1 Synchronous Serial Interface

1.1 Overview

The SSI is a full-duplex synchronous serial interface and can connect to a variety of external analog-to-digital (A/D) converters, audio and telecom codecs, and other devices that use serial protocols for transferring data. The SSI supports National's Microwire, Texas Instruments Synchronous Serial Protocol (SSP), and Motorola's Serial Peripheral Interface (SPI) protocol.

The SSI operates in master mode (the attached peripheral functions as a slave) and supports serial bit rates from 7.2 KHz to 24 MHz. Serial data formats may range from 2 to 17 bits in length. The SSI provides 16 entries deep x 17 bits wide transmit and receive data FIFOs.

The FIFOs may be loaded or emptied by the Central Processor Unit (CPU) using programmed I/O, or DMA transfers while receiving or transmitting.

Features:

- 3 protocols support: National's Microwire, TI's SSP, and Motorola's SPI
- Full-duplex or transmit-only or receive-only operation
- Programmable transfer order: MSB first or LSB first
- 16 entries deep x 17 bits wide transmit and receive data FIFOs
- Configurable normal transfer mode or Interval transfer mode
- Programmable clock phase and polarity for Motorola's SSI format
- Two slave select signal (SSI_CE_ / SSI_CE2_) supporting up to 2 slave devices
- Back-to-back character transmission/reception mode
- Loop back mode for testing

1.2 Pin Description

Table 1-1 Micro Printer Controller Pins Description

Name	I/O	Description
SSI_CLK	Output	Serial bit-rate clock
SSI_CE_	Output	First slave select enable
SSI_CE2_ / SSI_GPC	Output	Second slave select enable / General purpose control signal to external chip
SSI_DT	Output	Transmit data (serial data out)
SSI_DR	Input	Receive data (serial data in)

SSI_CLK is the bit-rate clock driven from the SSI to the peripheral. SSI_CLK is toggled only when data is actively being transmitted and received.

SSI_CE_ or SSI_CE2_ are the framing signal, indicating the beginning and the end of a serialized data word.

SSI_DT and SSI_DR are the Transmit and Receive serial data lines.

SSI_GPC is general-purpose control signal, synchronized with SSI_CLK, can be used for LCD control.

When the multiplexed pin is configured as SSI_GPC pin, SSI can't be configured for 17-bit (or multiples of it) data transfer. And the SSI can only perform transfer with the only slave select SPI_CE_.

SSI_GPC and SSI_CE2_ is a multiplexed pin.

1.3 Register Description

The SSI has seven registers: one data, two control, one status, one bit-rate control, and two interval control registers. The table list these registers.

Table 1-2 SSI Serial Port Registers

Name	RW	Reset Value	Address	Access Size
SSIDR	RW	0x??	0x10043000	32
SSICR0	RW	0x0000	0x10043004	16
SSICR1	RW	0x00007060	0x10043008	32
SSISR	RW	0x00000098	0x1004300C	32
SSIITR	RW	0x0000	0x10043010	16
SSIICR	RW	0x00	0x10043014	8
SSIGR	RW	0x0000	0x10043018	16

1.3.1 SSI Data Register (SSIDR)

SSIDR **0x10043000**

<table>
<tr><th>Bit</th><th>31</th><th>30</th><th>29</th><th>28</th><th>27</th><th>26</th><th>25</th><th>24</th><th>23</th><th>22</th><th>21</th><th>20</th><th>19</th><th>18</th><th>17</th><th>16</th><th>15</th><th>14</th><th>13</th><th>12</th><th>11</th><th>10</th><th>9</th><th>8</th><th>7</th><th>6</th><th>5</th><th>4</th><th>3</th><th>2</th><th>1</th><th>0</th></tr>
<tr><td></td><td colspan="15">Reserved</td><td>GPC/D16</td><td colspan="16">Transmit/Receive Data</td></tr>
<tr><td>RST</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>?</td><td>?</td><td>?</td><td>?</td><td>?</td><td>?</td><td>?</td><td>?</td><td>?</td><td>?</td><td>?</td><td>?</td><td>?</td><td>?</td><td>?</td><td>?</td><td>?</td></tr>
</table>

Bits	Name	Description	RW
31:17	Reserved		R
16	GPC/D16	This bit can be used as normal data bus bit 16 or GPC bit alternatively. When the multiplexed output pin is selected as SSI_CE2_, it is normal data bus bit and it's readable / writable; when multiplexed pin is selected as SSI_GPC, it is GPC bit for SSI_GPC pin output and it's write-only	RW
15:0	Transmit/ Receive Data	Data word to be written to/read from Transmit/Receive FIFO. When the transfer frame length is less than 17-bit, received data is automatically right justified in the receive-FIFO and the upper unused bits are filled with '0'. For transmission, the upper unused bits of the data written into SSIDR is ignored by the transmit logic. (Note: "upper unused bits" does not include the SSIDR.GPC bit.	RW

		National microwire format includes format 1 and format2, when national microwire format 2 is selected, Bit 16 of SSIDR is defined as read/write operation judge bit, if it is 0, bit 15~0 represent one read command; if it is 1, bit 15~0 represent one write command and following is the written data. So the maximum length of one command (is defined in MCOM) is 16, the maximum length of one written or read data (is defined in FLEN) can be 17. Transmit-FIFO only contain one read operation command once, or one write operation command and its data once, after transmit-FIFO is empty, next command can be filled in transmit-FIFO.	

1.3.2 SSI Control Register0 (SSICR0)

SSICR0 **0x10043004**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SSIE	TIE	RIE	TEIE	REIE	LOOP	RFINE	RFINC	Reserved	FSEL	Reserved			TFLUSH	RFLUSH	DISREV
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	RW
15	SSIE	This bit is used to enable/disable SSI module: 0 – disable; 1 – enable Clearing SSIE will not reset SSI FIFO, SSICR0, SSICR1, SSIGR, SSIITR and SSIICR automatically. Software should ensure the FIFOs/registers are properly configured and be flush/reset manually when necessary before enabling SSI.	RW
14	TIE	This bit enables/disables the transmit-FIFO half-empty interrupt TXI: 0 – disable; 1 – enable	RW
13	RIE	This bit enables/disables the receive-FIFO half-full interrupt RXI: 0 – disable; 1 – enable	RW
12	TEIE	This bit enables/disables the transmit-error interrupt TEI: 0 – disable; 1 – enable	RW
11	REIE	This bit enables/disables the receive-error interrupt REI: 0 – disable; 1 – enable	RW
10	LOOP	Used for test purpose. In loop mode, the output of SSI transmit shift register is connected to input of SSI receive shift register internally. The data received should be the same as the data transmitted. And do not output any valid signals on the pins. 0 – normal SSI mode; 1 – LOOP mode	RW

