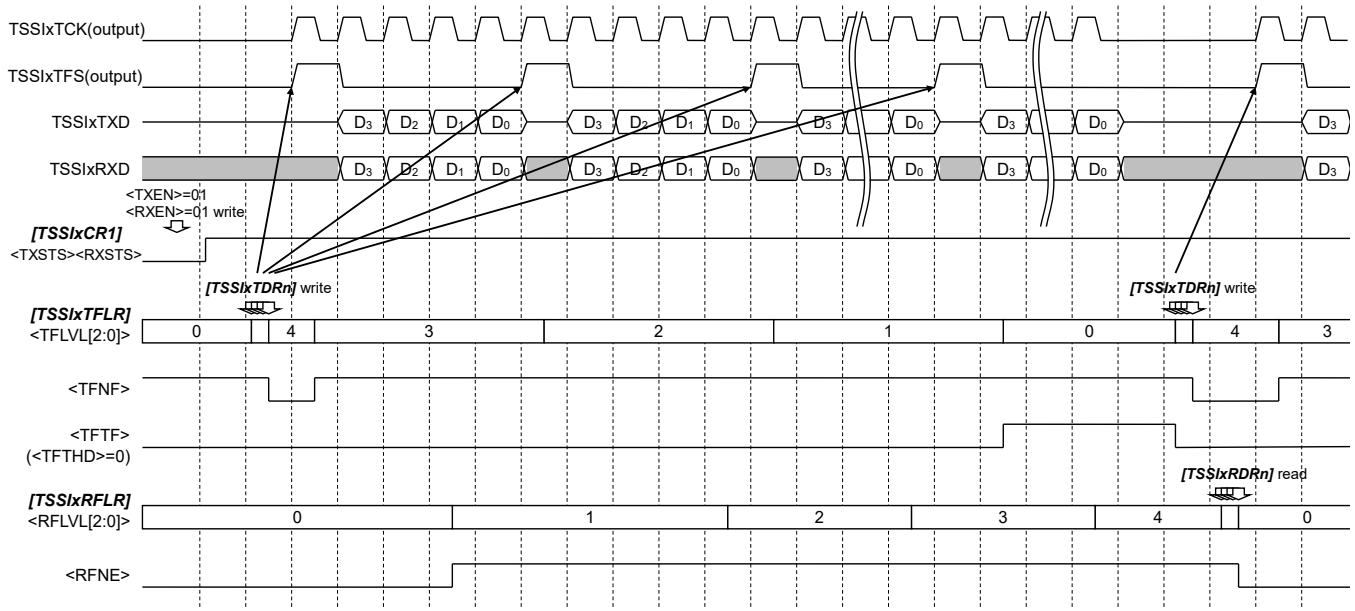
Set ***[TSSIxRCMR]***, ***[TSSIxRFMR]***, ***[TSSIxRFTLR]***, ***[TSSIxTCMR]***, ***[TSSIxTFMR]***, ***[TSSIxTFTLR]***. Set ***[TSSIxRIER]***, ***[TSSIxRDMACR]***, ***[TSSIxTIER]***, and ***[TSSIxTDMACR]*** for interrupts and DMA requests. Set ***[TSSIxRCR]*** to the expected value data in as necessary.

- Master transmit and receive

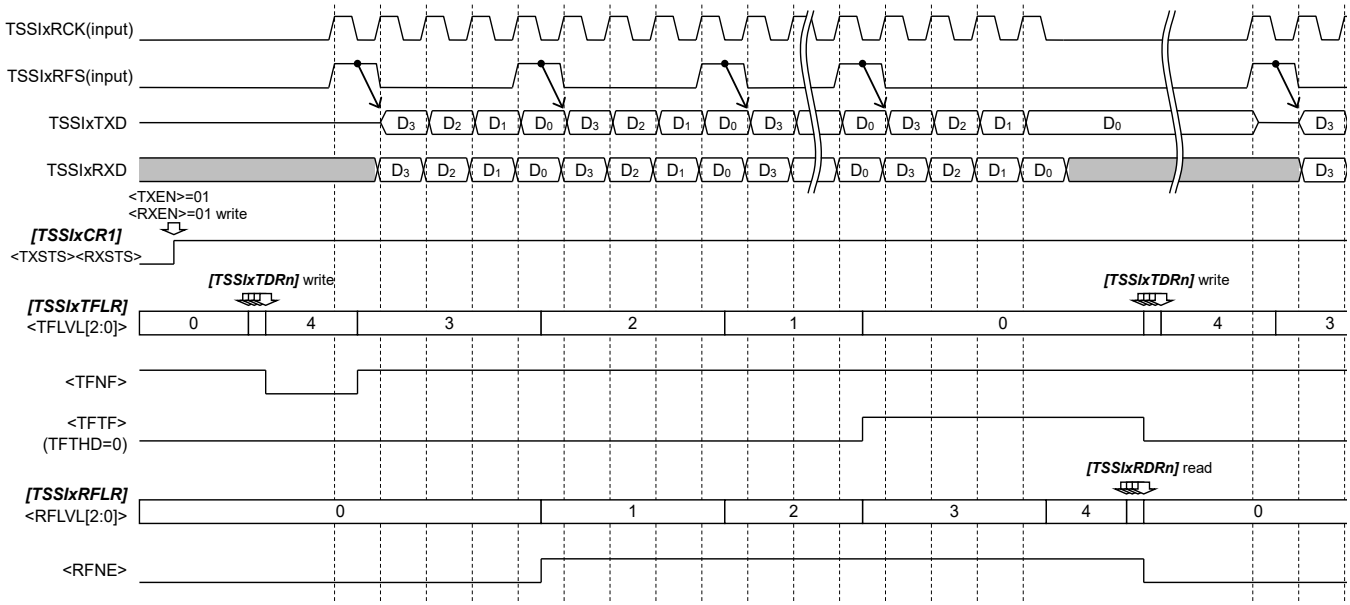
Set ***[TSSIxCR1]***<TXEN[1:0]> and ***[TSSIxCR1]***<RXEN[1:0]> to “01”. After checking that ***[TSSIxCR1]*** <TXSTS> and ***[TSSIxCR1]***<RXSTS> have become “1”, write transmit data to ***[TSSIxTDRn]*** to start transmit and receive. The output of the transfer clock(TSSIxTCK) is started, the transmit data is output to TSSIxTXD in synchronization with the rising edge of TSSIxTCK after the frame synchronization signal(TSSIxTFS) is output, and the receive data is detected from the TSSIxRXD input in synchronization with the falling edge of TSSIxTCK.

- Slave transmit and receive

Set ***[TSSIxCR1]***<TXEN[1:0]> and ***[TSSIxCR1]***<RXEN[1:0]> to “01”. After checking that ***[TSSIxCR1]*** <TXSTS> and ***[TSSIxCR1]***<RXSTS> have become "1", write the transmit data to ***[TSSIxTDRn]*** and detect the frame sync signal (TSSIxRFS) to start transmit and receive. After detecting the frame synchronization signal, the transmit data is output to TSSIxTXD in synchronization with the rising edge of TSSIxRCK, and the receive data is detected from the TSSIxRXD input in synchronization with the falling edge of TSSIxRCK.



(a) Master transmit and receive (***[TSSIxTCMR]***<TCKOUT[1:0]> = 10)



(b) Slave transmit and receive (with transfer clock only during transfer)

Figure 3.5 Operation example of full duplex communication

3.6.4. Receive data comparison

The receive data comparison function can be used for slave receive or slave transmit and receive.
Set ***[TSSIxRFMR]***<CMPDS[3:0]> to the comparison area size and ***[TSSIxRCR]*** <CMPDP[15:0]> to the comparison data pattern. After that, the receive data comparison function can be used by setting the receive start trigger selection (***[TSSIxRCMR]***<RSTART[1:0]>) to "11".

The communication of the data area can be controlled by comparing the comparison area of up to 16 bits after inputting the frame synchronization signal with the comparison data pattern (***[TSSIxRCR]*** <CMPDP[15:0]>).
When the comparison area matches, receive or transmit and receive the data area.
When the comparison area does not match, slave does not receive or transmit and receive the data area. When the TSSIxRFS input is asserted again, the next receive data is compared.

Note: Receive data in the comparison area is not treated as a data and is not stored in the receive FIFO.

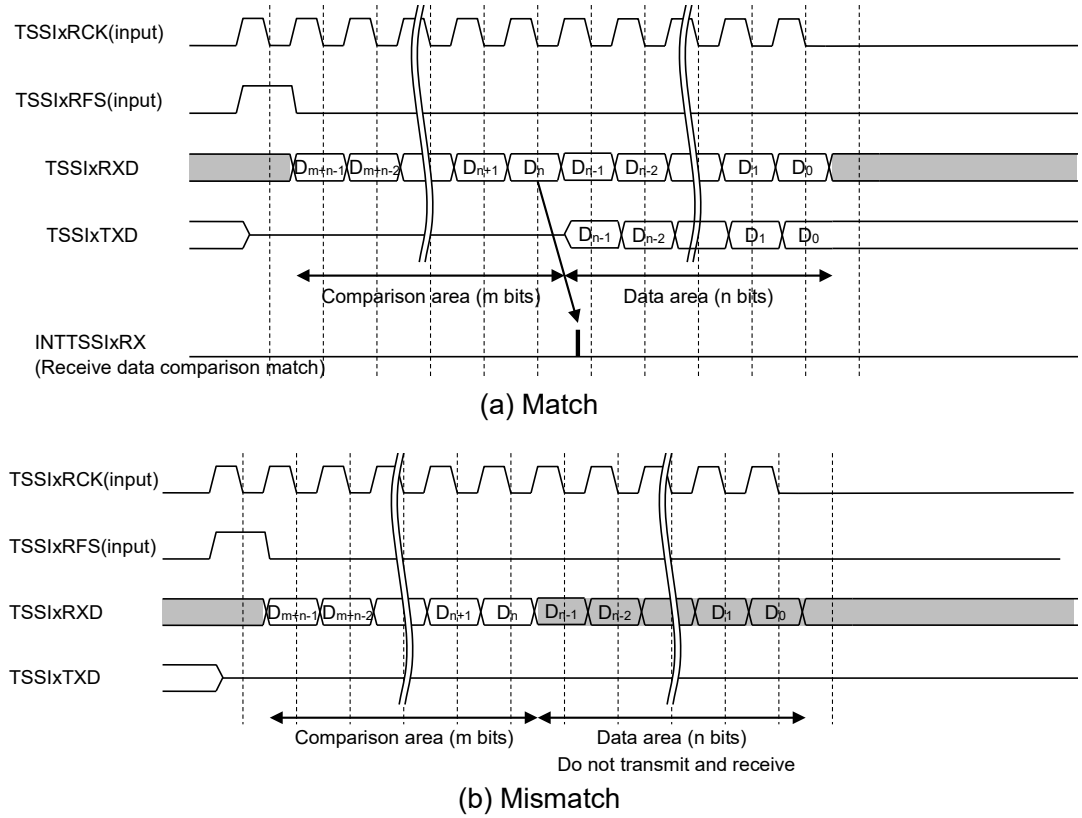


Figure 3.6 Transmit and receive waveform of using receive data comparison function

3.6.5. Suspending and resuming communication

In the case of the master device, communication can be suspended and resumed.

