



Faculty of Engineering
Cairo University



elctronics

SBE 202 B

electronics research

digital interface converter

Universal Serial Bus

submitted to:

dr/mohammed ahmed monir islam

submitted by:

mohamed abdelkarim seyam (sec 1 to sec 5)

mohamed ahmed abdelaziz (sec 5 to sec 18)

ramadan ibrahim moheyeldeen (sec 18 to sec 26)

Contents

1 Abstract	4
2 usb system	5
2.1 host	5
2.1.1 USB Host Hardware	5
2.1.2 USB Host Software	6
2.2 Hub	6
2.2.1 Root Hub	6
2.2.2 power Hub	6
2.3 USB Device	7
2.3.1 bus powered device	8
2.3.2 self powered device	8
2.4 USB Cables and Connectors	8
2.4.1 cables	8
2.4.2 connectors	9
3 USB Transfer Types	10
3.1 Control Transfers	10
3.1.1 setup stage	10
3.1.2 Data stage	10
3.1.3 Status stage	11
3.2 Bulk transfer	11
3.3 Isochronous Transfer	11
3.4 Interrupt Transfer	11
4 usb implementation	11
4.1 EFM32 USB Pin Descriptions	12
4.2 why EFM32?	12
4.3 EFM32 as USB Host	12

4.4	EFM32 as USB Device	13
4.4.1	Self Powered Device	14
4.4.2	Bus Powered Device	15
4.5	low speed device	15
4.6	EFM32 as USB On-The-Go Dual Role Device	16
5	High Speed Layout Guidelines	17
6	protocol specific layout guidelines	18
7	General High Speed Signal Routing	20
7.1	Trace Impedance	20
7.2	Trace Length Matching	20
8	Return Path	21
8.1	Via Stitch Ground Planes	23
9	High Speed Differential Signal Routing	25
9.1	Differential Signal Spacing	25
9.2	Symmetry In Differential Pairs	25
10	protocol	26
10.1	Connectors And Receptacles	26
10.2	Via Discontinuity Mitigation	27
10.3	Increase Via Anti-Pad Diameter	28
10.4	Equalize Via Count	28
10.5	Surface Mounted Devices Pad Discontinuity Mitigation	28
10.6	Signal Buildings	29
11	EMI and ESD considerations	29
12	Traffic On The bus	30
12.1	Protocol Analyzer	30

12.2 Packets	30
12.3 Control Transfer Type	31
12.3.1 SETUP Stage	31
12.3.2 DATA Stage	32
12.3.3 STATUS Stage	32
12.4 Interrupt Transfer Type	34
12.5 Bulk Transfer Type	35
13 Split Transaction	36
14 Device Enumeration	37
15 Descriptors	37
16 Low Voltage Differential Signaling (LVDS)	39
17 Signaling Levels	40
18 LVDS Termination	41
19 Common Mode Range	41
20 Failsafe Feature	42
21 Industry Standards for various LVDS Technologies	43
22 Point-to-Point Configuration	44
23 Bi-Directional Application on One TWP	44
24 Multi-Drop Configuration	45
25 Signal Quality Across Cable	45
26 conclusion	45

1 Abstract

usb stands for Universal serial bus and was invented and standardized in 1995. the first motivation is to allow the peripheral automatically configured when plugged in your own computer known as (plug and play) with out need to restart your computer.

usb has grown and become the most known interface for many industrial embedded application.and the different companies after that move the direction of saving power and reducing consumption by providing usb system sleep mode.

usb system consist of host and hubs, usb device can be self powered or bus powerd. Usb has many transfer types control, bulk, isochronous, and interrupt each for a specific purpose. different hardware implementation and PCB layout standards are required for high speed signaling. Enumeration is a process follow attaching the device to start the communication, descriptors store the host data during enumeration.

Implementation of PCB layout while using diffrent high speed signals. We will focus on high speed layout guidelines for USB, USB Hubs, HDMI, Displayport, PCIe, SATA. Concerns must be taken because high frequency signals are most impacted with other signals.

LVDS is a high speed general purpose interface with low power consumption that have a variety of applications.LVDS is characterized with low voltage swing in case of comparing it to other industrial data transmission standards

2 usb system

usb stands for Universal serial bus and was invented and standardized in 1995. the first motivation is to allow the peripheral automatically configured when plugged in your own computer known as (plug and play).usb has grown and become the most known interface for many industrial embedded application.

usb is a cable bus with single master called bus and multiple peripherals called devices.and it offers unique connector with different type of device , easy expandability using hubs. Figure(1)

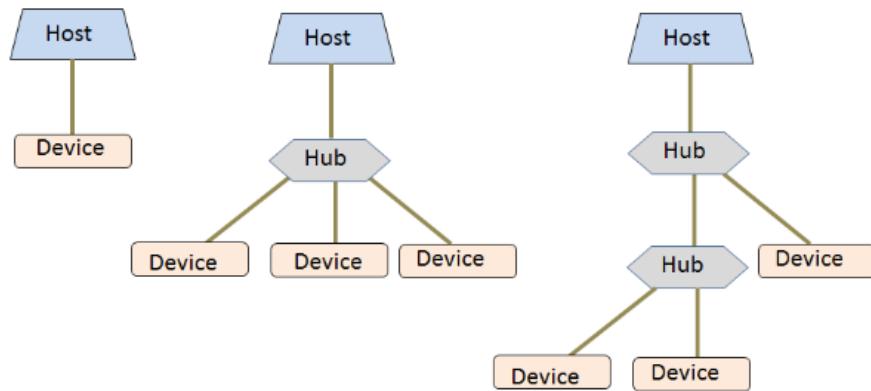


Figure 1: usb system

2.1 host

universal serial bus is a bus cable with single master called host which control all communication in the system as it sends the requests to the devices and the devices respond to it.USB Host designs consist of a USB Host controller hardware circuit and a software component.

2.1.1 USB Host Hardware

provide many functions as it detect the usb device removal and attachment and supply it with power ,manage flow of data between host and device ,provide error checking and

transaction.

2.1.2 USB Host Software

software handles the device connectivity ,enumeration and configuration

2.2 Hub

Hub expands the bus adding more attachment points (ports).Hub has one upstream port facing the host and multiple downstream ports facing the devices. to the host ,a hub will look like a device with multiple endpoints .each Hub reserves Endpoint 1 IN to inform the Host that a downstream device has either been removed or inserted from the bus in addition to downstream-port endpoints.

Hubs separate the ls ,hs ,fs using split transaction like if i have a low speed device and want to communicate with a high speed Host the hub communicate with each of them as its speed preventing combining speed which decrease the speed.

Hub also detect connect and speed of the device by differentiating between D+ and D- using chirp . the device pull up to 3.3 b on either D+ for Full speed and High speed or D- for low speed and this enabled when device sense Vbus.

2.2.1 Root Hub

the host can directly connected to only one device and if one more device want to connect it connects only through the hub.the Root Hub is the one directly connected to the Host also it can be also five additional hub connected to the root hub in series creating seperatetiers.

Figure(2)

2.2.2 power Hub

a Hub can be either a bus-powered or can be self-powered.bus-powered Hubs draw power from the usb Host bus on the other hand the self-powered hubs are capable of supplying power to the device to augment the power delivery of the Host.

During the enumeration process, a Host will consider the power characteristics of a Hub when deciding which downstream device configurations to activate.

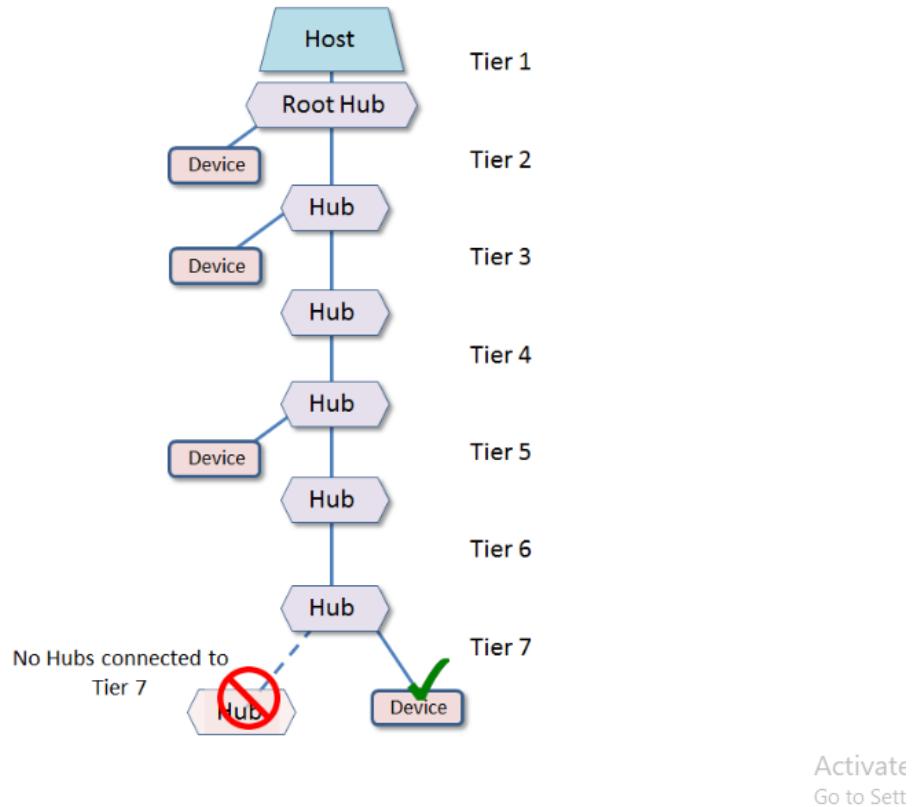


Figure 2: Usb Hub system

2.3 USB Device

the peripherals that attach to Usb Host called devices.it provides functionality to the end-user.Typical devices include products such as keyboards, mice and medical devices. every device has a unique address between 1 and 127 assigned by the host.every usb product need a unique pairs of IDS which is VID stands for vendor id and PID stands for product id.

the device uses to exchange data buffers called endpoints (EPS). it is two types one is responsible for receiving data from host called OUT EP and the other store data sent to the host called IN EP.

every device has EP0 IN and EP0 OUT used during enumeration.EPS with numbers

other than zeros used after enumeration for communication.

2.3.1 bus powered device

it sinking the current up to 500 mA after enumeration from Vbus only.

2.3.2 self powered device

it sink the current from the Vbus theoritically and an external power supply.

2.4 USB Cables and Connectors

2.4.1 cables

- traditional cables

Specification	Speed	Number of Lines	Maximum Length
1.0	Low	4 (VBUS, Gnd, D+, D-)	5
1.0	Full	4 (VBUS, Gnd, D+, D-)	5
2.0	High	4 (VBUS, Gnd, D+, D-)	5
3.0	Superspeed	9 (VBUS, Gnd, D+, D- SSRX-, SSRX+, Gnd, SSTX+, SSTX-)	3
3.1	Superspeed +	9 (VBUS, Gnd, D+, D- SSRX-, SSRX+, Gnd, SSTX+, SSTX-)	3

Figure 3: traditional cables

- type C USB *Type – CTM* calls for a cable with 24 signals.

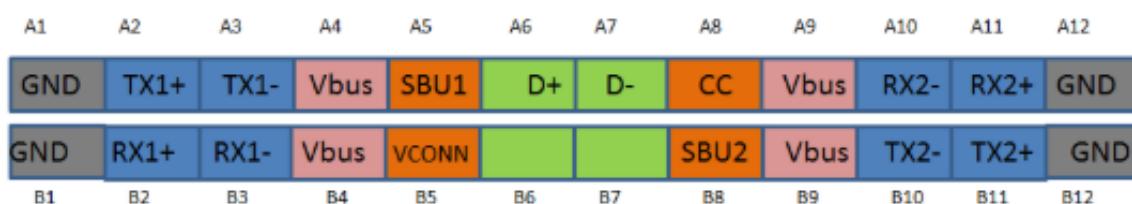


Figure 4: type C cables

2.4.2 connectors

all connectors of a device can be upstream (type B) meaning going up to host. all devices has type B connectors or downstream (type A) going down to a device (hubs will have type A too) and finally type C connectors which is symmetrical meaning that no difference between upstream and downstream . and reversible meaning that you can lift it and plugged the other site of it with no difference.