9	RFINE	This bit enables/disables receive finish control function: 0 – disable; 1 – enable. For SSICR1.FMAT = B'10 (National Microwire format 1 is selected), SSICR0.RFINE must be 0	RW
8	RFINC*	Receive finish control bit: 0 – receive continue; 1 – receive finished	RW

The receive finish condition list below:

RFINE	RFINC	Receive Finish Condition
0	x	Same as transmit completion condition (transmit-fifo is empty and SSICR1.UNFIN = 0)
1	0	Receive continue
1	1	Receive finish

7	Reserved		R
6	FSEL	This bit sets the frame signal to be used for slave select. The unselected frame signal always output invalid level. When multiplexed pin is used as SSI_GPC, only 0 can be set. 0 – SSI_CE_ is selected; 1 – SSI_CE2_ is selected.	RW
5:3	Reserved		R
2	TFLUSH	Flush the transmit FIFO when set to 1. Always return 0 when read.	RW
1	RFLUSH	Flush the receive FIFO when set to 1. Always return 0 when read.	RW
0	DISREV	This bit enables/disables receive function: 0 – enable; 1 – disable	RW

Note: *: 1) When transmitting finished or for receive-only operation, transmit function can be disabled and this bit is used to control receiving completion, and the SSI will consume less power.

2) When the finish condition is set, the receiving will complete after present character is completely shifted in, then the SSI will stop the SSI_CLK and negate the SSI_CE_ / SSI_CE2_ if necessary. To make sure present transfer is completed, user must read and get SSISR.END = 1 (or SSISR.BUSY = 0).

1.3.3 SSI Control Register1 (SSICR1)

SSICR1 **0x10043008**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	FRMHL		TFVCK		TCKFI		LFST	ITFRM	UNFIN	MULTS	FMAT						MCOM				TTRG		RTRG		FLEN						PHA	POL
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	0	0	0	0	0	1	1	0	0	0	0	0

<table>
<tr><th>Bits</th><th>Name</th><th>Description</th><th>RW</th></tr>
<tr><td>31:30</td><td>FRMHL</td><td>Frame valid level select, FRMHL [1: 0] correspond to SSI_CE2_ and SSI_CE_ respectively.
<table>
<tr><th>FRMHL[1:0]</th><th>Description</th><th></th></tr>
<tr><td>00</td><td>SSI_CE_ is low level valid and SSI_CE2_ is low level valid</td><td>Initial value</td></tr>
<tr><td>01</td><td>SSI_CE_ is high level valid and SSI_CE2_ is low level valid</td><td></td></tr>
<tr><td>10</td><td>SSI_CE_ is low level valid and SSI_CE2_ is high level valid</td><td></td></tr>
<tr><td>11</td><td>SSI_CE_ is high level valid and SSI_CE2_ is high level valid</td><td></td></tr>
</table></td><td>RW</td></tr>
<tr><td>29:28</td><td>TFVCK</td><td>Time from frame valid to clock start, that provide programmable time delay from frame (SSI_CE_ /SSI_CE2_) assert edge to SSI_CLK leading edge. When TFVCK = B’00, the time is fixed half SSI_CLK or one SSI_CLK cycle according to SSICR1.POL and SSICR1.PHA configuration.
For SSICR1.FMAT = B’01, SSICR1.TFVCK is ignored.
<table>
<tr><th>TFVCK[1:0]</th><th>Description</th><th></th></tr>
<tr><td>00</td><td>Ignore (default half or one SSI_CLK cycle delay time)</td><td>Initial value</td></tr>
<tr><td>01</td><td>1 more SSI_CLK cycle delay time is added</td><td></td></tr>
<tr><td>10</td><td>2 more SSI_CLK cycle delay time is added</td><td></td></tr>
<tr><td>11</td><td>3 more SSI_CLK cycle delay time is added</td><td></td></tr>
</table></td><td>RW</td></tr>
<tr><td>27:26</td><td>TCKFI</td><td>Time from clock stop to frame invalid, provide programmable time delay from SSI_CLK last edge to frame (SSI_CE_ /SSI_CE2_) negate edge. When TCKFI = B’00, the time is fixed one SSI_CLK or half SSI_CLK cycle according to SSICR1.POL and SSICR1.PHA configuration.
For SSICR1.FMAT = B’01, SSICR1.TFVCK is ignored.
<table>
<tr><th>TCKFI[1:0]</th><th>Description</th><th></th></tr>
<tr><td>00</td><td>Ignore (default half or one SSI_CLK cycle delay time)</td><td>Initial value</td></tr>
<tr><td>01</td><td>1 more SSI_CLK cycle delay time is added</td><td></td></tr>
<tr><td>10</td><td>2 more SSI_CLK cycle delay time is added</td><td></td></tr>
</table></td><td>RW</td></tr>
</table>

<table>
<tr><td></td><td></td><td><table><tr><td>11</td><td>3 more SSI_CLK cycle delay time is added</td><td></td></tr></table></td><td></td></tr>
<tr><td>25</td><td>LFST</td><td>Set to LSB first or MSB first when transfer: 0 – MSB first; 1 – LSB first</td><td>RW</td></tr>
<tr><td>24</td><td>ITFRM</td><td>Frame during interval, selects if the Frame (SSI_CE_ /SSI_CE2_) signal is negated or not during interval time at Interval Mode (SSICR1.FMAT = B'00 and SSIITR.IVLTM ≠ H'0000). It's ignored at Normal Mode.
0 – SSI_CE_ /SSI_CE2_ deassert during interval time at Interval Mode
1 – SSI_CE_ /SSI_CE2_ keeps asserted during interval time at Interval Mode</td><td>RW</td></tr>
<tr><td>23</td><td>UNFIN</td><td>This bit controls whether the SSI finishes transmission or wait for data filling (underrun happen) after all data in transmit-FIFO are sent out during transfer. This bit must be cleared to 0 when SSICR1.FMAT = B'01 (TI's SSP format).
0 – Transmit-FIFO empty means end of transmission;
1 – Transmission didn't finish when transmit-FIFO is empty, SSI underrun error would occur and SSI waits for data filling; SSI_CLK and SSI_CE_ /SSI_CE2_ keeps asserted, SSI_CLK stop at the current level.