- Transmit

  When “10” is written to ***[TSSIxCR1]***<TXEN[1:0]>, the transmitter is disabled when the current frame is completed, regardless of the presence or absence of data in the transmit FIFO. (Suspend)

  When “01” is written to ***[TSSIxCR1]***<TXEN[1:0]>, the remaining transmit FIFO data is transmitted. (Resume)

- Receive

  When “10” is written to ***[TSSIxCR1]***<RXEN[1:0]>, the receiver is disabled when the current frame is completed. (Suspend)

  When “01” is written to ***[TSSIxCR1]***<RXEN[1:0]>, the receive start. (Resume)

- Transmit and receive

  When “10” is written to ***[TSSIxCR1]***<TXEN[1:0]> and ***[TSSIxCR1]***<RXEN[1:0]> at the same time, the transmitter and receiver are disabled at the completion of the current frame regardless of the presence or absence of data in the transmit FIFO. (Suspend)

  When “01” is written to ***[TSSIxCR1]***<TXEN[1:0]> and ***[TSSIxCR1]***<RXEN[1:0]> at the same time, the remaining transmit FIFO data is transmitted and received at the same time. (Resume)

3.7. Transfer waveform

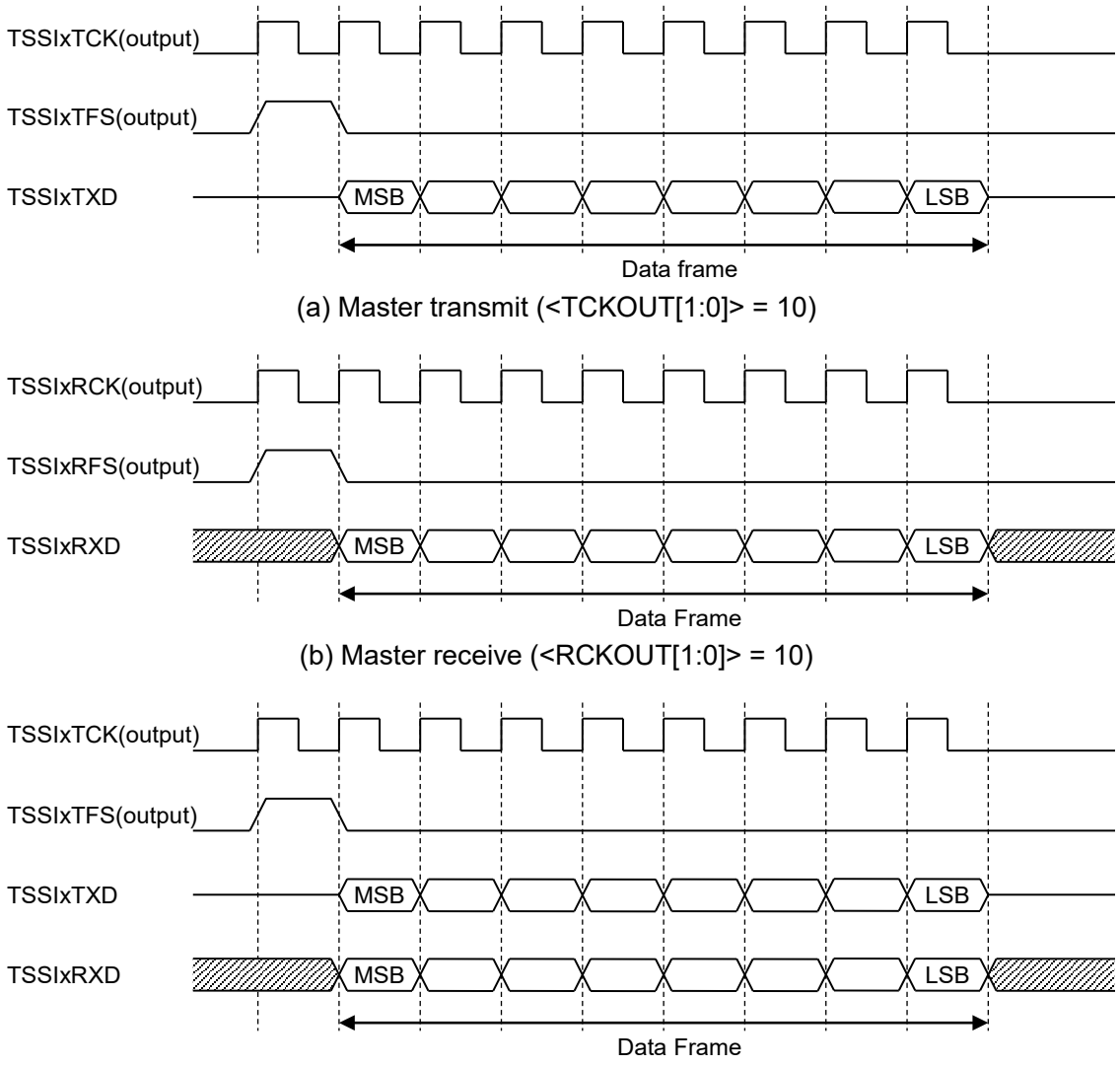


Figure 3.7 Single transfer waveform of master transfer

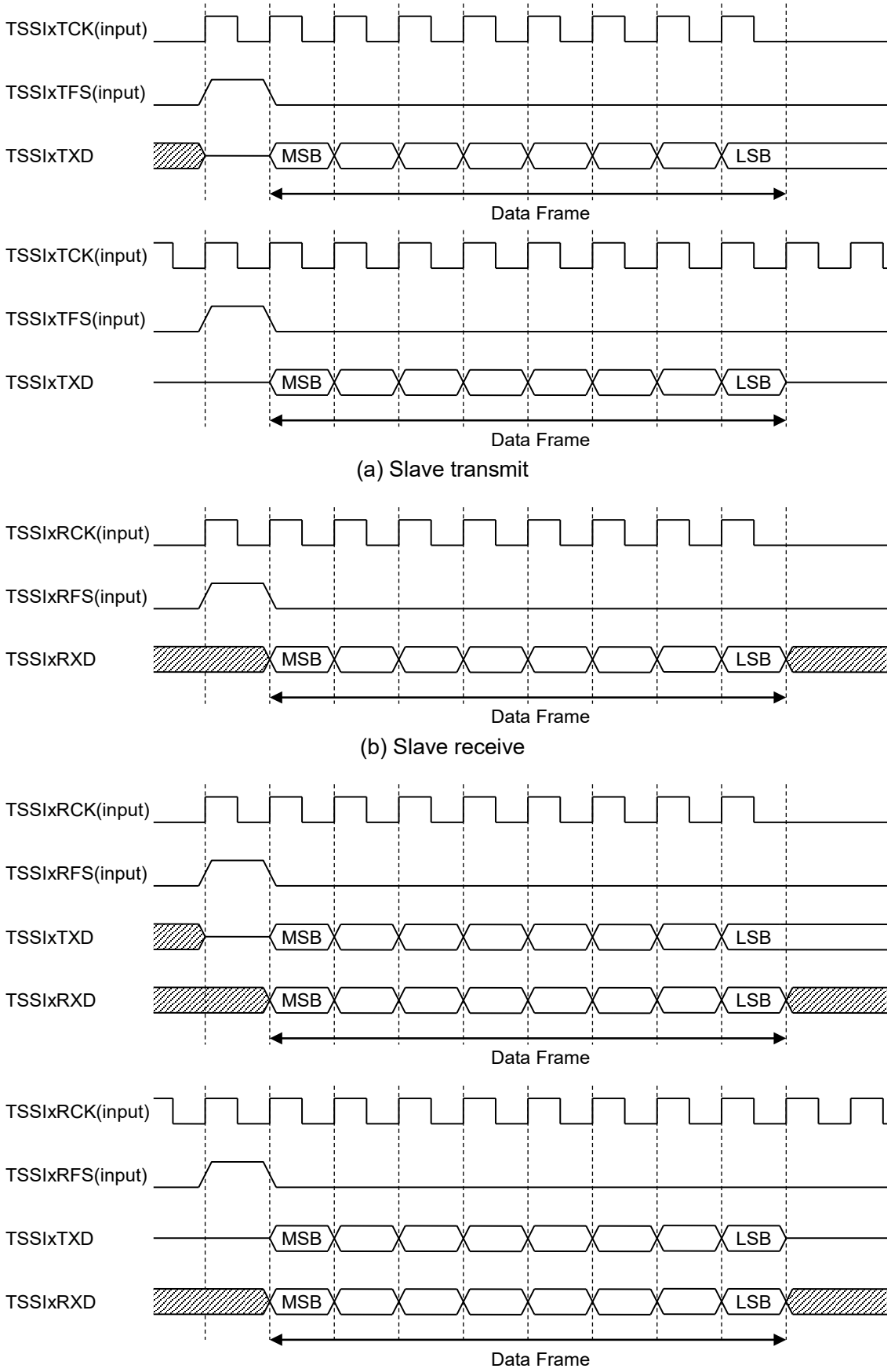
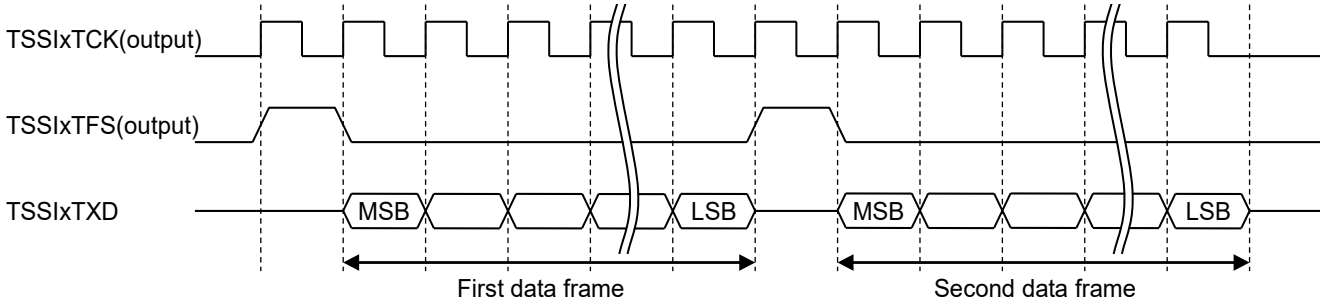
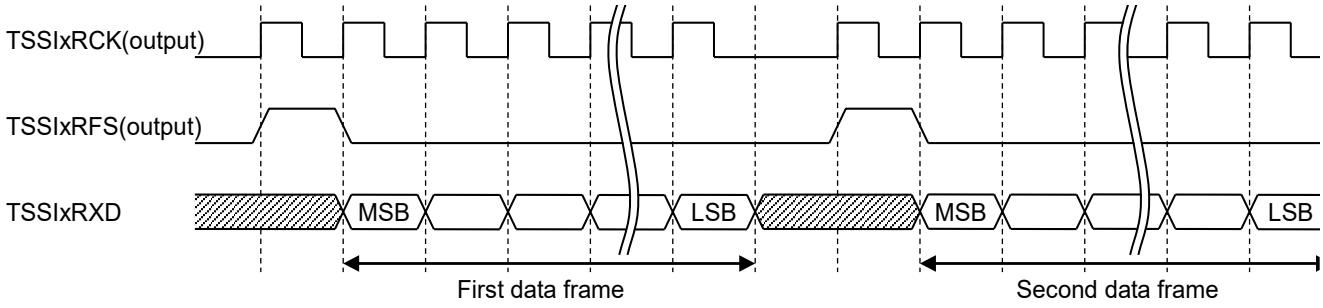


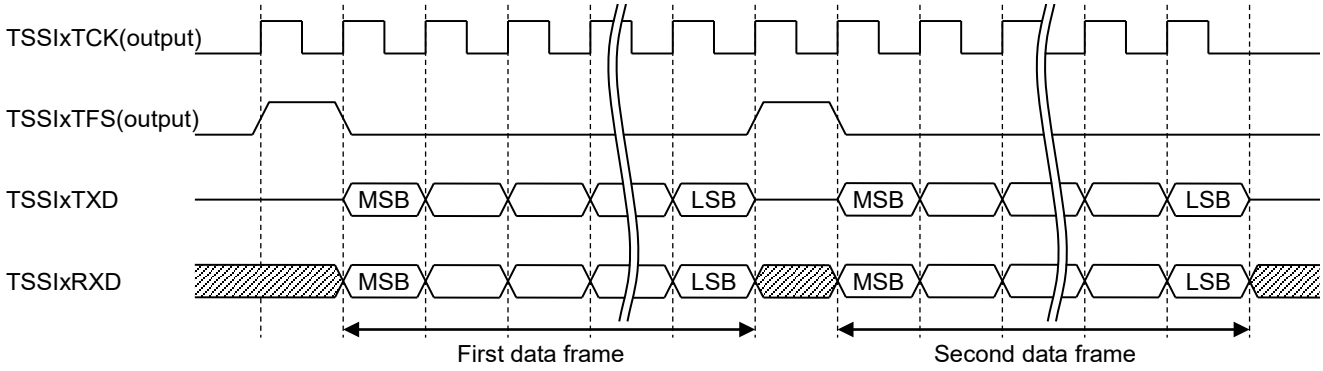
Figure 3.8 Single transfer waveform of slave transfer



(a) Master transmit (<TCKOUT[1:0]> = 10)



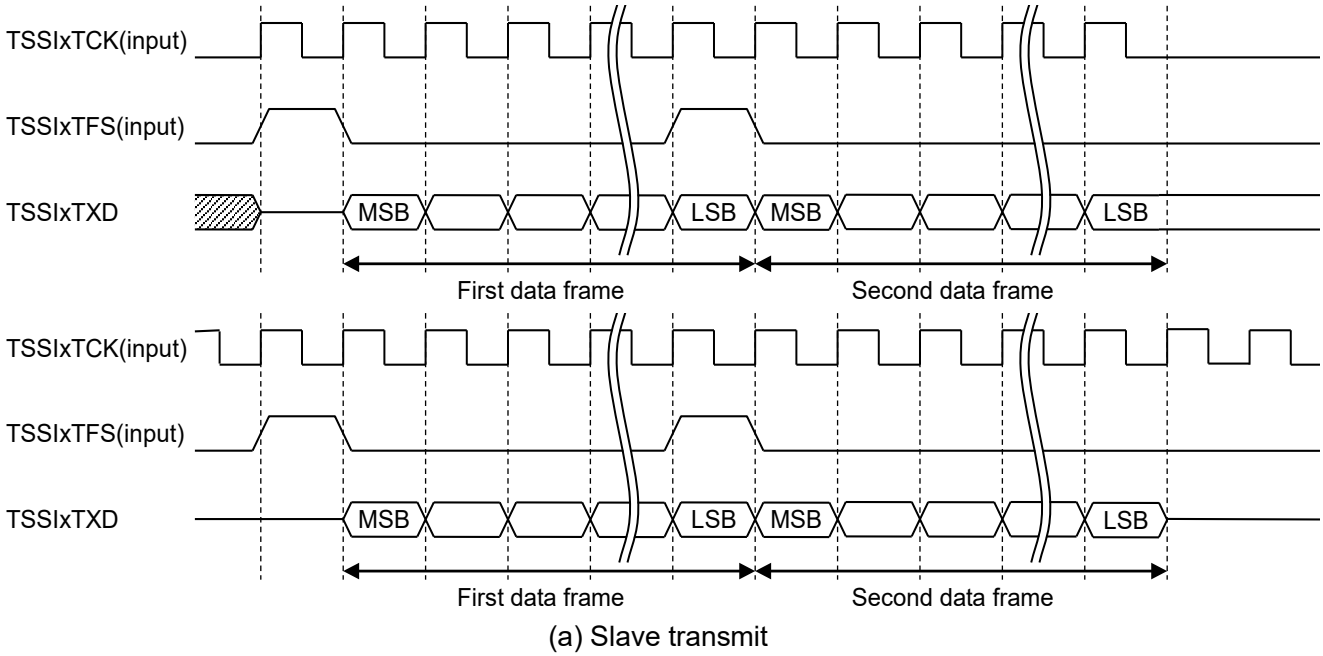
(b) Master receive (<RCKOUT[1:0]> = 10)



