

***Introduction to  
USB Development***

# USB Development

---

- 1 Introduction
- 1 Technical Overview
- 1 USB in Embedded Systems
- 1 Recent Developments
- 1 Extensions to USB
- 1 USB as compared to other technologies

# INTRODUCTION

## USB: Universal Serial Bus

- 1 A serial bus standard for connecting devices usually peripheral devices to computers.

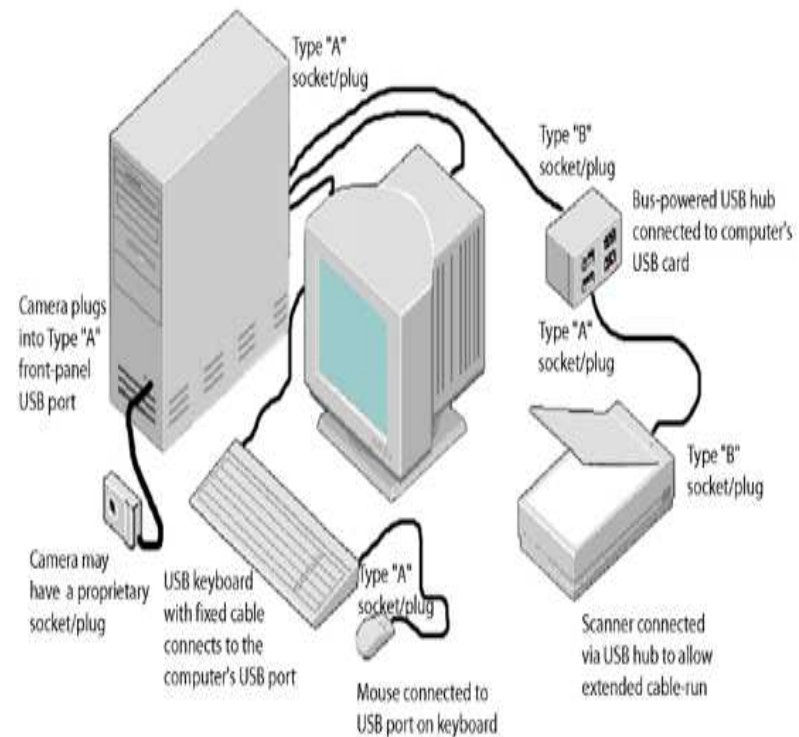
## Need for USB

- 1 To eliminate a proliferation of the cables
- 1 Eliminate the need for plug in expansion cards

## Standardization

- 1 Design of USB standardized by USB implementers forum

Why USB for Embedded Systems?



# INTRODUCTION

---

USB:

- ┆ Offers simple connectivity
- ┆ Low cost
- ┆ Ease of use
- ┆ Manages power effeciently
- ┆ Supports all kinds of Data

# TECHNICAL OVERVIEW

---

Serial Protocol and Physical Link uses two pairs of wires

- 1 Upstream Connection and Downstream Connection

Uses three types of cables and two types of connectors

- 1 High Speed cables at 480 Mbps
- 1 Full Speed cables at 12 Mbps and
- 1 Low Speed cables at 1.5 Mbps

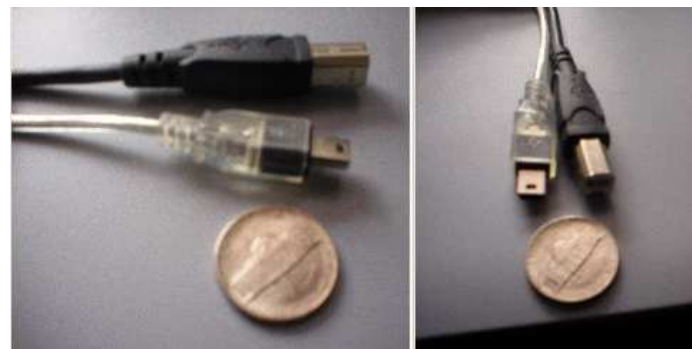
USB standard uses NZRI system to encode data

# Types of connectors

---

Two Types of Connectors:

- 1 Type A
- 1 Type B

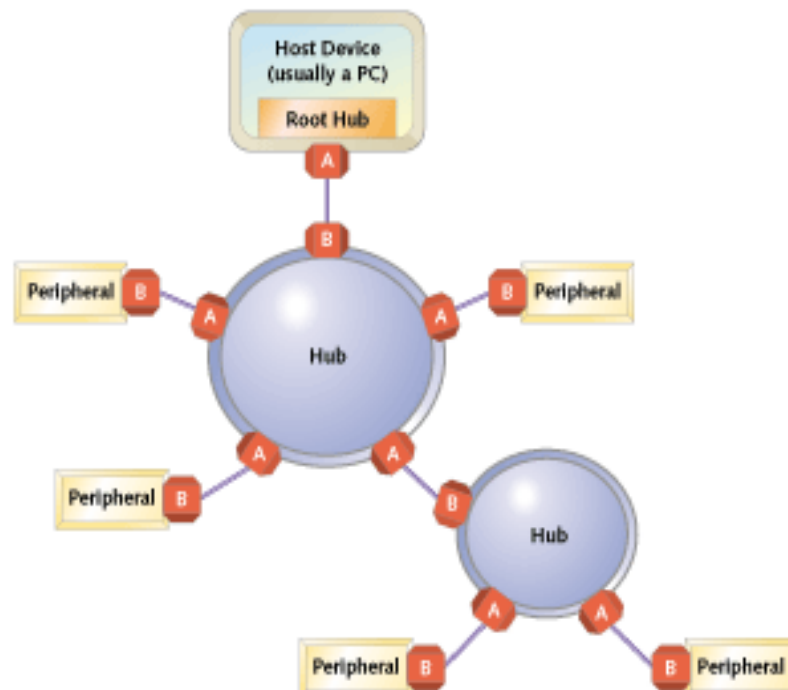


# ARCHITECTURE

---

Architecture of a USB network

- 1 Consists of one host device and multiple daisy chained devices



# ARCHITECTURE

---

Follows a Tiered star Topology and consists of:

- 1 Peripherals
- 1 Hubs
- 1 Host controller

Peripherals receive and respond to the commands from the host. E.g. Mice, Keyboard, Joysticks

Two types of Peripherals

- 1 Standalone and
- 1 Compound Device



# ARCHITECTURE

---

- 1 Host recognizes the peripheral through a process called **Enumerations**
- 1 Host communicates with the peripheral to learn its identity and identifies which device driver is required
- 1 Host supplies the peripheral with an address

## HUBS:

- 1 Allows many USB devices to share a single USB port
- 1 USB devices with some incorporated intelligence
- 1 Increase the logical and physical fan out
- 1 Single upstream connection and one-many down stream connection

# ARCHITECTURE

---

Two kinds of Hubs:

- Bus Powered Hub: Draws power from the host computers USB interface
- Self Powered Hub: Has a built in power supply.

Smart wire passing data between the peripheral and Host

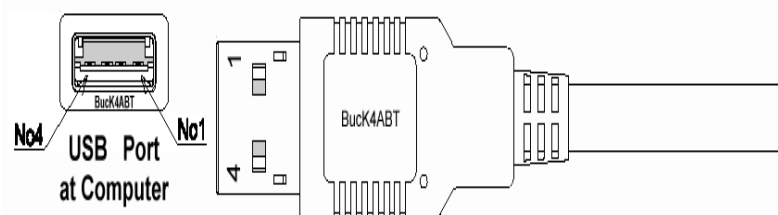
Direct connection exists between host and peripherals

Host is the controller of the entire network. E.g. PC



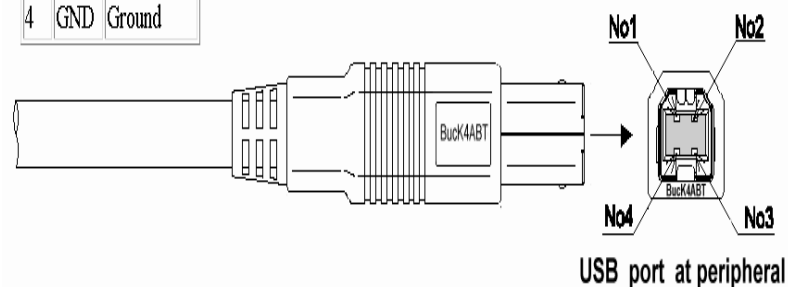
# Power Management

- 1 Peripherals connected regardless of the power state
- 1 A pair of wires to supply power to the peripherals
- 1 Manage power by enabling and disabling power to devices
- 1 Removes electrically ill behaved systems from the network