<b>Note:</b> For transmit-FIFO empty before any transfer after SSI enabled, if SSICR1.UNFIN = 1 or SSICR0.RFINE = 0, SSI will wait till transmit-FIFO isn't empty then start to transfer and no underrun error will occur; if SSICR1.UNFIN = 0 and SSICR0.RFINE = 1, after transmit-FIFO become empty, SSI will start a receive-only transfer.</td><td>RW</td></tr>
<tr><td>22</td><td>MULTS</td><td>This bit set the multiplexed pin function.
0 – Multiplexed pin is used as SSI_CE2_;
1 – Multiplexed pin is used as SSI_GPC.</td><td>RW</td></tr>
<tr><td>21:20</td><td>FMAT</td><td>These bits set the operating transfer format.<table><tr><th>FMAT[1:0]</th><th>Description</th><th></th></tr><tr><td>00</td><td>Motorola's SPI format</td><td>Initial value</td></tr><tr><td>01</td><td>TI's SSP format</td><td></td></tr><tr><td>10</td><td>National Microwire 1 format</td><td></td></tr><tr><td>11</td><td>National Micowire 2 format</td><td></td></tr></table></td><td>RW</td></tr>
<tr><td>19:16</td><td>Reserved</td><td></td><td>R</td></tr>
<tr><td>15:12</td><td>MCOM</td><td>When SSICR1.FMAT = B'10 or B'11 (National Microwire format 1 or 2 is selected), this bit decides the length of command from 1-bit to 16-bit. The length of written or read data is defined in FLEN. For SSICR1.FMAT ≠ B'10 or B'11, this bit is ignored.<table><tr><th>MCOM[1:0]</th><th>Description</th><th></th></tr><tr><td>0000</td><td>1-bit command selected</td><td></td></tr><tr><td>0001</td><td>2-bit command selected</td><td></td></tr><tr><td>0010</td><td>3-bit command selected</td><td></td></tr></table></td><td>RW</td></tr>
</table>

		0010	3-bit command selected		
		0011	4-bit command selected		
		0100	5-bit command selected		
		0101	6-bit command selected		
		0110	7-bit command selected		
		0111	8-bit command selected	Initial value	
		1000	9-bit command selected		
		1001	10-bit command selected		
		1010	11-bit command selected		
		1011	12-bit command selected		
		1100	13-bit command selected		
		1101	14-bit command selected		
		1110	15-bit command selected		
		1111	16-bit command selected		
11:10	TTRG	These bits set the transmit-FIFO half-empty threshold value, when equal or less characters left in transmit-FIFO, the SSISR.TFHE will be set to ‘1’.			RW
		TTRG[1:0]	**Description**		
		00	less than or equal to 1		
		01	less than or equal to 4		
		10	less than or equal to 8	Initial value	
		11	less than or equal to 14		
9:8	RTRG	Set the receive-FIFO half-full threshold value, when equal or more characters received in receive-FIFO, the SSISR.RFHF will be set to ‘1’.			RW
		RTRG[1:0]	**Description**		
		00	less than or equal to 1		
		01	less than or equal to 4	Initial value	
		10	less than or equal to 8		
		11	less than or equal to 14		
7:4	FLEN	These bits set the bit length of every character to be transmitted/received. The maximum data length can be configured is 17 bits. For data length longer than 17 bits (multiples of the SSICR1.FLEN configured length), the software should ensure properly processing. When SSI_GPC pin is used (SSICR1.MULTS = 1), the FLEN shouldn’t be configured as B’1111 (17-bit data). When TI SSP mode is selected (FMAT = 2’b01), 2-bit data length (FLEN = 4’b0000) isn’t supported.			RW
		MCOM[1:0]	**Description**		
		0000	2-bit data		
		0001	3-bit data		
		0010	4-bit data		
		0011	5-bit data		

<table>
<tr><td></td><td></td><td>
<table>
<tr><td>0011</td><td>5-bit data</td><td></td></tr>
<tr><td>0100</td><td>6-bit data</td><td></td></tr>
<tr><td>0101</td><td>7-bit data</td><td></td></tr>
<tr><td>0110</td><td>8-bit data</td><td>Initial value</td></tr>
<tr><td>0111</td><td>9-bit data</td><td></td></tr>
<tr><td>1000</td><td>10-bit data</td><td></td></tr>
<tr><td>1001</td><td>11-bit data</td><td></td></tr>
<tr><td>1010</td><td>12-bit data</td><td></td></tr>
<tr><td>1011</td><td>13-bit data</td><td></td></tr>
<tr><td>1100</td><td>14-bit data</td><td></td></tr>
<tr><td>1101</td><td>15-bit data</td><td></td></tr>
<tr><td>1110</td><td>16-bit data</td><td></td></tr>
<tr><td>1111</td><td>17-bit data</td><td></td></tr>
</table>
</td><td></td></tr>
<tr><td>3:2</td><td>Reserved</td><td></td><td>R</td></tr>
<tr><td>1</td><td>PHA</td><td>This bit sets the phase of the SSI_CLK from the beginning of a data frame for Motorola’s SPI format (SSICR1.FMAT = B’00).
0 – The leading edge of SSI_CLK is used to sample data from SSI_DR after the SSI_CE_ /SSI_CE2_ goes valid, it is initial value;
1 – The leading edge of SSI_CLK is used to drive data onto SSI_DT after the SSI_CE_ /SSI_CE2_ goes valid.</td><td>RW</td></tr>
<tr><td>0</td><td>POL</td><td>This bit sets SSI_CLK’s idle state polarity for Motorola’s SPI format (SSICR1.FMAT = B’00).
0 – SSI_CLK keeps low level when idle, when SSI_CE_ /SSI_CE2_ goes valid the leading clock edge is a rising edge, it is initial value;
1 – SSI_CLK keeps high level when idle, when SSI_CE_ /SSI_CE2_ goes valid the leading clock edge is a falling edge.</td><td>RW</td></tr>
</table>

1.3.4 SSI Status Register1 (SSISR)

SSISR 0x1004300C

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
															TFIFO-NUM					RFIFO-NUM					END	BUSY	TFF	RFE	TFHE	RFHF	UNDR	OVER
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	1	1	0	0	0