- traditional connector

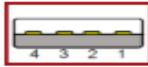
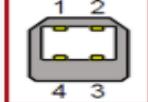
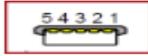
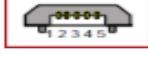
Connector Type	Pin out	Picture	Notes
Type A			Operates in USB 2.0 mode, if plugged into a USB 3.0 A receptacle.
Type B			Operates in USB 2.0 mode, if plugged into a USB 3.0 B receptacle.
Mini B			
Micro A			
Micro B			Operates in USB 2.0 mode, if plugged into a USB 3.0 Micro B receptacle.
3.0 Type A			Operates in USB 2.0 mode, if plugged into a USB 2.0 A receptacle.
3.0 Type B			
3.0 Micro B			

Figure 5: traditional cables

- type C

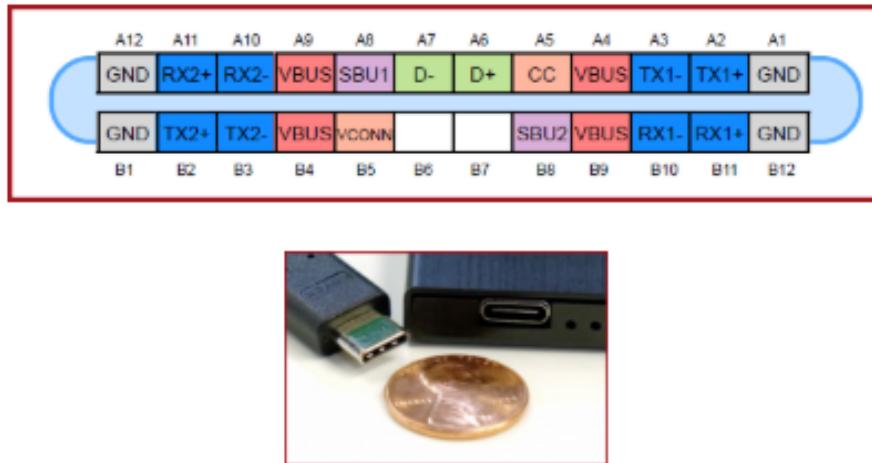


Figure 6: type C connector

3 USB Transfer Types

Universal Serial Bus (USB) transfer types refer to the mode of communication used between the Host and a device's endpoints. The transfer type determines the transaction length, frequency and CRC of packet attached. Transfer types are set by the device and are communicated to the Host during the enumeration process. There are four types of transfer.

3.1 Control Transfers

The control transfer used by the host during device configuration. It also supports command and status type communication flow. The default control endpoint is always zero. There are 3 stages

3.1.1 setup stage

Host sends request to USB device

3.1.2 Data stage

In and Out transaction are made

3.1.3 Status stage

it occurs in order to present a status to the request made by the host.

3.2 Bulk transfer

Bulk transfer used when throughput matters.it's max data is 64 for the full speed and 512 for the high speed, but transfer are scheduled over available band width.the typical devices that uses the bulk transfer is Mass Storage devices and Printers.

3.3 Isochronous Transfer

it is used in real time application like audio and video streaming when you don't want to lose any frame work.there shouldn't be any delay in data transmission.it is only supported by high and full speed devices.

3.4 Interrupt Transfer

it is used when latency matters (mouse).Interrupt transfer guarantee a max latency though polling ,with max data point of 64 (for low speed and full speed) and 1024 (for high speed) byte.

4 usb implementation

USB can be operated in 2 different modes; host or device, the hub is special case of the device.and it also can be operated as "on to Go" mode OTG meaning that you can work as a host or a device depending on which kind of controller is in the other end of the cable.for example your smart phone act a host when connecting a memory card to it and act as a device when connecting it to computer.we will use EFM32 microcontroller for implementation.

4.1 EFM32 USB Pin Descriptions

the following pins abbreviation stands for aspecific names

abbreviation	stands for
USB_DP	Data line
USB_DM	Inverted data line
USB_VBUS	Sensing if VBUS is connected.
USB_VBUSEN	VBUS Enable
USB_DMPU	Data Minus Pull-Up
USB_ID	ID for determining which device should act as bus master.
USB_VREGI	Voltage regulator input.
USB_VREGO	Voltage regulator output.

4.2 why EFM32?

EFM32 micro controllers have many features as they are the major in the market thanks to its sleep mode option as the enable energy efficient, autonomous behavior while the CPU is sleeping. it is ideal for battery-operated applications and other systems requiring high performance and low-energy consumption,LESENSE demo ready and its software packages are available and introduces high features.

4.3 EFM32 as USB Host

In host mode, the microcontroller acts as the bus master and it is responsible for inquiring the devices connected for configuration information and give them an address on the USB bus in an operation called enumeration.it also control data flow on the bus.no device can transmit on the bus without a host request.the host must provided power to the connected devices through the +5 V VBUS line.

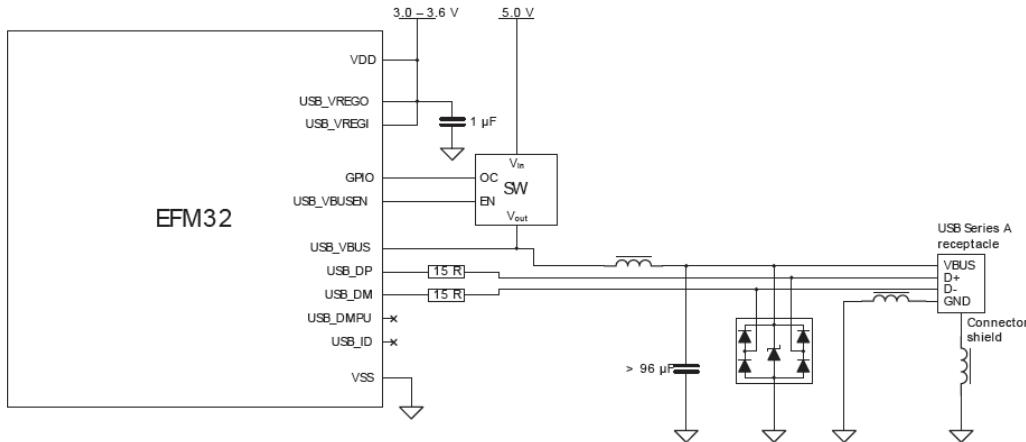


Figure 7: USB Host Schematics

design consideration

- choose a 48 MHz (2500 ppm) crystal.
- Use ferrite beads for VBUS, GND and receptacle shield. Place near receptacle.
- choose a switch that can shut off VBUS if current exceeds 500 mA.
- Provide at least 96 uF decoupling capacitance on VBUS. Place near USB receptacle.
- Terminate D+ and D- with 15 ohm serial resistors. Place near EFM32.
- choose an ESD protection device. Place near USB receptacle.
- Select a USB Series A type receptacle.

4.4 EFM32 as USB Device

to enable the host to configure the device the device should provide configuration information to the host depending on the device functionality it is divided into two types; hubs and functions as we said earlier the different between them.

the transmission of the data can't accomplished until requested by the host. microcontrollers EFM32 can not operate as a USB hub device.

the Usb device can be selfpowered as it get its power from external source or bus powered as it get its power from the USB host bus.

4.4.1 Self Powered Device

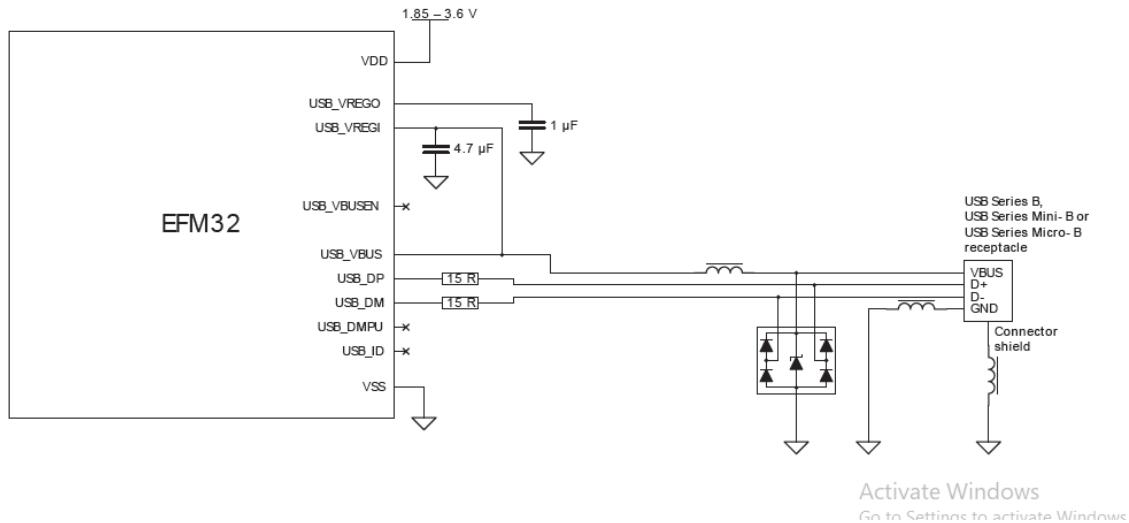


Figure 8: USB Self Powered Device Schematics

design consideration

- choose a 48 MHz (2500ppm) crystal.
- choose ferrite beads for VBUS, GND and receptacle shield. Place near receptacle.
- choose at least 4.7 μF decoupling capacitance on USB_VREGI. Place near EFM32.
- Keep total load capacitance on VBUS below 10 μF
- choose at least 1 μF decoupling capacitance on USB_VREGO. Place near EFM32.
- Terminate D+ and D- with 15 ohm serial resistors. Place near EFM32.
- choose an ESD protection device. Place near USB receptacle.

4.4.2 Bus Powered Device

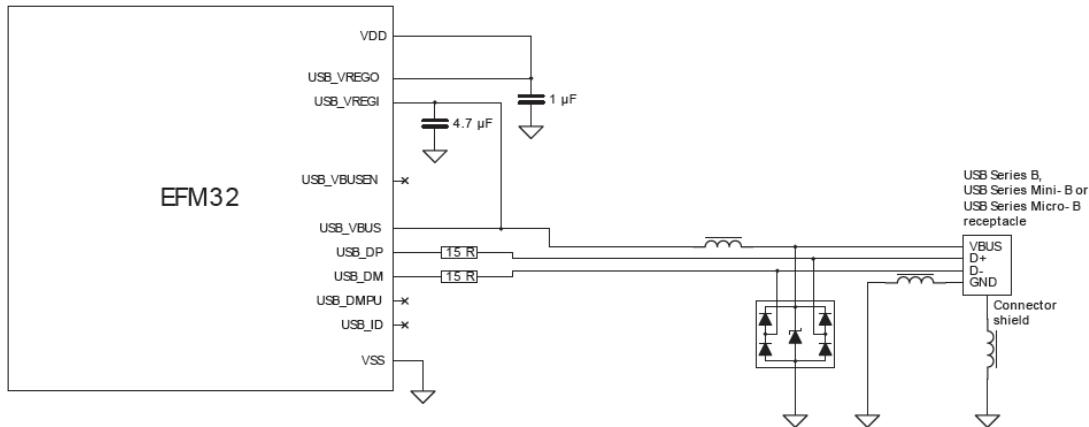


Figure 9: USB Bus Powered Device Schematics

design consideration

- select a 48 MHz (2500ppm) crystal.
- select ferrite beads for VBUS, GND and receptacle shield. Place near receptacle.
- select at least $4.7\mu F$ decoupling capacitance on USB_VREGI. Place near EFM32.
- select total load capacitance on VBUS below $10\mu F$
- select at least $1\mu F$ decoupling capacitance on USB_VREGO. Place near EFM32.
- Connect USB_VREGO to VDD.
- select decoupling capacitance on VDD as per AN0002 Hardware Design Considerations.
- Terminate D+ and D- with 15 ohm serial resistors. Place near EFM32.
- select an ESD protection device. Place near USB receptacle.

4.5 low speed device

by pulling up one of the data lines Speed identification of USB devices is done.

