

USB

Basic Objective behind the design of this USB is Unified Low, Medium and High Speed into a Common Bus.

USB is a peripheral BUS it is not a controller bus, USB controller is also referred as a Host controller.

BUS: Buses were designed based on the demand of the devices, some devices are Low Speed some devices are high and Medium speed.

- Low Speed device – UART
- Medium Speed device – Parallel
- High Speed device – PCI [Peripheral Component Interconnect]

In PCI :

- No hot plug
- No -autoconfiguration
- Limited System resources [IO addr., and Non-Shared irq lines]
- Limited standard connectors

To Over come those drawbacks we Use USB

=>Address : We have limited IO address space and limited virtual address, since the device come its OWN registers, we need to allocate some system resources for the device to work.

=> IRQ line : we have limited no. of irq lines we need to engage the one of the IRQ line for device [like shared irq].

=> Physical level: the connections are limited [More device of same type can not plug-in at a same point of time]

“USB is the solution for all the above problems”

USB is One Bus One Connector type

- we can connect the Low, medium and High speed device to same bus.

USB device doesn't take the address from the processor and irq also.

[there is no such thing called allocating the IO address, allocating memory address or IRQ line to USB device]

Key Benefits:

=> USB device are Hot Pluggable

- user need to configure it.

=> USB device don't consume the system resources

=> 127 device connected [including the HUB]

=> Low, Medium and High speed devices are connected to same USB port.

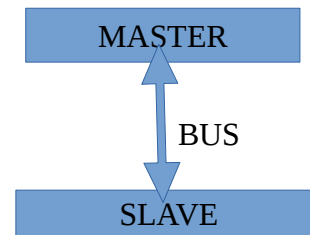
[all HID [Human Interface Devices] are Low speed:- keyboard, touchpad]

=>No IRQ --- No ISR routine/handler/function.

USB device has 3 Entities:--

- 1.Master – Slave Bus
- 2.Slave device
- 3.USB Host Controller [Master]

This 3 entities make up the entire USB system.



[Which you are connecting that Interface is internally connected to a BUS , which is connected to Controller and Controller internally connected to a Processor .]

[Host Controller job is to drive all command (control commands) , all data address to BUS , **Slave** Controller will respond to the requests which are sent by the Master]

Two kinds of Specifications for Implementation of Controller

- 1- Host controller
- 2- On The Go controller [It can be a Master it also can perform the services of a Slave depending on the connection Eg: Mobile Phones]
[this two are act as Master]
- 3- Device controller [Slave]
[This 3 Different species contain for building up a circuit called CONTROLLER]

USB Device Classes:

- 1.Audio
- 2.Communication [Modem]
- 3.HUB
- 4.HID
- 5.Printer
- 6.Mass Storage Classes

Vendor Specific Classes:

- Scanner, Video, Ether-net , Serial connectors

Q.when ever the Device changes the state , How will Software know about it and what are the ways?

There are Two ways:

- 1.polling
- 2.interrupt.

Q.How the polling work?

Keep looking into some registers to watchout for State Change (status registers).

Q.How does Interrupt will work?

Device will trigger a signal when ever it under goes in State Change .

Q.How will we write driver to know what the H/w is doing . [when USB device dont have I/O addr , IRQ. How will driver Shut Down State change.

[**Network logic]

->USB driver are more like Networking