Bits	Name	Description	RW
31:18	Reserved		R

17:13	TFIFO-NUM	These bits indicate the Characters Number in Transmit-FIFO.	R
12:8	RFIFO-NUM	These bits indicate the Characters Number in Transmit-FIFO.	R
7	END	This bit indicates transfer end status. It is the inverse of SSISR.BUSY when transfer is in process, but it'll keep cleared at interval time before transfer is completed. It'll be set when transfer finished.	R
6	BUSY	This bit indicates SSI's working status. 0 – SSI is idle or at interval time; 1 – Transmission and/or reception is in process.	R
5	TFF	This bit denotes transmit-FIFO is full or not. 0 – Transmit-FIFO is not full; 1 – Transmit-FIFO is full.	R
4	RFE	This bit denotes receive-FIFO is empty or not. 0 – Receive-FIFO is not empty; 1 – Receive-FIFO is empty.	R
3	TFHE	This bit denotes whether the characters number in transmit-FIFO being less or equal to SSICR1.TTRG. 0 – The data in transmit-FIFO is more than the condition set by SSICR1.TTRG; 1 – The data in transmit-FIFO meets the condition set by SSICR1.TTRG, If SSICR0.TIE = 1, it will generate SSI TXI interrupt.	R
2	RFHF	This bit denotes whether the characters number in receive-FIFO being more or equal to the number set by SSICR1.RTRG. 0 – The data in receive-FIFO is less than the condition set by SSICR1.RTRG 1 – The data in receive-FIFO meets the condition set by SSICR1.RTRG, If SSICR0.RIE = 1, it will generate SSI RXI interrupt.	R
1	UNDR	Transmit-FIFO underrun status. When underrun happens, SSI set this bit and keeps the current status of SSI_CLK and SSI_CE_/SSI_CE2_, waiting for transmit-FIFO filling. 0 – Underrun has not occurred; 1 – Underrun has occurred, when SSICR0.TEIE is set, it will generate SSI TEI interrupt. Write '0' to clear this bit, writing '1' has no effect.	RW
0	OVER	Receive-FIFO overrun status, new received data will lose. 0 – Overrun has not occurred; 1 – Overrun has occurred, When SSICR0.REIE is set, it will generate SSI REI interrupt. Write '0' to clear this bit, writing '1' has no effect.	RW

1.3.5 SSI Interval Time Control Register (SSIITR)

SSIITR **0x10043010**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CNTCLK	IVLTM														
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	RW
15	CNTCLK	Counting clock source select. 0 – Use SSI bit clock (SSI_CLK) as the interval counter clock source; 1 – Use 32K clock as the interval counter clock source.	RW
14:0	IVLTM	Interval time set, set the cycle number of counting clock source for desired interval time. When SSIITR.IVLTM = 0x0000, normal mode is selected, and SSIITR.CNTCLK and SSIICR are ignored. When SSIITR.IVLTM ≠ 0x0000, interval mode is selected. The interval time is calculated as follows: *Interval time ≈ [CNTCLK clock period] * [Value of IVLTM]* The actual interval time is as follow: When SSIITR.CNTCLK = 0: *Interval time = [CNTCLK clock period] * [Value of IVLTM] + 3 * device_clock period* When SSIITR.CNTCLK = 1: *Interval time ≥ [CNTCLK clock period] * [Value of IVLTM + 1] + 1 * device_clock period;* *Interval time ≤ [CNTCLK clock period] * [Value of IVLTM + 2] + 2 * device_clock period*	RW

1.3.6 SSI Interval Character-per-frame Control Register (SSIICR)

SSIICR 0x10043014

Bit	7	6	5	4	3	2	1	0
						ICC		
RST	0	0	0	0	0	0	0	0

Bits	Name	Description	RW
7:3	Reserved		R
2:0	ICC	Sets the fixed number of characters to be transmitted / received each time during SSI_CLK changing (and SSI_CE_ / SSI_CE2_ asserting) in interval mode for SSICR1.FMAT = B’00 (Motorola’s SPI format is selected). SSIICR is ignored for SSICR1.FMAT ≠ B’00. The desired transfer number of characters-per-frame is (SSIICR set value + 1).	RW

1.3.7 SSI Clock Generator Register (SSIGR)

SSIGR 0x10043018

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
												CGV				
RST	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Bits	Name	Description	RW
15:8	Reserved		R
7:0	CGV	Sets the frequency of serial bit clock (SSI_CLK). The serial bit clock (SSI_CLK) is generated by dividing device-clock as follows: F_{SSI_CLK} = [Frequency of device clock] / (2 * (CGV + 1)) Device clock is generated in CPM module. The value in SSIGR can be set from 0 to 255, and initialized to 0x0000 on power-on reset.	RW

1.4 Functional Description

Serial data is transferred between the processor and external peripheral through FIFO buffers in the SSI. Data transfers to system memory are handled by either the CPU (using programmed I/O) or by DMA. Operation is full duplex - separate buffers and serial data paths permit simultaneous transfers to and from the external peripheral.

Programmed I/O transmits and receives data directly between the CPU and the transmit/receive FIFO's. The DMA controller transfers data during transmit and receive operations between memory and the FIFO's.

Transmit data is written by the CPU or DMA to the SSI's transmit FIFO. The SSI then takes the data from the FIFO, serializes it, and transmits it via the SSI_DT signal to the peripheral. Data from the peripheral is received via the SSI_DR signal, converted to parallel words and is stored in the Receive FIFO. Read operations automatically target the receive FIFO, while write operations write data to the transmit FIFO. Both the transmit and receive FIFO buffers are 16 entries deep by 17 bits wide. As the received data fills the receive FIFO, a programmable threshold triggers an interrupt to the Interrupt Controller. If enabled, an interrupt service routine responds by identifying the source of the interrupt and then performs one or several read operations from the inbound (receive) FIFO buffer.

1.5 Data Formats

Four signals are used to transfer data between the processor and external peripheral. The SSI supports three formats: Motorola SPI, Texas Instruments SSP, and National Microwire. Although they have the same basic structure the three formats have significant differences, as described below.

SSI_CE_/SSI_CE2_ varies for each protocol as follows:

- For SPI and Microwire formats, SSI_CE_/SSI_CE2_ functions as a chip select to enable the external device (target of the transfer), and is held active-low during the data transfer.
- For SSP format, this signal is pulsed high for one serial bit-clock period at the start of each frame.

SSI_CLK varies for each protocol as follows:

- For Microwire, both transmit and receive data sources switch data on the falling edge of SSI_CLK, and sample incoming data on the rising edge.
- For SSP, transmit and receive data sources switch data on the rising edge of SSI_CLK, and sample incoming data on the falling edge.
- For SPI, the user has the choice of which edge of SSI_CLK to use for switching outgoing data, and for sampling incoming data. In addition, the user can move the phase of SSI_CLK, shifting its active state one-half period earlier or later at the start and end of a frame.

While SSP and SPI are full-duplex protocols, Microwire uses a half-duplex master-slave messaging protocol. At the start of a frame, a 1 or 2-byte control message is transmitted from the controller to the peripheral. The peripheral does not send any data. The peripheral interprets the message and, if it is a READ request, responds with requested data, one clock after the last bit of the requesting message.

The serial clock (SSI_CLK) only toggles during an active frame. At other times it is held in an inactive or idle state, as defined by its specified protocol.

1.5.1 Motorola's SPI Format Details

1.5.1.1 General Single Transfer Formats

The figures below show the timing of general single transfer format. SSI_GPC is also illustrated when the multiplexed pin is selected as SSI_GPC.