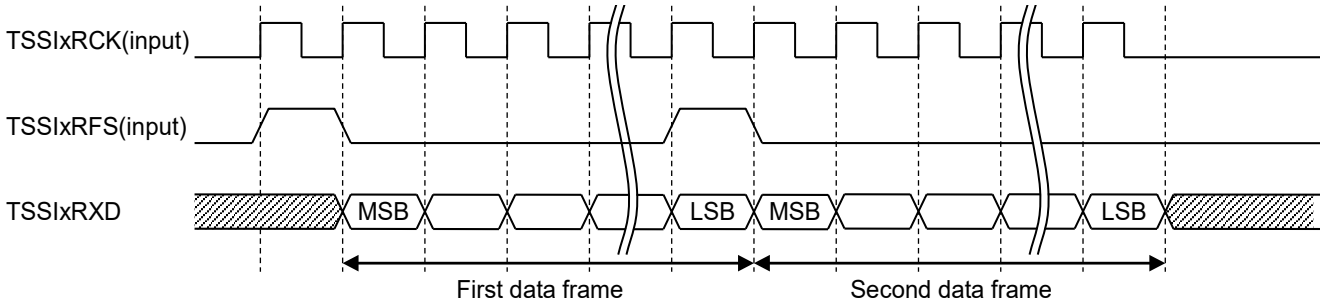
(c) Master transmit and receive (<TCKOUT[1:0]> = 10)

Figure 3.9 Continuous transfer waveform of master transfer

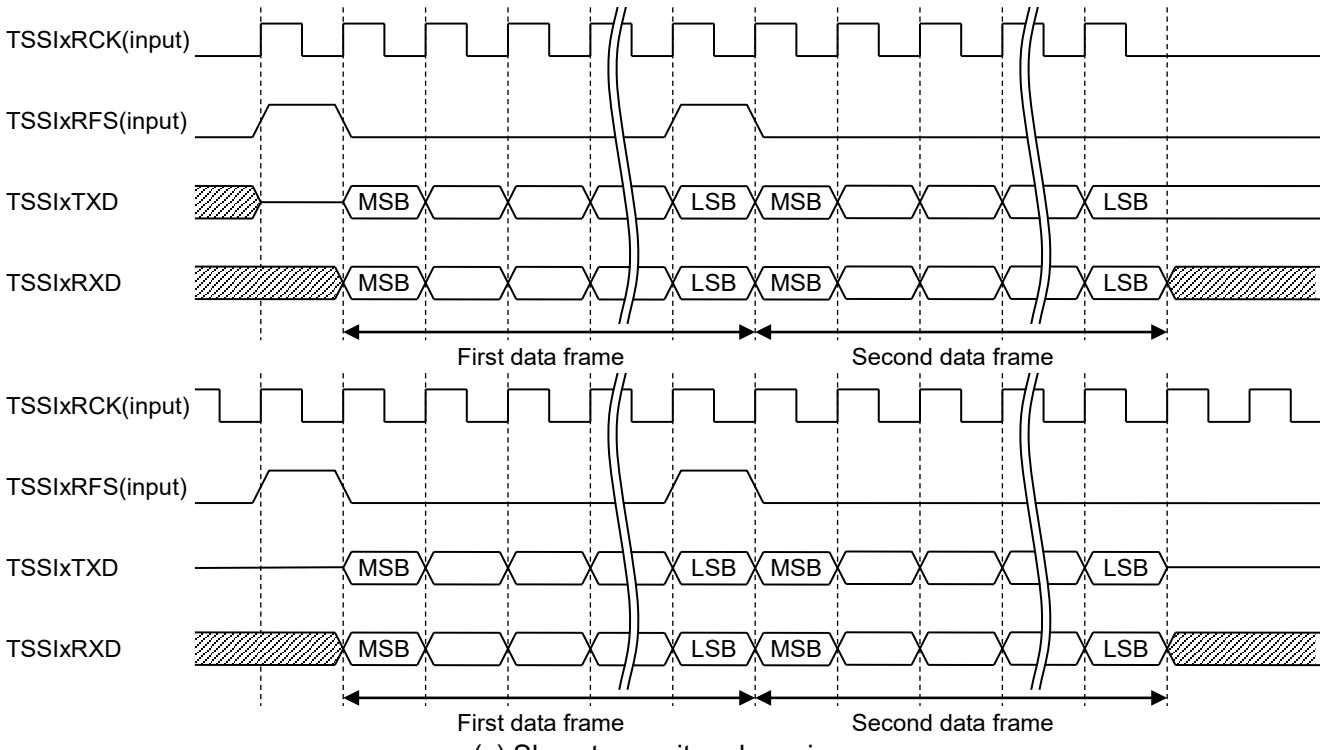
Note: In the master, an idle period of at least one communication serial clock cycle is inserted between data frames. This idle period is extended when the master cannot process the FIFO data in time.



(a) Slave transmit



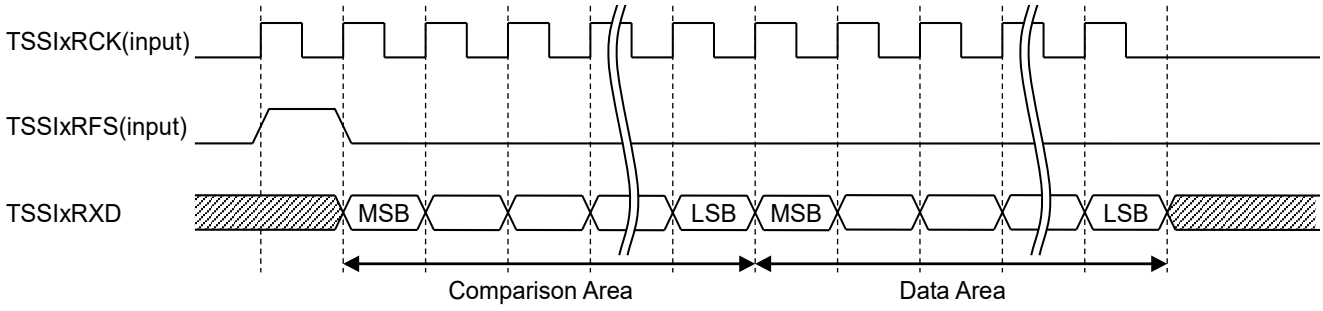
(b) Slave receive



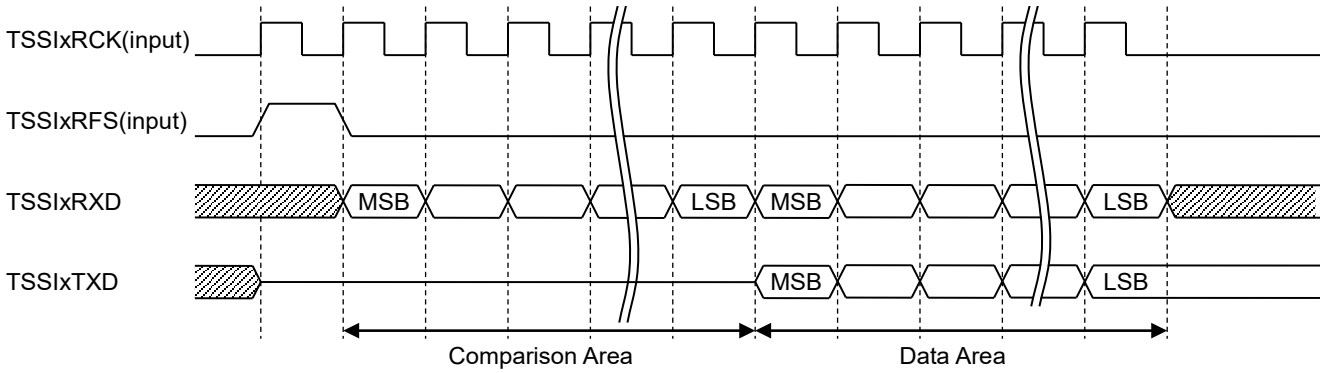
(c) Slave transmit and receive

Figure 3.10 Continuous transfer waveform of slave transfer

Note: The slave can communicate continuous data frames without idle periods. However, in the slave, when the external master device requests communication, the FIFO resources must be appropriately managed. Otherwise, a FIFO overrun or underrun error will occur.

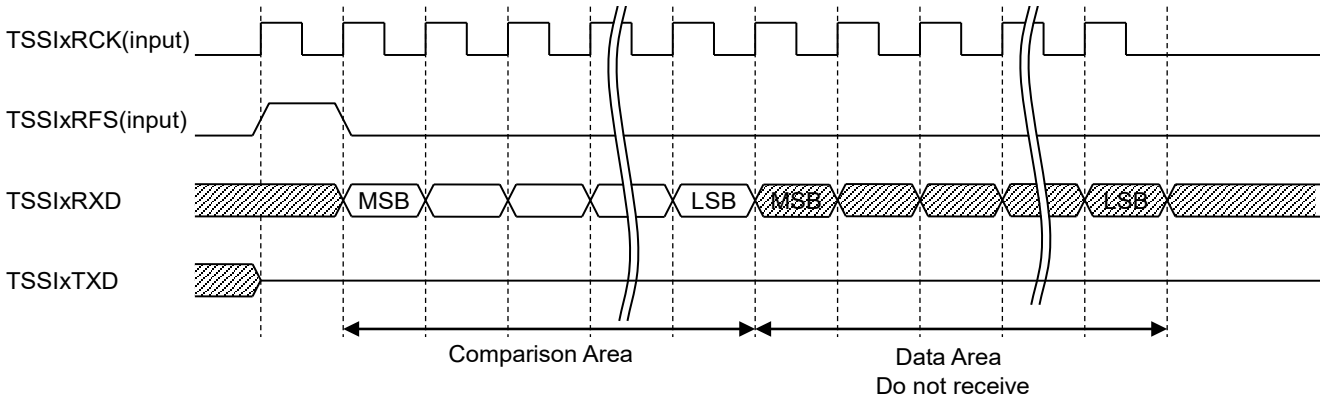


(a) Slave receive



(b) Slave transmit and receive

Figure 3.11 Transfer waveform of using receive data comparison function (Match)



(a) Slave transmit and receive

Figure 3.12 Transfer waveform of using receive data comparison function(Mismatch)

Note: The receive data comparison function can also be used for slave receive. In that case, TSSIxTXD is not output (output is disabled).

3.8. Interrupt request

The TSSI has three types of interrupt outputs: receive interrupt (INTTSSIxRX), transmit interrupt (INTTSSIxTX), and error interrupt (INTTSSIxERR). Each interrupt has multiple interrupt factors, and can be set enable / disable for each interrupt factor. Table 3.4 shows the relationship between interrupt outputs and interrupt factors.

INTTSSIxRX and INTTSSIxTX are pulse signals, and INTTSSIxERR is a level signal.

Table 3.4 Interrupt outputs and interrupt factors

Interrupt output	Interrupt factor	Status register	Enable register
Receive interrupt INTTSSIxRX	Receive FIFO not empty	***[TSSIxRSR]***<RFNE>	***[TSSIxRIER]***<RFNEIE>
	Receive FIFO threshold flag	***[TSSIxRSR]***<RFTF>	***[TSSIxRIER]***<RFTFIE>
	Receive data comparison match	-	***[TSSIxRIER]***<RCMIE>
	Receive FIFO transfer completed	-	***[TSSIxRIER]***<RFTEIE>
Transmit interrupt INTTSSIxTX	Transmit FIFO not full	***[TSSIxTSR]***<TFNF>	***[TSSIxTIER]***<TFNFIE>
	Transmit FIFO threshold flag	***[TSSIxTSR]***<TFTF>	***[TSSIxTIER]***<TFTFIE>
Error interrupt INTTSSIxERR	Receive FIFO overrun	***[TSSIxRSR]***<RFOR>	***[TSSIxRIER]***<RFORIE>
	Receive FIFO underrun	***[TSSIxRSR]***<RFUR>	***[TSSIxRIER]***<RFURIE>
	Transmit FIFO overrun	***[TSSIxTSR]***<TFOR>	***[TSSIxTIER]***<TFORIE>
	Transmit FIFO underrun	***[TSSIxTSR]***<TFUR>	***[TSSIxTIER]***<TFURIE>

3.8.1. Receive interrupt

(1) Receive FIFO not empty -----During receive data FIFO transfer

[TSSIxRSR]<RFNE> is set to “1” when the receive FIFO becomes not empty. When ***[TSSIxRIER]*** <RFNEIE> is set to “1”, a receive interrupt occurs.

