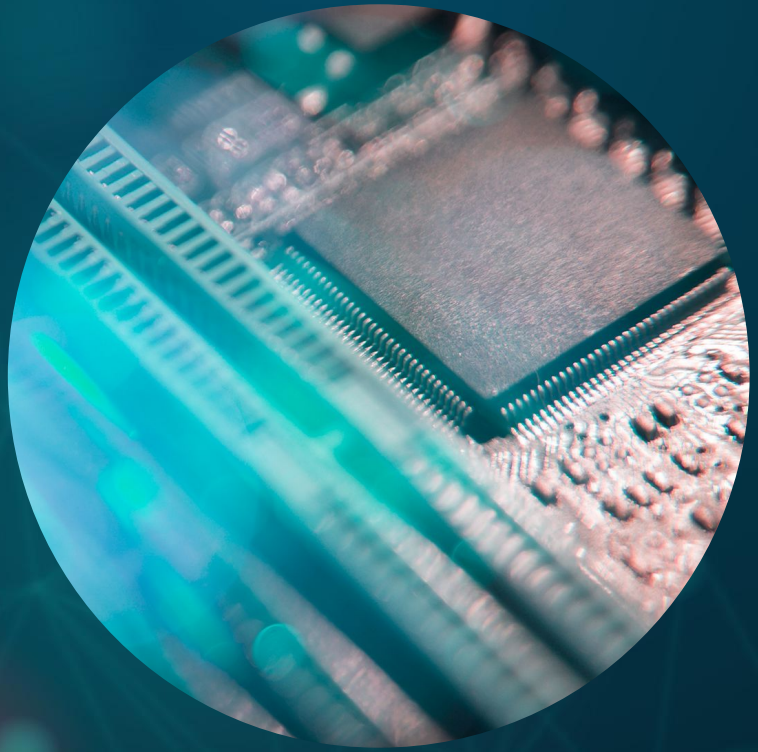
ARM Based Microcontroller

UART

Lecture 13



Advanced RISC Machines

IMT SCHOOL

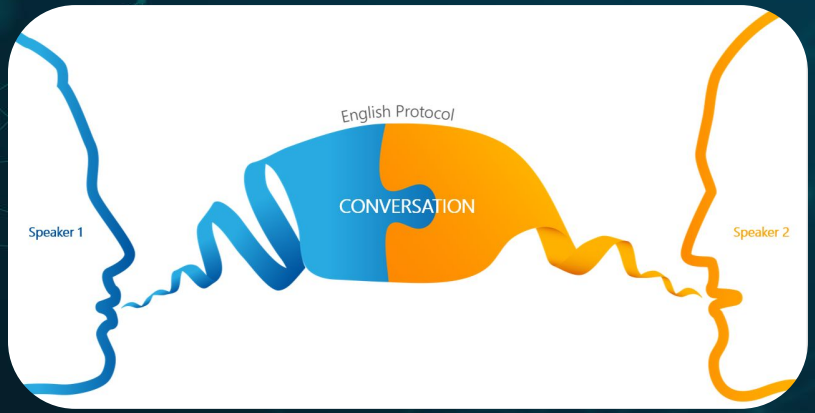
Introduction 01

Universal Asynchronous Rx, Tx

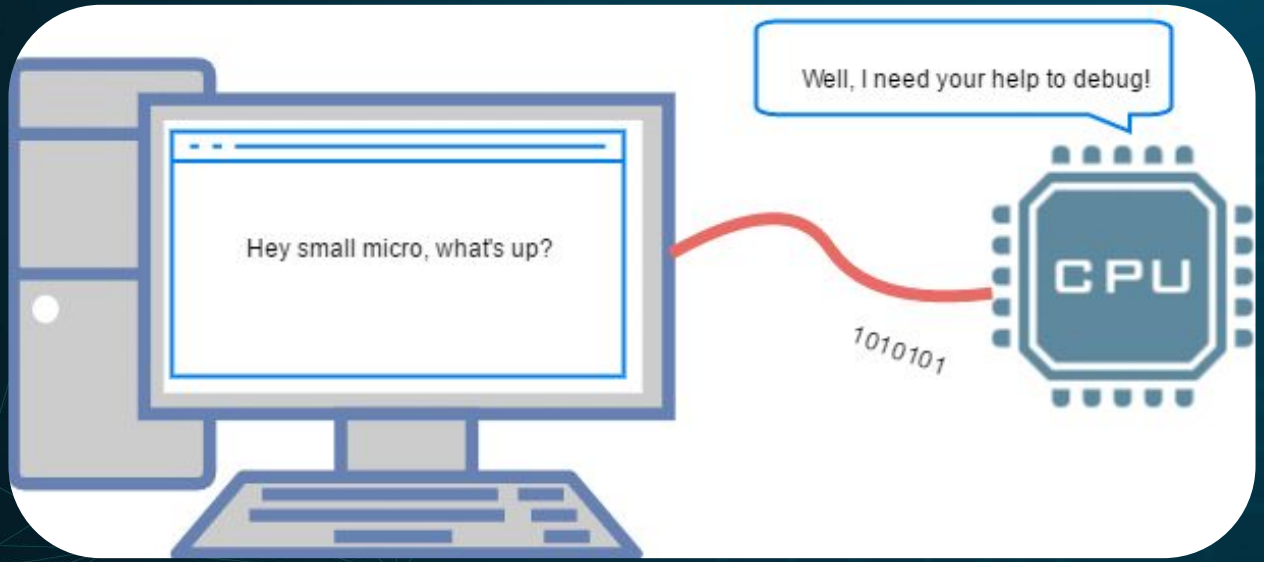
IMT SCHOOL

What is a communication Protocol ?