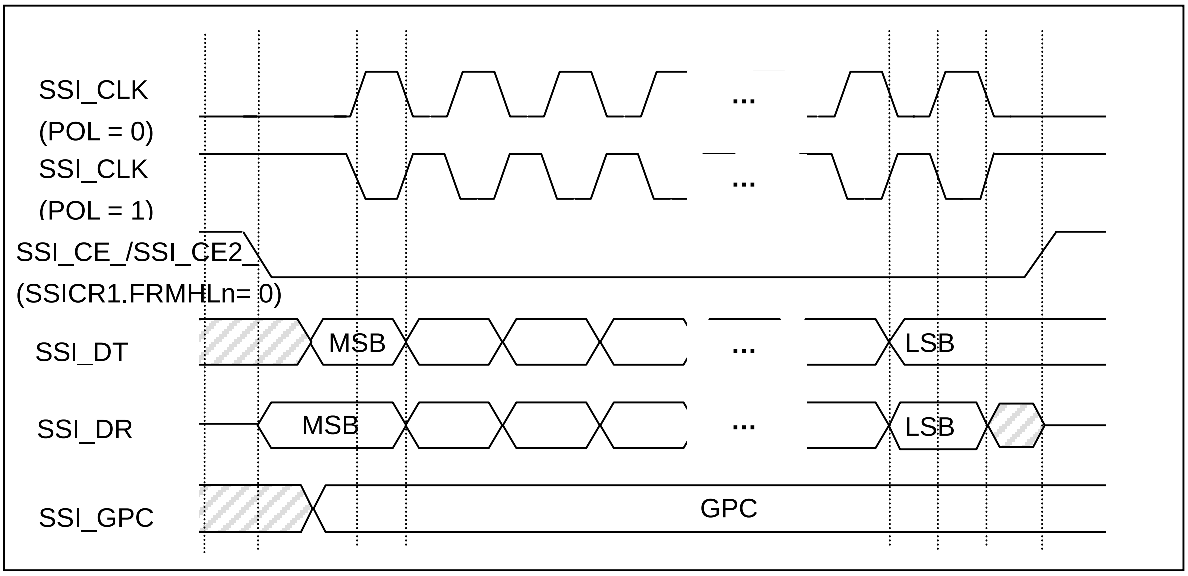

Figure 1-1 SPI Single Character Transfer Format (PHA = 0)

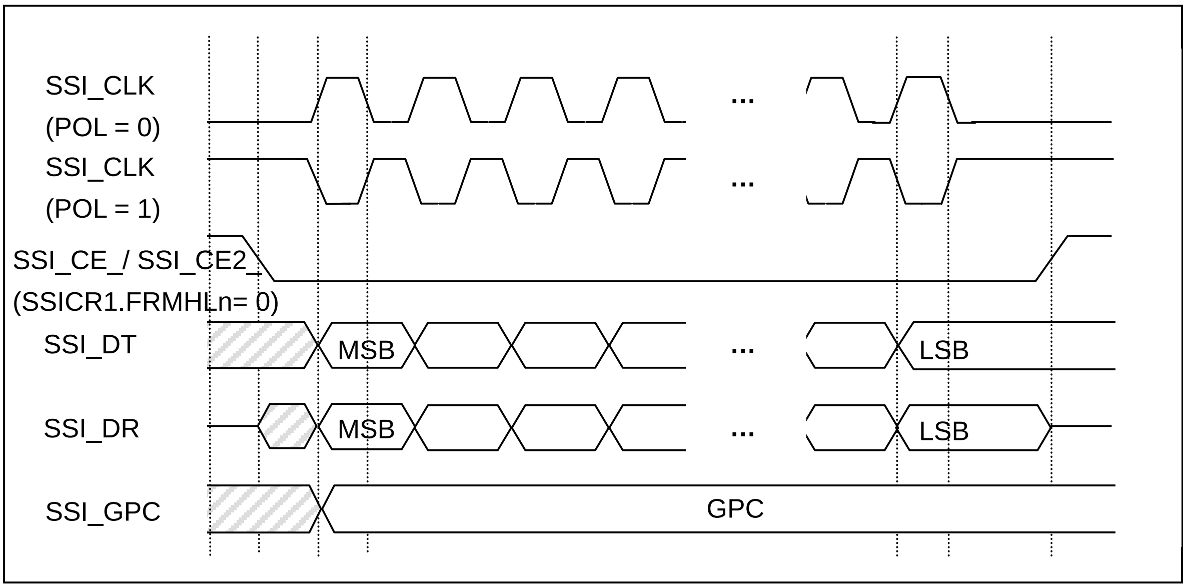

Figure 1-2 SPI Single Character Transfer Format (PHA = 1)

For SSICR1.PHA = 0, when SSICR1.TFVCK = B’00, hardware ensures the first clock edge appears one SSI_CLK period after SSI_CE_ / SSI_CE2_ goes valid; when SSICR1.TCKFI = B’00, hardware ensures the SSI_CE_ / SSI_CE2_ negated half SSI_CLK period after last clock change edge; when SSICR1.TFVCK ≠ B’00 or SSICR1.TCKFI ≠ B’00, 1/2/3 more clock cycles are inserted.

For SSICR1.PHA = 1, when SSICR1.TFVCK = B’00, hardware ensures the first clock edge appears half SSI_CLK period after SSI_CE_ / SSI_CE2_ goes valid; when SSICR1.TCKFI = B’00, hardware ensures the SSI_CE_ / SSI_CE2_ negated one SSI_CLK period after last clock change edge; when SSICR1.TFVCK ≠ B’00 or SSICR1.TCKFI ≠ B’00, 1/2/3 more clock cycles are inserted.

Data is sampled from SSI_DR at every rising edge (when PHA = 0, POL = 0 or PHA = 1, POL = 1) or at every falling edge (when PHA = 0, POL = 1 or PHA = 1, POL = 0). According to SPI protocol, input data on SSI_DR should be stable at every sample clock edge.

Drive data onto SSI_DT at every rising edge (when PHA = 0, POL = 1 or PHA = 1, POL = 0) or at every falling edge (when PHA = 0, POL = 0 or PHA = 1, POL = 1).

