

Embedded Systems Essentials with Arm: Get Practical with Hardware

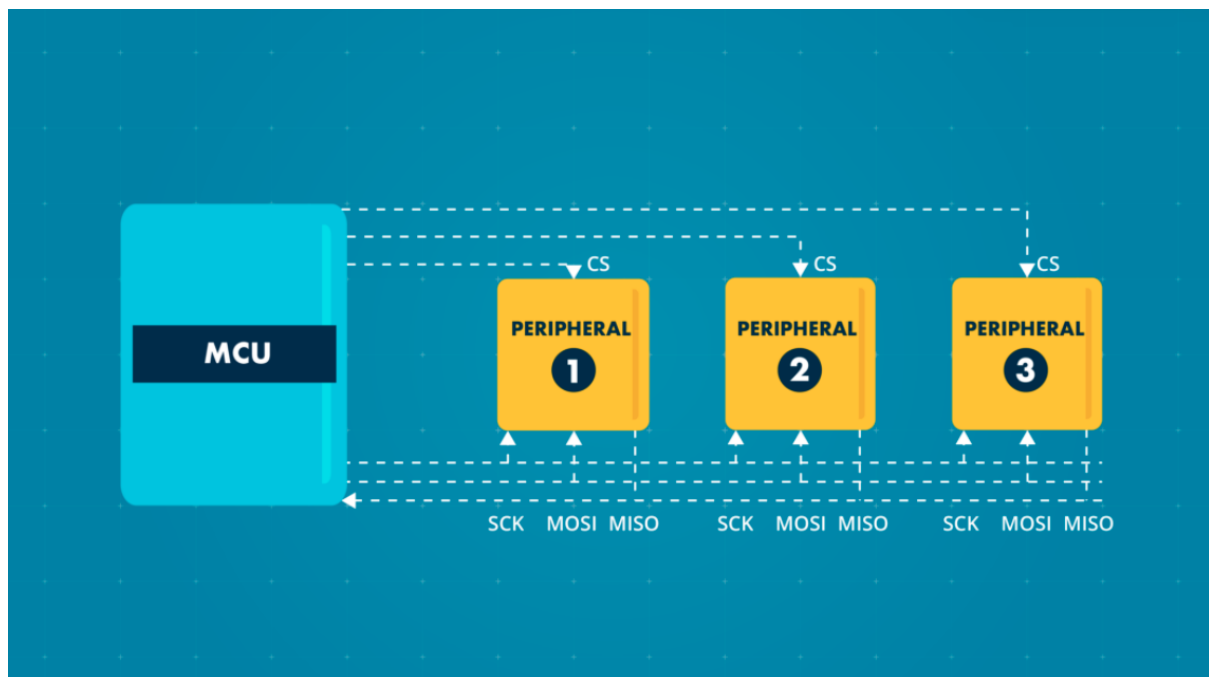
Module 1

KV2: Serial Peripheral Interface (SPI)

In this video, we're going to look at synchronous serial data communication. SPI (Serial Peripheral Interface) involves two shift registers working together. The first shift register is known as the master, and the second shift register is labelled slave. The two nodes can both receive and transmit data, but the master is more powerful because it controls the clock. When the clock is ticking, data moves. When the clock is not ticking, nothing moves. The output of one shift register is connected to the input of the other, and vice versa. That one clock, located in the master, controls both shift registers. If the clock ticks 8 times, 8 bits are moved from the master to the slave, and at the same time 8 bits are moved from the slave to the master.

The clock is labelled SCK (standing for serial clock), and the two data connections are MOSI (Master Out, Slave In) or MISO (Master In, Slave Out). The SPI Master can create a Chip Select line. This allows the SPI master to select or enable the Slave through the CS wire. This makes it possible to have multiple slaves connected to the SPI master.

That's exactly what we see in this diagram.



We have the interconnect lines, i.e. the serial clock, MOSI, and MISO. Those wires go to each of the peripherals, each acting as a slave, and the MCU (microcontroller unit), acting as the master. The master controls the CS lines and it selects one peripheral (one slave) at a time.

This then gives a 1-to-1 relationship between the master and that particular slave. The master can then choose to communicate with a different slave.

The advantages of SPI are:

- SPI is simple.
- It works well at transmitting data.
- It was created more than 40 years ago and people are still using it today and creating devices with SPI capability.

The downsides of SPI are:

- The master can send data, but there's no acknowledgement from the receiver. There is no way of being certain that data has been received correctly.
- There's no addressing. In a complicated system, you have to add a new CS line for every new node you introduce. This increases the complexity of the system and cancels out the advantage of SPI being a simple protocol. Remember that the advantage of serial was simplicity.
- There's no error-checking mechanism. You have no way of knowing if the data you send is corrupted by some form of interference.

Overall, SPI is a simple, convenient and low-cost option, but is not appropriate for complex or high-reliability systems.

Visit arm.com for more information.