Institut für Nanoelektronik

TUHH
Technische Universität Hamburg-Harburg

Circuit Design Laboratory: Digital Design (SS 16)

Prof. Dr.-Ing. W. Krautschneider
Instructors: Dipl.-Ing. Andreas Bahr / Dr.-Ing. Lait Abu Saleh

Lab Document **No.2:**

Lab Task: Design of a **S**erial **P**eripheral **I**nterface (SPI) Master

1. Task description:

Implement a SPI Master core. The interface signals to the top module are shown in Fig. 1. The following components are available for the complete system:

- 0.35µm CMOS Standard cells for 3.3V supply voltage.
- External oscillator (MCLK)

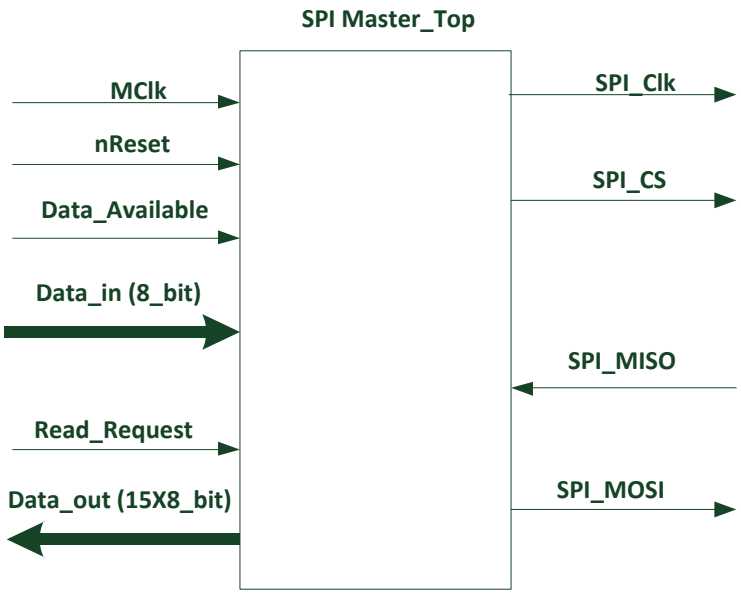

Fig. 1 Master Core Interface Signals

2. Communication Protocol:

The SPI communication is performed between two components, a Master and a Slave. The Master starts the communication as it selects the Slave via the “SPI_CS” signal and outputs the clock on the “SPI_CLK” signal. Hence, the address bits of the destination register are sent by the Master on the SPI_MOSI signal (Master_Out_Slave_In). Then depending on the chosen operation (transmit/receive), the data bits are either sent from the Master on the SPI_MOSI

signal for a transmit case or received from the Slave on the SPI_MISO (Master_In_Slave_Out) signal for a receive case.

In this design task, the SPI communication protocol depends on the following points:

- Data transmission and reception cannot happen at the same time.
- Data is transmitted or received in terms of blocks, each having 15 bytes.
- The master core has two separate 15-byte buffers (FIFOs), one for data transmission and one for data reception.
- The address of the destination register on the slave side, for both the transmission and reception operation, has to be set to 45 (0x2D)
- The address byte is divided into 1 control bit (write/read) and 7 bit register address

a. Data Transmission

The SPI_Master data transmission depends on the data availability in the transmission FIFO. At every positive edge of the asynchronous “Data_Available” signal, a new data Byte is available on the “Data_in” port which can be stored in the transmission FIFO, as in Fig. 2.

The SPI data transmission is performed in terms of blocks where each block contains 15 Bytes. Accordingly, before the data transmission starts, a block of 15 of data bytes has to be available in the transmission FIFO. The SPI data transmission has to be fast enough to transmit at least one byte before the next data_in byte arrives to prevent buffer overflow. The “Data_available” signal rate is about 1500 times slower than the MCLK.