1.5.1.2 Back-to-Back Transfer Formats

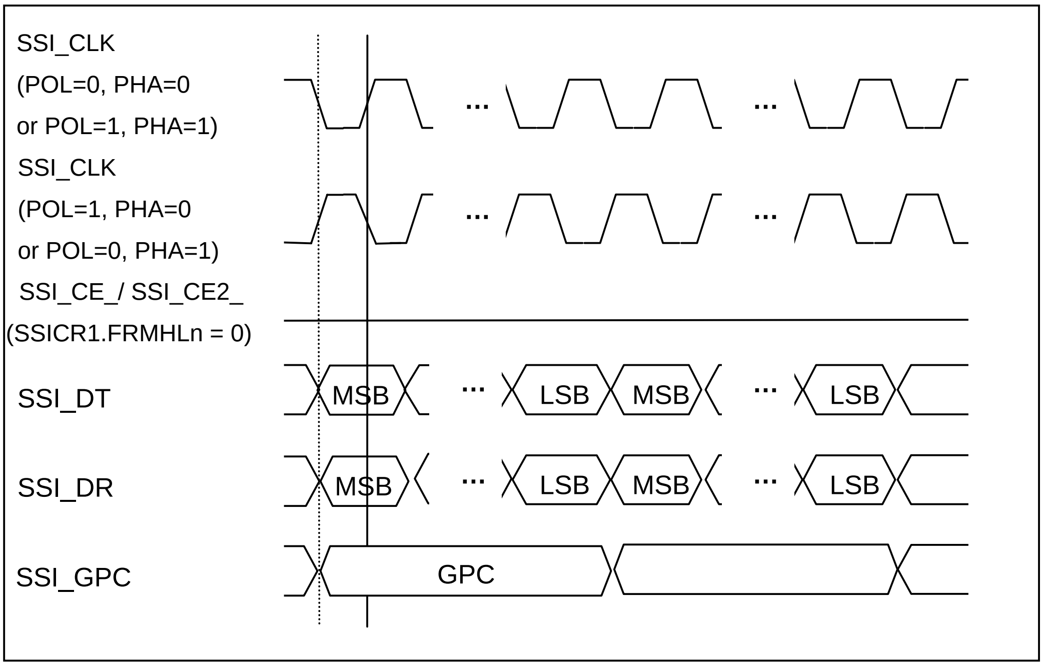

Figure 1-3 SPI Back-to-Back Transfer Format

For Motorola’s SPI format transfers those continuous characters are exchanged during SSI_CE_ / SSI_CE2_ being valid, the timing is illustrated in the figure (SSICR1.LFST = 0).

Back-to-back transfer is performed as transmit-only/full-duplex operation when transmit-FIFO is not empty before the completion of the last character’s transfer or performed as receive-only operation.

1.5.1.3 Frame Interval Mode Transfer Format

When in interval mode (SSIITR. IVLTM ≠ '0'), SSI always wait for an interval time (SSIITR.IVLTM), transfer fixed number of characters (SSIICR), then repeats the operation.

When SSICR0.RFINE = 1, if transmit-FIFO is still empty after the interval time, receive-only transfer will occur.

During interval-wait time, SSI stops SSI_CLK, and when SSICR1.ITFRM = 0 it negates the SSI_CE_ / SSI_CE2_, when SSICR1.ITFRM = 1 it keeps asserting the SSI_CE_ / SSI_CE2_.

For transfers finished with transmit-FIFO empty, if the SSI transmit-FIFO is empty before fixed number of characters being loaded to transfer (SSICR1.UNFIN must be 1), then the SSI will set SSISR.UNDR = 1; if enabled, it'll send out a SSI underrun interrupt. At the same time, SSI will hold the SSI_CE_ / SSI_CE2_ and SSI_CLK signals at current status and wait for the transmit-FIFO filling. The SSI will continue transfer after transmit-FIFO being filled. The SSI always stops after completion of fixed number of characters' transfer (SSICR1.UNFIN must be 0) with transmit-FIFO empty.

For transfers finished by SSICR0.RFINC being valid set, the SSI will stop after finished current character transfer and needn't wait for a whole completion of fixed number of characters' transfer.

Two Interval transfer mode are illustrated in the following figures. In these timing diagram, SSICR1.PHA = 0, SSICR1.POL = 0 and SSIICR = 0.

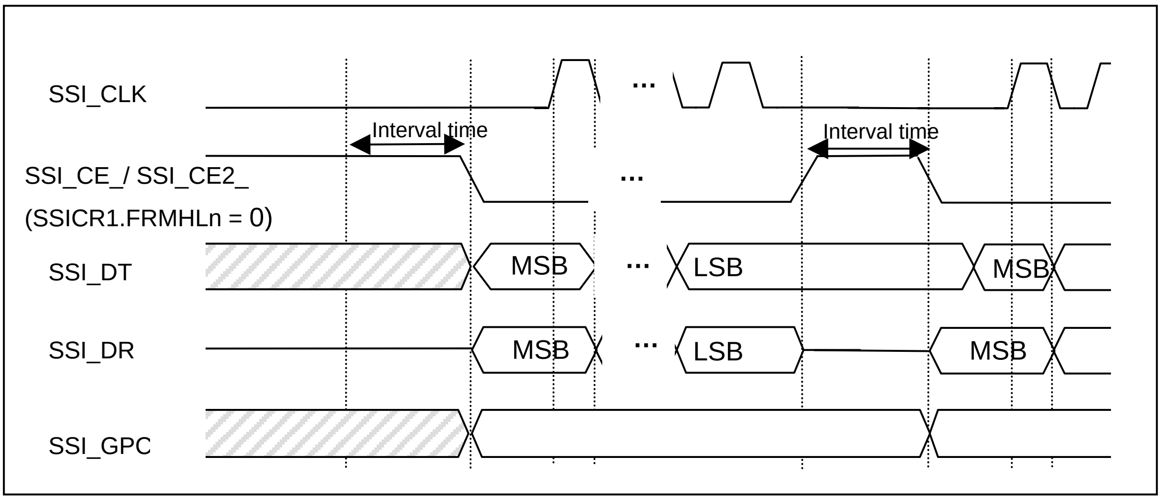

Figure 1-4 SPI Frame Interval Mode Transfer Format (ITFRM = 0, LFST = 0)

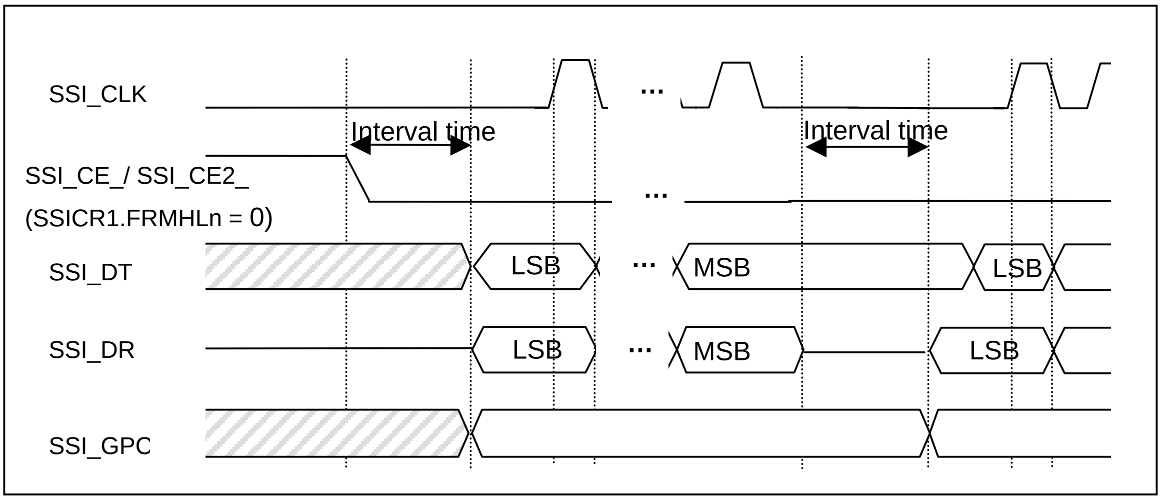

Figure 1-5 SPI Frame Interval Mode Transfer Format (ITFRM = 1, LFST = 1)