A communication protocol is a system of rules that allows two or more entities of a communications system to transmit information via any variation of a physical quantity. The protocol defines the rules, syntax, semantics and synchronization of communication and possible error recovery methods. Protocols may be implemented by hardware, software, or a combination of both.

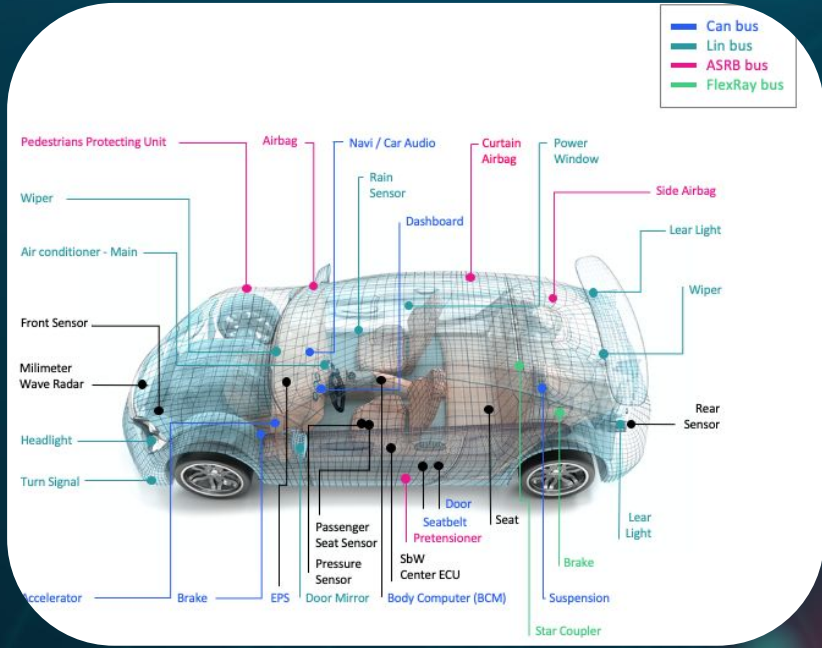


What is a communication Protocol ?



What is a communication Protocol ?

In Most of the complex systems, the functionality is divided into subsystems, each subsystem is an embedded system with microcontroller and it is called ECU (Electronic Control Unit). These ECUs need to share the data between each other, i.e. they need to communicate with each other !



Communication Protocol

Need for communications in Embedded Systems:

- Exchanging data between different subsystems within the same system.
- Reduce the complexity of a system by splitting it into different subsystems.
- transfer the data on different distances and on different mediums.

Standard Protocol:

A protocol is a defined method of communication by defining two main aspects:
Hardware Interface: This activity defines the hardware connections (wires) between the communicating nodes (ECUs)
Data Frame Format: This activity defines the data frame transmitted of the wires between the nodes including the number of bits and arrangement.

Communication Protocol

CLASSIFICATION

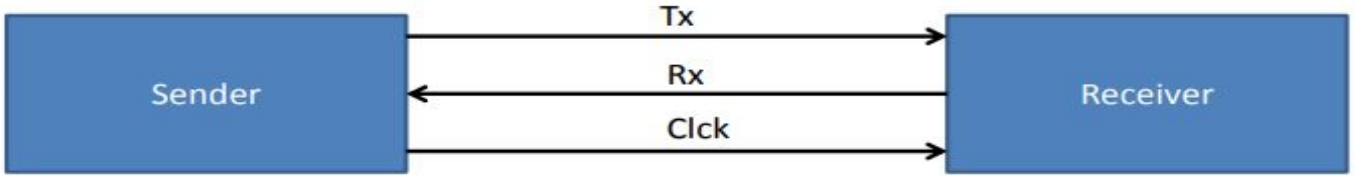
Modes of Data Transfer can be broadly divided into two types :