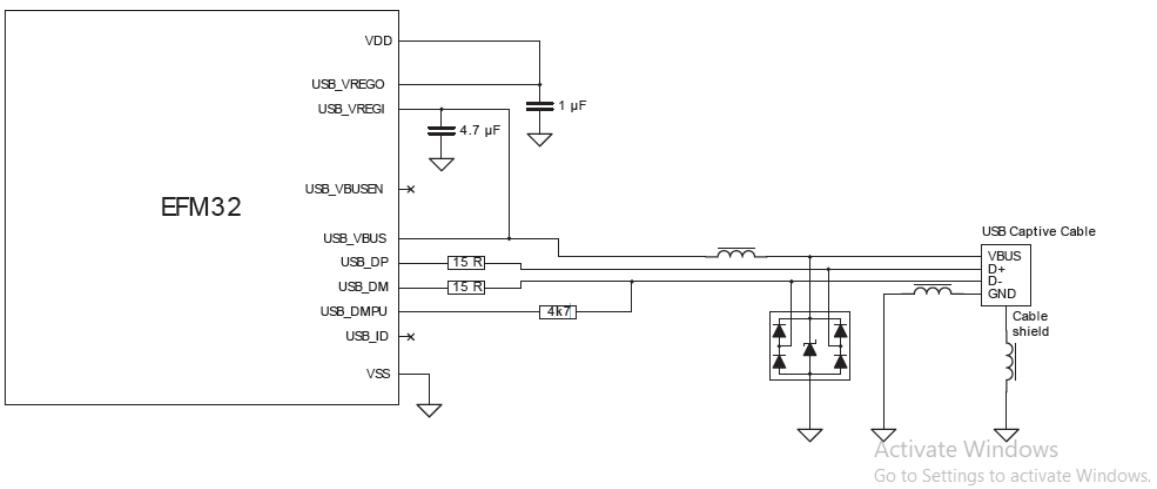


Figure 10: USB Low-speed Device Schematics

design consideration

- choose a 48 MHz 2500 ppm crystal.
- choose ferrite beads for VBUS, GND and receptacle shield. Place near receptacle.
- Connect a 4.7 *kohm* resistor between USB_DMPU and D- which work as apull up resistor.and choosing that value specifically that low speed device identified by pull up resistor= 1.5 *kohm* so 4.7 parrallel with the internal micro controller resistor which equal 2.2 *kohm* will give that value.
- Provide at least 4.7 uF decoupling capacitance on USB_VREGI. Place near EFM32.
- Keep total load capacitance on VBUS below 10 uF
- Provide at least 1uF decoupling capacitance on USB_VREGO. Place near EFM32

4.6 EFM32 as USB On-The-Go Dual Role Device

When operating as a OTG Dual Role Device, a USB product is capable of operating both as a USB host and a USB device. it uses micro-AB receptacle which can accept micro A or B plug.

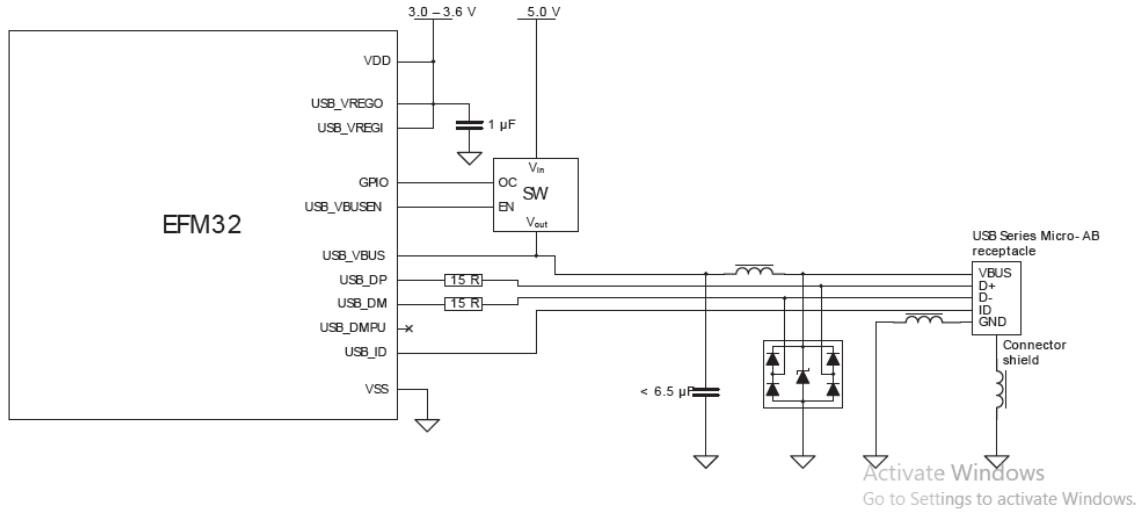


Figure 11: USB On-the-Go Dual Role Device Schematics

design consideration

- select a 48 MHz 2500 ppm crystal.
- select ferrite beads for VBUS, GND and receptacle shield. Place near receptacle.
- Connect the ID pin on the receptacle to USB_ID
- Ensure that total capacitance on VBUS is less than 6.5 uF
- Provide at least 1 uF decoupling capacitance on USB_VREGO. Place near EFM32.
- Terminate D+ and D- with 15 ohm serial resistors. Place near EFM32.
- select an ESD protection device. Place near USB receptacle or where the cable connects to the PCB.
- select a USB Series Micro-AB receptacle.

5 High Speed Layout Guidelines

Implementation of PCB layout while using different high speed signals. We will focus on high speed layout guidelines for USB, USB Hubs, HDMI, Displayport, PCIe, SATA.

Concerns must be taken because high frequency signals are most impacted with other signals.

Critical Signals

Signal Name	Description
DP/M	USB 2.0 differential data pair
SSTXP/N,SSRXP/N	SuperSpeed differential data pair
SATA_RXP/N, SATA_TXP/N	Serial ATA (SATA) differential data pair
PCIe_RXP/N, PCIe_TXP/N	PCI-Express (PCIe) differential data pair
HDMI_CLK+/-	High-Definition Multimedia Interface (HDMI) differential clock pair, positive or negative
HDMI_Data+/-	High-Definition Multimedia Interface (HDMI) differential data pair, positive or negative
DP_Lane#+/-	DisplayPort differential data pair, Lane 0 through 3, positive or negative

Figure 12: critical signals in use

6 protocol specific layout guidelines

many parameters are presented for various high speed standards, such as data rate, frequency, AC coupling, trace impedance, inter pair skew, intra pair skew. for better visualization for the differences

USB 2.0

Parameter	Value
Frequency	Low speed: 750 KHz (1.5 Mbps)
	Full Speed: 6 MHz (12 Mbps)
	High Speed: 240 MHz (480 Mbps)
AC Coupling Capacitors	No AC Capacitors allowed
Polarity Reversal	Not allowed
Trace Impedance	90 Ω ±15% differential, 45 Ω ±15% single ended
Max Cable Length	5 m

Figure 13: USB 2.0 Guidelines

USB 3.X

Parameter	Value
Frequency	SuperSpeed: 2.5 Ghz (5 Gbps)
	Superspeed+: 5 Ghz (10 Gbps)
AC Coupling Capacitors	AC capacitors required on the TX data lane. (Optional on the RX data lane)
Polarity Reversal	allowed on SSTX and SSRX
Max Intra-Pair Skew	15 ps/m (TI recommends 5 mils)
Max Inter-Pair Skew	N/A
Trace Impedance	90 Ω ±15% differential; 45 Ω ±15% single ended
Max Cable Length	3 m

Figure 14: USB 3.0 & 3.1 Guidelines

HDMI

Parameter	Value
Frequency	HDMI 1.4b: HDMI_CLK: up to 340 MHz
	HDMI 1.4b: HDMI_Data: up to 1.7 Ghz
	HDMI 2.0b: HDMI_CLK: up to 150 MHz
	HDMI 2.0b: HDMI_Data: up to 3 Ghz
AC Coupling Capacitors	No AC capacitors allowed
Polarity Reversal	Not allowed
Max Intra-Pair Skew for Source	0.15 * Tbit
Max Inter-Pair Skew for Source	0.20 * Tcharacter
Trace Impedance	100 Ω ±15% differential; 50 Ω ±15% single ended

Figure 15: HDMI Guidelines

DisplayPort

Parameter	Value
Frequency	DisplayPort 1.2: 2.7 GHz (5.4 Gbps)
	DisplayPort 1.4: 4.05 GHz (8.1 Gbps)
	DisplayPort 1.4: 4.05 GHz (8.1 Gbps)
AC Coupling Capacitors	AC capacitors required
Polarity Reversal	No built in support
Max Intra-Pair Skew	20 ps (~TI recommends about 5 mils)
Trace Impedance	100 Ω ±10% differential; 50 Ω ±15% single ended

Figure 16: Displayport Guidelines

PCIe

Parameter	Value
Frequency	PCIe Gen 1: 1.25 GHz (2.5 Gbps)
	PCIe Gen 2: 2.5 GHz (5 Gbps)
	PCIe Gen 3: 4 GHz (8 Gbps)
	PCIe Gen 4: 8 GHz (16 Gbps)
AC Coupling Capacitors	AC capacitors required
Polarity Reversal	allowed
Max Intra-Pair Skew	5 mils
Max Inter-Pair Skew	No Inter-pair specification
Trace Impedance	PCIe Gen 1&2 :100Ω ±5% differential; 50 Ω ±5% single ended
	PCIe Gen 3&4 :85Ω ±5% differential; 42.5 Ω ±5% single ended

Figure 17: PCI express Guidelines

SATA

Parameter	Value
Frequency	SATA-I: 750 MHz (1.5 Gbps)
	SATA-II: 1.5 GHz (3 Gbps)
	SATA-III: 3 Gbps (6 Gbps)
AC Coupling Capacitors	AC capacitors required
Max Intra-Pair Skew	5 mils
Polarity Reversal	Not allowed
Trace Impedance	100 $\Omega \pm 10\%$ differential; 50 $\Omega \pm 10\%$ single ended

Figure 18: SATA Guidelines

7 General High Speed Signal Routing

7.1 Trace Impedance

It is designed to reduce reflections in traces in high speed signals. It has two types:

- **single ended impedance** with reference to ground.
- **differential ended impedance** between two differential pair signal traces.

The trace impedance depend on the protocol(e.g. 50ohm $\pm 5\%$ single ended , 100 ohm $\pm 5\%$ differential for PCIe gen 1 & 2) The trace impedance value depend on geometry of the trace, permittivity of PCB materials and layers surrounding the trace. The trace length for signal pairs should be kept to minimum. The maximum trace ore cable length are subjected to standards as shown in protocol tables.

7.2 Trace Length Matching

- **Intra-Pair Skew** is the difference between two etch lengths of positive or negative lanes of differential pair.
- **Inter-Pair Skew** is the difference between the etch lengths of different differential pairs(e.g. USB 3.0 TX&RX do not need matching)



Figure 19: inter and intra-pair skew

- **Serpentine routing** is used to match intra-pair skew at mismatched ends($C > 3W$ & $B > 2A$) where
 - **A** distance between differential pairs.
 - **B** width of serpentine.
 - **C** length of serpentine.
 - **W** width of the trace.

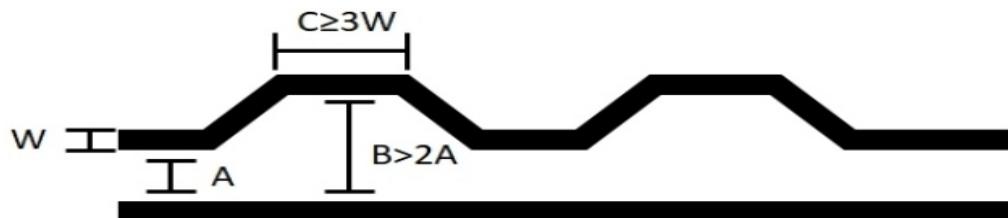


Figure 20: serpentine trace geometry

8 Return Path

The return current take the path with lowest resistance to DC signal in a closed loop system.



Figure 21: return path

For high frequency the lowest impedance path is the reference plane next to the signal(GND plane and power are put on a layer next to the signal layer) this reduce EMI and reduce impedance changes.

