IMT SCHOOL

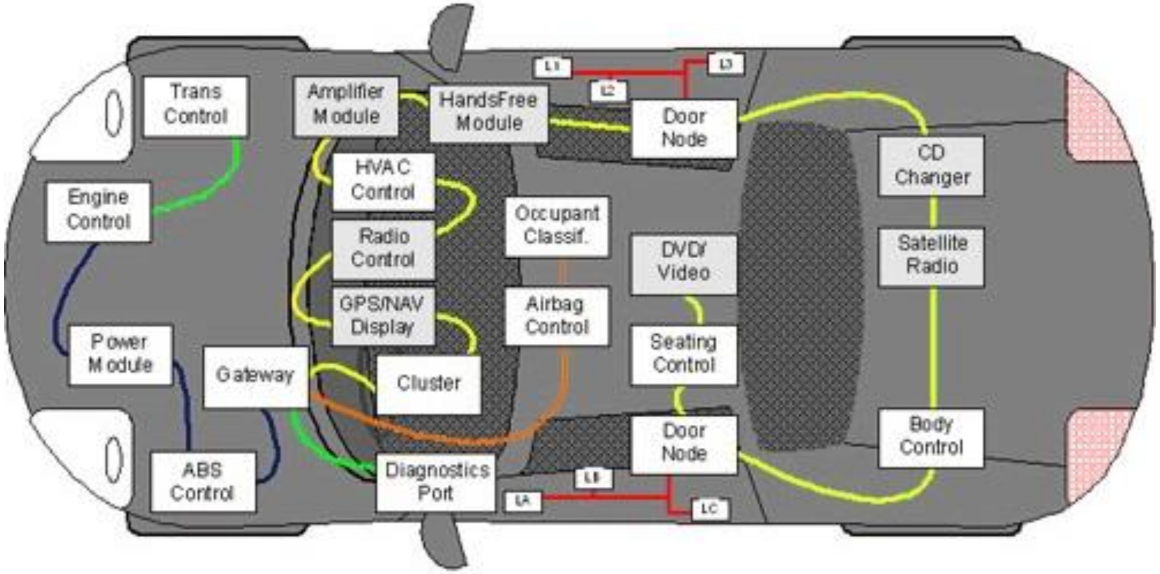
Embedded Systems Interfacing

Lecture thirteen

UART Serial Communication

Communication concept

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 concept

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 concept

Standard Protocol:

It is a fixed method of communication that is globally defined and documented. Any different systems that support this protocol shall easily communicate two each other without any constraints.

Communication protocols examples in embedded systems

- **UART** : Universal Asynchronous Receiver Transmitter
- **SPI** : Serial Peripheral Interface
- **I2C** : Inter-Integrated Circuit
- **CAN** : Controller Area Network
- **LIN** : Local Interconnect Network

Communication Protocols Specifications

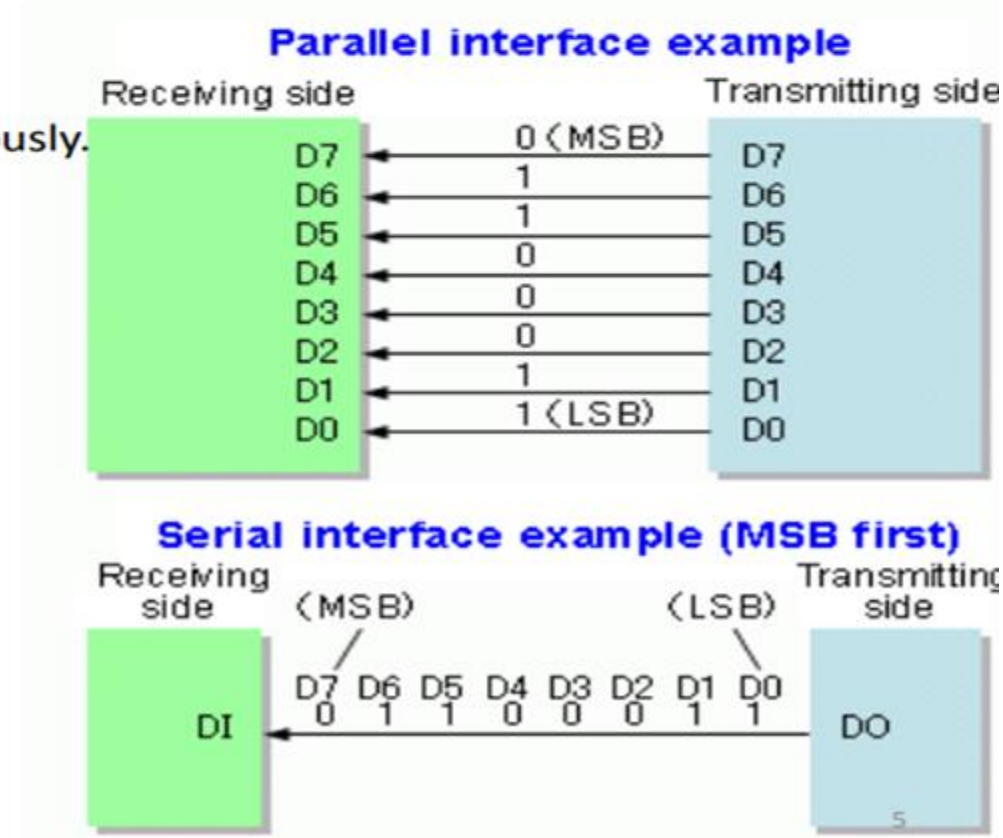
Serial Vs Parallel

- **Parallel Communication**

  Sending multiple bits simultaneously.