(2) Receive FIFO threshold flag-----During receive data FIFO transfer

When the number of receive FIFO entries is greater than the receive FIFO threshold register (***[TSSIxRFTLR]***<RFTHD[2:0]>), ***[TSSIxRSR]***<RFTF> is set to “1”. When ***[TSSIxRIER]***<RFTFIE> is set to “1”, a receive interrupt occurs.

(3) Receive data comparison match-----When the last bit of the comparison area is detected

When the receive data comparison function is used and ***[TSSIxRIER]***<RCMIE> is set to “1”, if the value of the receive data comparison area matches the value of the receive data comparison register (***[TSSIxRCR]***<CMPDP[15:0]>), a receive interrupt occurs.

(4) Receive FIFO transfer completed-----During receive data FIFO transfer

When ***[TSSIxRIER]***<RFTEIE> is set to "1", a receive interrupt occurs when the number of receive FIFO entries (***[TSSIxRFLR]***<RFLVL[2:0]>) exceeds ***[TSSIxRFMR]***<RDFC[3:0]>. However, when <RDFC[3:0]> = 1111, a receive interrupt occurs when transfer stop.

3.8.2. Transmit interrupt

(1) Transmit FIFO not full
It is enabled by setting ***[TSSIxTIER]***<TFNFIE> to "1".
When the transmit FIFO not full(***[TSSIxTSR]***<TFNF>) is set to “1”, a transmit interrupt occurs.

(2) Transmit FIFO threshold flag
When the number of transmit FIFO entries is less than or equal to the transmit FIFO threshold register(***[TSSIxTFTLR]***<TFTHD[2:0]>), ***[TSSIxTSR]***<TFTF> is set to "1".
When ***[TSSIxTIER]***<TFTFIE> is set to “1”, a transmit interrupt occurs.

3.8.3. Error interrupt

The following error interrupt occurs. When an error occurs, take appropriate action.
Since the error interrupt is a level output, the error interrupt is not released unless the status registers of all enabled interrupt factors are cleared.

(1) Receive FIFO overrun
It is enabled by setting ***[TSSIxRIER]***<RFORIE> to "1".
When detected the receive FIFO overrun (***[TSSIxRSR]***<RFOR> is set to “1”), an error interrupt occurs.
<RFOR> is cleared by writing “1”.

(2) Receive FIFO underrun
It is enabled by setting ***[TSSIxRIER]***<RFURIE> to “1”.
When detected the receive FIFO underrun (***[TSSIxRSR]***<RFUR> is set to “1”), an error interrupt occurs.
<RFUR> is cleared by writing “1”.

(3) Transmit FIFO overrun
It is enabled by setting ***[TSSIxTIER]***<TFORIE> to “1”.
When detected the transmit FIFO overrun (***[TSSIxTSR]***<TFOR> is set to “1”), an error interrupt occurs.
<TFOR> is cleared by writing “1”.

(4) Transmit FIFO underrun
It is enabled by setting ***[TSSIxTIER]***<TFURIE> to “1”.
When detected the transmit FIFO underrun (***[TSSIxTSR]***<TFUR> is set to “1”), an error interrupt occurs.
<TFUR> is cleared by writing “1”.

Table 3.5 About the occurrence of FIFO error

Operation mode		Transmit FIFO overrun	Transmit FIFO Underrun	Receive FIFO overrun	Receive FIFO Underrun
Master	transmit	Occur	not occur	-	-
	Receive	-	-	not occur	Occur
	Transmit and receive	Occur	Occur	Occur	Occur
Slave	transmit	Occur	Occur	-	-
	Receive	-	-	Occur	Occur
	Transmit and receive	Occur	Occur	Occur	Occur

3.9. DMA request

DMA has transmit DMA request and receive DMA request. DMA requests of the TSSI are single request.

- Transmit DMA request

  It is enabled by setting ***[TSSIxTDMACR]***<TDMAE> to “1”. When there is more than one empty stage in the transmit FIFO, a transmit DMA request occurs.

  Note1:When the transmit DMA request is enabled, do not write to the transmit FIFO by software.
  Note2:When the transmit DMA request is enabled, access to ***[TSSIxTDRn]*** may occur. Set ***[TSSIxTFMR]***<TDFS[4:0]> before enabling.

- Receive DMA request

  It is enabled by setting ***[TSSIxRDMACR]***<RDMAE> to “1” . When there is more than one data in the receive FIFO, a receive DMA request occurs.

  Note 1: When the receive DMA request is enabled, do not read from the receive FIFO by software.
  Note 2: When the receive DMA request is enabled, access to ***[TSSIxRDRn]*** may occur. Set ***[TSSIxRFMR]***<RDFS[4:0]> before enabling.

3.10. Software reset

Software reset is possible by writing to the TSSI control register 0(***[TSSIxCR0]***). The target areas are overall / receiver / receive FIFO / transmitter / transmit FIFO. Table 3.6 shows the registers initialized by each software reset.

When software reset, read ***[TSSIxCR0]*** to check the completion of software reset of the target area, and then access the target area.

Table 3.6 Software reset and initialization register

Register	Software reset				
	Overall *[TSSIxCR0]* <SWRST>	Receiver *[TSSIxCR0]* <RXSWRST>	Receive FIFO *[TSSIxCR0]* <RXFCLR>	Transmitter *[TSSIxCR0]* <TXSWRST>	Transmit FIFO *[TSSIxCR0]* <TXFCLR>
[TSSIxCR0]	-	-	-	-	-
[TSSIxCR1]	-	-	-	-	-
[TSSIxCPR]	-	-	-	-	-
[TSSIxRCMR]	✓	✓	-	-	-
[TSSIxRFMR]	✓	✓	-	-	-
[TSSIxRCR]	✓	✓	-	-	-
[TSSIxRDMACR]	✓	✓	-	-	-
[TSSIxRSR]	✓	✓	-	-	-
[TSSIxRIER]	✓	✓	-	-	-
[TSSIxRFTLR]	✓	✓	-	-	-
[TSSIxRFLR]	✓	✓	✓	-	-
[TSSIxRDRn]	✓	✓	✓	-	-
[TSSIxTCMR]	✓	-	-	✓	-
[TSSIxTFMR]	✓	-	-	✓	-
[TSSIxTDMACR]	✓	-	-	✓	-
[TSSIxTSR]	✓	-	-	✓	-
[TSSIxTIER]	✓	-	-	✓	-
[TSSIxTFTLR]	✓	-	-	✓	-
[TSSIxTFLR]	✓	-	-	✓	✓
[TSSIxTDRn]	✓	-	-	✓	✓

Note: ✓: Target, -: Not target

4. Register description

4.1. Register list

The control registers and addresses are as follows.

Peripheral functions		Channel / unit	Base address		
			TYPE1	TYPE2	TYPE3
Synchronous serial interface	TSSI	ch0	-	0x400CD000	0x4006D000
		ch1	-	0x400CD400	0x4006D400

Note: The channel / unit and base address type used differ depending on the product. For details, refer to reference Manual “Product Specific Information”.

Register name		Address (Base +)
TSSI Control Register 0	***[TSSIxCR0]***	0x0000
TSSI Control Register 1	***[TSSIxCR1]***	0x0004
TSSI Clock Division Register	***[TSSIxCPR]***	0x0010
TSSI Receive Clock / Mode Control Register	***[TSSIxRCMR]***	0x0040
TSSI Receive Data Frame Control Register	***[TSSIxRFMR]***	0x0044
TSSI Receive Data Comparison Register	***[TSSIxRCR]***	0x0048
TSSI Receive DMA Control Register	***[TSSIxRDMACR]***	0x004C
TSSI Receive Status Register	***[TSSIxRSR]***	0x0060
TSSI Receive Interrupt Enable Register	***[TSSIxRIER]***	0x0064
TSSI Receive FIFO Threshold Register	***[TSSIxRFTLR]***	0x0070
TSSI Receive FIFO Entry Register	***[TSSIxRFLR]***	0x0074
TSSI Receive Data Register 0	***[TSSIxRDR0]***	0x0080
TSSI Receive Data Register 1	***[TSSIxRDR1]***	0x0084
TSSI Receive Data Register 2	***[TSSIxRDR2]***	0x0088
TSSI Receive Data Register 3	***[TSSIxRDR3]***	0x008C
TSSI Transmit Clock / Mode Control Register	***[TSSIxTCMR]***	0x00A0
TSSI Transmit Data Frame Control Register	***[TSSIxTFMR]***	0x00A4
TSSI Transmit DMA Control Register	***[TSSIxTDMACR]***	0x00AC
TSSI Transmit Status Register	***[TSSIxTSR]***	0x00C0
TSSI Transmit Interrupt Enable Register	***[TSSIxTIER]***	0x00C4
TSSI Transmit FIFO Threshold Register	***[TSSIxTFTLR]***	0x00D0
TSSI Transmit FIFO Entry Register	***[TSSIxTFLR]***	0x00D4
TSSI Transmit Data Register 0	***[TSSIxTDR0]***	0x00E0
TSSI Transmit Data Register 1	***[TSSIxTDR1]***	0x00E4
TSSI Transmit Data Register 2	***[TSSIxTDR2]***	0x00E8
TSSI Transmit Data Register 3	***[TSSIxTDR3]***	0x00EC

4.2. Register details

4.2.1. *[TSSIxCR0]* (TSSI Control Register 0)

Bit	Bit Symbol	After Reset	Type	Function
31	SWRST	0	W	Software reset of the overall. (Note1), (Note2) The processing issued by this bit has priority over other software reset settings. 0: - 1: Overall software reset
			R	0: Not during overall software reset 1: During overall software reset
30:16	-	0	R	Read as “0”.
15	TXSWRST	0	W	Software reset of the transmitter. (Note1), (Note2) The processing issued by this bit has priority over the setting of <TXFCLR>. 0: - 1: Transmitter software reset
			R	0: Not during transmitter software reset 1: During transmitter software reset
14	TXFCLR	0	W	Software reset of the transmit FIFO. (Note1), (Note2) 0: - 1: Transmit FIFO software reset
			R	0: Not during transmit FIFO software reset 1: During transmit FIFO software reset
13:8	-	0	R	Read as “0”.
7	RXSWRST	0	W	Software reset of the receiver. (Note1), (Note2) The processing issued by this bit has priority over the setting of <RXFCLR>. 0: - 1: Receiver software reset
			R	0: Not during the receiver software reset 1: During receiver software reset
6	RXFCLR	0	W	Software reset of the receive FIFO. (Note1), (Note2) 0: - 1: Receive FIFO software reset
			R	0: Not during receive FIFO software reset 1: During receive FIFO software reset
5:0	-	0	R	Read as “0”.

Note1: When performed software reset , read to check the completion of software reset.

Note2: For the registers subject to software reset, refer to "**Table 3.6 Software reset and initialization register**".

4.2.2. *[TSSIxCR1]* (TSSI Control Register 1)

Bit	Bit Symbol	After Reset	Type	Function
31:11	-	0	R	Read as “0”.
10	TXSTS	0	R	Transmitter status Indicates the status of the transmitter. This bit is affected by <TXEN[1:0]>. 0: Disable 1: Enable When the transmitter is enabled, writing to the following registers is ignored. ***[TSSIxTCMR]***, ***[TSSIxTFMR]***, ***[TSSIxTFTLR]***
9:8	TXEN[1:0]	00	W	Transmitter enable / disable control Enable or disable the transmitter (Note 1) (Note 2) 00: - 01: Transmitter enable 10: Transmitter disable (Wait for completion of data frame processing during communication with master) 11: Transmitter disable (does not wait for completion of processing during communication) Read as “00”.
7:3	-	0	R	Read as “0”.
2	RXSTS	0	R	Receiver status Indicates the status of the receiver. This bit is affected by <RXEN[1:0]>. 0: Disable 1: Enable When the receiver is enabled, writing to the following registers is ignored. ***[TSSIxRCMR]***, ***[TSSIxRFMR]***, ***[TSSIxRFTLR]***, ***[TSSIxRCR]***
1:0	RXEN[1:0]	00	W	Receiver enable / disable control Enable or disable the receiver (Note 1) (Note 2) 00: - 01: Receiver enable 10: Receiver disable (Wait for completion of data frame processing during communication with master) 11: Receiver disable (does not wait for completion of processing during communication) Read as “00”.

Note 1: When using <TXEN[1:0]> and <RXEN[1:0]> in transmit and receive mode and suspending / resuming, control them at the same time.

Note 2: When disabled by <TXEN[1:0]> = 11, <RXEN[1:0]> = 11, reset by software before resuming.

