Chapter 14 Serial Peripheral Interface (SPI)

SPI supports data interaction in a 3-wire synchronous serial mode, plus a chip selector line to support hardware switching between Master and Slave modes, and supports communication on a single data line.

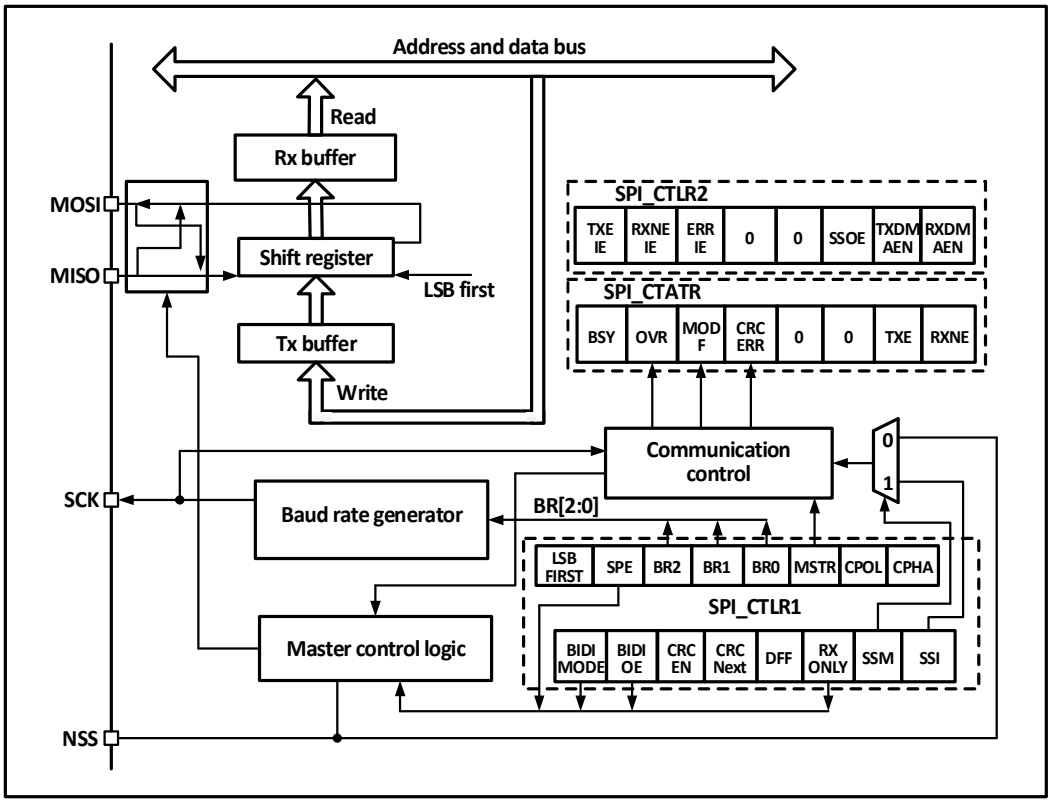
14.1 Main Features

- Support full-duplex synchronous serial mode
- Support single-line half-duplex mode
- Support Master mode and Slave mode, Multi-slave mode
- Support 8-bit or 16-bit data structures
- Maximum clock frequency supports up to half of F_{HCLK}
- Data order supports MSB or LSB first
- Support hardware or software control of NSS pins
- Hardware CRC checksum support for sending and receiving
- Transceiver buffers support DMA transfers
- Support modification of clock phase and polarity

14.2 Function Description

14.2.1 Overview

Figure 14-1 SPI structure block diagram



As can be seen from Figure 14-1, the four main SPI-related pins are MISO, M0SI, SCK and NSS. The MISO pin is the data input pin when the SPI module is operating in Master mode and the data output pin when it is operating in Slave mode. the MOSI pin is the data output pin when it is operating in Master mode and the data input pin when it is operating in Slave mode. the SCK is the clock pin, the clock signal is always output by the host and the slave receives the clock signal and synchronizes the data sending and receiving. the NSS pin is the chip select pin with the following usage.