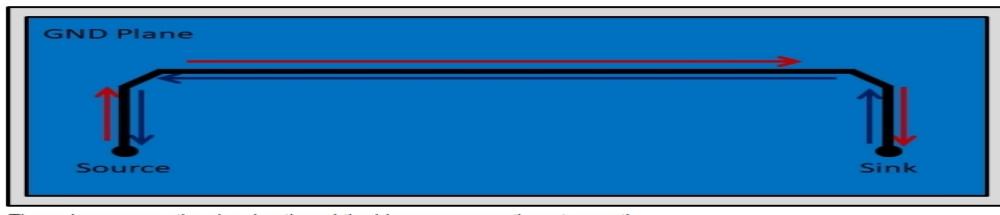


Figure 22: high frequency return path

Signals shouldn't cross over a plane split or void unless it is inevitable because it causes:

- current flow around the split or void.
- emission radiation.
- increase inductance with signal delay.
- crosstalk with near signals
- low integrity(more jitter, less amplitude)

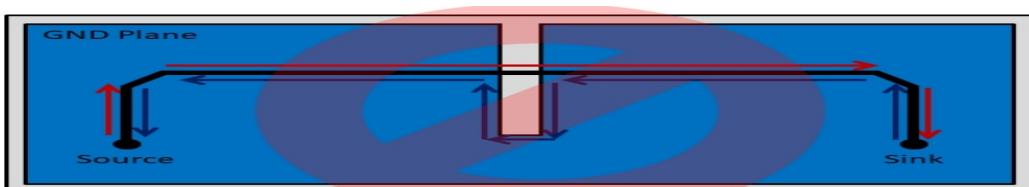
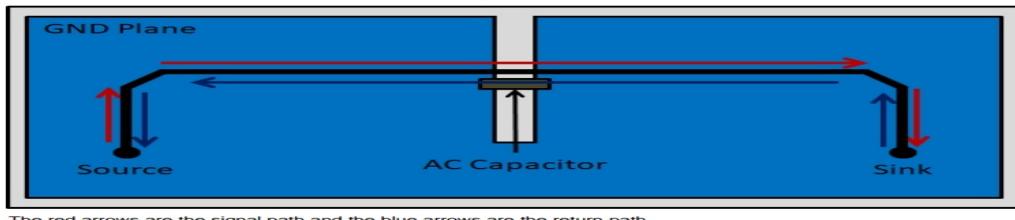


Figure 23: routing across split plane

Stitching capacitors reduce current loop area and impedance discontinuity(1 mu F or lower to plane crossing)



The red arrows are the signal path and the blue arrows are the return path.

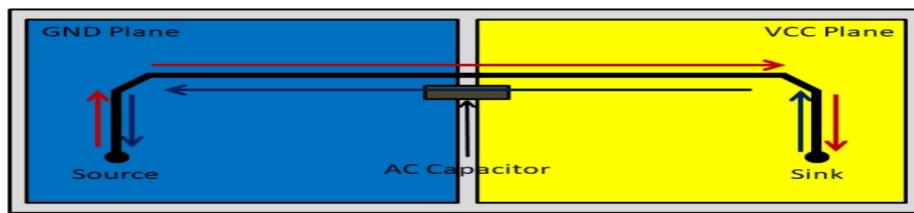
Figure 24: AC capacitor across a split plane

Not referenced planes should not overlap or capacitance, RF emissions, and EMI between them will develop.



The red arrows are the signal path and the blue arrows are the return path.

Figure 25: routing across different reference planes



The red arrows are the signal path and the blue arrows are the return path.

Figure 26: routing across different reference planes with stitch capacitor

8.1 Via Stitch Ground Planes

Via stitch any ground planes for continuous grounding and uniform impedance. Via should be 200 mils center to center or less.

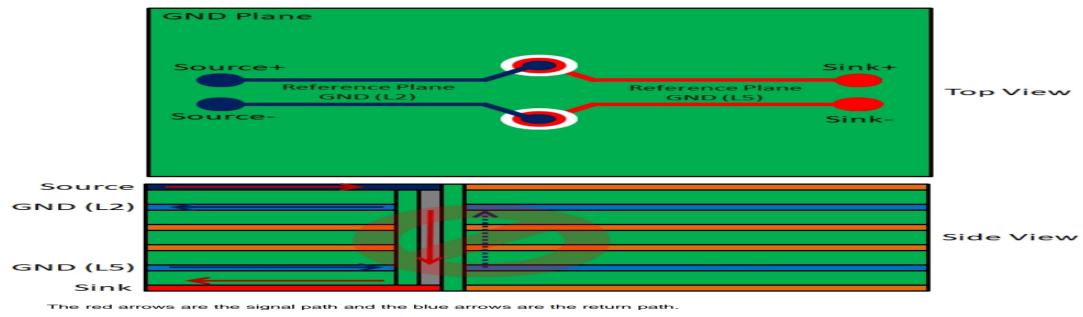


Figure 27: differential pair via return path without GND vias

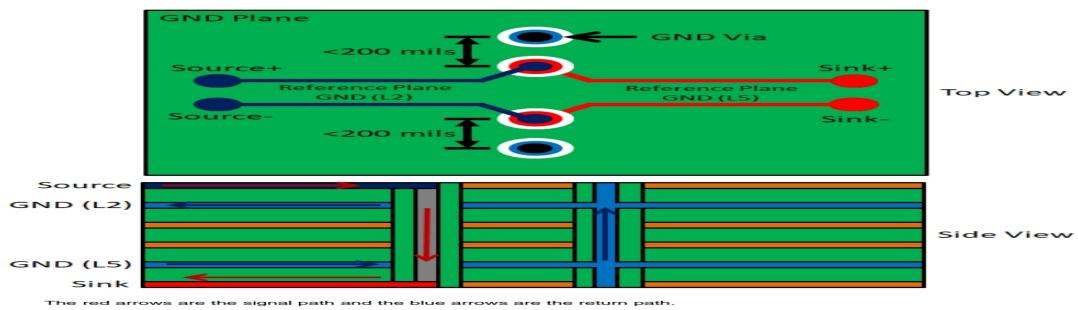


Figure 28: differential pair via return path with GND vias

High speed signal references to power planes is not preferred, but if inevitable We should use GND vias, Vcc vias, and stitching capacitors to maintain the path from sink to source.

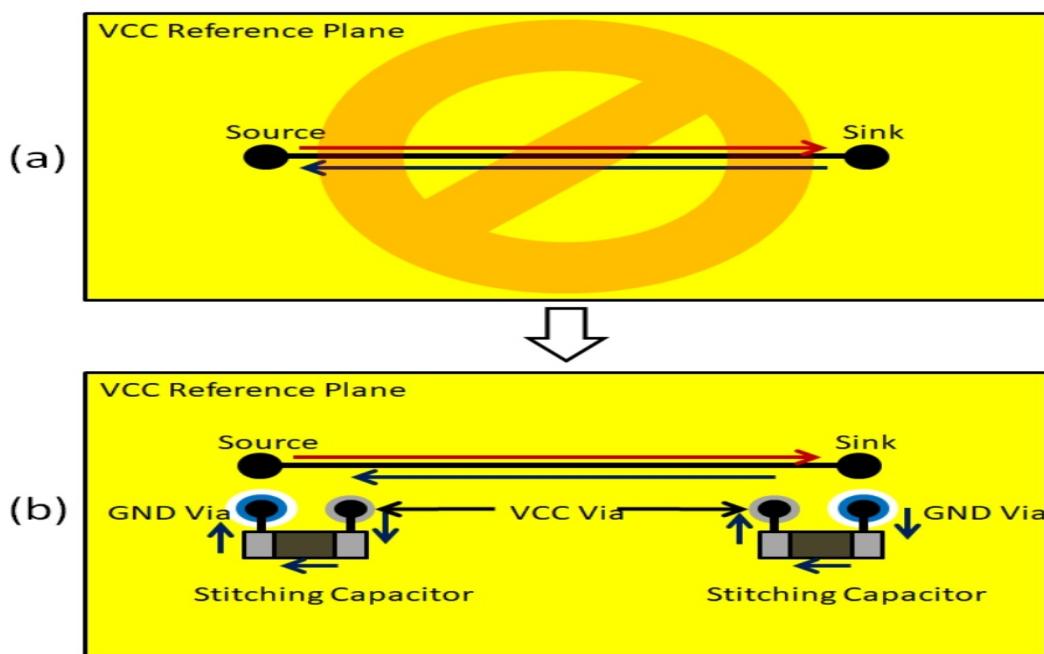


Figure 29: Vcc reference plane

9 High Speed Differential Signal Routing

9.1 Differential Signal Spacing

Differential pairs follow 5W rule(W is trace width) from any other signal or at least 50 mils if the next signal is clock or periodic signal. This is done to reduce cross talk.

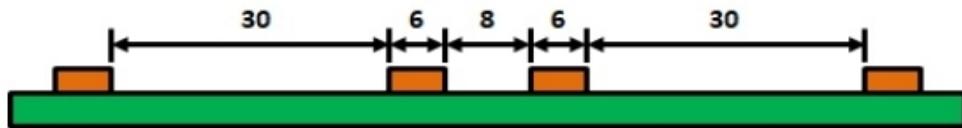


Figure 30: differential pair spacing next to other signals

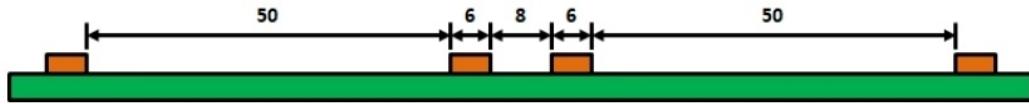


Figure 31: differential pair spacing next to clock or periodic signal

Additional rules for differential pair

- probes should not be placed on high speed differential pairs.
- do not route under or near crystals, oscillators, clock signal, mounting holes or magnetic devices
- differential pairs should be routed on top or bottom layer next to GND layer (>90 mils from the edges, >1.5W from voids in reference plane)

9.2 Symmetry In Differential Pairs

differential pairs must be symmetrical and parallel as short as possible only mismatch is during package break to connector pins.

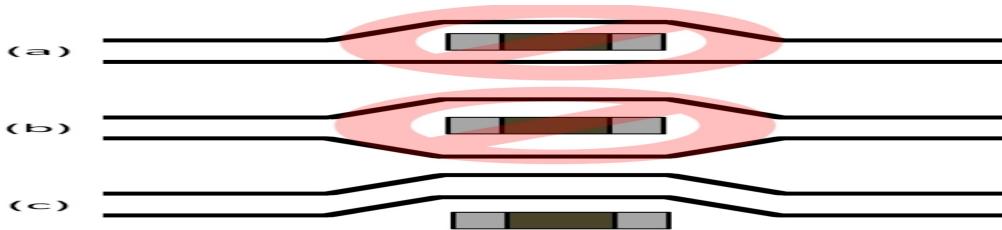
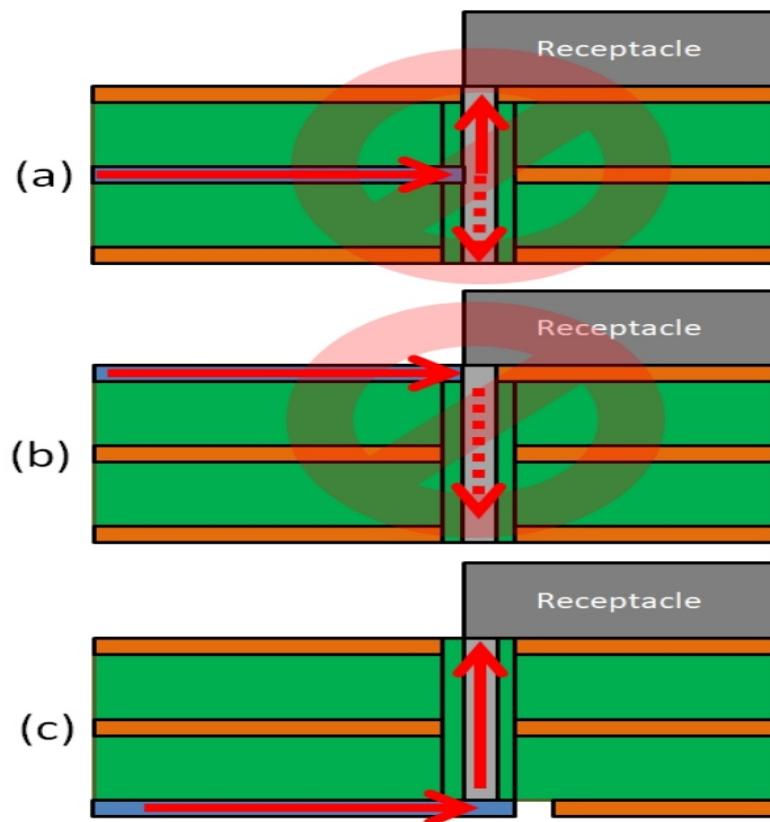


Figure 32: differential pair symmetry

10 protocol

10.1 Connectors And Receptacles

Making a through hole receptacle is preferred in the bottom layer to avoid stubs, such as USB-A. For surface mounted receptacles, such as USB-B & USB-AB is preferred at the top layer which eliminate the need for vias.