- **Serial communication**

  Sending bit by bit.



Serial Vs Parallel

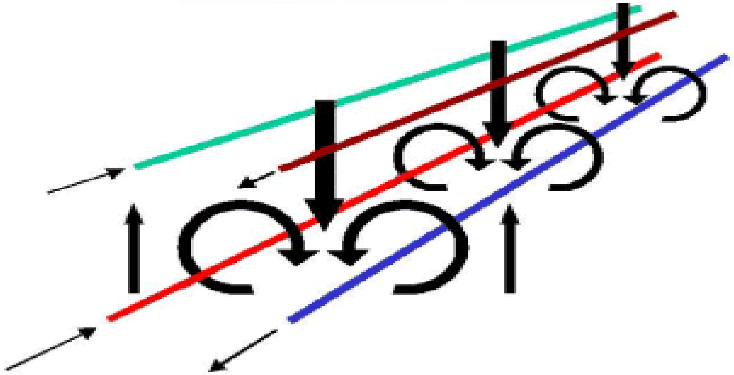
Previously, it was known that the parallel communication is faster than serial one, but as the data rates increased, Some problems raised with the parallel communications that made a limit for the speed.

Parallel Communication Problems

1- Complex connections

2- Cross talk

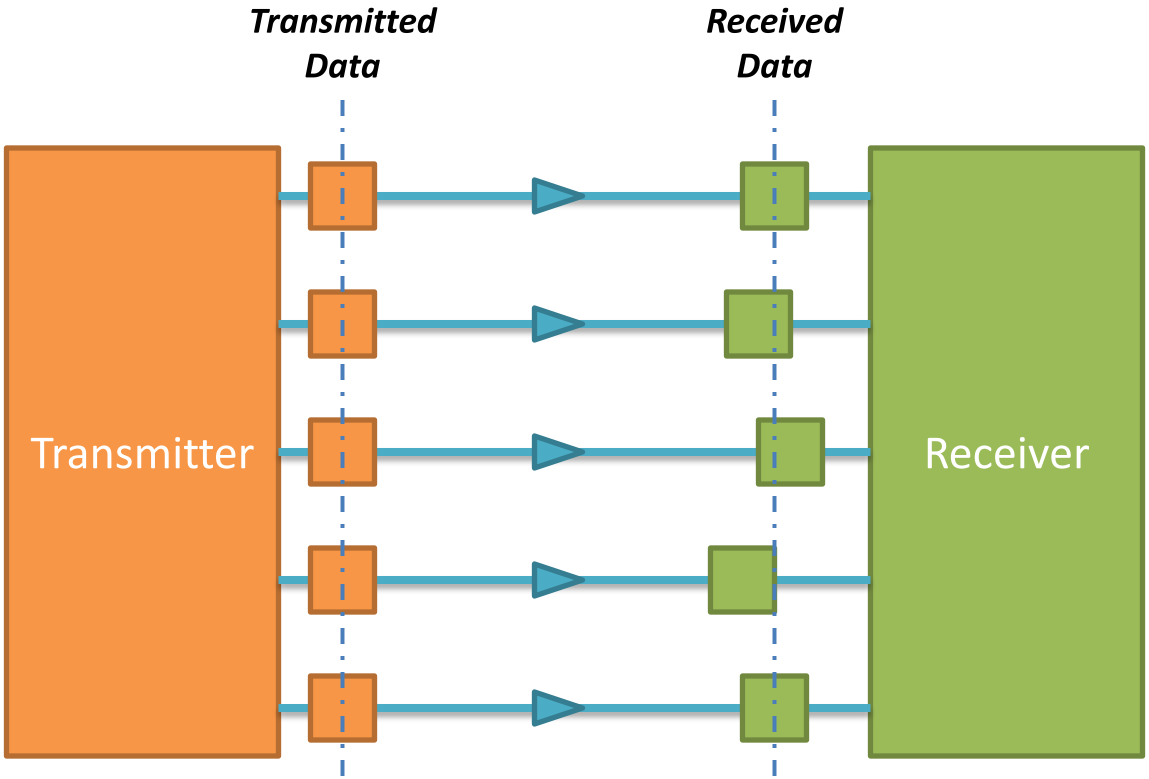
Cross Talk is the phenomenon where a wire that having electrical signal passing through it generates a magnetic field that affects other wires on the same cable !



