

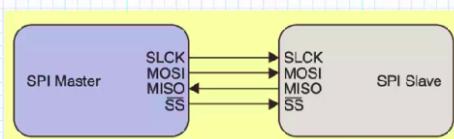
SERIAL PERIPHERAL INTERFACE (SPI)

The SPI Protocol

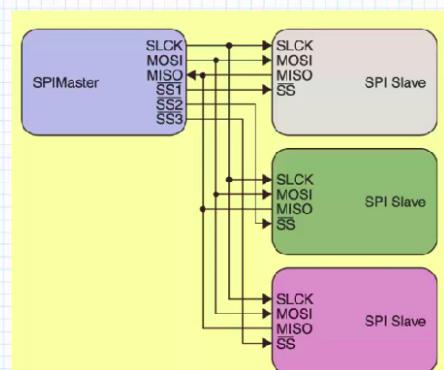
- Synchronous and full-duplex communication between a master and several slave devices.
- Uses two pins for data transfer-**SDIN** and **SDO**.
- There is a **SCLK** pin to synchronize data transfer between two chips.
- There is also a **CE** pin which is used to initiate and terminate the data transfer.
- Four (4) pins in total.

The SPI Protocol- Pin Names

Single Slave



Multiple Slaves

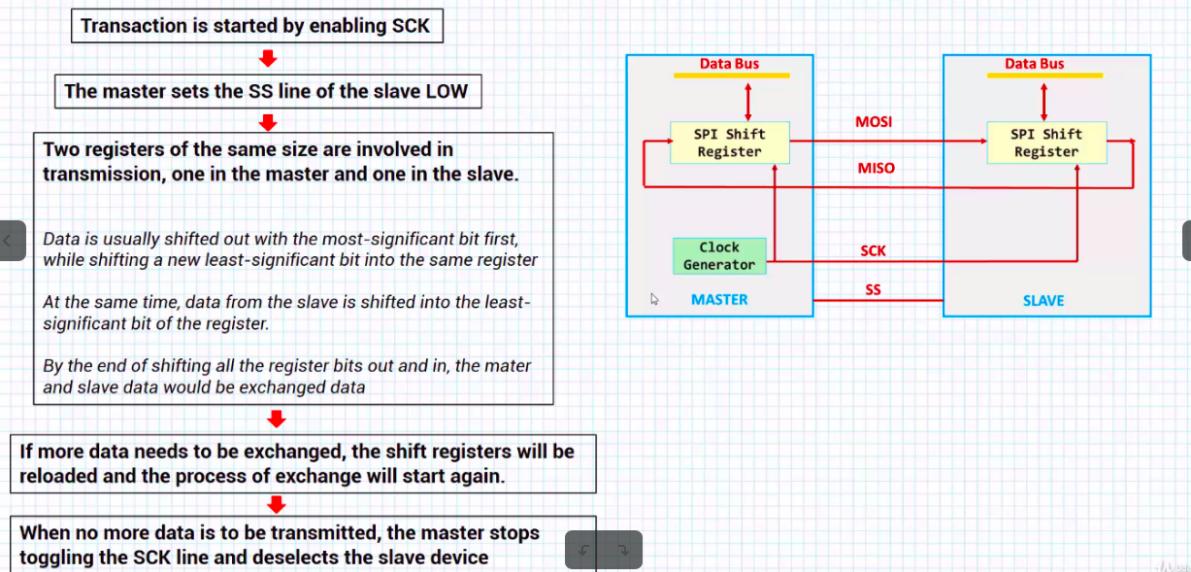


- **SDI** = **MOSI**
- **SDO** = **MISO**
- **SCLK** = **SCK**
- **CE** = **SS**

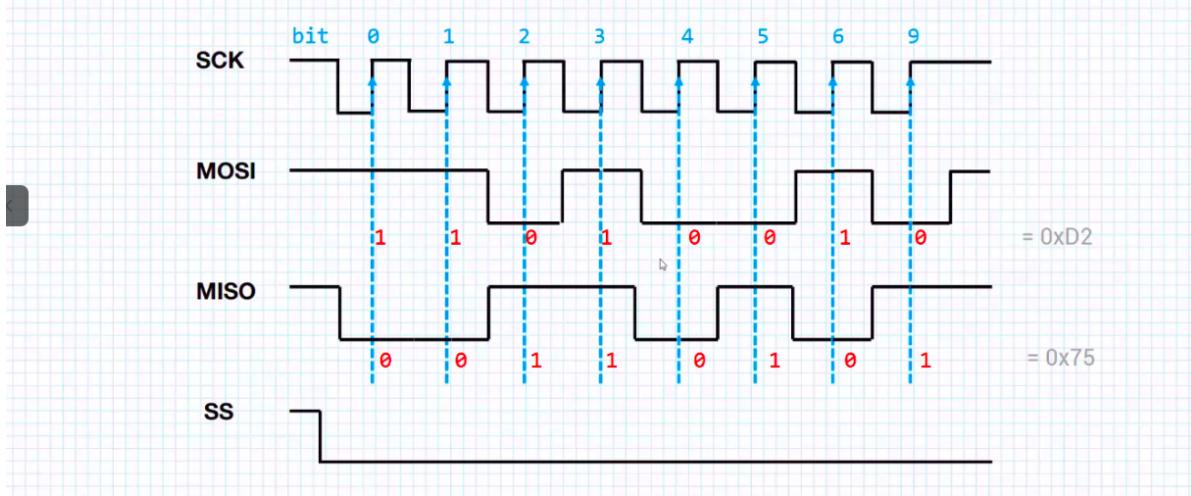
The SPI Protocol- Pin Functions

- **MOSI** : *Master Out Slave In*, used to send data from master to slave
- **MISO**: *Master In Slave Out*, used to send data from slave device to master
- **SCK**: *Serial Clock*, used to generate clock to synchronize data transfer between the master and slave device. Generated by master.
- **SS**: *Slave Select*, used to select a particular slave in the SPI multiple slaves configuration.

The SPI Protocol- How it works



The SPI Protocol- *Transmission*



The SPI Protocol- *Clock Phase and Polarity*

- Master and Slave device must agree on *clock phase* and *polarity*
- CPOL : Clock Polarity
- CPHA: Clock Phase