Pin	Name	Description
1	VCC	+5 VDC
2	D-	Data -
3	D+	Data +
4	GND	Ground

## Universal Serial Bus (USB)



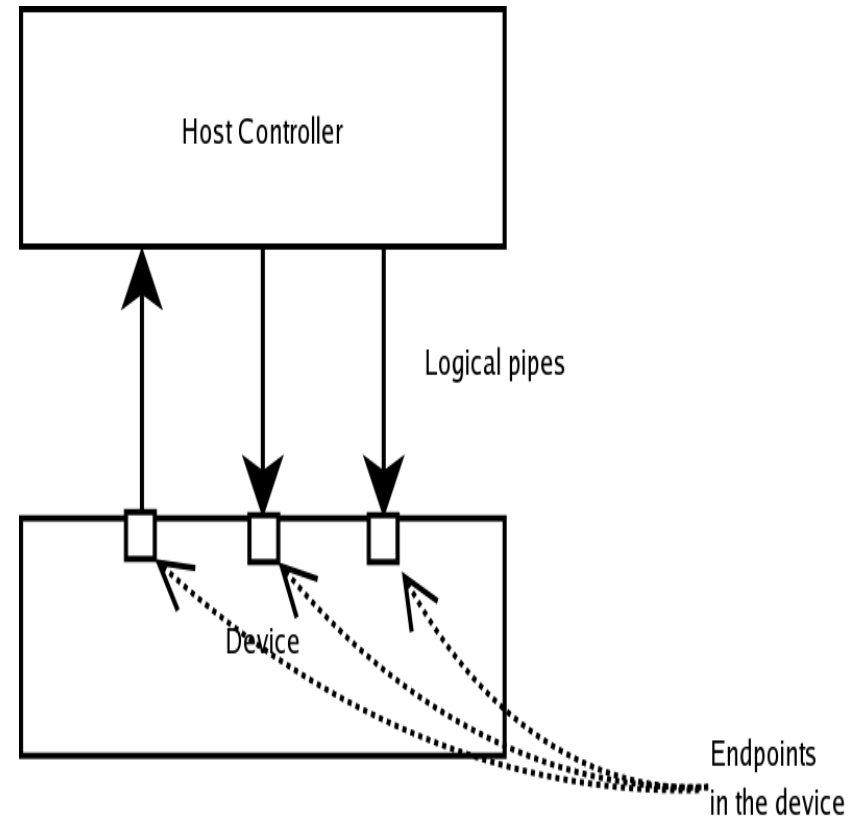
# USB Communication Overview

---

- 1 Communication between the host and the end points located in the peripheral
- 1 End point is a unique point in the device which is the source or the receiver of the data
- 1 End point has a definite address associated with it
- 1 Codes indicate the type of transfer
- 1 16 end points within each device each end point has a 4 bit address
- 1 End point “0” reserved for control transfers

# USB Communication Overview

- 1 Transactions between the host and end point take place through virtual pipes
- 1 Pipes are logical channels which connect the host to the end points
- 1 Once the communication is established the end points return a descriptor
- 1 Descriptor is a data structure tells the host about the end points configuration and expectations



# USB Communication Overview

---

USB supports four transfer types of data:

- 1 Control Transfers: exchange information such as configuration, command information , set up between host and end point
- 1 Bulk Transfers: Supports bulk amounts of data when timely delivery isn't critical.E.g. Printers and Scanners
- 1 Isochronous transfers: Handle transfers like streaming data
- 1 Interrupt transfers: Poll devices to see if they need service