3- Timing Skew

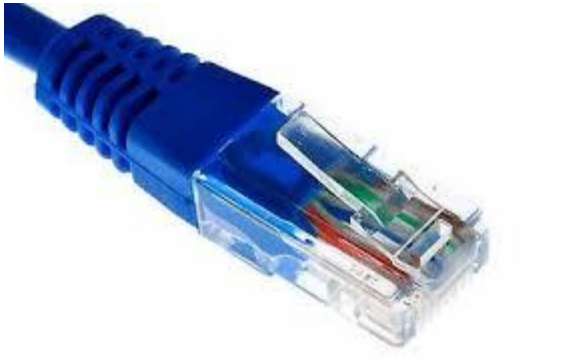
Timing Skew (Data Skew) is a phenomenon where the data transmitted on the parallel cables arrives at different time which lead to wrong data sampling. The skew is directly proportional with the data transmission speed, which means increasing data rate increases the skew !

`



Serial Vs Parallel

Because of the previous problems, the parallel communications comes to an end ! The serial communication doesn't have these problems, So higher data rates have been achieved. And now all modern communication protocols are serial like: UART, SPI, I2C, CAN, LIN, Ethernet, USB, …. Etc.

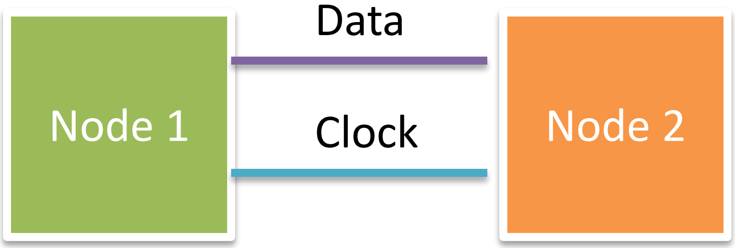


Communication Protocols Specifications

Synchronous Vs Asynchronous

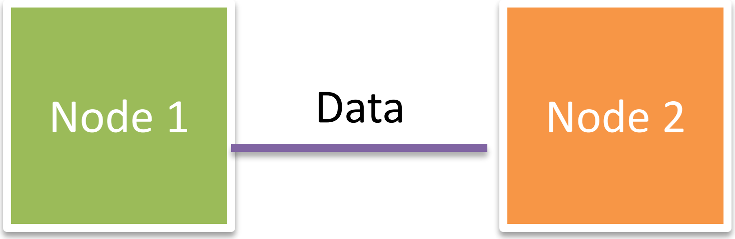
- ***Synchronous Communication***

  It is a type of communication the nodes of the network share the same clock.



- ***Synchronous Communication***

  It is a type of communication that doesn’t have a shared clock between the communicating nodes, instead the nodes are configured with the communicating rate and each node is having it own clock generator system that generates this clock.