- Combinations of CPOL and CPHA are referred to as *SPI bus modes*.

The SPI Protocol- *Bus Modes*

	CPOL	CPHA
Mode0	0	0
Mode1	0	1
Mode2	1	0
Mode3	1	1

- **CPOL = 0**
 - Active state of clock = 1
 - Idle state of clock = 0
 - CPHA = 0**
 - Data is captured on the *rising edge*
 - Data is output on the *falling edge*
 - CPHA = 1**
 - Data is captured on the *falling edge*
 - Data is output on the *rising edge*
- **CPOL = 1**
 - Active state of clock = 0
 - Idle state of clock = 1
 - CPHA = 0**
 - Data is captured on the *falling edge*
 - Data is output on the *rising edge*
 - CPHA = 1**
 - Data is captured on the *rising edge*
 - Data is output on the *falling edge*

The SPI Protocol- *Bus Modes*

- **CPOL = 0 Means sampling on the first edge**
- **CPOL = 1 Means sampling on the second edge**

The SPI Protocol- *NSS Management*

- NSS software mode :
SS line is driven internally by the firmware.
- NSS hardware mode :
A dedicated GPIO pin is used to drive the SS line
 - NSS Output Enabled:
Used only when device operates in master mode
 - NSS Output Disabled:
Allows multi-master capability for devices operating in master mode

The SPI Protocol- *TI Mode*

- NSS hardware mode must be used.
- CPHA and CPOL are forced to conform to Texas Instrument (TI) protocol requirements.
- In this mode NSS signal pulses at the end of every transmitted byte.