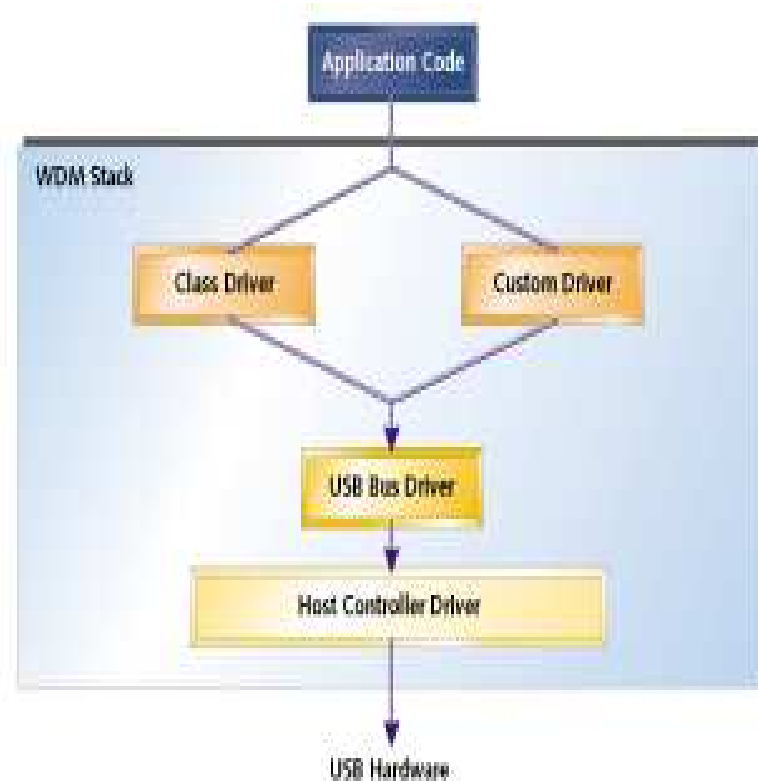
# HOST DEVICE DRIVERS

---

- 1 USB is a complex standard requires enormous amount of software support both of firmware side and the host support
- 1 USB communication model relies on the Win32 Driver model
- 1 Win32 driver model layers different processes of communication into a stack
- 1 Its organized into Application Layer, Driver classes, Bus Driver and Host controller driver

# Host Device Driver

- Application Code reacts with the drivers using API calls
- Within the stack itself the communication is through Interrupt I/O request packets
- USB Bus driver manages Enumeration, power management and other USB transactions





# Types of Drivers

---

1 Two types of drivers. Class Drivers and Custom Drivers

1 Drivers are organized into classes where hardware that falls into a single class share similar interfaces

E.g. HID Class which supports devices like mice, Joy sticks

Monitor class which supports image position, alignment

1 Custom Driver exploits capabilities of a particular piece of Hardware

E.g. A Data Acquisition system

# USB in Embedded Systems

---

- 1 Processor based USB controller adding Keyboard and mouse to an existing design
- 1 USB Host controller will communicate with the existing design via an UART
- 1 Code for the Host USB controller will contain a Library of routines and an Application layer Firmware
- 1 Code from the Library will take care of enumerating newly attached or removed devices, communication to and from the devices, Memory management, etc.

# USB in Embedded Systems

---

- 1 While creating the code the first step is to determine how many and what type of devices will be supported
- 1 The next step is to create application code for each device.
- 1 The driver template will include start, stop and run functionality
- 1 Addition of Hub Support is the key design consideration

# EXTENSIONS TO USB

---

- 1 USB 2.0
- 1 PictBridge Standard to communicate imaging devices
- 1 Microsoft X box console
- 1 IBM Ultraport
- 1 USB 1.0 OTG
- 1 USB 1.0a supplement OTG
- 1 Wireless USB

# USB On-the-Go Technology

---

- 1 USB On-The-Go Technology is used to provide dual role to the peripherals
- 1 Enables direct communication between the hosts without involving the processor
- 1 Incorporates Mini A , Mini B, Mini AB plugs and receptacles
- 1 Highly complex design

# USB On The Go

---

## Advantages:

- 1 Provides Dual Role Devices
- 1 Introduces new connector types, Mini A, Mini B, Mini AB
- 1 Provides with Aggressive Power Management

## On the Go Functionality of the USB can be implemented:

- 1 Using a Full solution Approach
- 1 Using a USB microcontroller
- 1 Designing a custom IC

# Wireless USB

---

- 1 A Paradigm developed by Cypress that allows devices to be connected but appear as if they are connected to the host over normal USB connectivity
- 1 Addresses many of the Design issues of Wireless networking
- 1 An evolution that relies on familiar and existing technologies
- 1 Desirable for point to point devices
- 1 Features of Wireless USB are its Ease of use, simple connectivity and conservation of the battery power

# USB as compared to other Standards (USB Vs FireWire)

---

## USB

- § Connects peripherals to the computers
- § Operates at low data rates
- § Cannot be intended to act as an internal bus for storage
- § Follows a tiered star topology
- § USB uses a speak when spoken to protocol
- § Relies on a single host to control the network

## FireWire

- A technology to interconnect mass storage devices
- Has higher data rates
- Uses a repeater based topology
- Can connect with any node at a time
- Any capable node can control the network
- FireWire ports are expensive



# REFERENCES

---

- 1 <http://en.wikipedia.org/wiki/USB>
- 1 [http://www.embedded-computing.com/articles/wright et al/](http://www.embedded-computing.com/articles/wright_et_al/)
- 1 <http://www.embedded.com/2000/0003/0003ia2.htm>
- 1 Introduction to USB for Design Engineers- A Webminar :
- 1 <http://www.pmc-sierra.com/webinars/usb.html>

# QUESTIONS??

---