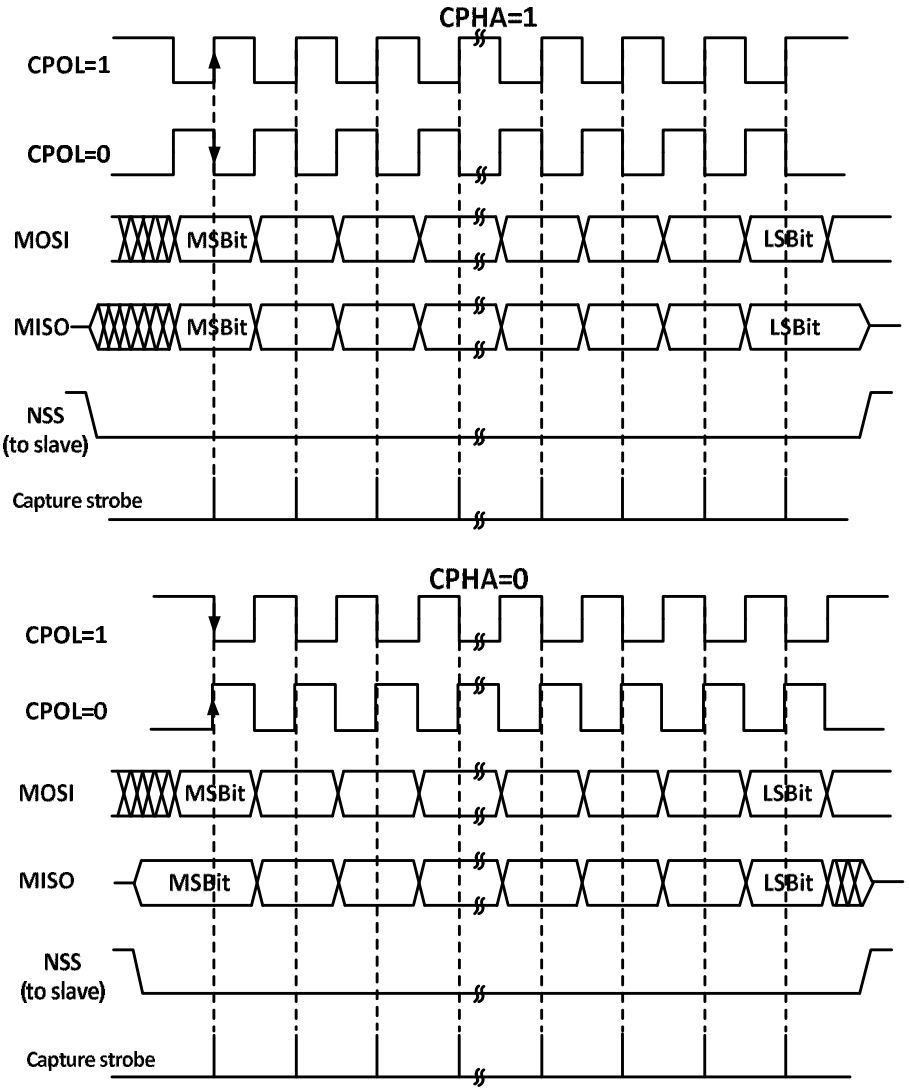
1) NSS controlled by software: When SSM is set and the internal NSS signal is output high or low as

determined by SSI, this case is generally used in SPI Master mode.

2) NSS is controlled by hardware: When the NSS output is enabled, i.e., when SSOE is set, the NSS pin will be actively pulled down when the SPI host sends outputs outward, and if it does not succeed in pulling down the NSS pin, which indicates that there is another master device on the main line that is communicating, a hardware error will be generated; SSOE is not set, it can be used in multi-master mode, and if it is pulled down it will be forced to enter the slave mode, and the MSTR bit will be cleared automatically.

CPHA is set to indicate that the module samples data on the second edge of the clock and the data is latched, while CPHA is not set to indicate that the SPI module samples data on the first edge of the clock and the data is latched, and CPOL indicates whether the clock is held high or low when there is no data. See Figure 14-2 below for details.

Figure 14-2 SPI Mode



The host and device need to be set to the same SPI mode, and the SPE bit needs to be cleared before configuring the SPI mode. the DEF bit determines whether the individual data length of the SP is 8 bits or 16 bits. LSBFIRST controls whether a single data word is preceded by the high bit or the low bit.

14.2.2 Master Mode

The serial clock is generated by SCK when the SPI module is operating in master mode. The following steps are performed to configure into master mode.

Configure the BR[2:0] field of the control register to determine the clock.
Configure the CPOL and CPHA bits to determine the SPI mode.
Configure DEF to determine the data word length.
Configure LSBFIRST to determine the frame format;
Configure the NSS pin, for example by setting the SSOE bit and letting the hardware set the NSS. it is also possible to set the SSM bit and set the SSI bit high.
To set the MSTR bit and the SPE bit, you need to make sure that the NSS is already high at this time.
When you need to transmit data you just need to write the data to be transmitted to the data register. SPI will transmit the data from the transmit buffer to the shift register in parallel and then transmit the data from the shift register according to the setting of LSBFIRST. When the data is already in the shift register, the TXE flag will be set. If the TXEIE is already set, then an interrupt will be generated. If the TXE flag position bit needs to be filled with data into the data register to maintain the complete data flow.
When the receiver receives data, when the last sample clock edge of the data word comes, the data is transferred from the shift register to the receive buffer in parallel, the RXNE bit is set, and an interrupt is generated if the RXNEIE bit was previously set. At this time, the data register should be read as soon as possible to take away the data.

