Serial Peripheral Interface

SPI provides:

- Speed improvements over UART and I2C.
- Full-duplex communication, meaning data can be sent and received simultaneously.

SPI is a four-wire serial communication protocol typically used between a controller (master) and its peripherals (slaves).

- The master controls the communication and can connect to multiple slaves.

The four wires of an SPI line are as follows:

1. **Chip Select (CS)**: Used to indicate to the slaves that data is incoming.
2. **Source Clock (SCLK)**: Provides timing and synchronization.
3. **Master Out Slave In (MOSI)**: Used for the master to send data to the slave.
4. **Master In Slave Out (MISO)**: Used for the master to receive data from the slave.

Chip Select (CS)

- Typically active low, indicated as $\overline{\text{CS}}$ (or CSN, Chip Select Not).
- Allows the master to address multiple slaves, with each having a unique CS line.

SCLK

- The clock signal is generated by the master, so slaves don't need their own clocks.
- Clock speed is usually in the MHz range, much faster than UART or I2C.
- Indicates when data should be sampled (when to read voltage levels on data lines).
- One bit is read per clock signal.
- **Clock Polarity (CPOL):**
  - CPOL=0: Active high (idle low).
  - CPOL=1: Active low (idle high).
- **Clock Phase (CPHA):**
  - CPHA=0: Data sampled on the leading edge.
  - CPHA=1: Data sampled on the trailing edge.
- The combination of CPOL and CPHA determines the SPI mode (0–3):
  - **Mode 0** (CPOL=0, CPHA=0) is the most common but the mode of a peripheral is often specified in it's datasheet.
- In multi-slave setups, all peripherals must share the same mode.

MOSI

- Sends data from master to slave(s).
- Data is typically sent in bytes, using either LSB or MSB first.
- Multiple bytes can be sent, with implementation left to the user.
- CS may remain low between bytes.

MISO

- Sends data from slave(s) to the master.
- Often used as a response to commands or queries sent via MOSI.
- Can be omitted if not needed.

Communication Overview

1. The master pulls the CS line low.
2. The clock starts on SCLK.
3. Data is exchanged between the master and slave via MOSI and MISO.
4. Communication ends when the master pulls CS high and stops the clock.

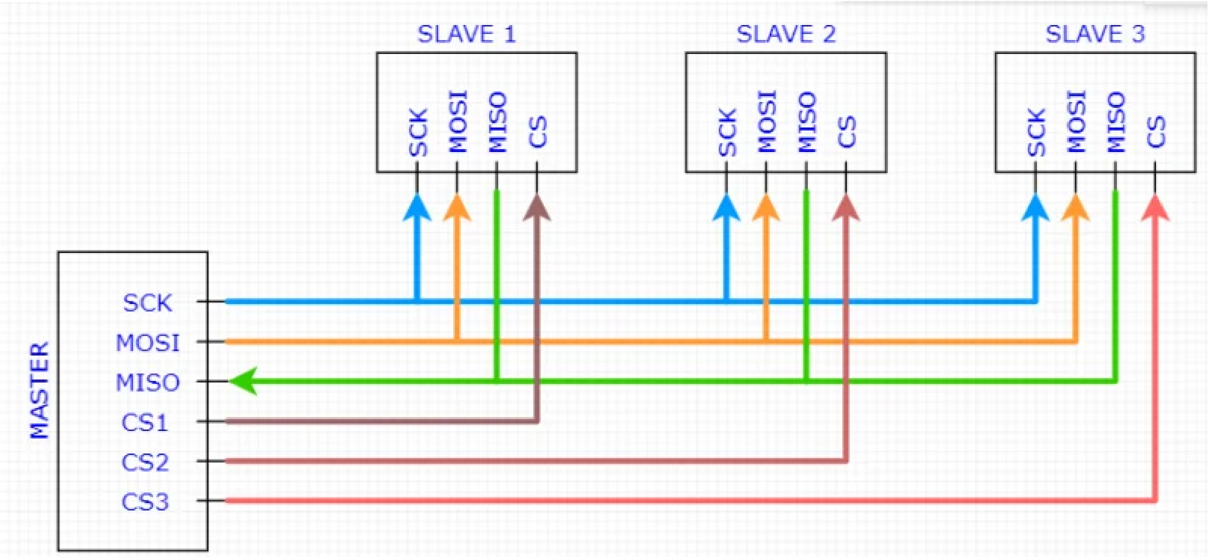
Because the “data frame” is determined by the CS line, this allows **dynamic frame lengths**, as the SPI master can control the duration of the communication session by keeping the CS line low for as long as needed.