(a): Signal coming from the middle of the PCB
 (b): Signal coming from the top of the PCB
 (c): Signal Coming from the bottom of the PCB

Figure 33: receptacles stub mitigation

10.2 Via Discontinuity Mitigation

Via produce capacitive and inductive discontinuity that result in reflections and signal degradation. Longer via stubs should be less than 15 mils or it will cause resonant and insertion loss. back drill with larger diameter than the drill used in original via.



Figure 34: via with long stub



Figure 35: via with short stub



Figure 36: long vias with backdrill

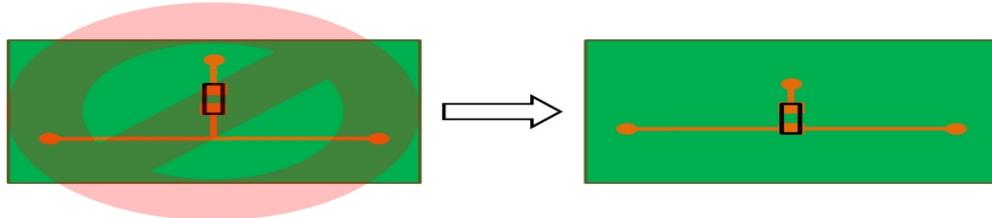


Figure 37: reducing stub length to reduce insertion loss

10.3 Increase Via Anti-Pad Diameter

Increase via anti pad reduce insertion loss and capacitive effect 30 mils The trace connecting the via contain only copper allowed in this area.

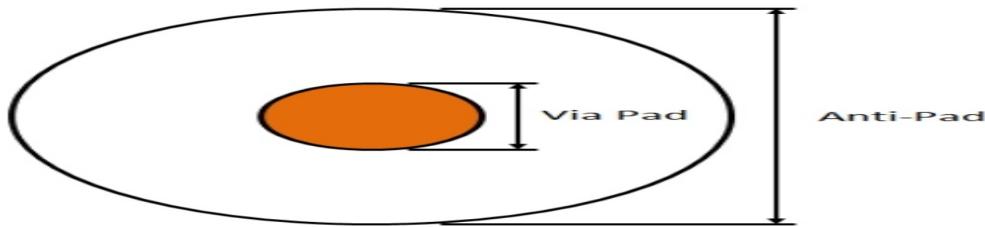


Figure 38: via anti-pad

10.4 Equalize Via Count

Via count on each differential pair is equal in count and spacing. Via length is considered when different lanes are matched.

10.5 Surface Mounted Devices Pad Discontinuity Mitigation

SMD on high speed trace is avoided because it reduce signal quality, If inevitable 0603 size is the maximum limit 0402 size or small is preferred also placement of AC coupling capacitor as symmetric as possible to reduce reflections.

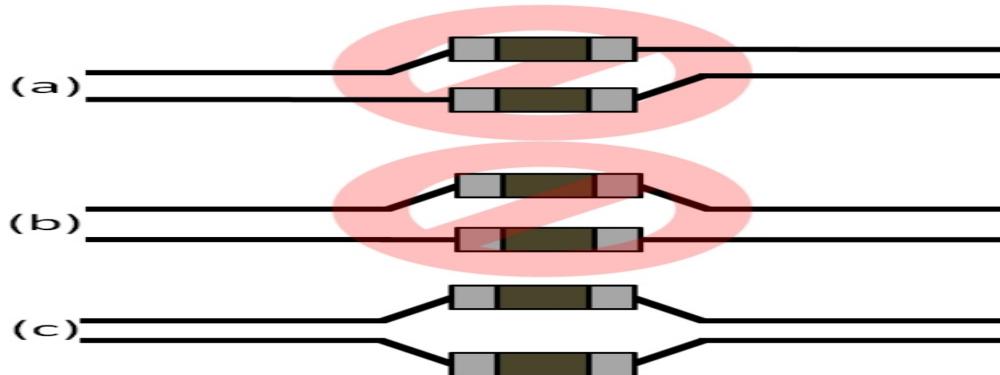


Figure 39: AC coupling capacitor placement

void two layers deep below SMD is required to minimize discontinuity

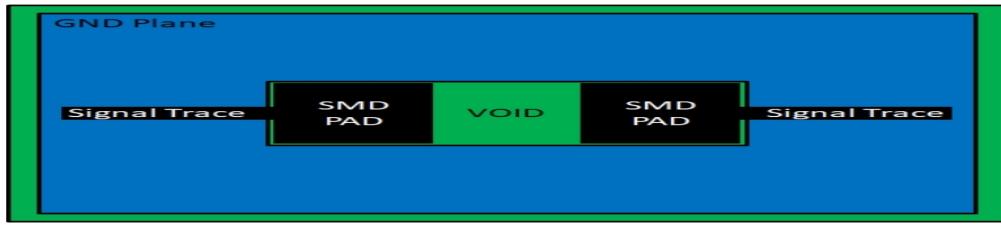


Figure 40: void below SMD

10.6 Signal Buildings

Bending with angle greater than 135 degree is required to ensure the bending is as loose as possible.

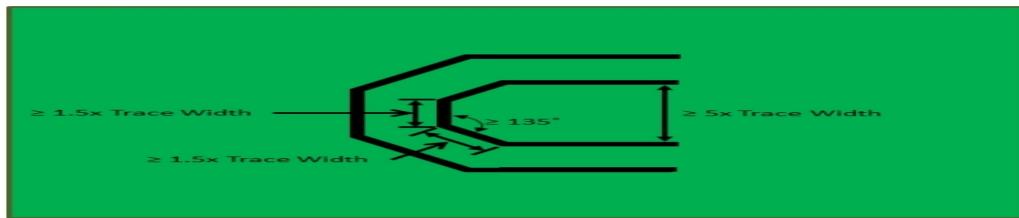


Figure 41: signal bending rules

11 EMI and ESD considerations

ESD and EMI layout rules

- ESD and EMI protection devices is placed as close as possible to the connector
- Keep unprotected trace away from the protected one to minimize EMI coupling.
- 60 % void under ESD and EMI components to reduce losses.
- Use 0402 0ohm resistor for common mode filter(CMF).
- AC coupling capacitor on the protected side of CMF.
- Via for transition should be close to CMF.
- Routing AC coupling capacitor + CMF + ESD protection devices close to the connector.

12 Traffic On The bus

12.1 Protocol Analyzer

It is a device put between the host and device to capture the traffic and display it on GUI(e.g. Mercury T2)

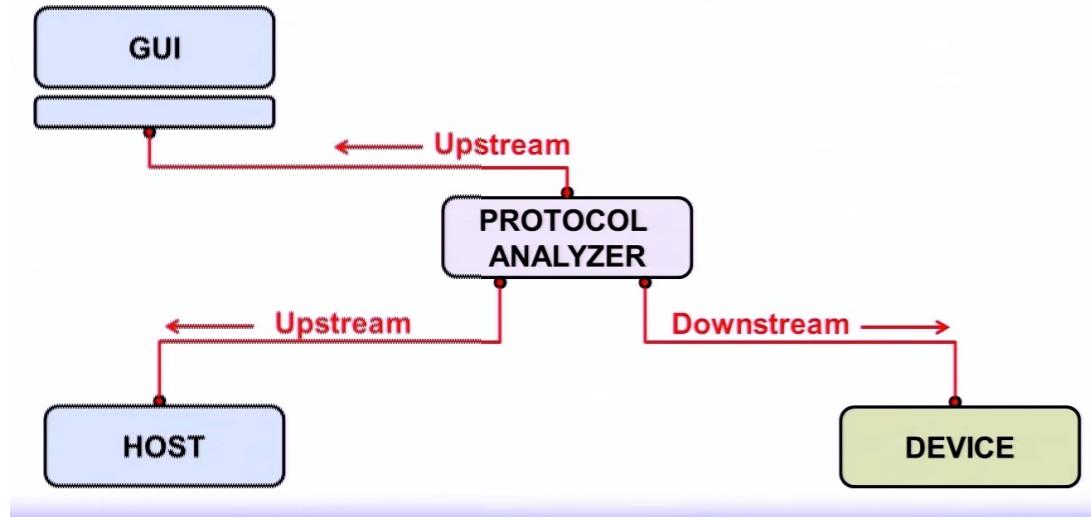


Figure 42: protocol analyzer

The analyzer break down the transfer into transactions and packets.

Transfer	H	Control	ADDR	ENDP	bRequest	wValue	wIndex	wLength	Time Stamp			
0	S	SET	0	0	SET ADDRESS	New address 2	0x0000	0	4 532 037 232			
Transaction	H	SETUP	ADDR	ENDP	T	R	bRequest	wValue	wIndex	wLength	ACK	Time Stamp
0	S	0xB4	0	0	0	H->D	S	D	0x0002	0x0000	0	4 532 037 232
Packet	H	SETUP	ADDR	ENDP	CRC5	Pkt Len		Duration	Idle	Time Stamp		
276	H	0xB4	0	0	0x08	8		133.333 ns	200.660 ns	4 532 037 232		
Packet	H	DATA0	Data	CRC16	Pkt Len			Duration	Idle	Time Stamp		
277	H	0xC3	9 bytes	0xD768	16			266.667 ns	349.330 ns	4 532 037 566		
Packet	H	ACK	Pkt Len					Duration	Time	Time Stamp		
278	D	0x4B	6					100.000 hs	124.034 us	4 532 038 182		

Figure 43: protocol analyzer GUI

12.2 Packets

Packets combine in specific sequence to form transactions and transfers between device and host. Types

- Token
 - (IN, OUT, SETUP) to start transaction.

- SOF(start of frame) time base and keep alive on the bus.
- **Data** DATA0, DATA1 alternated during a transfer.
- **Hand Shake** ACK, NAK, STALL report outcome of a transaction.
- **Special**
 - SSPLIT, CSPLIT for split transactions.
 - PING, NYET used by bulk transfer.

Token, data, special (except NYET) have CRC protection.

12.3 Control Transfer Type

Used during device enumeration to send a request for providing configuration data (EP0 IN) or accept configuration setting (EP0 OUT). It is made of three stages each build over one or more transaction.

12.3.1 SETUP Stage

It has one transaction.

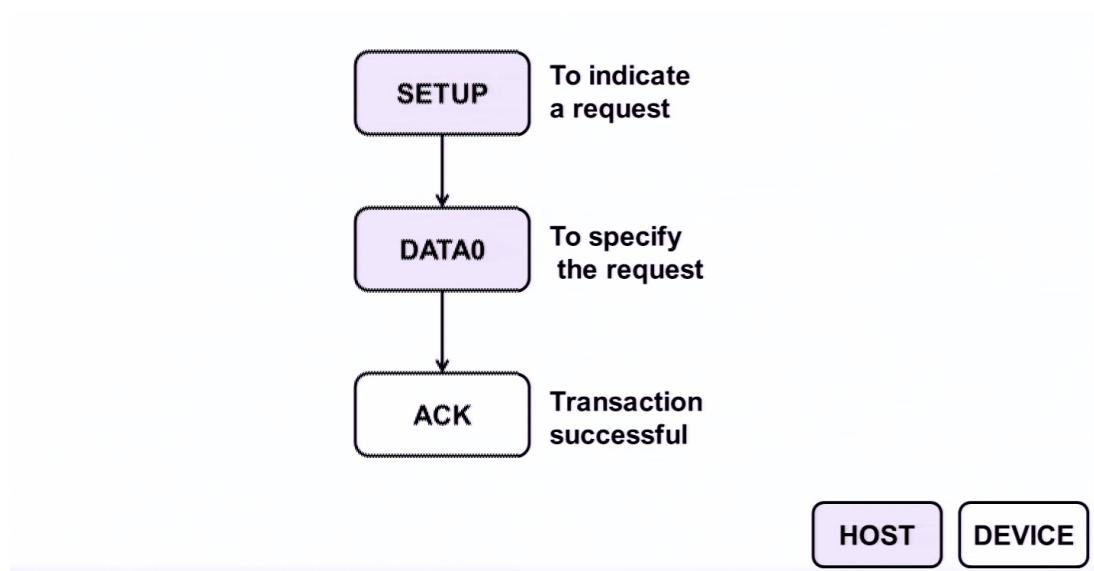


Figure 44: setup stage transaction

12.3.2 DATA Stage

It is an optional stage which receive data request or send data settings. Formed by more than one transaction.