1.5.2 TI's SSP Format Details

In this format, each transfer begins with SSI_CE_ pulsed high for one SSI_CLK period. Then both master and slave drive data at SSI_CLK's rising edge and sample data at the falling edge. Data are transferred with MSB first or LSB first. At the end of the transfer, SSI_DT retains the value of the last bit sent through the next idle period.

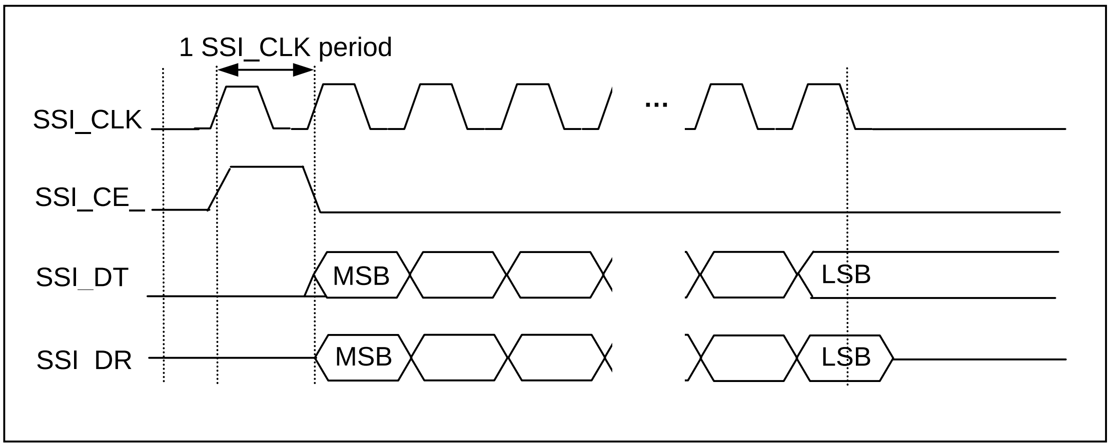

Figure 1-6 TI's SSP Single Transfer Format

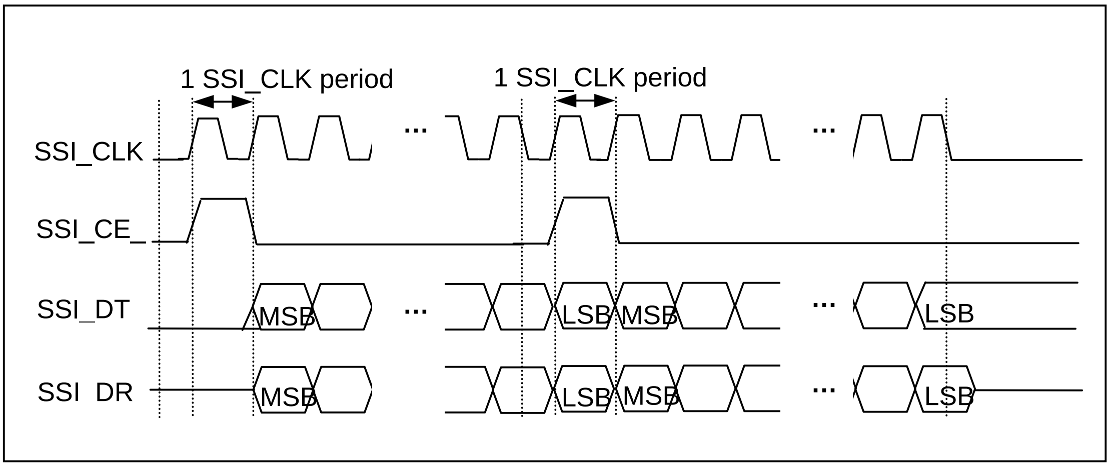

Figure 1-7 TI's SSP Back-to-back Transfer Format

1.5.3 National Microwire Format Details

It supports format 1 and format 2. If format 1 is selected, both master and slave drive data at SSI_CLK falling edge and sample data at the rising edge. If format 2 is selected, master drive and sample data at SSI_CLK falling edge, slave drive and sample data at SSI_CLK rising edge. SSI_CLK goes high midway through the command's most significant bit (or LSB) and continues to toggle at the bit rate. One bit clock (format 1) or half one bit clock (format 2) period after the last command bit, the external slave must return the serial data requested, with most significant bit first (or LSB first) on SSI_DR. SSI_CE_ / SSI_CE2 deasserts high half clock (SSI_CLK) period (and 1/2/3 additional clock periods) later. Format 1 support back-to-back transfer, the start and end of back-to-back transfers are similar to those of a single transfer. However, SSI_CE_ / SSI_CE2 remains asserted throughout the transfer. The end of a character data on SSI_DR is immediately followed by the start of the next command byte on SSI_DT.