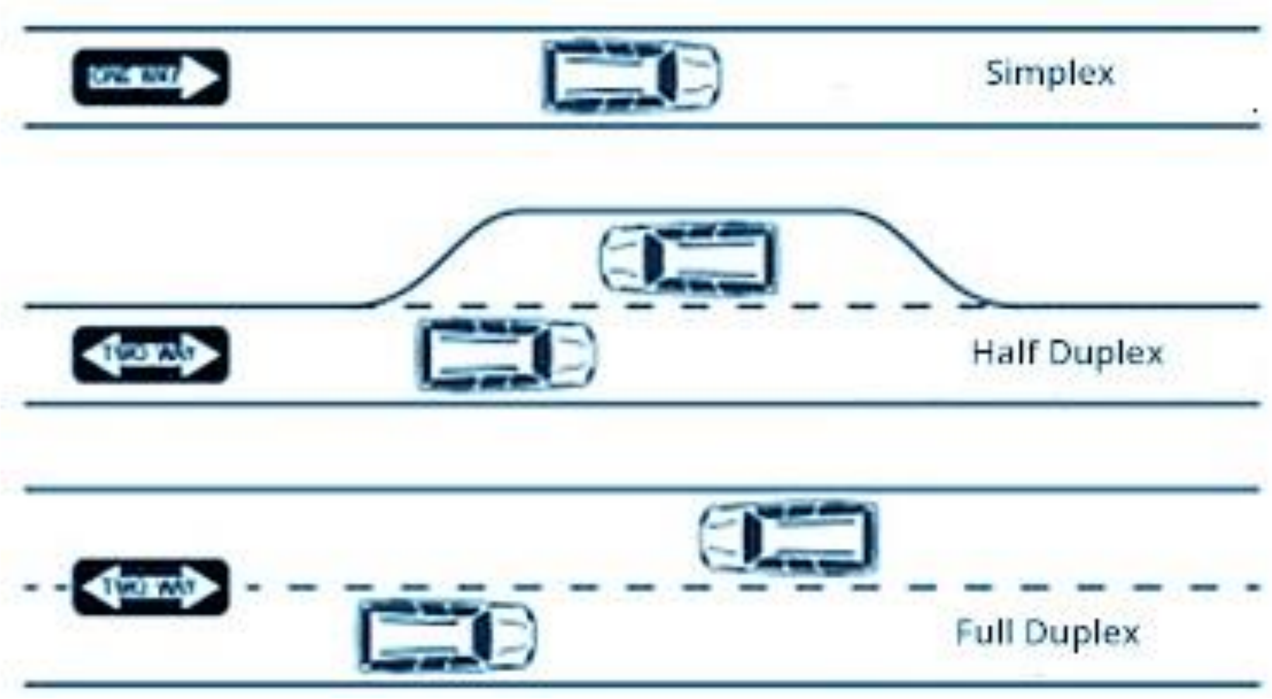
Communication Protocols Specifications

Simplex Vs *half duplex* Vs *Full duplex*

Simplex Channel: The data is in one direction ,from a transmitter to receiver

Half duplex channel: The data is bidirectional, each node can transmit and receive but not in the same time !

Full duplex channel: The data is bidirectional and each node can transmit and receive at the same time.



Communication Protocols Specifications

Master Salve Vs Peer to Peer

❖ ***Peer To Peer communication***

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

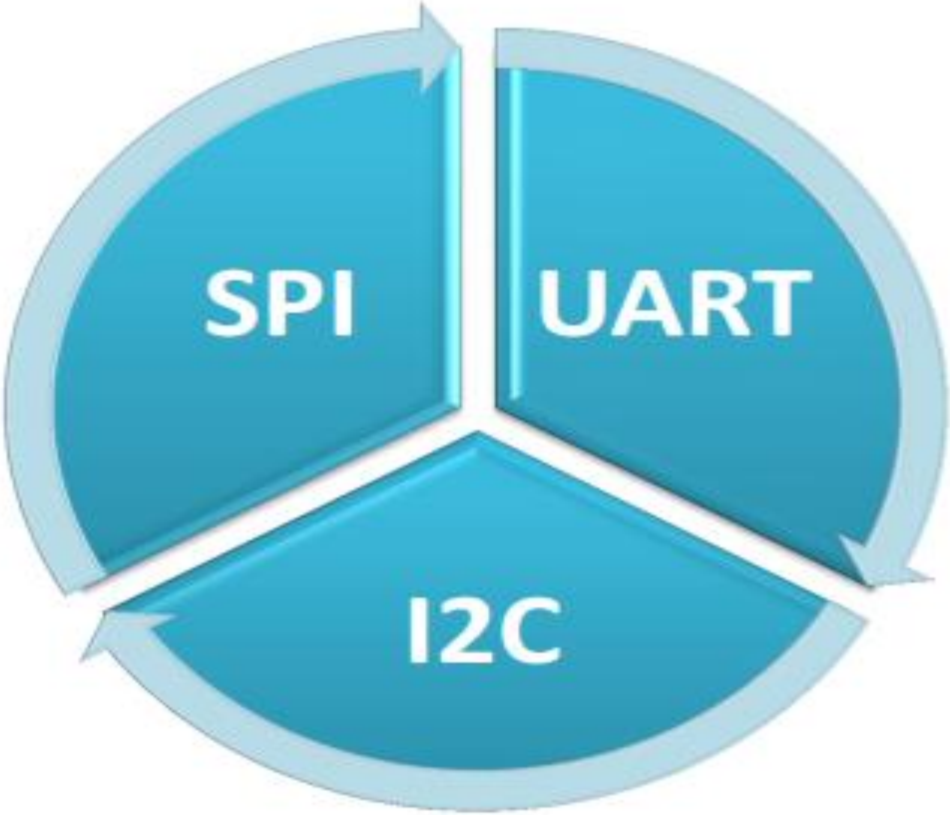
❖ ***Master Slave Communication***

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.