14.2.3 Slave Mode

When the SPI module is operating in slave mode, SCK is used to receive the clock from the host and its own baud rate setting is invalid. To configure into slave mode, proceed as follows.
Configure the DEF bit to set the data bit length.
Configure the CPOL and CPHA bits to match the host mode.
Configure LSBFIRST to match the host data frame format;
The NSS pin needs to be held low in hardware management mode, if NSS is set to software management (SSM set), then keep SSI unset.
Clear the MSTR bit and set the SPE bit to enable SPI mode. In transmitting, when the first slave receive sample edge appears in SCK, the slave starts to transmit. The process of sending is to move the data in the transmit buffer t the transmit shift register. When the data in the transmit buffer is moved to the shift register, the TXE flag will be set, and if the TXEIE bit was set before, then an interrupt will be generated.
During reception, after the last clock sample edge, the RXNE bit is set, the bytes received by the shift register are transferred to the receive buffer, and the read operation of the read data register can obtain the data in the receive buffer. If RXNEIE is set before RXNE is set, then an interrupt is generated.

14.2.4 Simplex Mode

The SPI interface can operate in half-duplex mode, where the master device uses the MOSI pin and the slave device uses the MISO pin for communication. When using half-duplex communication, you need to set BIDIMODE and use BIDIOE to control the transmission direction.
Setting the RXONLY bit in normal full-duplex mode sets the SPI module to receive-only simplex mode, releasing a data pin after RXONLY is set. The SPI can also be set to transmit only mode by ignoring the received data.

14.2.5 CRC

The SPI module uses CRC checksum to ensure the reliability of full-duplex communication, and separate CRC calculators are used for data sending and receiving. the polynomial for CRC calculation is determined by the polynomial register, and different calculations are used for 8-bit data width and 16-bit data width, respectively. Setting the CRCEN bit will enable CRC checksum and at the same time will reset the CRC calculator. After the last data byte is sent, setting the CRCNEXT bit will send the TXCRCR calculator calculation after the current byte is sent, while the CRCERR bit will be set if the last received receive shift register value does not match the locally calculated RXCRCR calculation. Using the CRC checksum requires setting the polynomial calculator and setting the CRCEN bit when configuring the SPI operating mode, and setting the CRCNEXT bit on the last word or half-word to send the CRC and perform the receive CRC checksum. Note that the

polynomial for the CRC calculation should be unified for both sending and receiving.

14.2.6 DMA

The SPI module supports the use of DMA to speed up data communication, either by using DMA to fill the transmit buffer or by using DMA to pick up data from the receive buffer in a timely manner. DMA will pick up or send data in a timely manner using RXNE and TXE as signals. DMA can also operate in simplex or CRC mode.

14.2.7 Errors

- Master mode fault (MODF)

When the SPI is operating in NSS pin hardware management mode, an external pull-down of the NSS pin occurs; or in NSS pin software management mode, the SSI bit is cleared; or the SPE bit is cleared, causing the SPI to be shut down; or the MSTR bit is cleared and the SPI enters slave mode. If the ERRIE bit is already set, an interrupt is also generated. Steps to clear the MODF bit: First perform a read or write operation to R16_SPI1_STATR, and then write R16_SPI1_CTLR1.

- Overrun condition

If the host sends data and there is unread data in the receive buffer of the slave device, an overflow error occurs, the OVR bit is set, and an interrupt is also generated if ERRIE is set. Sending an overflow error should restart the current transmission. Reading the data register and then reading the status register will eliminate this bit.

- CRC error

When the received CRC word and the value of RXCRCR do not match, a CRC error will be generated and the CRCERR bit will be set.

14.2.8 Interrupts

The SPI module supports five interrupt sources, among which the TXE and RXNE events are set when the TXEIE and RXNEIE bits are set respectively. In addition to the above three errors will also generate interrupts, namely MODF, OVR and CRCERR, after enabling the ERRIE bit, these three errors will also generate error interrupts.