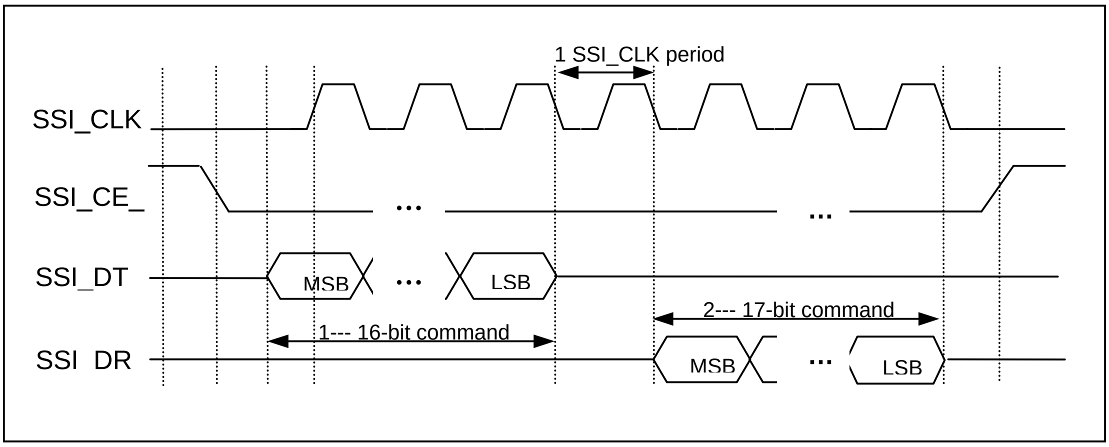

Figure 1-8 National Microwire Format 1 Single Transfer

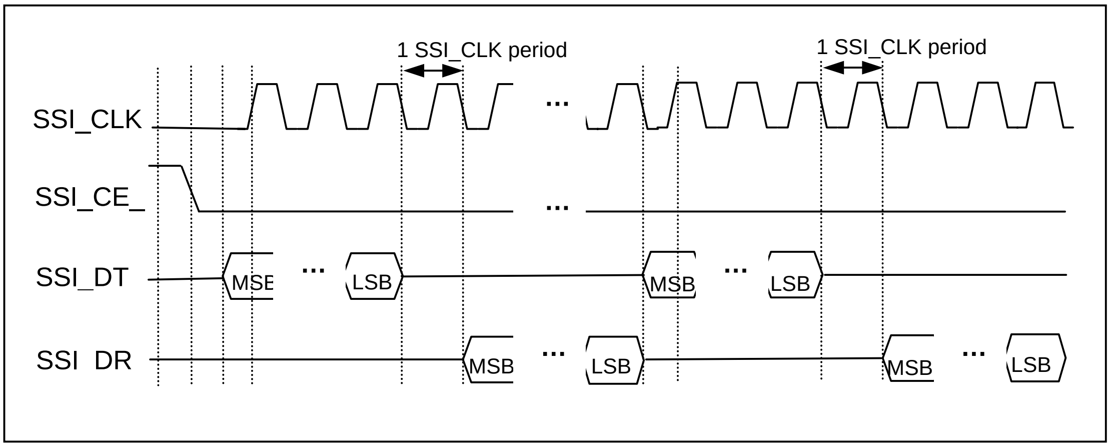

Figure 1-9 National Microwire Format 1 Back-to-back Transfer

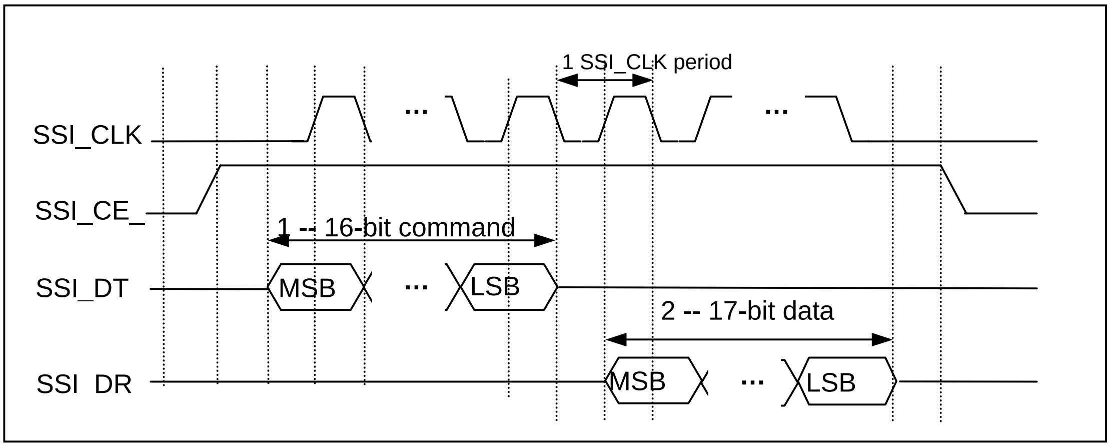

Figure 1-10 National Microwire Format 2 Read Timing

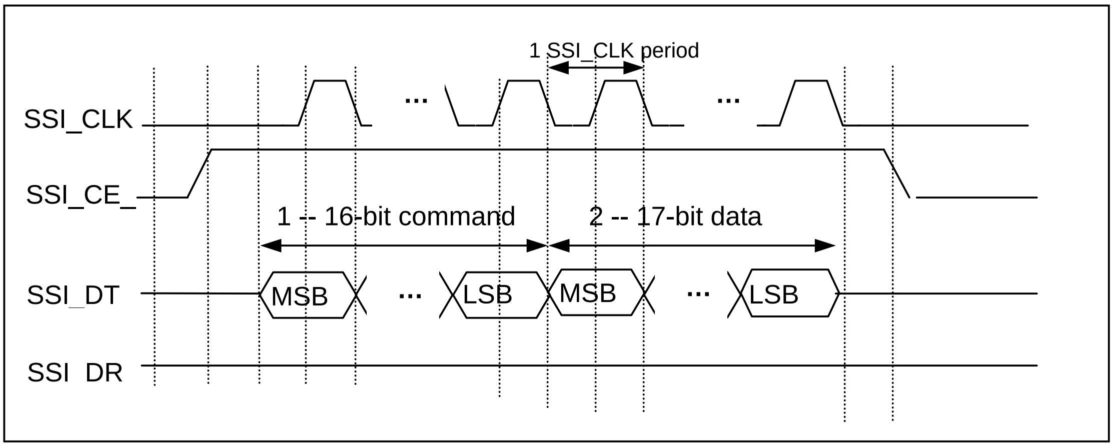

Figure 1-11 National Microwire Format 2 Write Timing

1.6 Interrupt Operation

In SSI, there are TXI, RXI, TEI and REI total 4 interrupts, all these interrupts are combined together to make one SSI interrupt, which can be masked by writing ‘1’ into corresponding mask bit in INTC interrupt mask register (IMR).

Table 1-3 SSI Interrupts

Operation	Condition	Flag Bit	Mask Bit	Interrupt	DMAC Activation
Transmit	T-FIFO is half-empty or less	SSISR.TFHE	SSICR0.TIE	**TXI**	Possible
	Transmit underrun error	SSISR.UNDR	SSICR0.TEIE	**TEI**	Impossible
Receive	R-FIFO is half-full or more	SSISR.RFHF	SSICR0.RIE	**RXI**	Possible
	Receive overrun error	SSISR.OVER	SSICR0.REIE	**REI**	Impossible

Either SSISR.TFHE or SSISR.RFHF can activate DMA transferring when corresponding individual interrupt mask bit in SSICR0 is cleared (masked) and DMA is enabled and configured.