Wired

Wireless

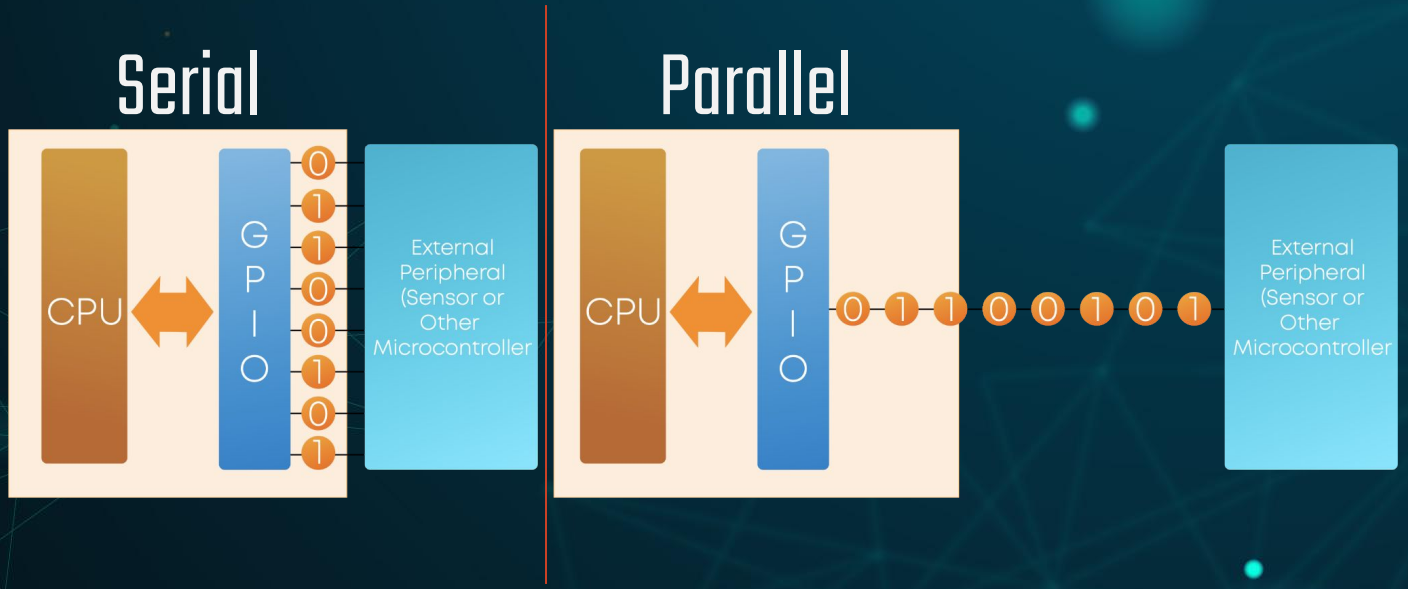
Communication Protocol Classification

- **Synchronous serial communication**
  - Synchronous serial communication describes a serial communication protocol in which data is sent in a continuous stream at a constant rate.
  - Synchronous communication requires that the clocks in the transmitting and receiving devices are synchronized – running at the same rate – so the receiver can sample the signal at the same time intervals used by the transmitter so that there is an extra wire for clock carrying.



Communication Protocol Classification

Modes of Data Transfer can also be divided into



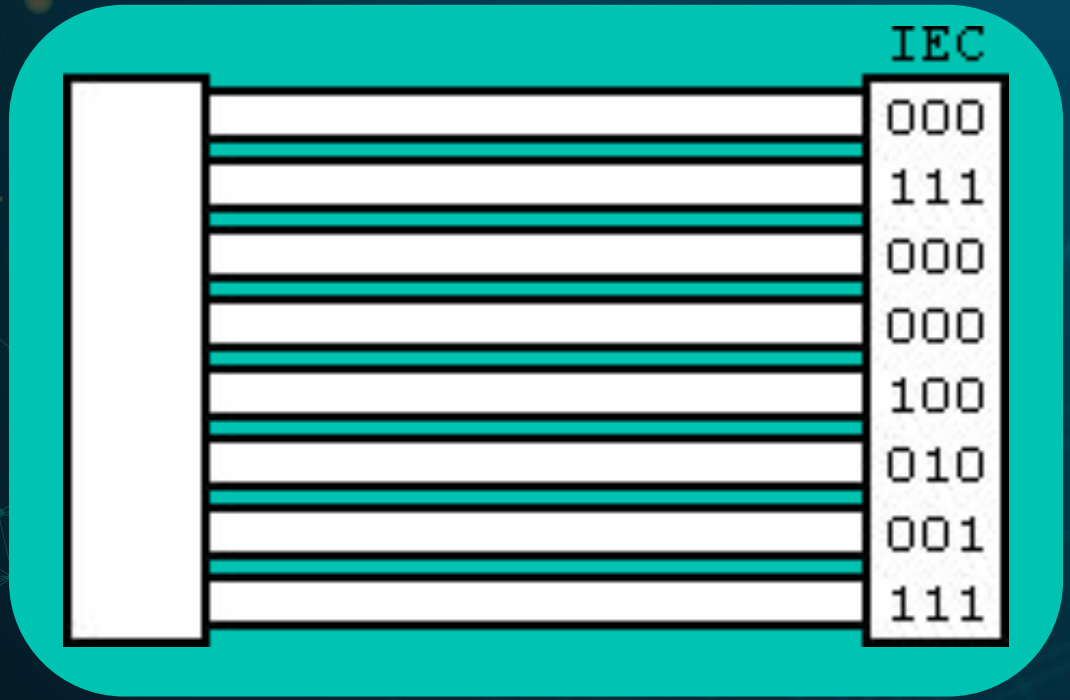
Communication Protocol

Parallel Communication

In data transmission, parallel communication is a method of conveying multiple binary digits (bits) simultaneously. It contrasts with serial communication, which conveys only a single bit at a time; this distinction is one way of characterizing a communications link.

The basic difference between a parallel and a serial communication channel is the number of electrical conductors used at the physical layer to convey bits. Parallel communication implies more than one such conductor. For example, an 8-bit parallel channel will convey eight bits (or a byte) simultaneously, whereas a serial channel would convey those same bits sequentially, one at a time. If both channels operated at the same clock speed, the parallel channel would be eight times faster. A parallel channel may have additional conductors for other signals, such as a clock signal to pace the flow of data, a signal to control the direction of data flow, and handshaking signals.

Parallel Communication



Parallel Communication Problems

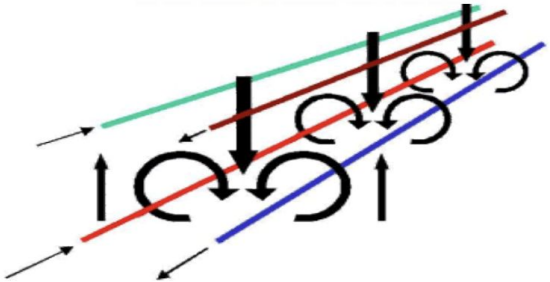
Previously, it was known that the parallel communication is faster than serial one, but as the data rates increased, Some problems raised with the parallel communications that made a limit for the speed.