4.2.3. *[TSSIxCPR]* (TSSI Clock Division Register)

Bit	Bit Symbol	After Reset	Type	Function
31:8	-	0	R	Read as “0”.
7:0	DIV[7:0]	00000001	R/W	Divider setting (Note 2) Divide ΦT0 to generate a divided clock (SCLK). 0x01 to 0xFF: SCLK frequency = ΦT0 frequency / (<DIV[7:0]>+1) <DIV[0]> is “1” regardless of the written value.

Note 1: Set this register when ***[TSSIxCR1]***<RXSTS> = ***[TSSIxCR1]***<TXSTS> = 0.

Note 2: SCLK frequency must be equal to or less than 1/2 of fsys frequency.

4.2.4. *[TSSIxRCMR]* (TSSI Receive Clock / Mode Control Register)

Bit	Bit Symbol	After Reset	Type	Function
31:10	-	0	R	Read as “0”.
9:8	RSTART[1:0]	00	R/W	Receive start trigger selection 00: When ***[TSSIxCR1]***<RSTS> = 1 and the receive FIFO becomes empty, it starts immediately. (master receive) 01: Cooperate operation with transmitter (master transmit and receive) (Note 2) 10: TSSIxRFS input (slave receive, slave transmit and receive) 11: TSSIxRFS input (use receive data comparison function, slave receive, slave transmit and receive)
7	-	0	R/W	Write “0”.
6:4	-	0	R	Read as “0”.
3:2	RCKOUT[1:0]	00	R/W	Receive clock output mode selection Select the input / output direction of TSSIxRCK and the gating method at the time of output. (Note 3) 00: TSSIxRCK is input, TSSIxRFS is input (Slave receive, slave transmit and receive, master transmit and receive) 01: TSSIxRCK is always clock output, TSSIxRFS is output (Master receive) 10: TSSIxRCK is clock output during transfer, TSSIxRFS is outputs (Master receive) 11: Reserved
1:0	RCKSEL[1:0]	00	R/W	Receive clock selection Select the clock used for receive. 00: SCLK (master) 01: Reserved 10: TSSIxRCK (slave) 11: Reserved

Note 1: For details on this register setting, refer to “Table 3.1 Operation mode settings and combinations of pins used”.

Note 2: When the receiver is operated cooperatively

- Do not set ***[TSSIxTCMR]***<TSTART[1:0]> to “01”.
- Set ***[TSSIxRCMR]***<RCKSEL[1:0]> = ***[TSSIxTCMR]***<TCKSEL[1:0]>.

Note 3: Make port settings. For details on port settings, refer to reference manual “Input / output ports”.

Note 4: When the receive is enabled (***[TSSIxCR1]***<RXSTS> = 1), this register is write-protected.

Note 5: When ***[TSSIxRCMR]***<RSTART[1:0]> = 01, do not rewrite ***[TSSIxTCMR]*** even during receive operation.

Note 6: When ***[TSSIxTCMR]***<TSTART[1:0]> = 01, do not rewrite ***[TSSIxRCMR]*** and ***[TSSIxRCR]*** during transmit operation.

4.2.5. *[TSSIxRFMR]* (TSSI Receive Data Frame Control Register)

Bit	Bit Symbol	After Reset	Type	Function
31:16	-	0	R	Read as “0”.
15:12	CMPDS[3:0]	0000	R/W	Comparison data size The receive data comparison function controls the data size for matching. Compares the receive data and ***[TSSIxRCR]*** for the number of bits set in this field. Available when ***[TSSIxRCMR]***<RSTART[1:0]> = 11. 0x0: 1bit 0x1: 2bits 0x2: 3bits 0x3: 4bits 0xE: 15bits 0xF: 16bits
11:8	RDFC[3:0]	0000	R/W	Number of receive data frames In master transmit and receive mode or master receive mode, specifies the number of data frames to transfer. Set this field to “0xF” in transmit and receive mode. In the receive mode, select any value within the settable range. 0xF: In receive mode, receive continues until stop processing is performed. In transmit and receive mode, communication start / stop control is the same as in transmit mode. Others: The data frame of <RDFC[3:0]> +1 is transferred. The FIFO transfer completion interrupt occurs when the number of receive to the FIFO matches this register setting. However, when this register is set to “0xF”, an interrupt occur at the time of transfer stop processing.
7:5	-	0	R	Read as “0”.
4:0	RDFS[4:0]	00000	R/W	Receive data frame size Specifies the data frame size of the receive frame. When comparing the receive data (***[TSSIxRCMR]***<RSTART[1:0]> = 11), set this field to the data area size of the receive data comparison format .(Note 2) The data size is (<RDFS[4:0]> +1) bits, but setting is prohibited for 3 bits or less. In the transmit and receive mode, set this field to the same value as ***[TSSIxTFMR]***<TDFS[4:0]>. When comparing receive data (***[TSSIxRCMR]***<RSTART[1:0]> = 11), set this field to the data area size in <RDFS[4:0]> and set ***[TSSIxTFMR]*** <TDFS[4:0]> to the data frame size.. 00000 to 00010: Reserved 00011: 4bits 00100: 5bits 11110: 31bits 11111: 32bits

Note1: This register is write-protected when the receiver is enabled (***[TSSIxCR1]***<RXSTS> = 1).

Note2: For the receive data comparison format, refer to “3.2 Data frame format”.

Note3: ***[TSSIxRDRn]*** (n = 0,1,2,3) can be accessed after setting this register.

4.2.6. *[TSSIxRCR]* (TSSI Receive Data Comparison Register)

Bit	Bit Symbol	After Reset	Type	Function
31:16	-	0	R	Read as “0”.
15:0	CMPDP[15:0]	0x0000	R/W	Comparison data pattern This value is compared with the receive data when using the receive data comparison function. Set the bits from the MSB side that is out of comparison target by ***[TSSIxRFMR]*** <CMPDS[3:0]> to “0”.

Note 1: The L bits(= ***[TSSIxRFMR]***<CMPDS[3:0]> +1) receive data is not stored in the receive FIFO and is compared with the expected value of ***[TSSIxRCR]***<CMPDP[15:0]> [L-1: 0]. When the comparison results match, the subsequent data is treated as a data area.

Note 2: When the receiver is enabled (***[TSSIxCR1]***<RXSTS> = 1), this register is write-protected.

Note 3: When ***[TSSIxTCMR]***<TSTART[1:0]> = 01, rewriting is prohibited even during transmit operation.

4.2.7. *[TSSIxRDMACR]* (TSSI Receive DMA Control Register)

Bit	Bit Symbol	After Reset	Type	Function
31:1	-	0	R	Read as “0”.
0	RDMAE	0	R/W	Receive DMA request control 0: Disable 1: Enable If there is an entry in the receive FIFO, generates a receive DMA request (TSSIxRXDMAREQ). The type of request is a single transfer request. Note: If this register is enabled, access to ***[TSSIxRDRn]*** (n = 0, 1,2,3) may occur.

Note1: This register can be rewritten when no access by DMAC occurs.

Note2: Set ***[TSSIxRFMR]***<RDFS[4:0]> before enabling this register.

4.2.8. *[TSSIxRSR]* (TSSI Receive Status Register)

The interrupt signal can be cleared by writing “1” to this register (writing “0” is not effective).

Bit	Bit Symbol	After Reset	Type	Function
31:6	-	0	R	Read as “0”.
5	RFUR	0	R	Receive FIFO underrun Clearing by software is required. 0: No underrun has occurred 1: Underrun occurred
			W	0: - 1: clear
4	RFOR	0	R	Receive FIFO overrun Clearing by software is required. 0: No overrun has occurred 1: Overrun occurred No overrun occurs during master receive.
			W	0: - 1: clear
3	-	0	R	Read as “0”.
2	RFTF	0	R	Receive FIFO threshold flag Shows the status according to the number of FIFO entries and the threshold setting. 0: ***[TSSIxRFLR]***<RFLVL[2:0]> ≤ ***[TSSIxRFTLR]***<RFTHD[2:0]> 1: ***[TSSIxRFLR]***<RFLVL[2:0]> > ***[TSSIxRFTLR]***<RFTHD[2:0]>
1	RFNE	0	R	Receive FIFO not empty 0: Receive FIFO is empty 1: Receive FIFO is not empty
0	RBSY	0	R	Receiver operating status 0: Waiting 1: Operating

4.2.9. *[TSSIxRIER]* (TSSI Receive Interrupt Enable Register)

Bit	Bit Symbol	After Reset	Type	Function
31:8	-	0	R	Read as “0”.
7	RFTEIE	0	R/W	Enable receive interrupt (INTTSSIxRX) output when receive FIFO transfer is completed. 0: Disable 1: Enable
6	-	0	R	Read as “0”.
5	RFURIE	0	R/W	Enable error interrupt (INTTSSIxERR) output when receive FIFO underrun 0: Disable 1: Enable
4	RFORIE	0	R/W	Enable error interrupt (INTTSSIxERR) output when receive FIFO overrun 0: Disable 1: Enable
3	RCMIE	0	R/W	Enable receive interrupt (INTTSSIxRX) output when receive data comparison matches 0: Disable 1: Enable
2	RFTFIE	0	R/W	Enable receive interrupt (INTTSSIxRX) output when receive FIFO threshold flag 0: Disable 1: Enable
1	RFNEIE	0	R/W	Enable receive interrupt (INTTSSIxRX) output when receive FIFO not empty 0: Disable 1: Enable
0	-	0	R	Read as “0”.

Note: Set this register when ***[TSSIxCR1]***<RXSTS> = 0.

4.2.10. *[TSSIxRFTLR]* (TSSI Receive FIFO Threshold Register)

Bit	Bit Symbol	After Reset	Type	Function
31:3	-	0	R	Read as “0”.
2:0	RFTHD[2:0]	000	R/W	Receive FIFO threshold Set the threshold for the number of receive FIFO entries. 000 to 011 When the number of entries in the receive FIFO (***[TSSIxRFLR]*** <RFLVL[2:0]>) is greater than <RFTHD[2:0]>, the receive FIFO threshold flag (***[TSSIxRSR]***<RFTF>) is set to “1”. 100～111: Reserved

Note: This register is write-protected when the receiver is enabled (***[TSSIxCR1]***<RXSTS> = 1).

4.2.11. *[TSSIxRFLR]* (TSSI Receive FIFO Entry Register)

Bit	Bit Symbol	After Reset	Type	Function
31:3	-	0	R	Read as “0”.
2:0	RFLVL[2:0]	000	R	Number of receive FIFO entries 000 to 100 Others: Reserved

4.2.12. *[TSSIxRDR0]* (TSSI Receive Data Register 0)

This is an example of ***[TSSIxRDR0]***. ***[TSSIxRDR1]*** to ***[TSSIxRDR3]*** have the same configuration.

When reading receive data, no matter which address of the receive data registers (***[TSSIxRDR0]*** to ***[TSSIxRDR3]***) are read, the data held first will be read. Therefore, for example, a continuous read from ***[TSSIxRDR0]*** and sequential read from ***[TSSIxRDR0]*** to ***[TSSIxRDR3]*** are same operation.

Bit	Bit Symbol	After Reset	Type	Function
31:0	RDAT0[31:0]	0x00000000	R	Receive data Reads receive data from receive FIFO. When the data frame size is 31 bits or less, the receive data is justified at the LSB and “0” is read from the MSB. Writing to this register is not effective.

Note 1: This register can be accessed after setting ***[TSSIxRFMR]***<RDFS[4:0]> correctly.

Note 2: When the receive DMA request is enabled, do not read this register by software.

4.2.13. *[TSSIxTCMR]* (TSSI Transmit Clock / Mode Control Register)

Bit	Bit Symbol	After Reset	Type	Function
31:10	-	0	R	Read as “0”.
9:8	TSTART[1:0]	00	R/W	Transmit start trigger selection 00: When ***[TSSIxCR1]***<TSTS> = 1 and data is prepared in the transmit FIFO, it starts immediately. (master transmit mode) 01: Cooperative operation with receiver (Note 2) (Slave transmit and receive mode) 10: TSSIxTFS input (slave transmit mode) 11: Reserved
7	-	0	R/W	Write “0”
6:4	-	0	R	Read as “0”.
3:2	TCKOUT[1:0]	00	R/W	Transmit clock output mode selection Select the input / output direction of TSSIxTCK and the gating method at the time of output. (Note 3) 00: TSSIxTCK is input, TSSIxTFS is input (slave) 01: TSSIxTCK is always clock output, TSSIxTFS is output (master) 10: TSSIxTCK is clock output during transfer, TSSIxTFS is output(master) 11: Reserved
1:0	TCKSEL[1:0]	00	R/W	Transmit clock selection Select the clock used for transmit. 00: SCLK (master) 01: TSSIxTCK (slave) 10: TSSIxRCK (slave) 11: Reserved

Note 1: For details on this register setting, refer to “Table 3.1 Operation mode settings and combinations of pins used”.

Note 2: When the transmitter is operated cooperatively

- Do not set ***[TSSIxRCMR]***<RSTART[1:0]> to “01”.
- Set ***[TSSIxRCMR]***<RCKSEL[1:0]> = ***[TSSIxTCMR]***<TCKSEL[1:0]>.

Note 3: Make port settings. For details on port settings, refer to reference manual “Input / output ports”.