Multi-Slave Configuration

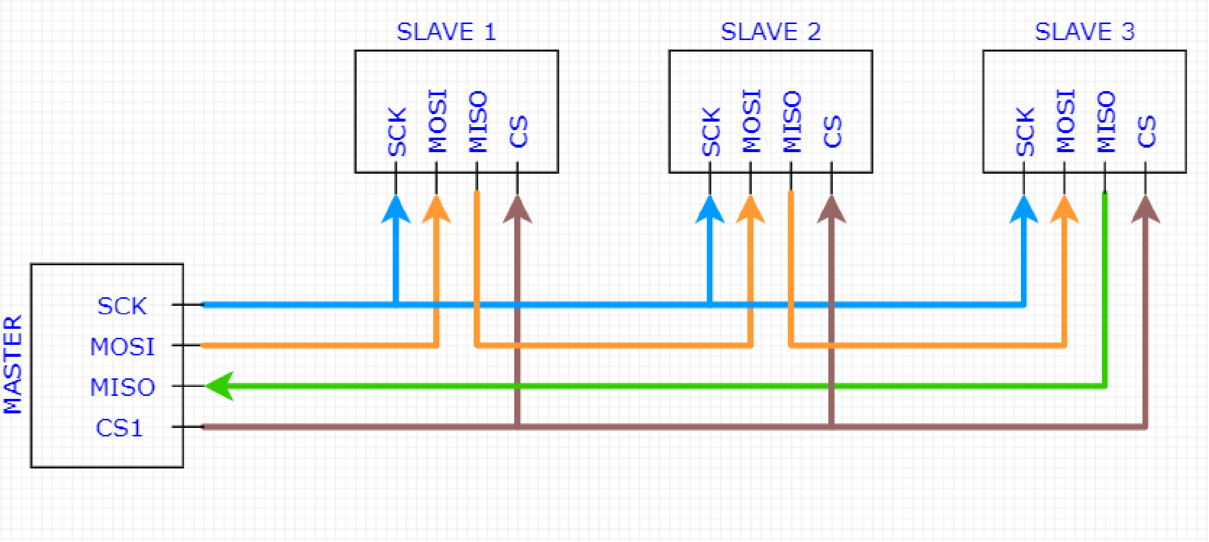
There are two ways to configure a multi-slave setup:

1. **Independent Slaves**
    - Each slave has its own separate CS (Chip Select) line, while the other lines (MOSI, MISO, and SCLK) are shared among all slaves.
    - This configuration is simple to set up but doesn't scale well for setups with many peripherals, as each new slave requires an

additional CS line.



2. **Cooperative Slaves (Daisy Chain)**
   - All slaves share the clock (SCLK) and CS lines.
   - Each slave's MISO (output) connects to the next slave's MOSI (input), and the MISO of the last slave routes back to the master.
   - Data is shifted through the slaves in a chain, with the master sending data to the first slave, which passes it along to the next, allowing sequential communication through the chain.
   - This configuration is more scalable than independent slaves and requires only 4 wires per peripheral.


SLAVE 1
SLAVE 2
SLAVE 3
SCK
MOSI
MISO
CS
MASTER
SCK
MOSI
MISO
CS1