Parallel Communication Problems

1- Complex connections

2- Cross talk

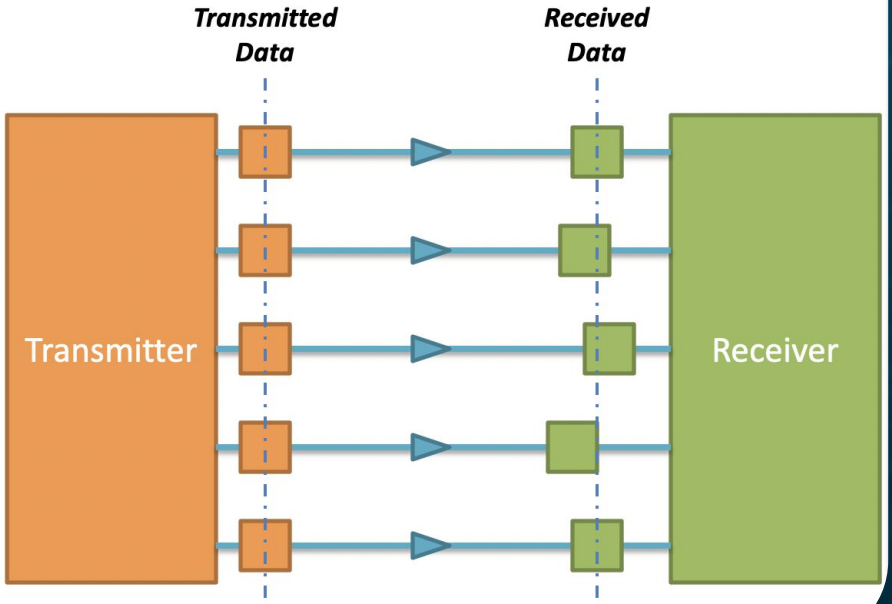
Cross Talk is the phenomenon where a wire that having electrical signal passing through it generates a magnetic field that affects other wires on the same cable !



Parallel Communication Problems

3- Timing Skew

Timing Skew (Data Skew) is a phenomenon where the data transmitted on the parallel cables arrives at different time which lead to wrong data sampling. The skew is directly proportional with the data transmission speed, which means increasing data rate increases the skew !



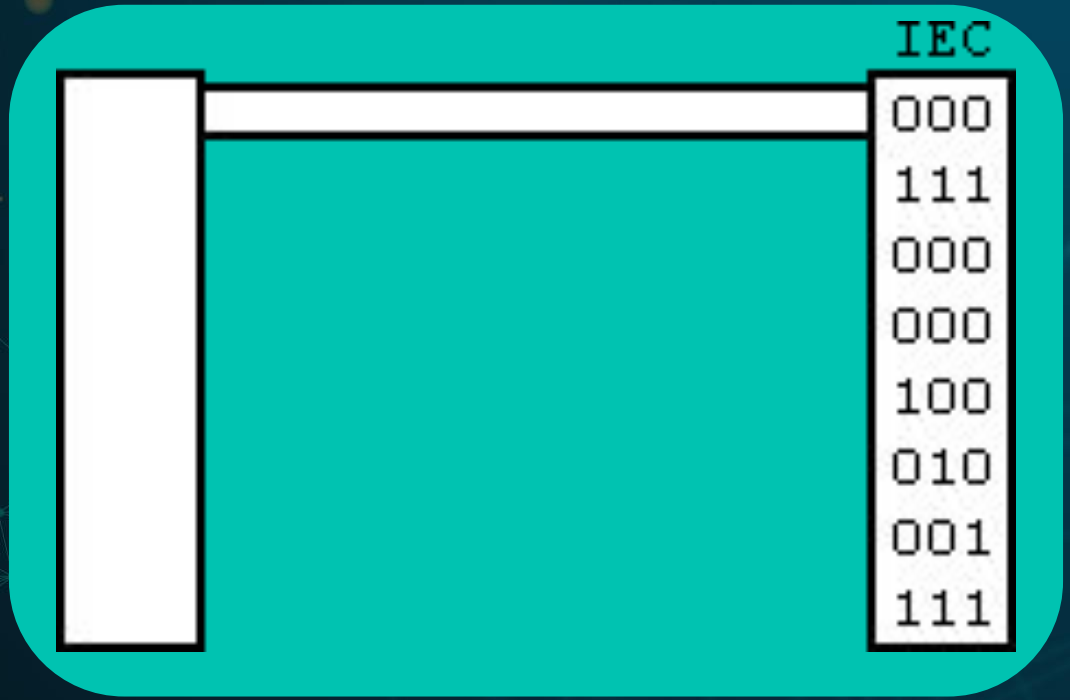
Communication Protocol

Serial Communication

In telecommunication and data transmission, serial communication is the process of sending data one bit at a time, sequentially, over a communication channel or computer bus. This is in contrast to parallel communication, where several bits are sent as a whole, on a link with several parallel channels.

Serial communication is used for all long-haul communication and most computer networks, where the cost of cable and synchronization difficulties make parallel communication impractical. Serial computer buses are becoming more common even at shorter distances, as improved signal integrity and transmission speeds in newer serial technologies have begun to outweigh the parallel bus's advantage of simplicity (no need for serializer and deserializer, or SerDes) and to outstrip its disadvantages (clock skew, interconnect density). The migration from PCI to PCI Express is an example.