Note 4: When the transmit is enabled (***[TSSIxCR1]***<TXSTS> = 1), this register is write-protected.

Note 5: When ***[TSSIxRCMR]***<RSTART[1:0]> = 01, do not rewrite during receive operation.

Note 6: When ***[TSSIxTCMR]***<TSTART[1:0]> = 01, do not rewrite ***[TSSIxRCMR]*** and ***[TSSIxRCR]*** during transmit operation.

4.2.14. *[TSSIxTFMR]* (TSSI Transmit Data Frame Control Register)

Bit	Bit Symbol	After Reset	Type	Function
31:5	-	0	R	Read as “0”.
4:0	TDFS[4:0]	00000	R/W	Transmit data frame Set the data frame size of the transmit frame. The data size is (<TDFS[4:0]> +1) bits, but setting is prohibited for 3 bits or less. In transmit and receive mode, set the same value as ***[TSSIxRFMR]***<RDFS[4:0]>. When comparing receive data (***[TSSIxRCMR]***<RSTART[1:0]> = 11), set ***[TSSIxRFMR]*** <RDFS[4:0]> to the data area size of the receive data comparison format (Note 2) and set <TDFS[4:0]> to the data frame size. 00000 to 00010: Reserved 00011: 4bits 00100: 5bits 11110: 31bits 11111: 32bits

Note1: This register is write-protected when the transmitter is enabled (***[TSSIxCR1]***<TXSTS> = 1).

Note2: For the receive data comparison format, refer to “3.2. Data frame format”.

Note3: ***[TSSIxTDRn]*** (n = 0,1,2,3) can be accessed after setting this register.

4.2.15. *[TSSIxTDMACR]* (TSSI Transmit DMA Control Register)

Bit	Bit Symbol	After Reset	Type	Function
31:1	-	0	R	Read as “0”.
0	TDMAE	0	R/W	Transmit DMA request control 0: Disable 1: Enable When there is not full in the transmit FIFO, generate a transmit DMA request (TSSIxTXDMAREQ). The type of request is a single transfer request. Note: When this register is enabled, access to ***[TSSIxTDRn]*** (n = 0, 1,2,3) may occur.

Note1: This register can be rewritten when no access by DMAC occurs.

Note2: Set ***[TSSIxTFMR]***<TDFS[4:0]> before enabling this register.

4.2.16. *[TSSIxTSR]* (TSSI Transmit Status Register)

The interrupt signals can be cleared by writing “1” to this register (writing “0” is not effective).

Bit	Bit Symbol	After Reset	Type	Function
31:6	-	0	R	Read as “0”.
5	TFUR	0	R	Transmit FIFO underrun In the slave, an underrun occurs even if there is no entry in the transmit FIFO when the transmitter is enabled. 0: No underrun has occurred 1: Underrun occurred Underrun does not occur when master transmit.
			W	0: - 1: clear
4	TFOR	0	R	Transmit FIFO overrun 0: No overrun has occurred 1: Overrun occurred
			W	0: - 1: clear
3	-	0	R	Read as “0”.
2	TFTF	1	R	Transmit FIFO threshold flag Shows the status according to the number of FIFO entries and the threshold setting. 0: ***[TSSIxTFLR]***<TFLVL[2:0]> > ***[TSSIxTFTLR]***<TFTHD[2:0]> 1: ***[TSSIxTFLR]***<TFLVL[2:0]> ≤ ***[TSSIxTFTLR]***<TFTHD[2:0]>
1	TFNF	1	R	Transmit FIFO not full 0: Transmit FIFO is full 1: Transmit FIFO has one or more empty stages
0	TBSY	0	R	Transmitter operating status 0: Waiting 1: Operating

4.2.17. *[TSSIxTIER]* (TSSI Transmit Interrupt Enable Register)

Bit	Bit Symbol	After Reset	Type	Function
31:6	-	0	R	Read as “0”.
5	TFURIE	0	R/W	Enable error interrupt (INTTSSIxERR) output by transmit FIFO underrun 0: Disable 1: Enable
4	TFORIE	0	R/W	Enable error interrupt (INTTSSIxERR) output by transmit FIFO overrun 0: Disable 1: Enable
3	-	0	R	Read as “0”.
2	TFTFIE	0	R/W	Enable transmit interrupt (INTTSSIxTX) output by transmit FIFO threshold flag 0: Disable 1: Enable
1	TFNFIE	0	R/W	Enable transmit interrupt (INTTSSIxTX) output by transmit FIFO not full 0: Disable 1: Enable
0	-	0	R	Read as “0”.

Note: Set this register when ***[TSSIxCR1]***<TXSTS> = 0.

4.2.18. *[TSSIxTFTLR]* (TSSI Transmit FIFO Threshold Register)

Bit	Bit Symbol	After Reset	Type	Function
31:3	-	0	R	Read as “0”.
2:0	TFTHD[2:0]	000	R/W	Transmit FIFO threshold Set this field to the threshold for the number of transmit FIFO entries. 000 to 011 Others: Reserved When the number of entries in the transmit FIFO (***[TSSIxTFLR]*** <TFLVL[2:0]>) is <TFTHD[2:0]> or less, the transmit FIFO threshold flag (***[TSSIxTSR]***<TFTF>) is set to “1”.

Note: This register is write-protected when the transmitter is enabled (***[TSSIxCR1]***<TXSTS> = 1).

4.2.19. *[TSSIxTFLR]* (TSSI Transmit FIFO Entry Register)

Bit	Bit Symbol	After Reset	Type	Function
31:3	-	0	R	Read as “0”.
2:0	TFLVL[2:0]	000	R	Number of transmit FIFO entries 000 to 100 Others: Reserved

4.2.20. *[TSSIxTDR0]* (TSSI Transmit Data Register 0)

This is an example of ***[TSSIxTDR0]. [TSSIxTDR1]*** to ***[TSSIxTDR3]*** have the same configuration.

When writing transmit data, no matter which address of the transmit data registers (***[TSSIxTDR0]*** to ***[TSSIxTDR3]***) is written, the data will be stored at the end of the FIFO. Therefore, for example, continuous writing to ***[TSSIxTDR0]*** and sequential writing from ***[TSSIxTDR0]*** to ***[TSSIxTDR3]*** are same operation.

Bit	Bit Symbol	After Reset	Type	Function
31:0	TDAT0[31:0]	0x00000000	W	Transmit data Write transmit data to the transmit FIFO. When the data frame size is less than 32 bits, justify it to the LSB side. Read as “0x00000000”.

Note 1: This register can be accessed after setting ***[TSSIxTFMR]***<TDFS[4:0]> correctly.

Note 2: When the transmit DMA request is enabled, do not write to this register by software.

5. Example of usages

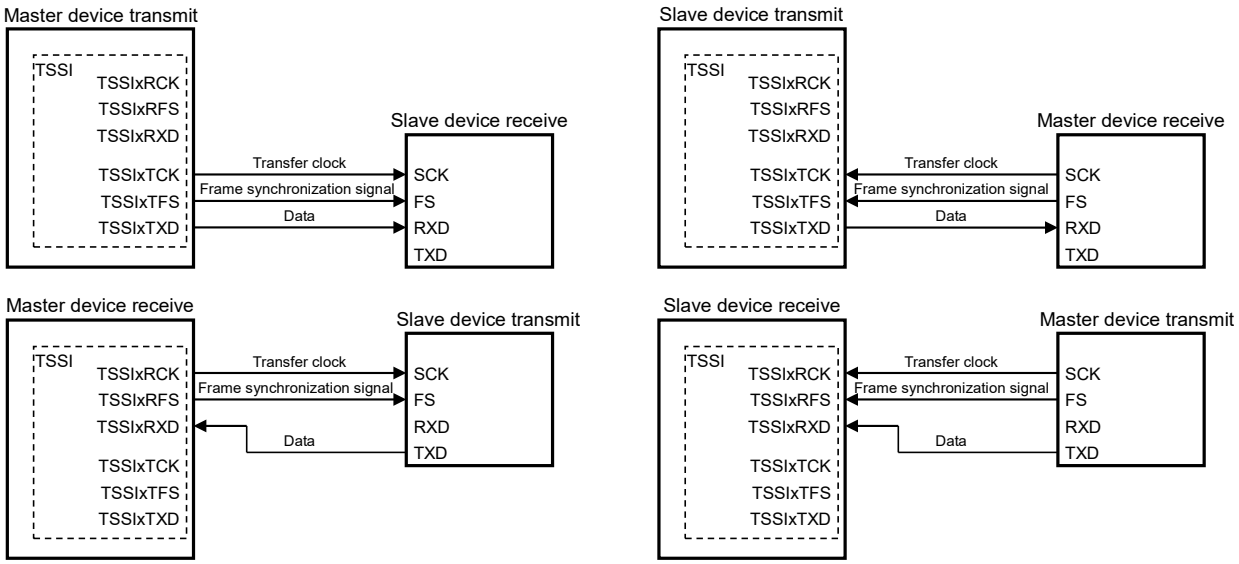


Figure 5.1 Connection example of transmit, receive

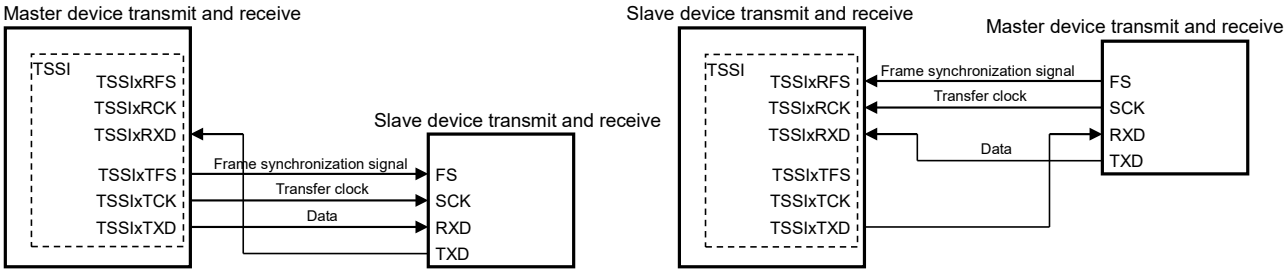


Figure 5.2 Connection example of transmit and receive

5.1. Master transmit

Table 5.1 shows the procedure when using the TSSI for master transmit.

Table 5.1 Setup procedure for master transmit

Step	Operation	Description	Note
1	Ceck the disable of transmitter	If either ***[TSSIxCR1]***<TXSTS> or ***[TSSIxTSR]***<TBSY> is “1”, go to Step-1a. Otherwise, go to Step-2.	-
1a	Processing of disable at transmitter	If ***[TSSIxCR1]***<TXSTS> = 1, write ***[TSSIxCR1]***<TXEN[1:0]> = 10,11. Wait until ***[TSSIxCR1]***<TXSTS> = 0 and ***[TSSIxTSR]***<TBSY> = 0. When “11” is written to ***[TSSIxCR1]***<TXEN[1:0]>, perform a software reset on the transmitter.	-
2	Check and clear error status	If any bit of ***[TSSIxTSR]*** is read as “1”, perform error processing as necessary. When error handling is not required, write “0xFFFFFFFF” to this register to clear the error flag.	(Note1)

3	In the master Transmit settings	Please set registers as follows. Set ***[TSSIxCPR]***<DIV[7:0]> to dividing value ***[TSSIxTCMR]***<TCKSEL[1:0]> = 00 (Bit clock is SCLK) ***[TSSIxTCMR]***<TCKOUT[1:0]> = 01,10 (TSSIxTCK, TSSIxTFS output) ***[TSSIxTCMR]***<TSTART[1:0]> = 00 (Transfer starts when ready) Set ***[TSSIxTFMR]***<TDFS[4:0]> to data frame size Set ***[TSSIxTFTLR]*** <TFTHD[2:0]> in the transmit FIFO threshold value if necessary Set ***[TSSIxTIER]*** with interrupt notification if necessary ***[TSSIxTDMACR]***<TDMAE>=0	-
4	Processing of enable at transmitter	Write “01” to ***[TSSIxCR1]***<TXEN[1:0]> and wait for ***[TSSIxCR1]*** <TXSTS> = 1.	-
5	Transmit processing	Write to ***[TSSIxTDR0]*** for the required number of data frames, checking that the transmit FIFO has stages. Check the stages of the transmit FIFO by using ***[TSSIxTSR]***<TFTF>, <TFNF> or the corresponding interrupt. When that data is stored in the transmit FIFO, the TSSI outputs communication data to the outside in a sequential order.	(Note2)
6	Check transfer completion	Check that ***[TSSIxTSR]***<TBSY> = 0.	-
7	Transmit processing (repetition)	To perform the transmit process again, repeat Steps 5 and 6.	(Note2)

Note1: When an error has occurred, a software reset is required for the corresponding function block.

Note2: In master transmit, transmit processing can be suspended. When “10” is written to ***[TSSIxCR1]*** <TXEN[1:0]>, the transmitter stops after transmit current processing data frame. To restart, write “01” to ***[TSSIxCR1]***<TXEN[1:0]>.

5.2. Master receive

Table 5.2 shows the procedure when using the TSSI for master receive.