14.3 Register Description

Table 14-1 SPI-related registers list

Name	Access address	Description	Reset value
R16_SPI_CTLR1	0x40013000	SPI Control register1	0x0000
R16_SPI_CTLR2	0x40013004	SPI Control register2	0x0000
R16_SPI_STATR	0x40013008	SPI Status register	0x0002
R16_SPI_DATAR	0x4001300C	SPI Data register	0x0000
R16_SPI_CRCR	0x40013010	SPI Polynomial register	0x0007
R16_SPI_RCRCR	0x40013014	SPI Receive CRC register	0x0000
R16_SPI_TCRCR	0x40013018	SPI Transmit CRC register	0x0000
R16_SPI_HSCR	0x40013024	SPI High-speed control register	0x0000

14.3.1 SPI Control Register 1 (SPI_CTLR1)

Offset address: 0x00

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
BIDIMODE	BIDIOE	CRCEN	CRCNEXT	DFF	RXONLY	SSM	SSI	LSBFIRST	SPE	BR[2:0]			MSTR	CPOL	CPHA

Bit	Name	Access	Description	Reset value
15	BIDIMODE	RW	Bidirectional data mode enable bit.	0

			1: Selection of 1-line bidirectional mode. 0: Selection of 2-line bi-directional mode.	
14	BIDIOE	RW	Output enable in bidirectional mode bit, used in conjunction with BIDImode. 1: Enable output, transmit only. 0: Disable output, receive only.	0
13	CRCEN	RW	Hardware CRC checksum enable bit, this bit can only be written when SPE is 0. This bit can only be used in full-duplex mode. 1: Initiate CRC calculation. 0: CRC calculation is disabled.	0
12	CRCNEXT	RW	After the next data transfer, send the value of the CRC register. This should be set immediately after the last data is written to the data register. 1: Sending CRC checksum results. 0: Continue to send data from the data register.	0
11	DFF	RW	Data frame format bit, this bit can only be written when SPE is 0. 1: Sending and receiving using 16-bit data length. 0: Use 8-bit data length for sending and receiving.	0
10	RXONLY	RW	The receive-only bit in 2-wire mode is used in conjunction with BIDIMODE. Setting this bit allows the device to receive only and not transmit. 1: Receive only, simplex mode. 0: Full-duplex mode.	0
9	SSM	RW	Software slave management bit, this bit determines whether the level of the NSS pin is controlled by hardware or software. 1: Software control of the NSS pins. 0: Hardware control NSS pins.	0
8	SSI	RW	Internal slave select bit, with SSM set, this bit determines the level of the NSS pin. 1: NSS is high. 0: NSS is low.	0
7	LSBFIRST	RW	Frame format control bit. It is not possible to modify this bit during communication. 1: LSB is transmitted first; 0: MSB is transmitted first. Note: LSB is only supported by SPI as host.	0
6	SPE	RW	SPI enable bit. 1: Enable SPI. 0: Disable SPI.	0
[5:3]	BR[2:0]	RW	Baud rate setting field, this field cannot be modified during communication. 000: F_{HCLK} /2; 001: F_{HCLK} /4. 010: F_{HCLK} /8; 011: F_{HCLK} /16. 100: F_{HCLK} /32; 101: F_{HCLK} /64. 110: F_{HCLK} /128; 111: F_{HCLK} /256.	0
2	MSTR	RW	Master-slave setting bit, this bit cannot be modified during communication. 1: Configured as a master device. 0: Configured as a slave device.	0b
1	CPOL	RW	Clock polarity selection bit, this bit cannot be modified during communication. 1: SCK is held high in idle state. 0: SCK is held low in idle state.	0
0	CPHA	RW	Clock phase setting bit, this bit cannot be modified during communication. 1: Data sampling starts from the second clock	0

			edge. 0: Data sampling starts from the first clock edge.	

14.3.2 SPI Control Register 2 (SPI_CTLR2)

Offset address: 0x04

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved								TXEIE	RXNEIE	ERRIE	Reserved		SSOE	TXDMAEN	RXDMAEN

Control register 2

Bit	Name	Access	Description	Reset value
[15:8]	Reserved	RO	Reserved	0
7	TXEIE	RW	Tx buffer empty interrupt enable bit. Setting this bit allows an interrupt to be generated when TXE is set.	0
6	RXNEIE	RW	RX buffer not empty interrupt enable bit. Used to generate an interrupt request when the RXNE flag is set.	0
5	ERRIE	RW	Error interrupt enable bit. Setting this bit allows interrupts to be generated when errors (CRCERR, OVR, MODF) are generated.	0
[4:3]	Reserved	RO	Reserved	0
2	SSOE	RW	SS output enable bit. Disabling SS output can work in multi-master mode. 1: Enable the SS output. 0: Disable SS output in Master mode.	0
1	TXDMAEN	RW	Tx buffer DMA enable bit. 1：Enable Tx buffer DMA. 0：Disable Tx buffer DMA.	0
0	RXDMAEN	RW	Rx buffer DMA enable bit. 1：Enable Rx buffer DMA. 0：Disable Rx buffer DMA.	0

