Serial Communication

Making Wonders

By Vincent Bett, ECE '19

Introduction
Serial communication is the process of sending data one bit at a time over a computer bus.

It is the most widely used approach when transferring information between a data processing equipment and peripherals.

Protocols govern the execution of serial data transfer. A protocol is a secure and reliable form of communication that has a set of rules that are followed by both the sender and the receiver. [1]

A sender is the source device that encodes and transmits information while a receiver is the device that receives and decodes any incoming data stream.

Data can be transmitted either synchronously or asynchronously using a serial communication protocol. Synchronous transmission involves transmitting the data in a continuous stream at a constant rate with the use of a clock signal while asynchronous communication involves transmitting data without the use of a synchronization clock signal. Instead the synchronization information is contained in the data stream itself.

Synchronous serial communication
A Synchronous serial communication interface involves using a clock to make sure that the correct data is received by the receiver. To achieve this, the clocks in the transmitter and the receiver are synchronized to run at the same rate. This allows the receiver to sample the incoming signals at the same time intervals used by the transmitter.

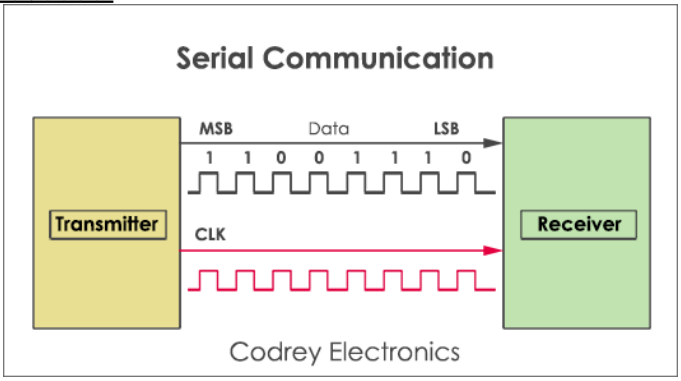

Figure 1: Synchronous serial communication. [2]

In the figure 1 above, the transmitter sends 1 bit every clock pulse and at the same time while the receiver receives 1 bit every clock cycle.

Protocols
The main Synchronous communication protocols include I2C and SPI.

Inter-Integrated Circuit (I2C)
I2C also known as Two Wire Interface uses a bus configuration to allow two connections between the devices. A serial clock connection for the controller to send a clock signal and a Serial Data connection for information exchange.

Serial Peripheral Interface(SPI)
SPI involves connecting the communicating devices with 4 wires.

- A serial data in on which the controller sends data to the receiver.
- A serial data out on which the peripheral device sends data to the controller.
- A clock connection for the controller to synchronize with the receiver and
- A chip select connection that the controller uses to tell the receiver to either listen for or ignore incoming data streams.

Asynchronous serial communication
Asynchronous serial communication does not involve a synchronization clock. Instead, this information is stored in the data stream itself in the form of start and end signals. The data to be transmitted is located between these two signals. The start signal tells the receiver to start receiving information while the end signal tells the receiver to stop receiving and to go back to waiting mode.

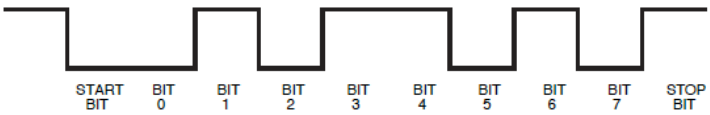

Figure 2: Asynchronous data transmission. [3]

Figure 2 shows how a transmitter transmits 8 bits of information.

The speed at which the data is transmitted is governed by the baud rate which is the speed at which the data is transmitted measured in bits per second. Both the sender and the receiver have to have the same baud rate in order for the data to be valid.

Protocols
Some asynchronous serial communication protocols are RS 232, RS 422, RS 485 where RS stands for Recommended Standard.

These protocols contain the rules governing the nature of signals when transmitting data between a transmitter and a receiver. Some of these rules include the signal voltage levels, the impedance of the wire used to transfer the data and the speed at which the bits are transferred (Baud rate).

The different protocols each have different implementations that make them incompatible with one another.

To resolve this, engineers have developed the Universal Serial Bus interface; a set of interface specifications that would remove the need for the different implementations.

Universal Serial Bus (USB)
USB is a set of interface specifications for high speed wired data transfer between computers and peripherals.

This new interface eliminates the need for different connectors for different different devices, supports various data types from mouse movements, compressed audio to video. It also allows for hot swapping which means that the peripheral device can be plugged and unplugged without rebooting the computer or turning off the device. The interface also provides power to the peripheral devices removing the need for a different power source. [4]

Implementation
We used the USB interface to implement a two-way communication channel between the BBC's Micro:bit [6] and an Apple MacBook computer.

The first half of the communication channel involved transmission of information from the Micro:bit to the computer. The interface would allow for transfer of control signals from the Micro:bit to the computer. These signals would then be decoded and used to trigger actions on the Cue robot.

The second half involved transmission of information from the computer to the Micro:bit. The computer would send back data that from the Cue sensors to be displayed on the Micro:bit .

Conclusion
Serial communication is frequently used to transmit data between devices at short distances.

This mode of communication is robust due to its built in error detection and correction capabilities. It therefore allows for a reliable method of communication.

References
[1] Codrey, "What is serial communication and how it works." [Online]. Available: https://www.codrey.com/embedded-systems/serial-communication-basics/

[2] Codrey, "What is serial communication and how it works." [Online]. Available: https://www.codrey.com/embedded-systems/serial-communication-basics/

[3] Embedded lab, "Synchronous serial transmission." [Online]. Available: http://embedded-lab.com/blog/lab-8-asynchronous-serial-communication/

[4] EEHerald, “USB interface tutorial.” [Online]. Available: http://www.eeherald.com/section/design-guide/esmod14.html

[5] USB Organization, “USB interface specs.” [Online]. Available: https://www.usb.org/documents

[6] Ashish Neupane: “Senior Capstone Tech Note”