Table 5.2 Setup procedure for master receive

Step	Operation	Description	Note
1	Check the disable of receiver	If either ***[TSSIxCR1]***<RXSTS> or ***[TSSIxRSR]***<RBSY> is "1", go to Step-1a. Otherwise, go to Step-2.	-
1a	Processing of disable at receiver	If ***[TSSIxCR1]***<RXSTS> = 1, write ***[TSSIxCR1]***<RXEN[1:0]> = 10,11. Wait until ***[TSSIxCR1]***<RXSTS> = 0 and ***[TSSIxRSR]***<RBSY> = 0. When "11" is written to ***[TSSIxCR1]***<RXEN[1:0]>, perform a software reset on the receiver.	-
2	Check and clear error status	If any bit of ***[TSSIxRSR]*** is read as "1", perform error processing as necessary. When error handling is not required, write "0xFFFFFFFF" to this register to clear the error flag.	(Note1)
3	In the master receive settings	Please set registers as follows. Set ***[TSSIxCPR]***<DIV[7:0]> to dividing value ***[TSSIxRCMR]***<RCKSEL[1:0]> = 00 (Bit clock is SCLK) ***[TSSIxRCMR]***<RCKOUT[1:0]> = 01,10 (TSSIxRCK, TSSIxRFS output) ***[TSSIxRCMR]***<RSTART[1:0]> = 00 (Transfer starts when ready) Set ***[TSSIxRFMR]***<RDFS[4:0]> to data frame size Set ***[TSSIxRFMR]***<RDFC[3:0]> in the number of data frames Set ***[TSSIxRFTLR]***<RFTHD[2:0]> in the receive FIFO threshold if necessary Set ***[TSSIxRIER]*** with interrupt notification if necessary ***[TSSIxRDMACR]***<RDMAE>=0	-
4	Processing of enable at receiver	Write "01" to ***[TSSIxCR1]***<RXEN[1:0]> and wait for ***[TSSIxCR1]*** <RXSTS> = 1.	-
5	Receive processing	TSSI communicates with the outside and receive data when the receive FIFO has empty stages and the specified number of receive data frames has not been reached. The software reads from ***[TSSIxRDR0]*** for the required number of data frames, checking that there is an entry in the receive FIFO. Check the status of the receive FIFO entry using ***[TSSIxRSR]*** <RFTF>, <RFNE> or the corresponding interrupt.	(Note2)
6	Receive end processing	When ***[TSSIxRFMR]***<RDFC[3:0]> = 0xF is set, the receiver completes the receive process by writing ***[TSSIxCR1]***<RXEN[1:0]> = 10. When ***[TSSIxRFMR]***<RDFC[3:0]> ≠ 0xF, the receiver stops after storing the specified number of data frames to the receive FIFO.	-
7	Check receive completion	Check that ***[TSSIxRSR]***<RBSY> = 0.	-
8	Receive processing (repetition)	To perform the receiving process again, repeat Steps 4, 5, 6, and 7.	-

Note 1: When an error has occurred, a software reset is required for the corresponding function block.

Note 2: In master receive, receive processing can be suspended. When "10" is written to ***[TSSIxCR1]*** <RXEN[1:0]>, the receiver stops after receive current processing data frame. To restart, write "01" to ***[TSSIxCR1]***<RXEN[1:0]>.

5.3. Master transmit and receive

Table 5.3 shows the procedure when using the TSSI for master transmit and receive.

Table 5.3 Setup procedure for master transmit and receive

Step	Operation	Description	Note
1	Check the disable at transmitter and receiver	If any of ***[TSSIxCR1]***<TXSTS>, ***[TSSIxTSR]***<TBSY>, ***[TSSIxCR1]*** <RXSTS>, ***[TSSIxRSR]***<RBSY> is “1”, go to Step-1a. Otherwise, go to Step-2.	-
1a	Processing of disable at transmit and receive	If ***[TSSIxCR1]***<TXSTS> = 1, write ***[TSSIxCR1]***<TXEN[1:0]> = 10,11. Wait until ***[TSSIxCR1]***<TXSTS> = 0 and ***[TSSIxTSR]***<TBSY> = 0. If “11” is written to ***[TSSIxCR1]***<TXEN[1:0]>, perform a software reset on the transmitter. When ***[TSSIxCR1]***<RXSTS> = 1, write ***[TSSIxCR1]***<RXEN[1:0]> = 10,11. Wait until ***[TSSIxCR1]***<RXSTS> = 0 and ***[TSSIxRSR]***<RBSY> = 0. When “11” is written to ***[TSSIxCR1]***<RXEN[1:0]>, perform a software reset on the receiver.	-
2	Check and clear error status	Read ***[TSSIxTSR]*** and ***[TSSIxRSR]***, and if there is a bit that becomes "1", perform error processing as necessary. When error processing is not required, write "0xFFFFFFFF" to these registers to clear the error flag.	(Note)
3	In the master transmit and receive settings	Please set registers as follows. Set ***[TSSIxCPR]***<DIV[7:0]> to dividing value ***[TSSIxTCMR]***<TCKSEL[1:0]> = 00 (Bit clock is SCLK) ***[TSSIxRCMR]***<RCKSEL[1:0]> = 00 (Bit clock is SCLK) ***[TSSIxTCMR]***<TCKOUT[1:0]> = 01,10 (TSSIxTCK, TSSIxTFS output) ***[TSSIxRCMR]***<RCKOUT[1:0]> = 00 (TSSIxRCK, TSSIxRFS input: not used) ***[TSSIxTCMR]***<TSTART[1:0]> = 00 (Transfer starts when the transmitter is ready) ***[TSSIxRCMR]***<RSTART[1:0]> = 01 (Use trigger of transmitter) Set ***[TSSIxTFMR]***<TDFS[4:0]> to data frame size Set ***[TSSIxRFMR]*** <RDFS[4:0]> in the same value as ***[TSSIxTFMR]***<TDFS[4:0]> ***[TSSIxRFMR]***<RDFC[3:0]> = 0xF (no specification of the number of receive data frames) Set ***[TSSIxTFCR]***<TFTHD[2:0]> in the transmit FIFO threshold if necessary. Set ***[TSSIxRFCR]*** <RFTHD[2:0]> in the same value as ***[TSSIxTFCR]***<TFTHD[2:0]> if necessary. Set ***[TSSIxTIER]*** with interrupt notification if necessary. Set ***[TSSIxRIER]*** with interrupt notification if necessary. ***[TSSIxTDMACR]***<TDMAE>=0 ***[TSSIxRDMACR]***<RDMAE>=0	-
4	Enable of transmitter and receiver	Write "01" to ***[TSSIxCR1]***<TXEN[1:0]> and <RXEN[1:0]> each. After that, check that both ***[TSSIxCR1]***<TXSTS> and <RXSTS> are "1".	-
5	Processing of transmit and receiver	Write to ***[TSSIxTDR0]*** for the required number of data frames, checking that the transmit FIFO has stages. Check the stages of the transmit FIFO by using ***[TSSIxTSR]*** <TFTF>, <TFNF> or the corresponding interrupt. When that data is stored in the transmit FIFO, the TSSI outputs communication data to the outside in a sequential order. In transmit and receive mode, the TSSI receives and transmits communication data at the same time. The software reads from ***[TSSIxRDR0]*** for the required number of data frames, checking that there is an entry in the receive FIFO. Check the status of the receive FIFO entry using ***[TSSIxRSR]***<RFTF>, <RFNE> or the corresponding interrupt.	-

6	Check the transmit and receive completion	Check that ***[TSSIxTSR]***<TBSY> = 0 and ***[TSSIxRSR]***<RBSY> = 0.	-
7	Transmit and receive processing (repetition)	To perform transmit and receive processing again, repeat Steps 5 and 6.	-

Note: When an error has occurred, a software reset is required for the corresponding function block.

5.4. Slave transmit

Table 5.4 shows the procedure when using the TSSI for slave transmit.

Table 5.4 Setup procedure for slave transmit

Step	Operation	Description	Note
1	Check the disable at transmitter	If either ***[TSSIxCR1]***<TXSTS> or ***[TSSIxTSR]***<TBSY> is "1", go to Step-1a. Otherwise, go to Step-2.	-
1a	Processing of disable at transmitter	Wait for ***[TSSIxTSR]***<TBSY> = 0. When ***[TSSIxCR1]***<TXSTS> = 1, write ***[TSSIxCR1]***<TXEN[1:0]> = 10,11. Wait until ***[TSSIxCR1]***<TXSTS> = 0. When "11" is written to ***[TSSIxCR1]***<TXEN[1:0]>, perform a software reset on the transmitter.	(Note1)
2	Check and clear error status	If any bit of ***[TSSIxTSR]*** is read as "1", perform error processing as necessary. When error handling is not required, write "0xFFFFFFFF" to this register to clear the error flag.	(Note2)
3	In the slave transmit settings	Please set registers as follows. ***[TSSIxTCMR]***<TCKSEL[1:0]> = 01 (Bit clock input is TSSIxTCK) ***[TSSIxTCMR]***<TCKOUT[1:0]> = 00 (TSSIxTCK, TSSIxTFS input) ***[TSSIxTCMR]***<TSTART[1:0]> = 10 (Transfer starts when TSSIxTFS is detected) Set ***[TSSIxTFMR]***<TDFS[4:0]> to data frame size Set ***[TSSIxTFCR]***<TFTHD[2:0]> in the transmit FIFO threshold if necessary. Set ***[TSSIxTIER]*** with the interrupt if necessary. ***[TSSIxTDMACR]***<TDMAE> = 0	-
4	Processing of enable at transmitter	Write "01" to ***[TSSIxCR1]***<TXEN[1:0]> and wait for ***[TSSIxCR1]*** <TXSTS> = 1.	(Note1)
5	Transmit processing	Write to ***[TSSIxTDR0]*** for the required number of data frames, checking that the transmit FIFO has stages. Check the stages of the transmit FIFO by using ***[TSSIxTSR]*** <TFTF>, <TFNF> or the corresponding interrupt.	(Note3)
6	Check the tranmit completion	If necessary, check that ***[TSSIxSR]***<TBSY> = 0 after writing data of the required number of data frames to the transmit FIFO.	-
7	Transmit processing (repetition)	To perform the transmit request from the external master repeatedly, repeat Steps 5 and 6.	(Note3)

Note 1: Enable / disable the function on the slave when the external master is stopped.

Note 2: When an error has occurred, a software reset is required for the corresponding function block.

Note 3: The slave transmitter transmits the transmit FIFO data according to a communication request from the external master. Therefore, when the transmit FIFO is not ready for the communication request of the external master, an underrun error will occur.

5.5. Slave receive

Table 5.5 shows the procedure when using the TSSI for slave receive.

Table 5.5 Setup procedure for slave receive

Step	Operation	Description	Note
1	Check the disable at receiver	If either ***[TSSIxCR1]***<RXSTS> or ***[TSSIxRSR]***<RBSY> is “1”, go to Step-1a. Otherwise, go to Step-2.	-
1a	Processing of disable at receiver	Wait for ***[TSSIxRSR]***<RBSY> = 0. When ***[TSSIxCR1]***<RXSTS> = 1, write ***[TSSIxCR1]***<RXEN[1:0]> = 10,11. Wait until ***[TSSIxCR1]***<RXSTS> = 0. When “11” is written to ***[TSSIxCR1]***<RXEN[1:0]>, perform a software reset on the receiver.	(Note1)
2	Check and clear error status	If any bit of ***[TSSIxRSR]*** is read as “1”, perform error processing as necessary. When error handling is not required, write "0xFFFFFFFF" to this register to clear the error flag.	(Note2)
3	In the slave receive settings	Please set registers as follows. ***[TSSIxRCMR]***<RCKSEL[1:0]> = 10 (Bit clock is TSSIxRCK input) ***[TSSIxRCMR]***<RCKOUT[1:0]> = 00 (TSSIxRCK, TSSIxRFS input) ***[TSSIxRCMR]***<RSTART[1:0]> = 10 (Transfer starts when TSSIxRFS is detected) Set ***[TSSIxRFMR]***<RDFS[4:0]> to data frame size Set ***[TSSIxRFCR]***<RFTHD[2:0]> in the receive FIFO threshold if necessary. Set ***[TSSIxRIER]*** with interrupt notification if necessary. ***[TSSIxRDMACR]***<RDMAE> = 0	-
4	Processing of enable at receiver	Write "01" to ***[TSSIxCR1]***<RXEN[1:0]> and wait for ***[TSSIxCR1]*** <RXSTS> = 1.	(Note1)
5	Receive processing	The software reads from ***[TSSIxRDR0]*** for the required number of data frames, checking that there is an entry in the receive FIFO. Check the status of the receive FIFO entry using ***[TSSIxRSR]*** <RFTF>, <RFNE> or the corresponding interrupt.	(Note3)
6	Check the receive completion	If necessary, read the required number of data frames from the receive FIFO, and then check that ***[TSSIxRSR]***<RBSY> = 0.	-
7	Receive processing (repetition)	To perform the receive request from the external master repeatedly, repeat Steps 5 and 6.	(Note3)

Note 1: Enable / disable the function on the slave when the external master is stopped.