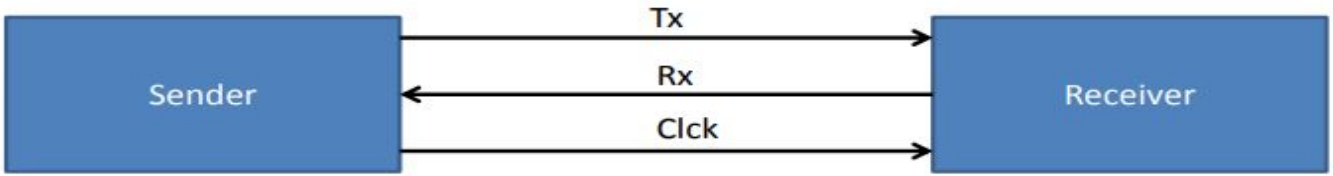
Serial Communication



Communication Protocol

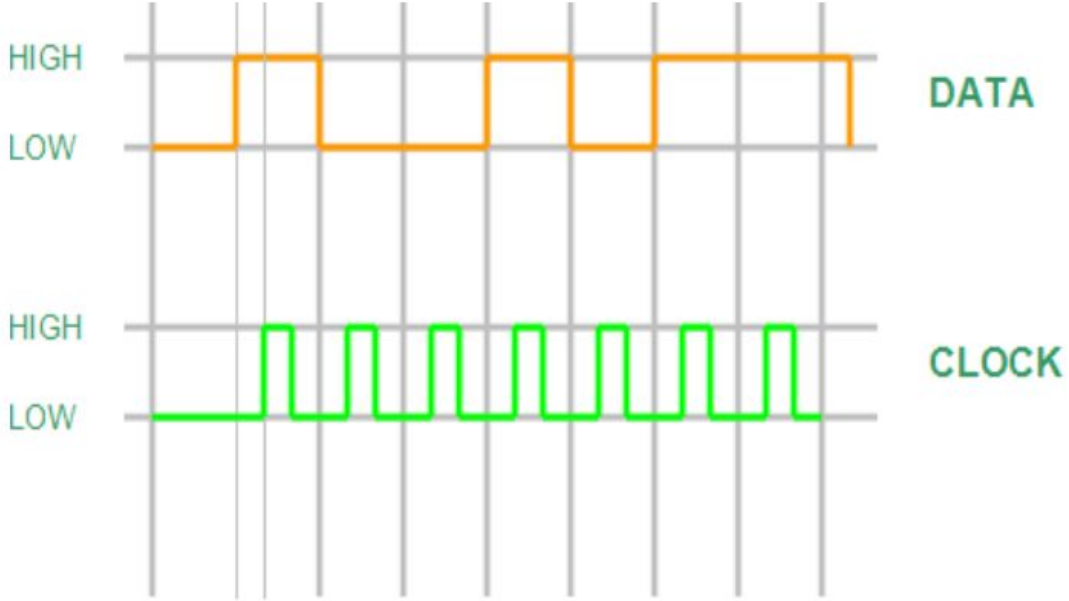
Synchronous serial communication

- Synchronous serial communication describes a serial communication protocol in which data is sent in a continuous stream at a constant rate.
- Synchronous communication requires that the clocks in the transmitting and receiving devices are synchronized – running at the same rate – so the receiver can sample the signal at the same time intervals used by the transmitter so that there is an extra wire for clock carrying.



Synchronous Serial Communication

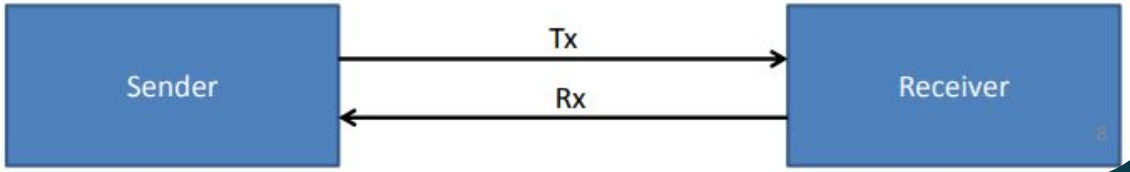
The diagram corresponds to the transfer of the data 10010111. It corresponds to the value of the data at every rising edge of the clock.



Communication Protocol

Asynchronous serial communication

- Asynchronous serial communication is a form serial communication in which the communicating endpoints' interfaces are not continuously synchronized by a common clock signal.
- Instead of a common synchronization signal, the data stream contains synchronization information in form of start and stop signals, before and after each unit of transmission, respectively. The start signal prepares the receiver for arrival of data and the stop signal resets its state to enable triggering of a new sequence.



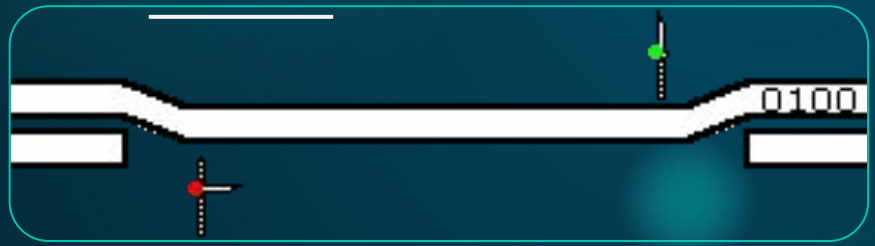
Communication Protocol Classification

simplex

In simplex operation, a network cable or communications channel can only send information in one direction; it's a “one-way street”. A good example would be your keyboard to your CPU. The CPU never needs to send characters to the keyboard but the keyboard always sends characters to the CPU. In many cases, Computers almost always send characters to printers, but printers usually never send characters to computers (there are exceptions, some printers do talk back). Simplex requires only one lane (in the case of serial).

Communication Protocol Classification

Half Duplex



Technologies that employ half-duplex operation are capable of sending information in both directions between two nodes, but only one direction or the other can be utilized at a time. This is a fairly common mode of operation when there is only a single network medium (cable, radio frequency and so forth) between devices. The only advantage that Half-Duplex would have is the single lane or single track is cheaper than the double lane or double track.

For example, in conventional Ethernet networks, any device can transmit, but only one may do so at a time.