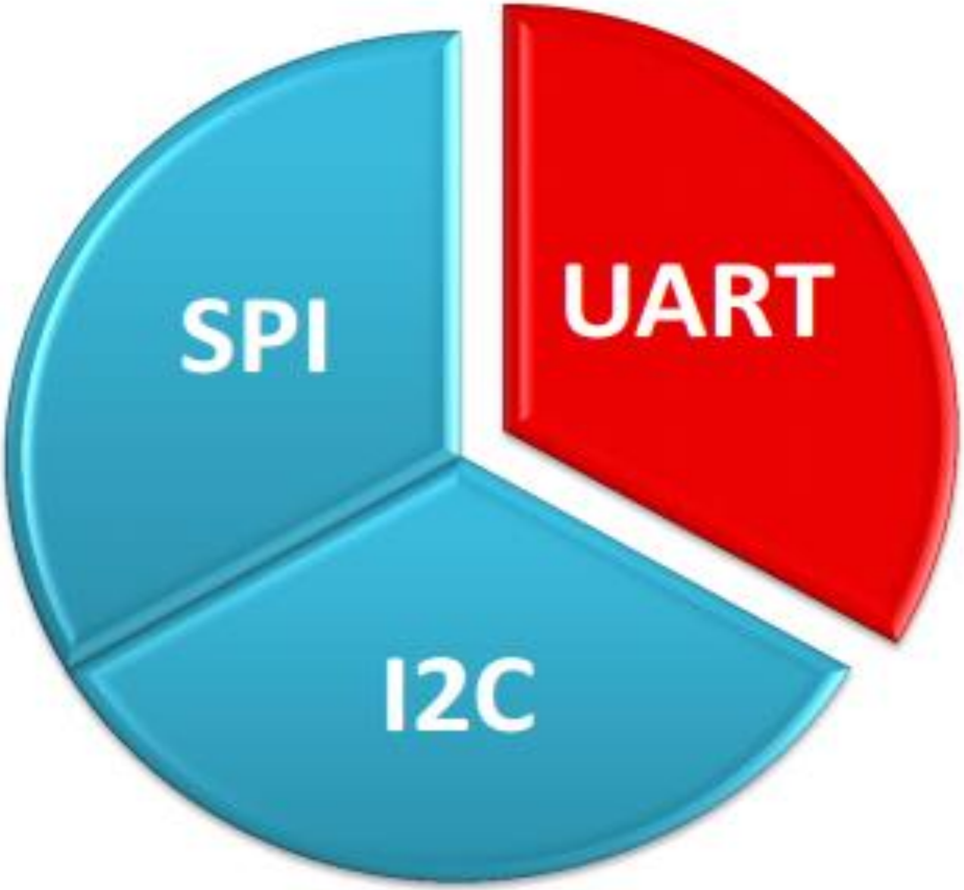
The Master / Slave network can be divided to:

❑ Single Master Single Slave (SMSS)
❑ Single Master Multi Slave (SMMS)
❑ Multi Master Multi Slave (MMMS)

Basic Communication protocols



Basic Communication protocols

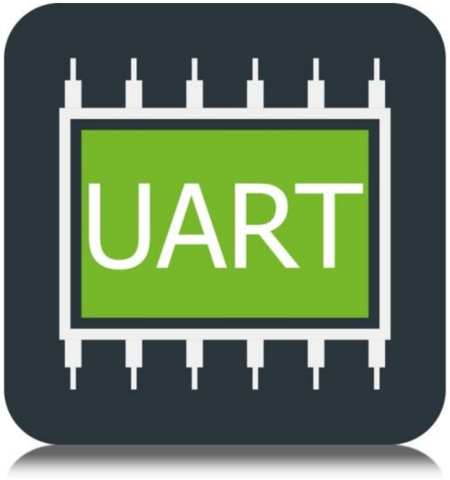


UART concept

What is UART ?

UART stands for Universal Asynchronous Receiver Transmitter. It is a serial communication protocol that consists of 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 concept