Note 2: When an error has occurred, a software reset is required for the corresponding function block.

Note 3: The slave receiver stores data in the receive FIFO according to a communication request from the external master. Therefore, when the receive FIFO is not ready for the communication request from the external master, an overrun error will occur.

5.6. Slave transmit and receive

Table 5.6 shows the procedure when using the TSSI for slave transmit and receive.

Table 5.6 Setup procedure for slave transmit and receive

Step	Operation	Description	Note
1	Check the disable at transmitter and receiver	If any of ***[TSSIxCR1]***<TXSTS>, ***[TSSIxTSR]***<TBSY>, ***[TSSIxCR1]*** <RXSTS>, ***[TSSIxRSR]***<RBSY> is “1”, go to Step-1a. Otherwise, go to Step-2.	-
1a	Processing of disable at transmit and receive	Wait for ***[TSSIxTSR]***<TBSY> = 0. When ***[TSSIxCR1]***<TXSTS> = 1, write ***[TSSIxCR1]***<TXEN[1:0]> = 10,11. Wait until ***[TSSIxCR1]***<TXSTS> = 0. When “11” is written to ***[TSSIxCR1]***<TXEN[1:0]>, perform a software reset on the transmitter. Wait for ***[TSSIxRSR]***<RBSY> = 0. When ***[TSSIxCR1]***<RXSTS> = 1, write ***[TSSIxCR1]***<RXEN[1:0]> = 10,11. Wait until ***[TSSIxCR1]***<RXSTS> = 0. When “11” is written to ***[TSSIxCR1]***<RXEN[1:0]>, perform a software reset on the receiver.	(Note1)
2	Check and clear error status	Read ***[TSSIxTSR]*** and ***[TSSIxRSR]***, and when there is a bit that is set to “1”, perform error processing as necessary. When error handling is not required, write “0xFFFFFFFF” to this register to clear the error flag.	(Note2)
3	In the slave transmit and receive settings	Please set registers as follows. ***[TSSIxTCMR]***<TCKSEL[1:0]> = 10 (Bit clock is TSSIxRCK input) ***[TSSIxRCMR]***<RCKSEL[1:0]> = 10 (Bit clock is TSSIxRCK input) ***[TSSIxTCMR]***<TCKOUT[1:0]> = 00 (TSSIxTCK, TSSIxTFS input: not used) ***[TSSIxRCMR]***<RCKOUT[1:0]> = 00 (TSSIxRCK, TSSIxRFS input) ***[TSSIxTCMR]***<TSTART[1:0]> = 01 (Use trigger of receiver) ***[TSSIxRCMR]***<RSTART[1:0]> = 10 (Transfer starts when TSSIxRFS is detected) Set ***[TSSIxTFMR]***<TDFS[4:0]> to data frame size Set ***[TSSIxRFMR]*** <RDFS[4:0]> in the same value as ***[TSSIxTFMR]***<TDFS[4:0]> Set ***[TSSIxTFCR]***<TFTHD[2:0]> in the transmit FIFO threshold if necessary. Set ***[TSSIxRFCR]*** <RFTHD[2:0]> in the same value as ***[TSSIxTFCR]***<TFTHD[2:0]> if necessary. Set ***[TSSIxTIER]*** with interrupt notification if necessary. Set ***[TSSIxRIER]*** with interrupt notification if necessary. ***[TSSIxTDMACR]***<TDMAE> = 0 ***[TSSIxRDMACR]***<RDMAE> = 0	-
4	Enable of transmitter and receiver	Write “01” to ***[TSSIxCR1]***<TXEN[1:0]> and <RXEN[1:0]>, and wait for each of ***[TSSIxCR1]***<TXSTS> and <RXSTS> to become “1”.	(Note1)
5	Processing of transmit and receiver	Write to ***[TSSIxTDR0]*** for the required number of data frames, checking that the transmit FIFO has stages. Check the stages of the transmit FIFO by using ***[TSSIxTSR]*** <TFTF>, <TFNF> or the corresponding interrupt. In transmit and receive mode, the TSSI receives and transmits communication data at the same time. The software reads from ***[TSSIxRDR0]*** for the required number of data frames, checking that there is an entry in the receive FIFO. Check the existence of the receive FIFO entry using ***[TSSIxRSR]***<RFTF>, <RFNE> or the corresponding interrupt.	(Note3)

6	Check the transmit and receive completion	If necessary, after writing data of the required number of data frames to the transmit FIFO and reading data of the required number of data frames from the receive FIFO, check that both ***[TSSIxTSR]***<TBSY> and ***[TSSIxRSR]*** <RBSY> are "0".	-
7	Transmit and receive processing (repetition)	To perform the transmit and receive request from the external master repeatedly, repeat Steps 5 and 6.	(Note3)

Note 1: Enable / disable the function on the slave when the external master is stopped.

Note 2: When an error has occurred, a software reset is required for the corresponding function block.

Note 3: The slave transmitter and the receiver transmits the data of the transmit FIFO and store the receive data in the receive FIFO according to the communication request from the external master. Therefore, when the FIFO is not ready for the communication request of the external master, a transmit FIFO underrun or receive FIFO overrun error will occur.

5.7. Transfer procedure by using DMAC

The difference from the procedure shown in chapters 5.1 to 5.6 is shown below.

(1) Step-1 to Step-4 of each procedure, the setup procedure are basically the same as when using a CPU. However FIFO not full / not empty interrupts are not used.
(2) Set the transfer settings on the DMAC as Step-5.
(3) As Step-6, write “1” to ***[TSSIxTDMACR]***<TDMAE> in transmit mode, write “1” to ***[TSSIxRDMACR]*** <RDMAE> in receive mode, and both in transmit and receive mode.
(4) As Step-7, there is the judgment of transfer completion. In receive mode or transmit and receive mode, the completion of entire transfer is checked by DMAC transfer completion. In the transmit mode, check the transfer completion on the TSSI (***[TSSIxTSR]***<TBSY>) after checking the DMAC transfer completion.

Note1: The TSSI generates a single transfer request to the DMAC according to the empty stages of the transmit FIFO and the receive FIFO by setting ***[TSSIxTDMACR]***<TDMAE> = 1 or ***[TSSIxRDMACR]***<RDMAE> = 1. This DMA request does not depend on the function status of the transmitter and receiver (***[TSSIxCR1]*** <TXSTS>, <RXSTS>). Even when each function block is disabled, the DMAC can access the FIFO by setting ***[TSSIxTDMACR]***<TDMAE> or ***[TSSIxRDMACR]***<RDMAE>.

Note2: In the case of the master, communication can be suspended setting ***[TSSIxCR1]***<TXEN[1:0]>, <RXEN[1:0]>.

Note3: In the case of a slave, enable / disable the function block by ***[TSSIxCR1]***<TXEN[1:0]>, <RXEN[1:0]> when the external master is not communicating.

5.8. Receive data comparison function in slave receive or transmit and receive

In the slave receive mode, the receive data comparison function can be used by setting ***[TSSIxRCMR]*** <RSTART[1:0]> = 11 (use the receive data comparison function).
In the slave transmit and receive mode, the receive data comparison function can be used by setting ***[TSSIxTCMR]***<TSTART[1:0]> = 01 (use the trigger of the receiver) and ***[TSSIxRCMR]***<RSTART[1:0]> = 11 (use receive data comparison function).

The receive data comparison function receives data after detecting assertion of the TSSIxRFS input. The receive data of m (= ***[TSSIxRFMR]***<CMPDS[3:0]> +1) bits is not stored in the receive FIFO, but is compared with [m-1: 0] of the receive data comparison register ***[TSSIxRCR]***<CMPDP[15:0]>. When the comparison results match, the subsequent data is treated as a data area.

When there is a mismatch, the TSSI ignores subsequent data. When the TSSIxRFS input is asserted again, the next receive data is compared.

6. Precautions and requests for use

- Do not access addresses to which no registers have been assigned.

7. Revision History

Table 7.1 Revision History

Revision	Date	Description
1.0	2020-11-16	First release

RESTRICTIONS ON PRODUCT USE

Toshiba Corporation and its subsidiaries and affiliates are collectively referred to as “TOSHIBA”.
Hardware, software and systems described in this document are collectively referred to as “Product”.

- TOSHIBA reserves the right to make changes to the information in this document and related Product without notice.
- This document and any information herein may not be reproduced without prior written permission from TOSHIBA. Even with TOSHIBA's written permission, reproduction is permissible only if reproduction is without alteration/omission.
- Though TOSHIBA works continually to improve Product's quality and reliability, Product can malfunction or fail. Customers are responsible for complying with safety standards and for providing adequate designs and safeguards for their hardware, software and systems which minimize risk and avoid situations in which a malfunction or failure of Product could cause loss of human life, bodily injury or damage to property, including data loss or corruption. Before customers use the Product, create designs including the Product, or incorporate the Product into their own applications, customers must also refer to and comply with (a) the latest versions of all relevant TOSHIBA information, including without limitation, this document, the specifications, the data sheets and application notes for Product and the precautions and conditions set forth in the "TOSHIBA Semiconductor Reliability Handbook" and (b) the instructions for the application with which the Product will be used with or for. Customers are solely responsible for all aspects of their own product design or applications, including but not limited to (a) determining the appropriateness of the use of this Product in such design or applications; (b) evaluating and determining the applicability of any information contained in this document, or in charts, diagrams, programs, algorithms, sample application circuits, or any other referenced documents; and (c) validating all operating parameters for such designs and applications. **TOSHIBA ASSUMES NO LIABILITY FOR CUSTOMERS' PRODUCT DESIGN OR APPLICATIONS.**
- **PRODUCT IS NEITHER INTENDED NOR WARRANTED FOR USE IN EQUIPMENTS OR SYSTEMS THAT REQUIRE EXTRAORDINARILY HIGH LEVELS OF QUALITY AND/OR RELIABILITY, AND/OR A MALFUNCTION OR FAILURE OF WHICH MAY CAUSE LOSS OF HUMAN LIFE, BODILY INJURY, SERIOUS PROPERTY DAMAGE AND/OR SERIOUS PUBLIC IMPACT ("UNINTENDED USE").** Except for specific applications as expressly stated in this document, Unintended Use includes, without limitation, equipment used in nuclear facilities, equipment used in the aerospace industry, lifesaving and/or life supporting medical equipment, equipment used for automobiles, trains, ships and other transportation, traffic signaling equipment, equipment used to control combustions or explosions, safety devices, elevators and escalators, and devices related to power plant. **IF YOU USE PRODUCT FOR UNINTENDED USE, TOSHIBA ASSUMES NO LIABILITY FOR PRODUCT.** For details, please contact your TOSHIBA sales representative or contact us via our website.
- Do not disassemble, analyze, reverse-engineer, alter, modify, translate or copy Product, whether in whole or in part.
- Product shall not be used for or incorporated into any products or systems whose manufacture, use, or sale is prohibited under any applicable laws or regulations.
- The information contained herein is presented only as guidance for Product use. No responsibility is assumed by TOSHIBA for any infringement of patents or any other intellectual property rights of third parties that may result from the use of Product. No license to any intellectual property right is granted by this document, whether express or implied, by estoppel or otherwise.
- **ABSENT A WRITTEN SIGNED AGREEMENT, EXCEPT AS PROVIDED IN THE RELEVANT TERMS AND CONDITIONS OF SALE FOR PRODUCT, AND TO THE MAXIMUM EXTENT ALLOWABLE BY LAW, TOSHIBA (1) ASSUMES NO LIABILITY WHATSOEVER, INCLUDING WITHOUT LIMITATION, INDIRECT, CONSEQUENTIAL, SPECIAL, OR INCIDENTAL DAMAGES OR LOSS, INCLUDING WITHOUT LIMITATION, LOSS OF PROFITS, LOSS OF OPPORTUNITIES, BUSINESS INTERRUPTION AND LOSS OF DATA, AND (2) DISCLAIMS ANY AND ALL EXPRESS OR IMPLIED WARRANTIES AND CONDITIONS RELATED TO SALE, USE OF PRODUCT, OR INFORMATION, INCLUDING WARRANTIES OR CONDITIONS OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, ACCURACY OF INFORMATION, OR NONINFRINGEMENT.**
- Do not use or otherwise make available Product or related software or technology for any military purposes, including without limitation, for the design, development, use, stockpiling or manufacturing of nuclear, chemical, or biological weapons or missile technology products (mass destruction weapons). Product and related software and technology may be controlled under the applicable export laws and regulations including, without limitation, the Japanese Foreign Exchange and Foreign Trade Law and the U.S. Export Administration Regulations. Export and re-export of Product or related software or technology are strictly prohibited except in compliance with all applicable export laws and regulations.
- Please contact your TOSHIBA sales representative for details as to environmental matters such as the RoHS compatibility of Product. Please use Product in compliance with all applicable laws and regulations that regulate the inclusion or use of controlled substances, including without limitation, the EU RoHS Directive. **TOSHIBA ASSUMES NO LIABILITY FOR DAMAGES OR LOSSES OCCURRING AS A RESULT OF NONCOMPLIANCE WITH APPLICABLE LAWS AND REGULATIONS.**