Communication Protocol Classification

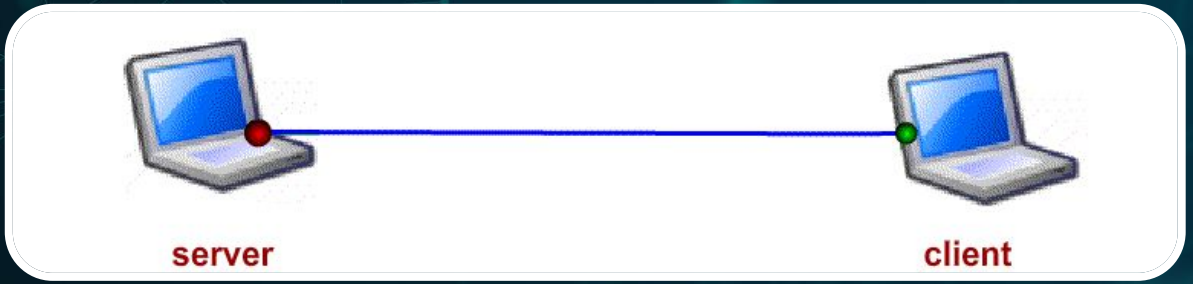
Full Duplex

In full-duplex operation, a connection between two devices is capable of sending data in both directions simultaneously.
Full-duplex channels can be constructed either as a pair of simplex links (as described above) or using one channel designed to permit bidirectional simultaneous transmissions. A full-duplex link can only connect two devices, so many such links are required if multiple devices are to be connected together.

Communication Protocol Classification

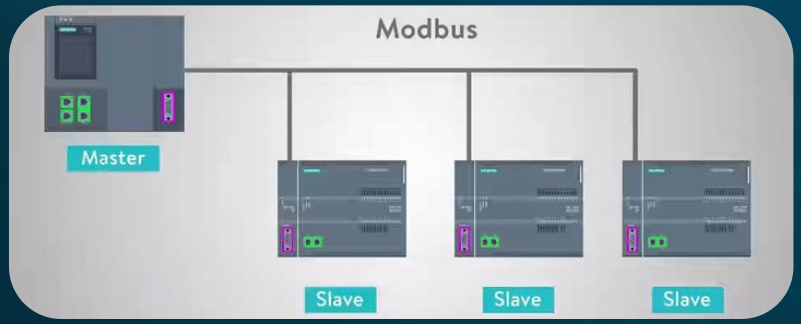
Pear to Pear

In this type of communication the communicating nodes can send to each other any time with no privileges.



Communication Protocol Classification

Master & Slave



In this type of communication there is a master node that can send data to any other nodes (Slaves). The master is the only node that can initiate the communication, the slave can never initiate the communication. The slave can send data to master only when the master permit the slave to send. The Master / Slave network can be divided to: Single Master Single Slave (SMSS) Single Master Multi Slave (SMMS) Multi Master Multi Slave (MMMS)

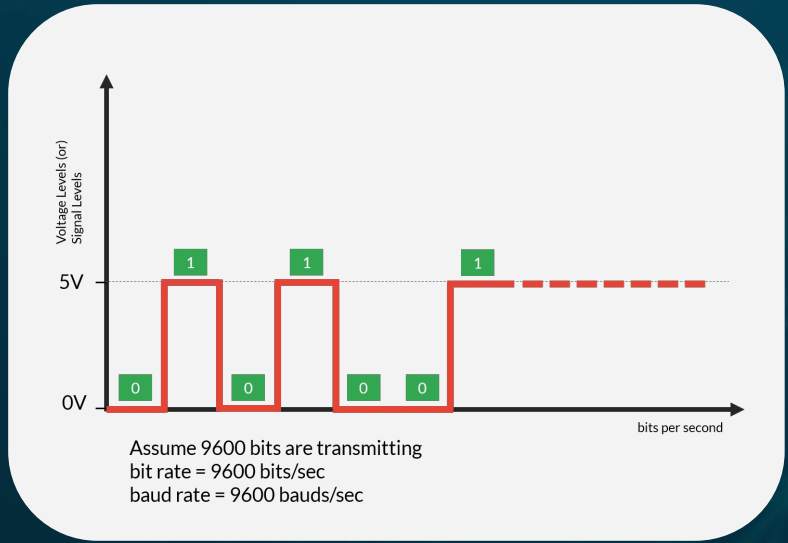
Communication Protocol Concepts

Bit rate:
Bit rate refers to Number of data bits transferred per second. Unit of Bit rate is bits per second(bps). For example bit rate is 10kbps means 10,000 bits are transferred within one second.

Baud rate:
Baud rate refers to number of signal or voltage level changes per second. Unit of baud rate is bauds per second. For example baud rate is 9600 means 9600 signal level changes are happening within a second.

Binary Signaling

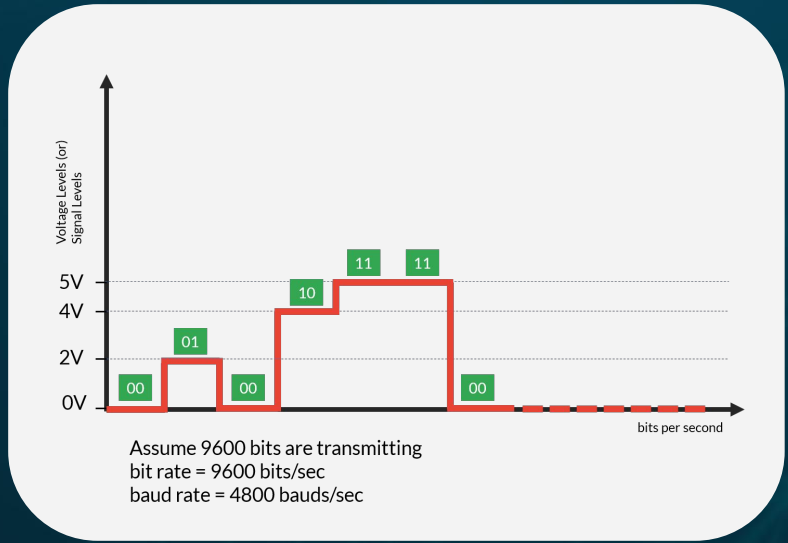
In binary signaling, we can represent two voltage or signal levels by using a single bit. Single bit has value either logic 0 or logic 1 where logic 0 represent '0' volt and logic 1 represent '5' volt. **In binary signaling, number of signal level changes are equal to number of bits transferred within a second because one bit at time represents either of two signal levels. That's why bit rate and baud rate are same in binary signaling.**



Multilevel Signaling

In multi-level signaling, there are more than two voltage or signal levels. To represent those signal levels, we require more than one bit. Number of bits required to represent voltage levels is obtained by using following formula.