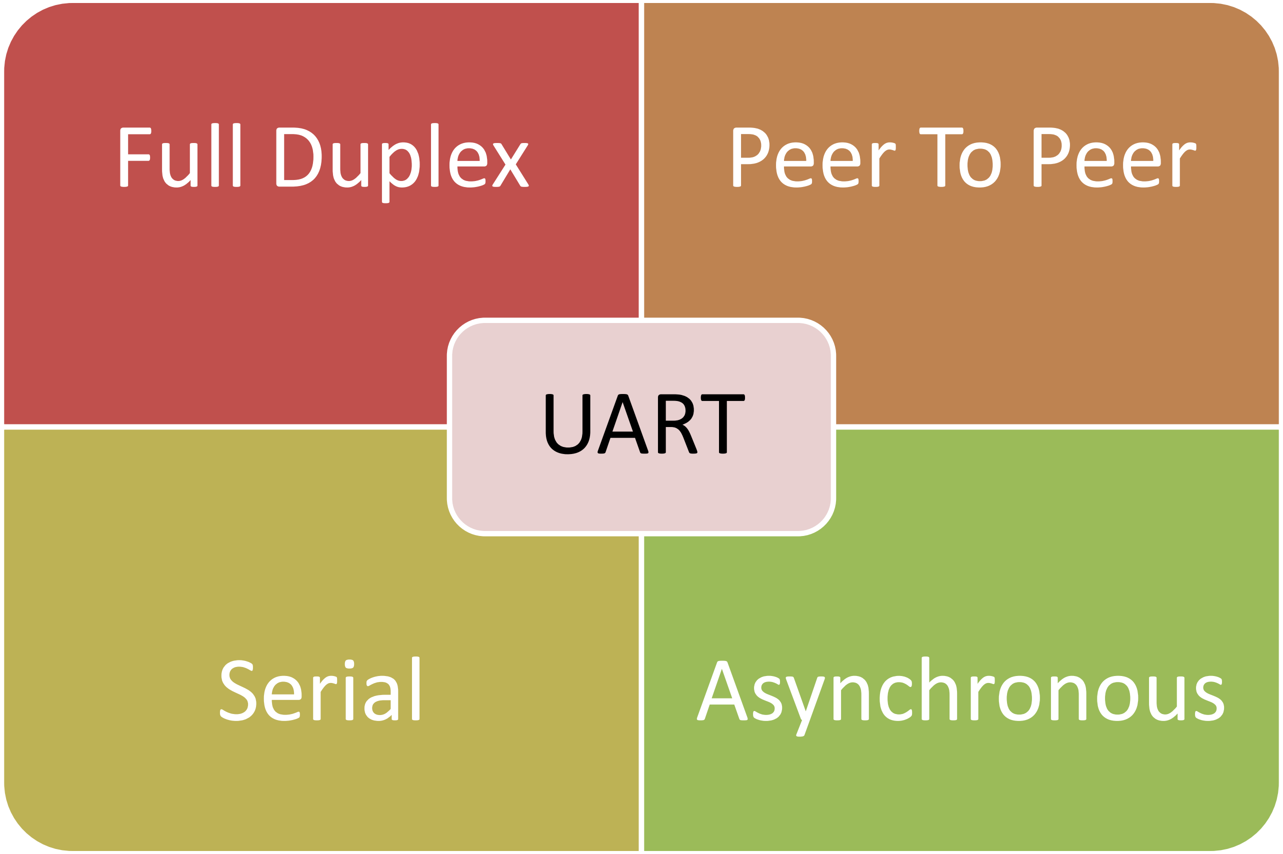
Note

What is the difference between Bit Rate and Baud Rate ... ?

The **bit rate** is the number of **bits** transmitted per second, whereas, the **baud rate** is the number of symbols transmitted per second. The symbol is a signal unit that is defined by the protocol, it may be one bit or more. Therefore, **baud rate** is always less than or equal to the **bit rate** but never greater.

In UART, the symbol is one bit, therefore the baud rate is equals to the bit rate in UART Communication protocol