14.3.3 SPI Status Register (SPI_STATR)

Offset address: 0x08

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved								BSY	OVR	MODF	CRCERR	UDR	CHSID	TXE	RXNE

Bit	Name	Access	Description	Reset value
[15:8]	Reserved	RO	Reserved	0
7	BSY	RO	Busy flag. This flag is set and cleared by hardware. 1：SPI is busy in communication or Tx buffer is not empty. 0：SPI (or I2S) not busy.	0
6	OVR	RWO	Overrun flag. This flag is set by hardware and reset by a software sequence. 1：Overrun occurred. 0：No overrun occurred.	0

5	MODF	RO	Mode fault. This flag is set by hardware and reset by a software sequence. 1：Mode fault occurred. 0：No mode fault occurred.	0
4	CRCERR	RW0	CRC error flag. This flag is set by hardware and reset by a software sequence. 1：CRC value received does not match the SPI_RXCRCR value. 0：CRC value received matches the SPI_RXCRCR value.	0
3	UDR	R0	Underrun flag. This flag is set by hardware and reset by a software sequence. 1：Underrun occurred. 0：No underrun occurred.	0
2	CHSID	RO	Channel side. This flag is set by hardware and reset by a software sequence. 1：Channel Right has to be transmitted or has been received. 0：Channel Left has to be transmitted or has been received.	0
1	TXE	RO	Transmit buffer empty. 1：Tx buffer empty. 0：Tx buffer not empty.	1
0	RXNE	RO	Receive buffer not empty. 1：Rx buffer not empty. 0：Rx buffer empty. Note: Read DATAR and auto-zero.	0

14.3.4 SPI Data Register (SPI_DATAR)

Offset address: 0x0C

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DR[15:0]															

Bit	Name	Access	Description	Reset value
[15:0]	DR[15:0]	RW	Data register. The data registers are used to store the received data or pre-store the data to be sent out, so the reading and writing of the data registers actually correspond to the operation of different areas, where the read pairs use the receive buffer and the write pairs correspond to the send buffer. Data can be received and sent in 8 or 16 bits, and it is necessary to determine how many bits of data to use before transmission. When using 8 bits for data transmission, only the lower 8 bits of the data registers are used, and the higher 8 bits are forced to 0 for reception. using a 16-bit data structure causes all 16 bits of the data registers to be used.	0

14.3.5 SPI1 Polynomial Register (SPI_CRCR)

Offset address: 0x10

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CRCPOLY[15:0]															

Bit	Name	Access	Description	Reset value
[15:0]	CRCPOLY[15:0]	RW	CRC polynomial. This register contains the polynomial for the CRC calculation.	7h

14.3.6 SPI1 Receive CRC Register (SPI_RCRCR)

Offset address: 0x14

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RXCRC[15:0]															

Bit	Name	Access	Description	Reset value
[15:0]	RXCRC[15:0]	RO	Rx CRC. Store the result of the calculated CRC checksum of the received byte. Setting CRCEN resets this register. The calculation method uses the polynomial used in CRCPOLY. 8-bit mode only the lower 8 bits are involved in the calculation, 16-bit mode all 16 bits are involved in the calculation. It is necessary to read this register when BSY is 0.	0

14.3.7 SPI1 Transmit CRC Register (SPI_TCRCR)

Offset address: 0x18

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TXCRC[15:0]															

Bit	Name	Access	Description	Reset value
[15:0]	TXCRC[15:0]	RO	Tx CRC. Store the result of the calculated CRC checksum of the bytes that have been sent out. Setting CRCEN resets this register. The calculation method uses the polynomial used in CRCPOLY. 8-bit mode only the lower 8 bits are involved in the calculation, while in 16-bit mode all 16 bits are involved. It is necessary to read this register when BSY is 0.	0

14.3.8 SPI High-speed Control Register (SPI_HSCR)

Offset address: 0x24

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved															HSRXEN

Bit	Name	Access	Description	Reset value
[15:1]	Reserved	RO	Reserved	0
0	HSRXEN	W0	Read enable in SPI high-speed mode. 1: Enable high-speed read mode. 0: Disable high-speed read mode. The fifth bit of the chip lot number is less than 2. This mode is supported only when the clock is divided by 2 (i.e., BR = 000 in the CTLR1 register) and is valid for all other lots without restriction, and the bit is write-only.	0