N=log2(m)
where N=Number of bits required to represent signal or voltage levels.
m=number of signal or voltage levels.
Example: To represent 4 voltage levels you required at least 2 bits. Put m=4 in above formula , you will get N as 2.



Multilevel Signaling

Assume in the above figure, Number of bits transferred per second are 9600 and there are 4 voltage or signal levels[0V, 2V, 4V, 5V]. To represent those 4 voltage or signal levels you required at least 2 bits (using N=log2(m)). Assume 00 represents 0V, 01 represents 2V, 10 represents 4V and 11 represents 5V.

If you observed carefully in the above figure, the number of bits transferred per second are 9600 but the number of signal level changes per second are 4800 because 2 bits are representing one voltage level. That's why in multi level signaling baud rate is not the same as bit rate. In multi level signaling baud rate always depends on the signal or voltage levels. In the above case, the Baud rate is half of bit rate as two bits are required to represent one signal or voltage level.

USART 01

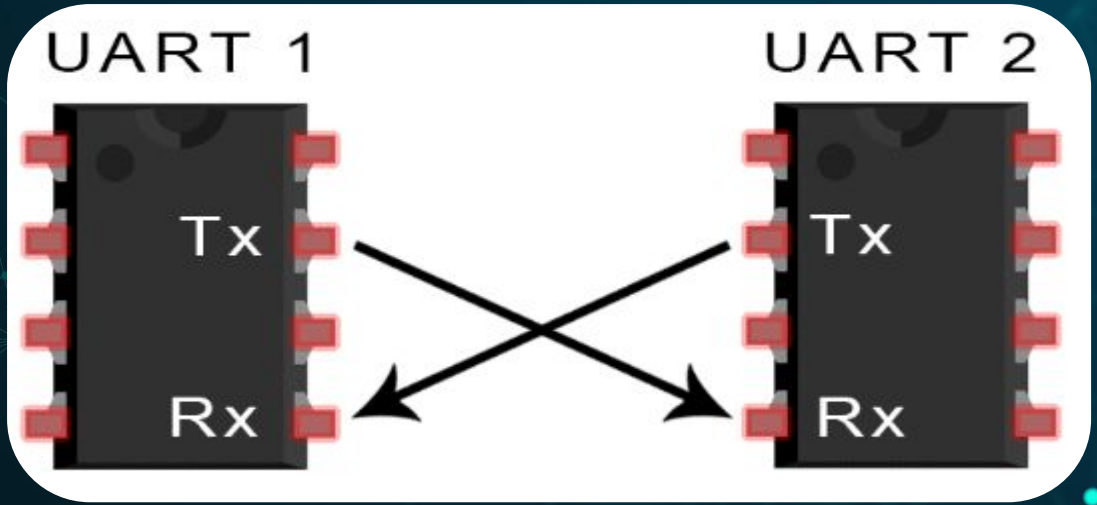
Universal Synchronous Asynchronous Rx, Tx