UART Specifications



Full Duplex

Peer To Peer

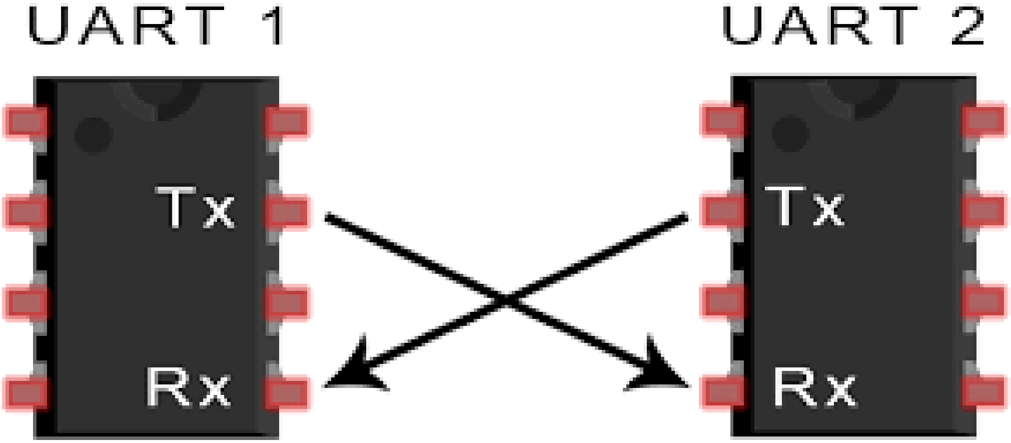
UART

Serial

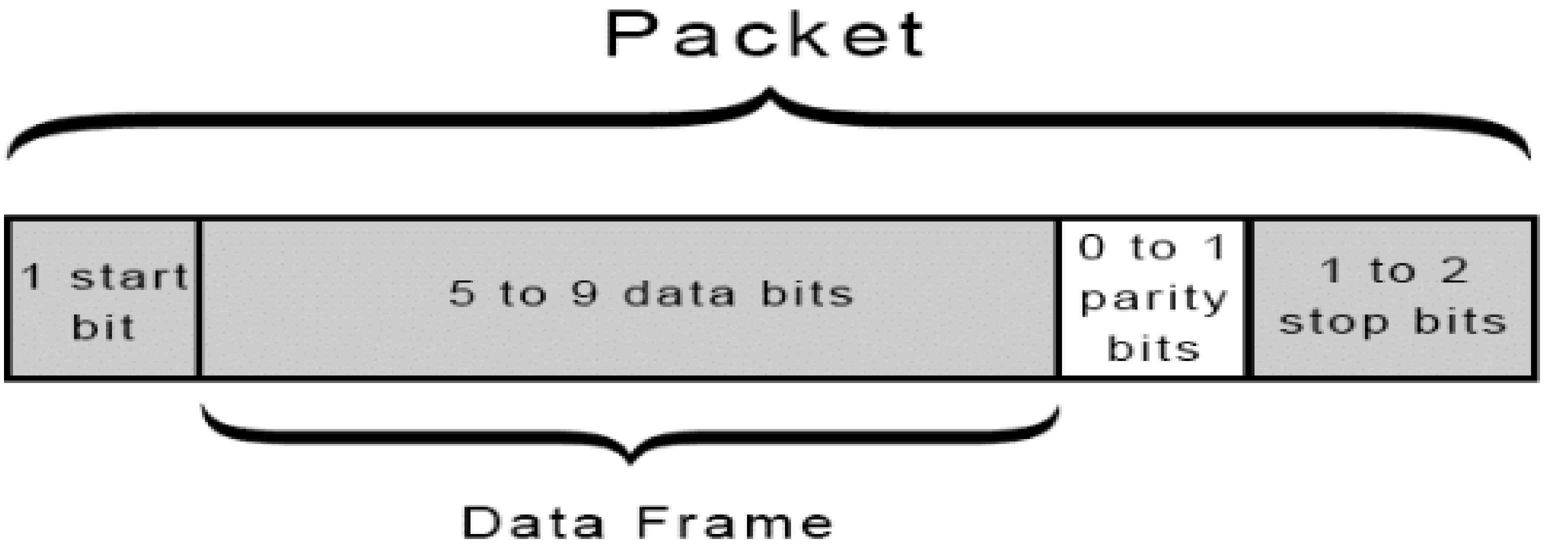
Asynchronous

UART Hardware Interface

Each node has a line called Tx (Transmission line) and another one called Rx (Receive Line). The Tx of one node shall be connected to Rx of the other node and vice versa.



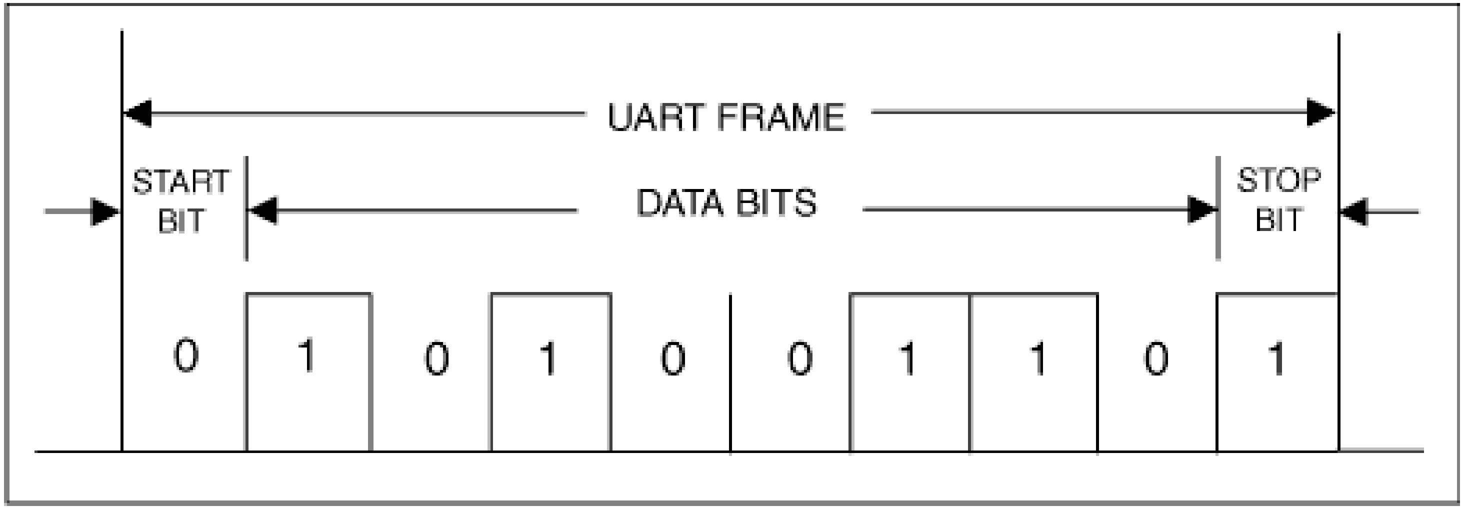
UART Data Frame Format



UART frame format

- Start bit: 1 bit indicates the start of a new frame, always logical 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.

UART Frame Example



This Frame has:

- ○ 8 bit Data Frame
- ○ No Parity bits
- ○ 1 Stop bit

UART Frame Example

Question …

what will be the value of the parity bit in the pervious example if it was enabled in the even mode … ?

UART Frame Example

Question …

what will be the value of the parity bit in the pervious example if it was enabled in the even mode … ?