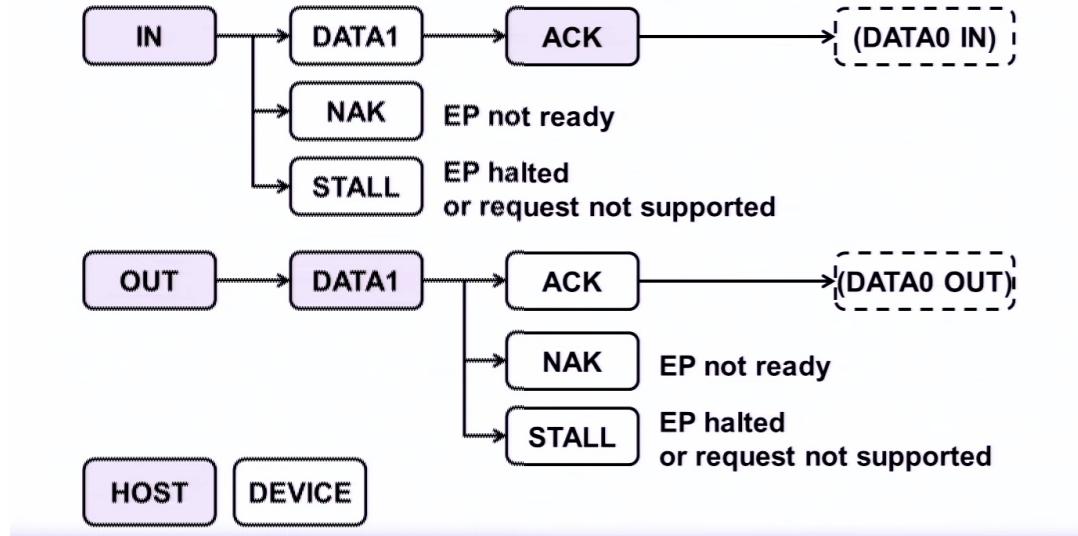


Figure 45: data stage transactions

12.3.3 STATUS Stage

It report the outcome of request. Formed by one transaction.

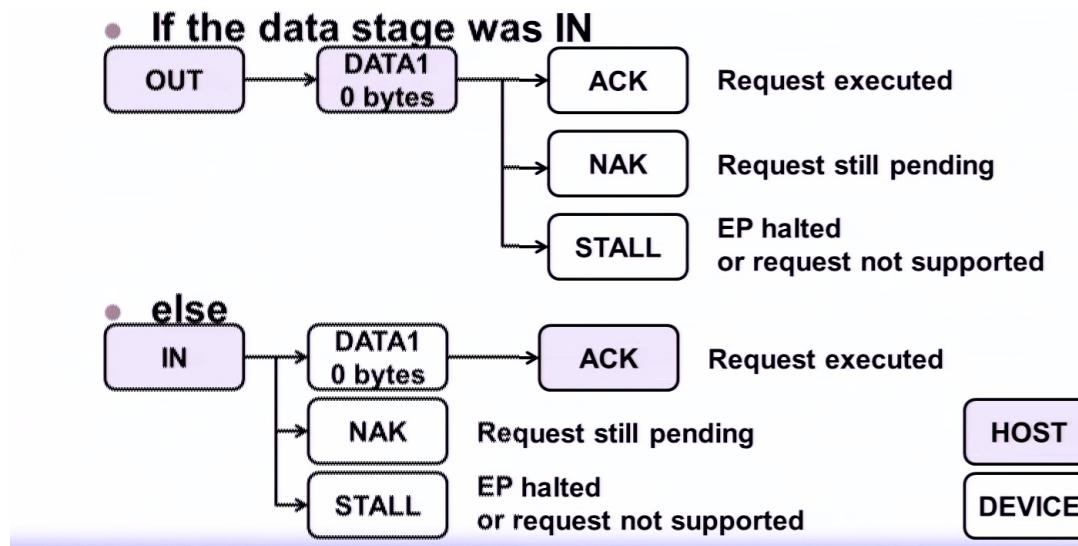


Figure 46: status stage transactions

For better visualization to all these steps we will use the help of a protocol analyzer.

Transfer	F	Control	ADDR	ENDP	bRequest	wValue	wIndex	wLength
12	S	SET	2	0	SET_CONFIGURATION	New Configuration 1	0x0000	0

- ..breaks down in setup and status stages (no data stage):

Transaction	F	SETUP	ADDR	ENDP	T	D	Tp	R	bRequest	wValue	wIndex	wLength	ACK	
57	S		0xB4	2	0	0	H->D	S	D	0x09	0x0001	0x0000	0	0x4B
58	F	IN	ADDR	ENDP	T	Data			ACK		Time		Time Stamp	
			0x96	2	0	1	0 bytes			0x4B	19.000 ms	5.	965 815 116	

- Setup stage (transaction 57):

Packet	H	F	Sync	SETUP	ADDR	ENDP	CRC5	EOP	Pkt Len		
302			00000001	0xB4		2	0	0x15	250.000 ns 35 Bits (5 Bytes)		
303	H	D	Sync	DATA0			Data	CRC16	EOP	Pkt Len	
			00000001	0xC3		0	8 bytes	0xE4A4	266.660 ns 100 Bits (13 Bytes)		
304	H	D	Sync	ACK			EOP		Pkt Len	Duration	Time
			00000001	0x4B		2	50.000 ns	19 Bits (3 Bytes)		1.583 us	988.200

Figure 47: using analyzer to visualize the control transfer

- Status stage (transaction 58):

Packet	H	F	Sync	IN	ADDR	ENDP	CRC5	EOP	Pkt Len		
306	H	D	Sync	00000001	0x96	2	0	0x15	250.000 ns 35 Bits (5 Bytes)		
307	H	D	Sync	DATA1			Data	CRC16	EOP	Pkt Len	
			00000001	0xD2		0	8 bytes	0x0000	250.000 ns 35 Bits (5 Bytes)		
308	H	D	Sync	ACK			EOP		Pkt Len	Duration	Time
			00000001	0x4B		2	50.000 ns	19 Bits (3 Bytes)		1.583 us	18.9

Figure 48: (cont.)status stage with no data stage

Transfer	F	Control	ADDR	ENDP	bRequest	wValue	wIndex	Descriptors
1	S	GET	2	0	GET_DESCRIPTOR	DEVICE type	0x0000	DEVICE Descriptor

- ..breaks down in setup, data and status stages:

Transaction	F	SETUP	ADDR	ENDP	T	D	Tp	R	bRequest	wValue	wIndex	wLength	ACK	
4	S		0xB4	2	0	0	D->H	S	D	0x06	0x0100	0x0000	8	0x4B
5	F	IN	ADDR	ENDP	T	Data			ACK		Time		Time Stamp	
			0x96	2	0	1	8 bytes			0x4B	999.868 us	5.	907 812 632	
6	F	OUT	ADDR	ENDP	T	Data			ACK		Time		Time Stamp	
			0x87	2	0	1	0 bytes			0x4B	1.000 ms	5.	908 812 500	

- Data stage (transaction 5):

Packet	H	F	Sync	IN	ADDR	ENDP	CRC5	EOP			
90	H	D	Sync	00000001	0x96	2	0	0x15 266.660 ns 3			
91	H	D	Sync	DATA1			Data	CRC16	EOP		
			00000001	0xD2		0	8 bytes	0xEAEC	250.000 ns 99		
92	H	D	Sync	ACK			EOP		Pkt Len	Duration	Time
			00000001	0x4B		2	50.000 ns	19 Bits (3 Bytes)		1.583 us	18.9

Figure 49: another example with data stage

- **Status stage (transaction 6):**

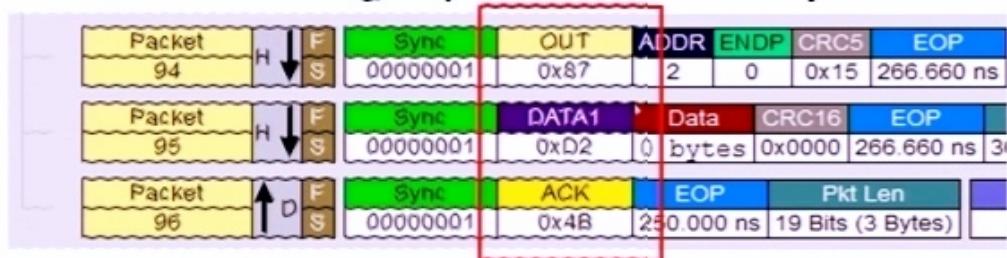


Figure 50: (cont.)status stage

12.4 Interrupt Transfer Type

IN endpoint (polled or asynchronous)

- DATA0 and DATA1 alternated
- Data payload according to the endpoint max packet size – more than one transaction if needed

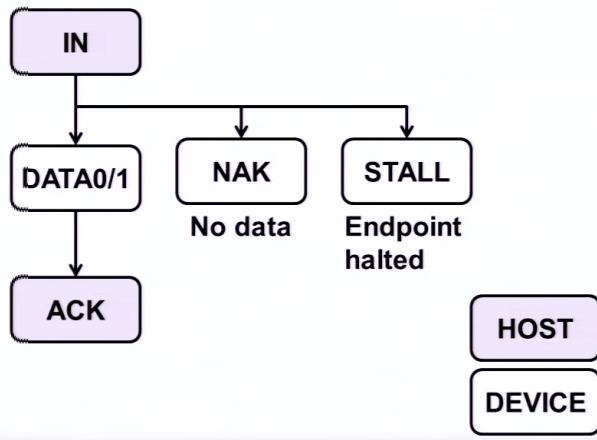


Figure 51: IN interrupt transaction

OUT endpoint (max period per enumeration)

- DATA0 and DATA1 alternated
- Data payload according to the endpoint max packet size – more than one transaction if needed

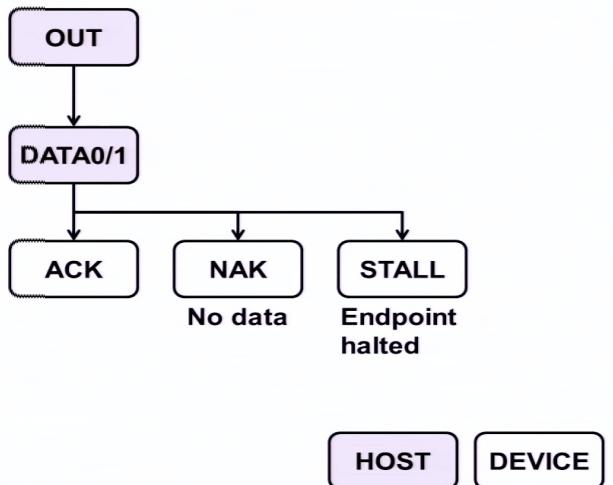


Figure 52: OUT interrupt transaction

The HID class uses interrupt transfer.

- **to toggle the LED**

When the GUI is idle, the host polls the IN interrupt endpoint (EP1 IN in this case) and keeps receiving NAKs

Toggle LED(s) is pressed on the GUI: interrupt OUT transfer (to EP1 OUT in this case) to toggle the LED.