IMT SCHOOL

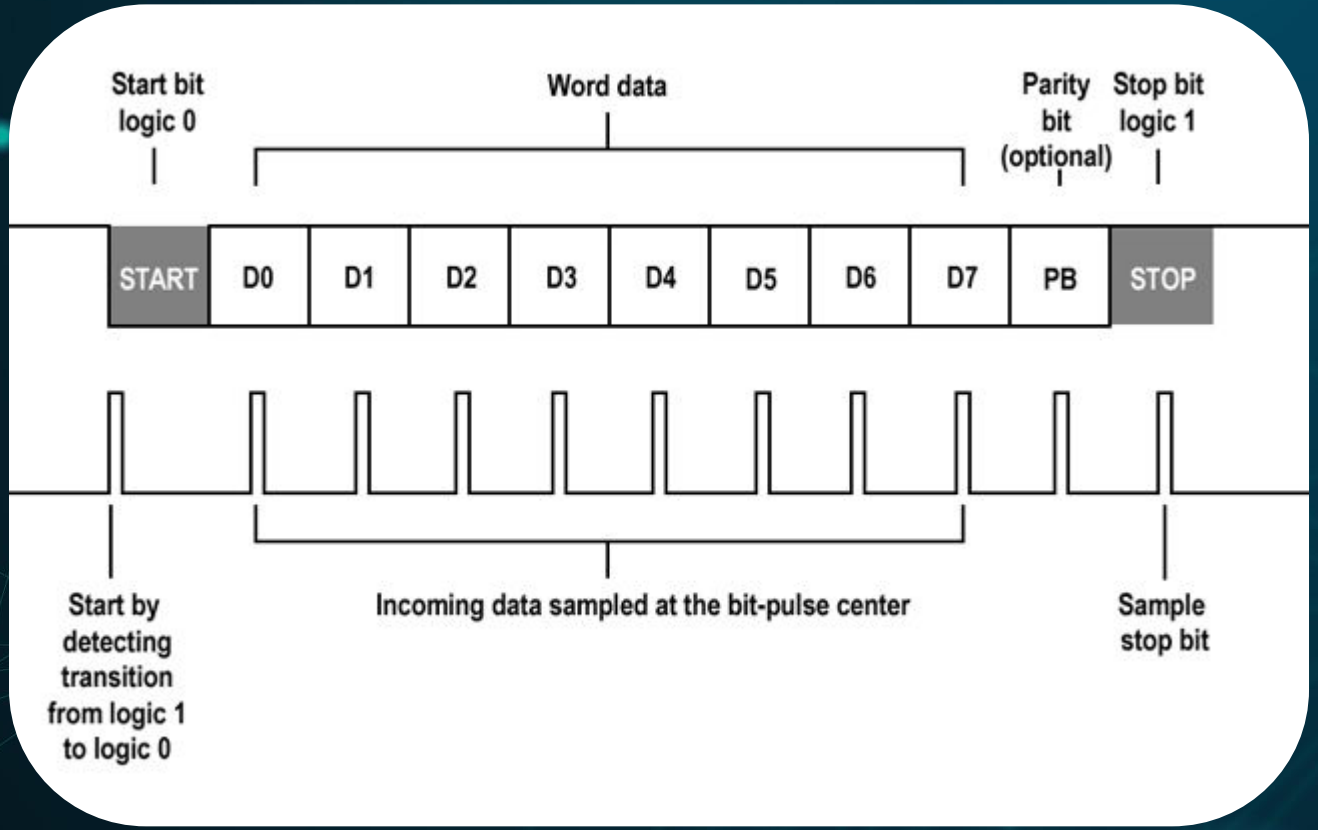
UART Features

What is UART

UART stands for Universal Asynchronous Receiver Transmitter. There is one wire for transmitting data, and one wire to receive data. A common parameter is the baud rate known as "bps" which stands for bits per second. If a transmitter is configured with 9600bps, then the receiver must be listening on the other end at the same speed.



UART Frame


Start bit logic 0
Word data
Parity bit (optional)
Stop bit logic 1
START
D0
D1
D2
D3
D4
D5
D6
D7
PB
STOP
Start by detecting transition from logic 1 to logic 0
Incoming data sampled at the bit-pulse center
Sample stop bit

UART Frame

- Start bit: 1 bit indicates the start of a new frame, always logic low.
- Data: 5 to 9 bits of sent data.
- Parity bit: 1 bit for error checking
    - Even parity: clear parity bit if number of 1s sent is even.
    - Odd parity: clear parity bit if number of 1s sent is odd.
- Stop bit: 1 or 2 bits indicate end of frame, always logic high.

What Will be the parity bit if the data =(10100110)

IMT SCHOOL

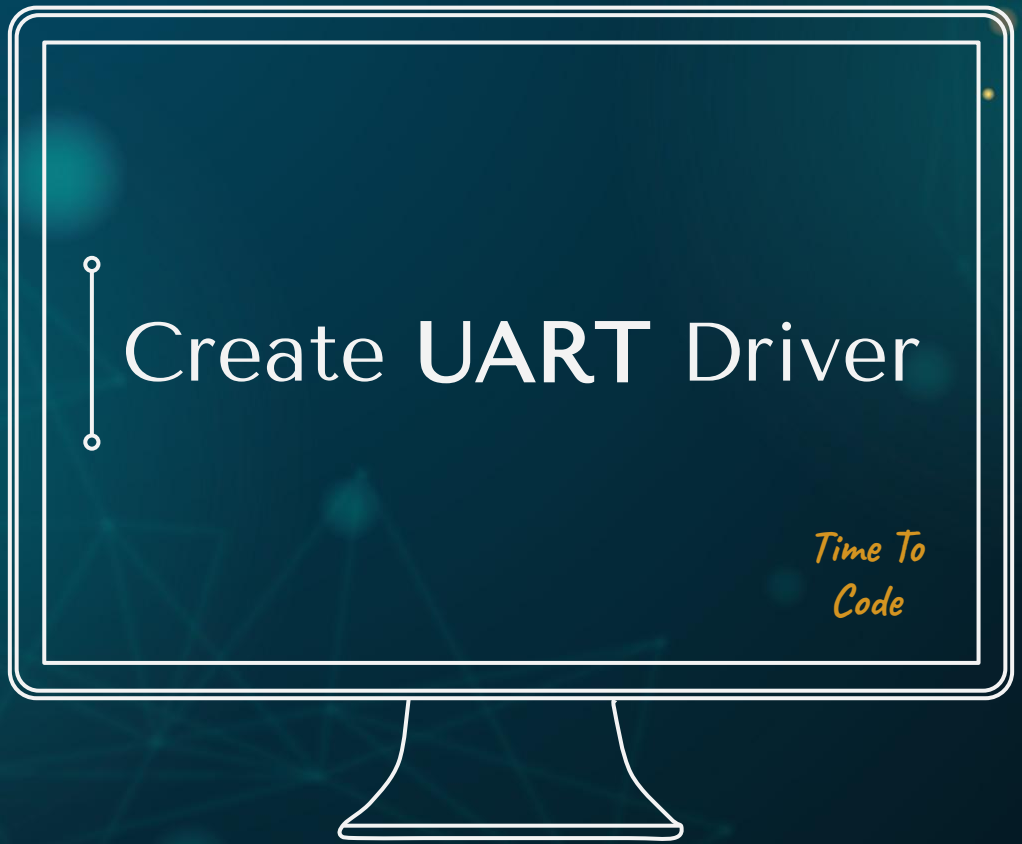
UART

This module converts USB port to a UART port. It helps connecting the microcontroller to your PC. Connect Rx
and Tx signals of the USB to TTL module with the UART port in your microcontroller.
Hint, Don’t forget connecting the common ground !
It is a mandatory for any communicating devices to have a common ground.
Use Putty software terminal to communicate with your microcontroller

IMT SCHOOL

STM32 Is AWESOME

Session LAb



IMT SCHOOL

THANKS!

Do you have any questions?

www.imtschool.com

www.facebook.com/imaketechnologyschool/