Answer …

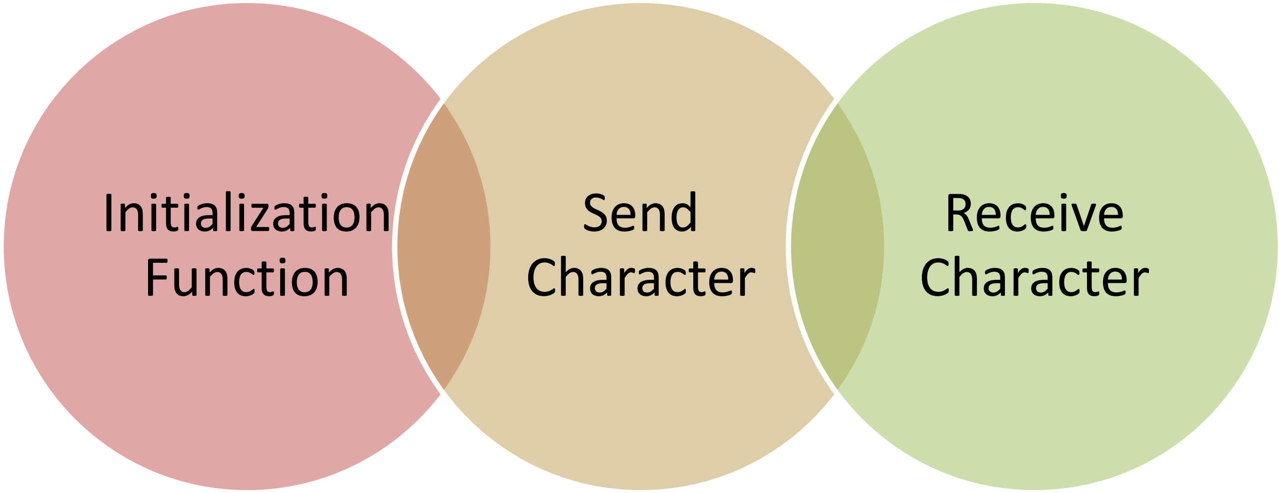
As the data frame has 4 bits with value 1, then the number of 1’s is even, then the parity bit shall be 0

UART Driver

It is the time to build the UART driver for our microcontroller AVR atmega32. In AVR, it called USART !

The difference between UART and USART is that the USART is supporting the synchronous mode.

UART Public Interfaces



Initialization Function

Send Character

Receive Character

USB to TTL Module

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 tour micrcontroller

LAB 1

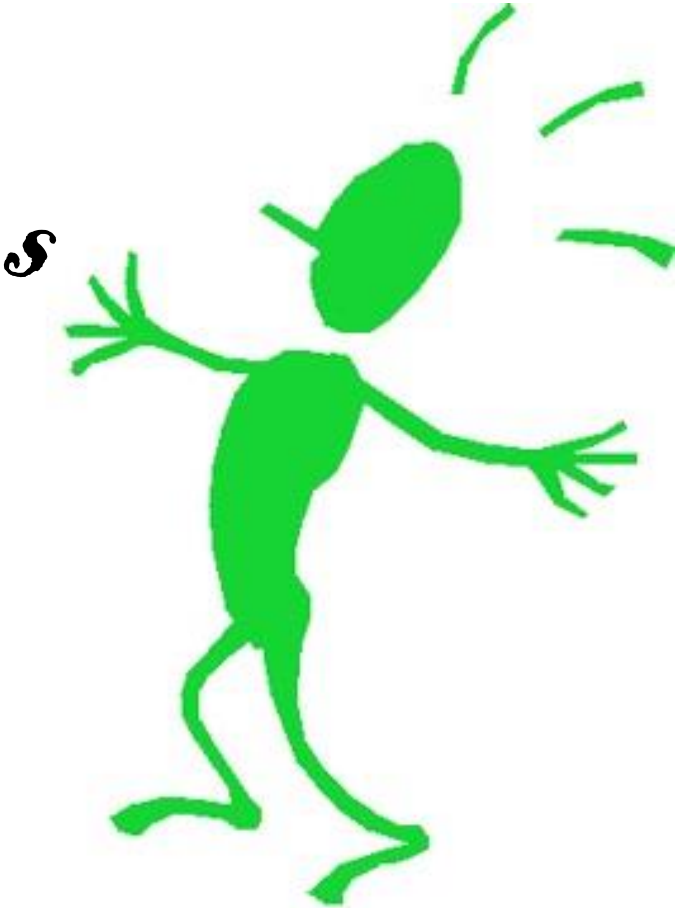
Control the LEDs on your kit by your computer !
Sending a number by the computer shall toggle the corresponding LED.
i.e sending ‘1’ toggles LED 1 , sending 2 toggles LED 2 and so on …

Time To

Code

The End …

Any questions … ?



IMT SCHOOL

🏠 **www.imtschool.com**

f **ww.facebook.com/imaketechnologyschool/**