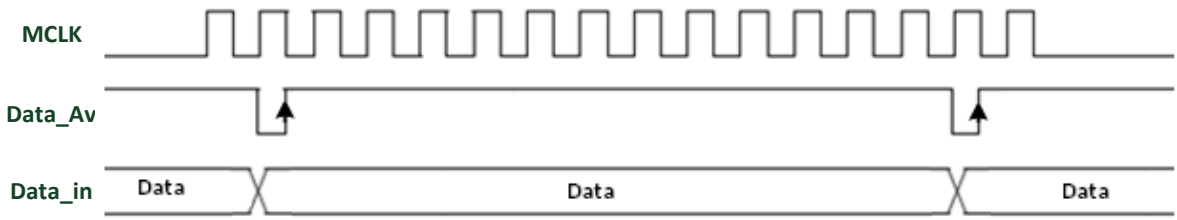

Fig. 2 Input Data for Transmission to SPI_Master

Fig. 3 shows the SPI data transmission signals, where timing is sequential from left to right. The data transmission starts with the address byte on the SPI_MOSI line, followed by 15 data bytes. The control bit of the address byte (A7) should be set to 0 to indicate a write operation.

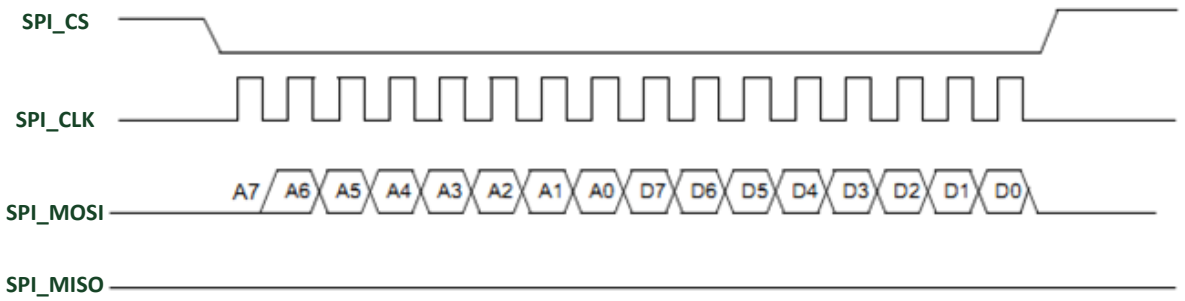

Fig. 3 Timing for SPI Transmission

b. Data Reception

The SPI_Master data reception is initiated by the positive edge of an asynchronous “Read_Request” interrupt signal, as in Fig. 4, which tells the Master that there is data available for reception. Hence, the master shall start the necessary communication to receive data form the Slave

Similar to the data transmission, the SPI data reception is performed in terms of blocks where each block contains 15 Bytes. The received data is stored in the receive FIFO and it can be made available at the interface in parallel via a 15-byte “Data_out” port

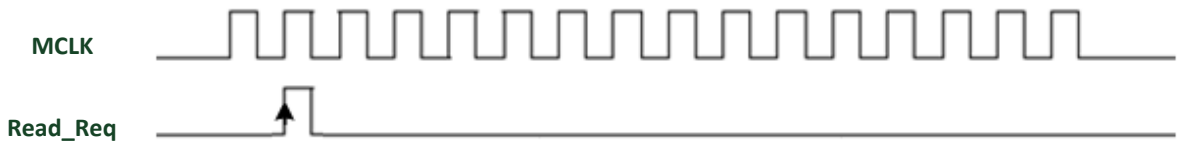

Fig. 4 Read_Request Interrupt for Data Reception

Fig. 5 shows the SPI data reception signals, where timing is sequential from left to right. The data reception starts with the address byte on the SPI_MOSI line. The control bit of the address byte (A7) should be set to 1 to indicate a read operation. Then, 15 data bytes are received on the SPI_MISO line.