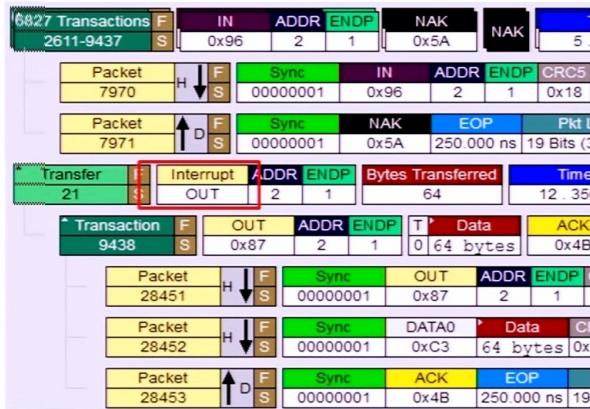


Figure 53: LED toggle bus traffic

- **to get the pushbutton state**

When Get Pushbutton State is pressed on the GUI, first a transfer is sent (to EP1 OUT in this case) to put the pushbutton state in EP1 IN (in this case)

Following that, an interrupt IN transfer (to EP1 IN in this case) is sent to retrieve the pushbutton state

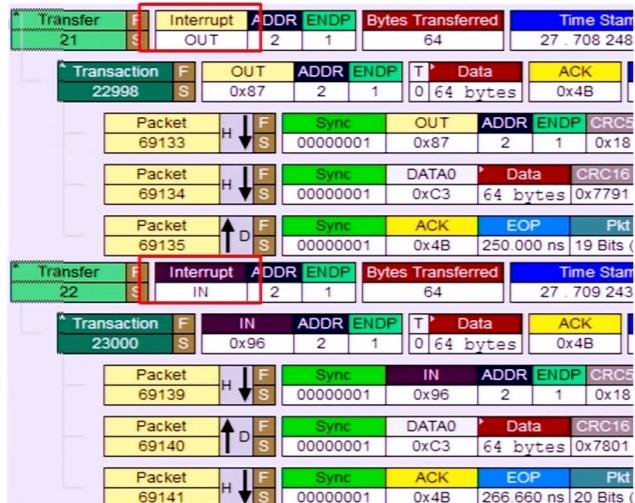


Figure 54: bushbutton bus traffic

12.5 Bulk Transfer Type

Transaction structure is similar to interrupt, but high speed OUT uses PING protocol which avoid waste in the bandwidth by the host sending OUT token followed by DATA0 or DATA1 to get NAK by the device. The device can answer with NYET packet which accept current data, ask to be PINGed and ACK before the next data transfer attempt.

**MPLAB Harmony MSD demo:
PIC32MZ Flash as a logical
SCSI storage**

OUT+NYET

ACKed PING
+ DATA OUT

Transfer	H	Bulk	ADDR	ENDP	* Mass	CBSU In Len	SCSI CDE
26	S	OUT	2	1	Storage	0x00000024	
Transaction	H	OUT	ADDR	ENDP	T	Data	NYET
101	S	0x87	2	1	0	31 bytes	0x69
Packet	H	OUT	ADDR	ENDP	CRC5	Pkt Len	
1220	S	0x87	2	1	0x18	8	
Packet	H	DATA0	*	Data	CRC16	Pkt Len	
1221	S	0xC3	31 bytes	0x2036	39		
Packet	H	NYET	*	Pkt Len	Duration	Tim	
1222	S	0x69	6	100.000 ns	35.00		
Transfer	H	Bulk	ADDR	ENDP	* Mass	CBS	
29	S	OUT	2	1	Storage	0x0	
Transaction	H	PING	ADDR	ENDP	ACK		
104	S	0x2D	2	1	0x4E		
Packet	H	PING	ADDR	ENDP			
1231	S	0x2D	2	1			
Packet	H	ACK	*	Pkt Len			
1232	S	0x4B	6	1			
Transaction	H	OUT	ADDR	ENDP	T	D	
105	S	0x87	2	1	1	31	
Packet	H	OUT	ADDR	ENDP			
1233	S	0x87	2	1			
Packet	H	DATA1	*	Data			
1234	S	0xD2	31 bytes				
Packet	H	NYET	*	Pkt Len			
1235	S	0x69	6	1			

Figure 55: bulk transfer analyzer

13 Split Transaction

It allow the hub to separate high speed from low and full speed traffic

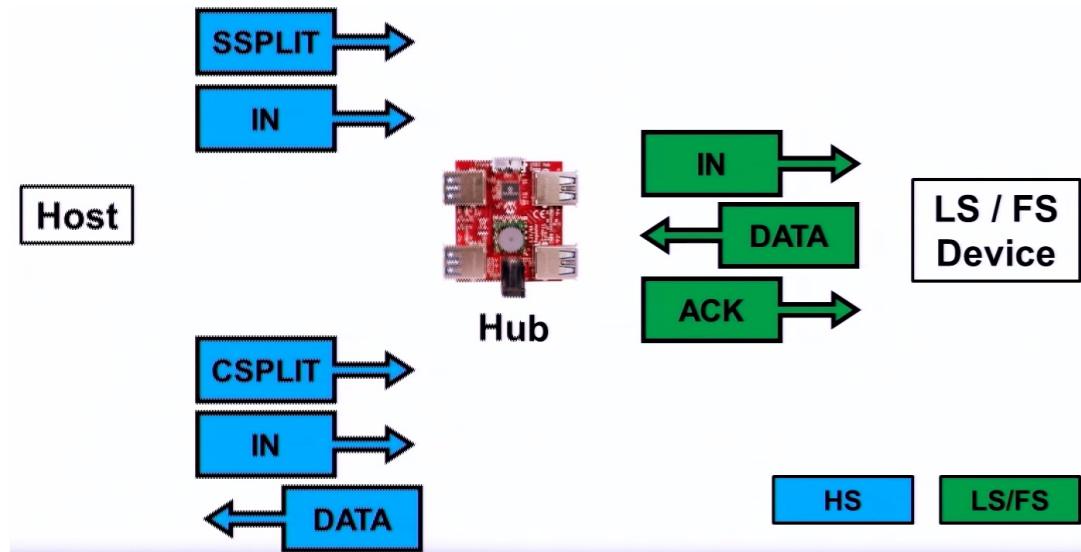


Figure 56: split transaction

14 Device Enumeration

It is a process following device attachment to the bus.

1. Device plugged to downstream port detect the port Vbus and assert pull up to either D+ or D-.
2. Device then detected by the hub and start signaling to reset the device with established speed.
3. EP0s are enabled and the host uses control transfer request.
 - Read device power and communication requirements.
 - Assign unique address to the device and accept its requirements if sustainable.
4. The device then is enabled and communication starts.

15 Descriptors

Host information is stored in data structure called descriptors during enumeration. It has two types:

- Class descriptors which is specific to certain applications.
- Standard descriptors common for every device.
 - device one descriptor per device.
 - configuration at least one descriptor per device.
 - * Interface at least one per configuration.
 - Endpoint at least one per interface.
 - String which provide human readable information as device serial number

Each descriptor has many fields for development and debugging.

- **Device Descriptor**

- bDeviceClass: the USB device class code(0 if class specified in the interface descriptor).
- bMaxPacketSize: maximum packet size for EP0.
- idVendor: vendor ID.
- idProduct: product ID.
- bNumConfigration: number of configurations for this device.

- **Configuration Descriptor**

- bNumInterfaces: number of interfaces in this configuration.
- Configuration Value: value used by the host to select this configuration.
- bmAttributes: self or bus-powered, remote wakeup enabled.
- bMaxPower: maximum current sank from Vbus after configuration.

- **Interface Descriptor**

- b Interface Number: number of interfaces.
- bNumEndpoints: number of endpoints in this interface without EP0.
- bInterfaceClass: USB interface class code.

- **Endpoint Descriptor**

- bEndPointAdress: endpoint number and direction.
- bmAttributes: transfer type used by the endpoint.
- wMaxPacketSize: maximum data packet size.
- bInterval: endpoint access period in micro frames.

- Descriptor values example

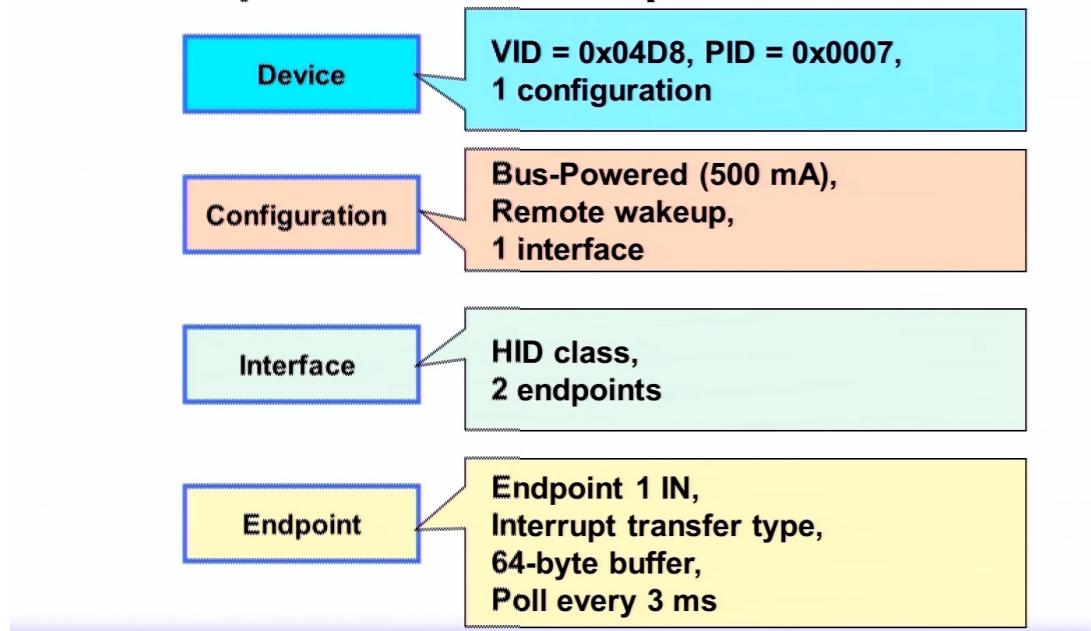


Figure 57: describtor values example

16 Low Voltage Differential Signaling (LVDS)

With the fast improvements and growth in technology, the demand for a high speed and bandwidth has increased substantially. The data transmission standards that already existed (i.e. SCSI, RS-422, and RS-485) were not good enough to transfer data as fast as desired, so the need for an improvement led in the year of (1994) to the creation of LVDS or the low voltage differential signaling, which is a high speed (greater than 155.5 Mbps), general purpose interface with low power consumption that have a variety of applications. We will take a closer look at its characteristics with the study of the DS90C031 and the DS90C032 which are LVDS Quad 5V CMOS Differential line Driver and Receiver.

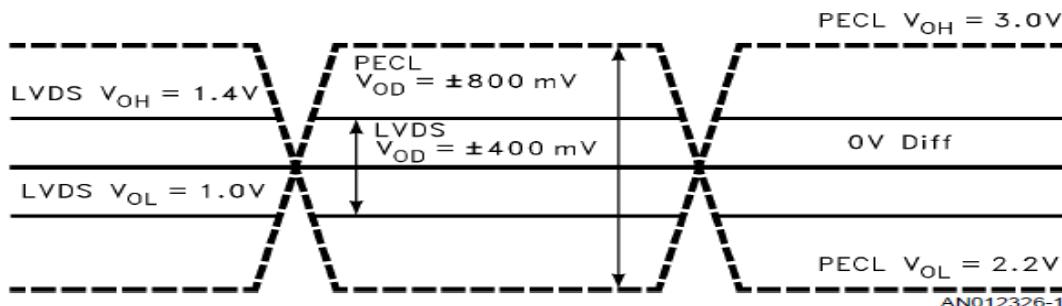


Figure 58: PECL vs LVDS Signal Swing

The Standard of the LVDS is based on one of the two standards, either the *ANSI/TIA/EIA-644* or the IEEE SCI (Scalable Coherent Interface). Because of its differential scheme, it is considered much less susceptible to the common mode noise (CMN), and that means that the receiver only responds to the differential voltage.

It does not depend on the power supply (i.e. 5V power supply), that means there is an easy way of changing the supply to values like +3.3V, and +2.5V or maybe even lower and still maintain the same performance level. Unlike the LVDS, ECL and PECL technologies depend on the power supply. While in the case of LVDS, it is more dependent on the technologies that derive the LVDS drivers. LVDS technology has an aggregate bandwidth in the Gbps range with keeping it a loss-less media. Depending on the kind of media being derived, the data rates is possible in the range of 500 to 1k Mbps.

17 Signaling Levels

From its name, it is obvious that LVDS is characterized with low voltage swing in case of comparing it to other industrial data transmission standards. Figure58 illustrates the signal levels and also compare between the PECL and the LVDS.

The Capability of the LVDS to provide low power and to maintain high data rates is a result of utilizing low voltage swings. We can notice that the signaling of the LVDS is about 50% smaller than that of the PECL. Electromagnetic interference (EMI) effects are also reduced because the swing of the signal is much smaller than the TTL, CMOS, or PECL, and it's due to the soft transitions, current mode drivers, low switching currents, and the use of differential data transmission.