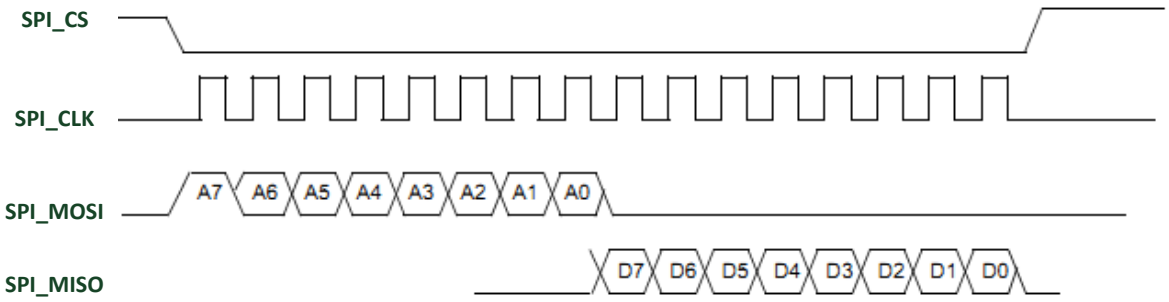

Fig. 5 Timing for SPI Reception

3. Timing Requirements:

Please consider the following points while designing:

- The SPI_CLK chosen for this design task is 1MHz
- The master clock signal (MCLK) is the main system clock and controls all its internal blocks. The MCLK rate is 20MHz, so an internal clock divider is needed to generate the SPI_CLK.
- The nReset signal is active low, and is assynchronous to the master clock (MCLK)

- The SPI_CLK clock polarity is 0, meaning that its idle state is 0. It is only active during the SPI communication.
- The SPI_CLK clock phase is 0, meaning that data is valid on the first SPI_CLK edge

The timing requirements and parameters for the SPI communication are shown in Table 1 and Fig. 6.

Parameter	Symbol	Min	Typ	Max	Unit
Data to SPI_CLK Setup	tDC	50	125		ns
SPI_CLK to Data Hold SPI_MOSI	tCDHI	50	125		ns
SPI_CLK to Data Hold SPI_MISO*	tCDHO				ns
SPI_CLK to Data Delay	tCDD	0	2	50	ns
SPI_CLK Low Time	tCL		125		ns
SPI_CLK High Time	tCH	120			ns
SPI_CLK Frequency**	fSPI_CLK			4	MHz
SPI_CLK Rise and Fall	tR, tF			25	ns
SPI_CS to SPI_CLK Setup	tCC	125			ns
SPI_CLK to SPI_CS Hold	tCCH	125			ns
SPI_CS Inactive Time	tCWH	250			ns
SPI_CS to Output High-Z	tCDZ			300	ns

Table 1 Timing Requirements

* Depends on SPI_CLK frequency. Data will be valid until new is driven (see tCDD) or SPI_CS is inactive (high)

**in this design task, we will use a 1 MHz SPI_CLK

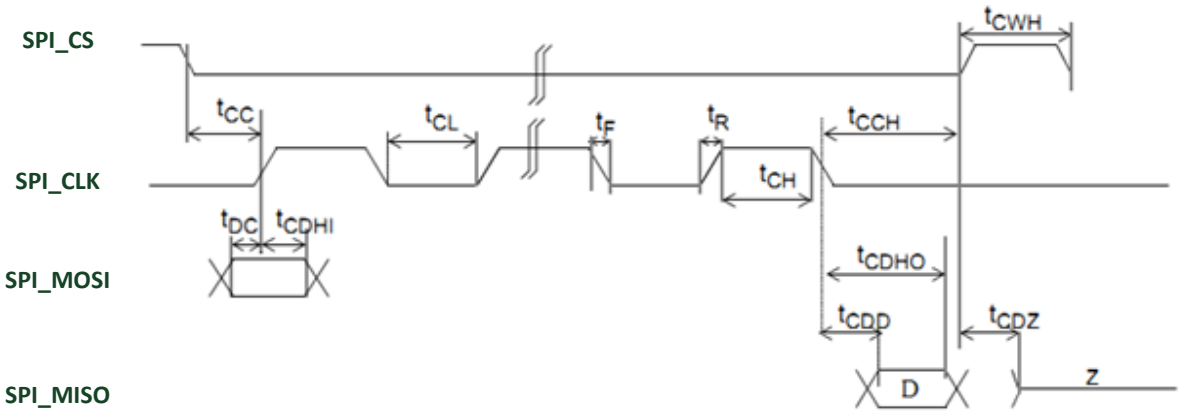

Fig. 6 Timing Parameters