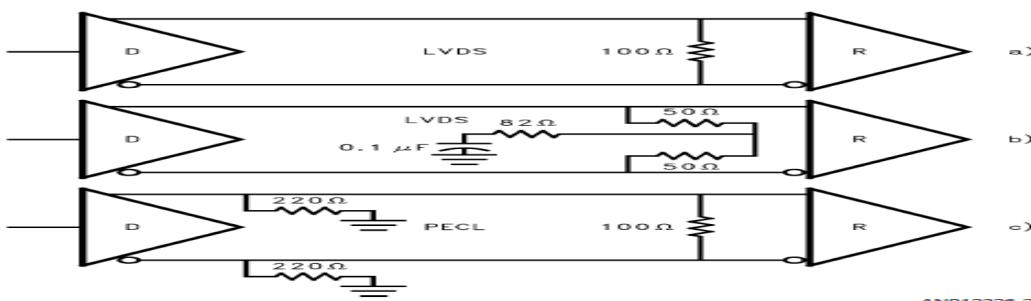


Figure 59: Termination Schemes

18 LVDS Termination

In order to obtain its diverse features, LVDS uses constant current mode driver. The nominal value of the current source is 3.5mA (of a maximum 4.5mA) and in order not to cause any reflections and also to reduce any undesired electromagnetic emissions, a termination impedance has to be added and is typically 100Ω . This termination resistance is very important as it causes a differential output voltage (VOD) to appear at the receiver input across the termination resistor load. see figure59A. It is not recommended to transmit data from the driver to the receiver without the use of termination resistance. In order to reduce the effects of electrical parasitics, we derive the use of PCB's surface mount components.

Using a termination resistance is considered the simplest method to implement most applications, but it is not the only method of doing it. We can use a capacitor to ground with a cable damping resistor as illustrated in fig59B. The advantage of this method is that it gives more common mode termination, but it is not popular approach.

If we want to make a comparison between the LVDS and the ECL or PECL terminations, we can find that the later ones require even more complex termination than that of the LVDS. For example: the PECL drivers require along with the 100Ω resistor across the output, a more extra 220Ω pull down resistor with each driver as illustrated in fig59c, and this method increase the cost of the system as it requires extra PCB space.

19 Common Mode Range

There is +1.2V driver voltage offset in any typical LVDS and the receiver of an LVDS can tolerate a minimum ground shift of $\pm 1V$ between the receiver's and the driver's grounds, and the common mode voltage at the receiver input pins with respect to ground is known as the sum of driver offset voltage, ground shifting and any longitudinal coupled noise. The receiver's common mode ranges from +0.2V to +2.2V, while the recommended input voltage for the receiver ranges from ground to +2.4V.

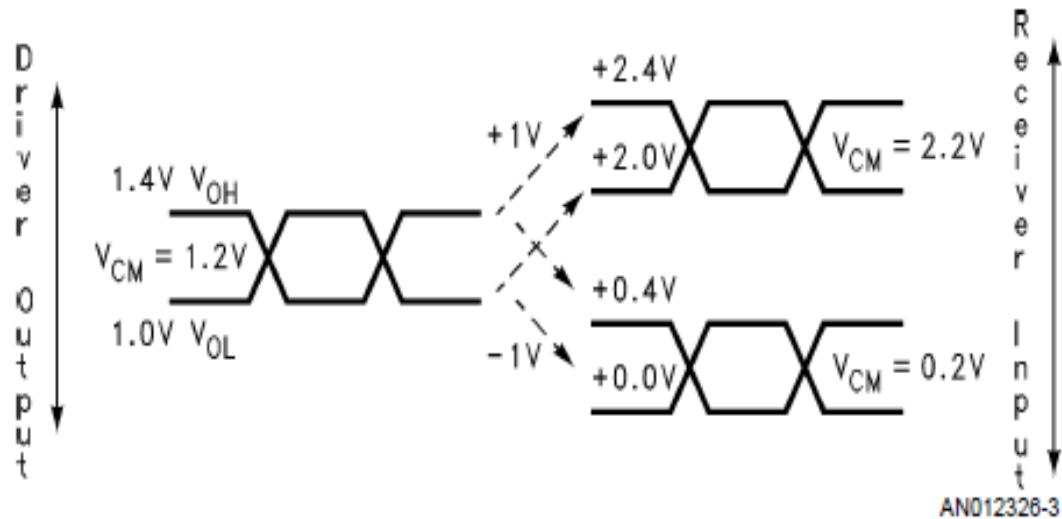


Figure 60: Common Mode Voltage Range

For example, if the driver has a $V_{OH} = 1.4V$ and $V_{OL} = 1.0V$ (with respect to the ground of the driver), and a ground shift = $+1.0V$ (the ground of the driver is $+1.0V$ higher than the ground of the receiver), this will produce $V_{IL} = 1.4 + 1.0 = +2.4V$ and $V_{IH} = 1.0 + 1.0 = +2.0V$ and $V_{CM} = +2.2V$. in the same way, but with the ground shift = $-1V$ will produce $V_{IL} = 1.4 - 1.0 = +0.4V$ and $V_{IH} = 1.0 - 1.0 = 0.0V$ and $V_{CM} = +0.2V$. This is shown in fig(60)

20 Failsafe Feature

This is a receiver feature, which allows us to set the output at a specific fault condition to a known logic state (HIGH in this case). This occurs whenever the receiver's inputs are one of three cases, shorted, opened, or terminated.

The first issue has to do with the open circuit. In some conditions, we have quad receivers (i.e. in the DS90C032 LVDS), but not all of them been used, so the unused are open circuits. If the device doesn't support the failsafe with open input (fig61)A, any external noise can trigger the output once it exceeds the receiver's threshold and cause error in the communication line. And fortunately, the DS90C032 supports the failsafe in case of open input circuit, the receiver output will be HIGH.

The second issue is the shorted circuit, if in some conditions the input accidentally

gets shorted fig(61)B, the receiver output will be HIGH as well and not a floating point (unknown state).

The third and last issue is that if the driver is either in Tri-State (Powered off) or even is removed while the receiver is still powered on. In this case, the receiver will also provide a HIGH logic in the same conditions mentioned above.

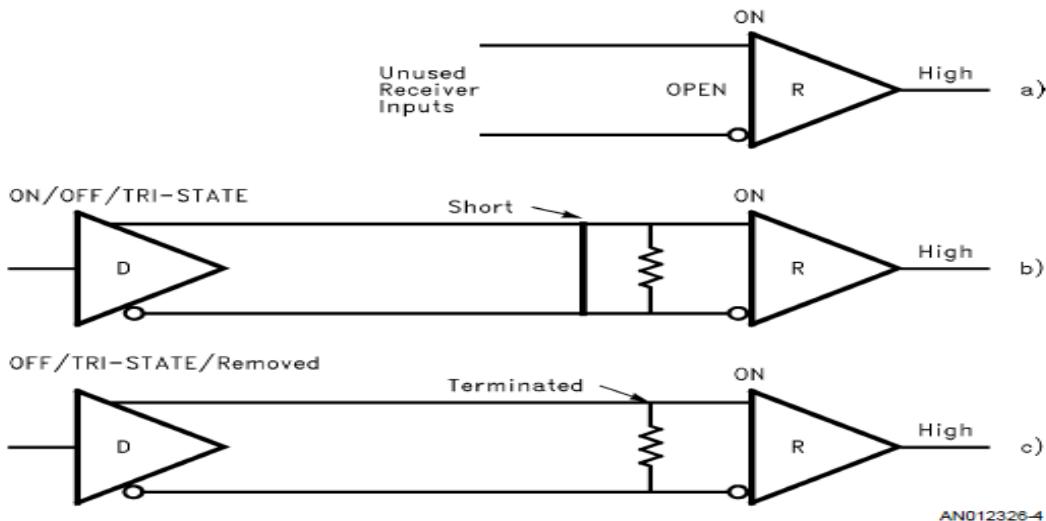


Figure 61: Failsafe Operation

21 Industry Standards for various LVDS Technologies

Fortunately, there is a wide variety of VLDS technologies to choose from according to the application. fig(62) illustrates some of the different technologies used and their parameters. We will take the Point-to-Point, Bi-Directional, and Multi-Drop configurations for these technologies into our account.

	Industry Standard	Maximum Data Rate	Output Swing (V_{OD})	Power Consumption
LVDS	TIA/EIA-644	3.125 Gbps	± 350 mV	Low
LVPECL	N/A	10+ Gbps	± 800 mV	Medium to High
CML	N/A	10+ Gbps	± 800 mV	Medium
M-LVDS	TIA/EIA-899	250 Mbps	± 550 mV	Low
B-LVDS	N/A	800 Mbps	± 550 mV	Low

Figure 62: Industry Standards for Various LVDS Technologies

22 Point-to-Point Configuration

From its name, it is obvious that point-to-point configuration has only one driver and one receiver and therefore, this configuration requires a minimal number of transitions (i.e. from circuit board to a cable). Due to that the signal path impedance will be well-controlled, allowing very high rates of signaling. The configuration has to be terminated with a 100Ω resistor at the far end. This termination provides a differential voltage for the current mode driver (VOD), and the driver in turn can drive a twisted pair (TWP) wire (10m) at about 155 Mbps (77 MHz). see fig(63).

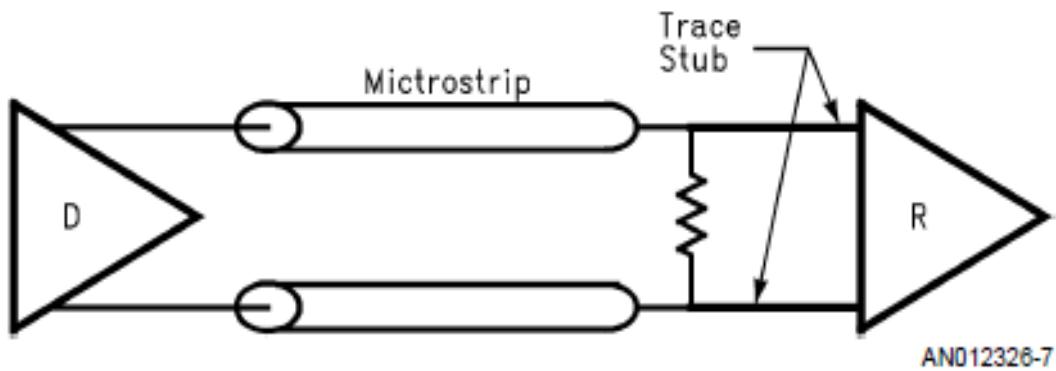


Figure 63: Point to point Configuration Using LVDS

23 Bi-Directional Application on One TWP

As the name dictates, the bi-directional configuration, also called "half-duplex" allows the signal to flow in either direction (see fig(64)), but only at one direction at a time over the single TWP. Because there are two receivers, we need two termination resistors at both ends. But, because the drivers are current mode devices (The Current of the device is 4 mA) The 2 resistors are calculated as parallel which would cut the signal in half ($100\Omega \parallel 100\Omega = 50\Omega$), and this would result a reduction in the noise margin (to about 25 mV) which attenuate the driver output swing severely, so this configuration is not recommended with the standard LVDS devices.

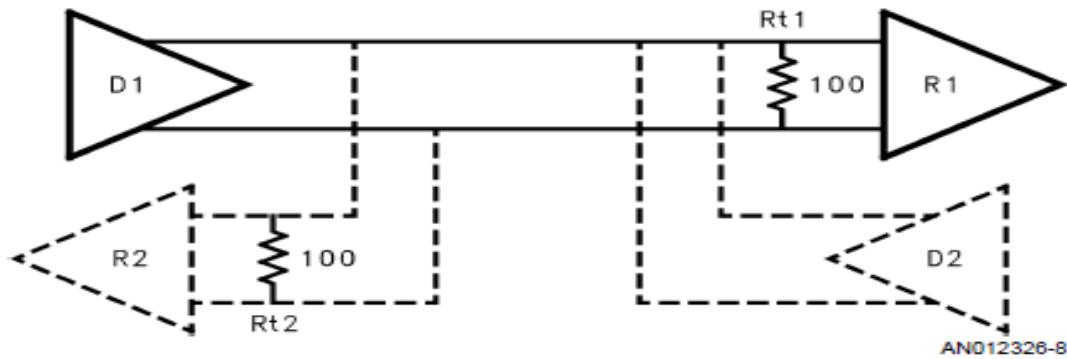


Figure 64: Bi-Directional Application Over One Pair of Twp

24 Multi-Drop Configuration

Multi-drop configuration has more than 2 receivers (up to 10 and beyond) connected to the bus (see fig(65)).

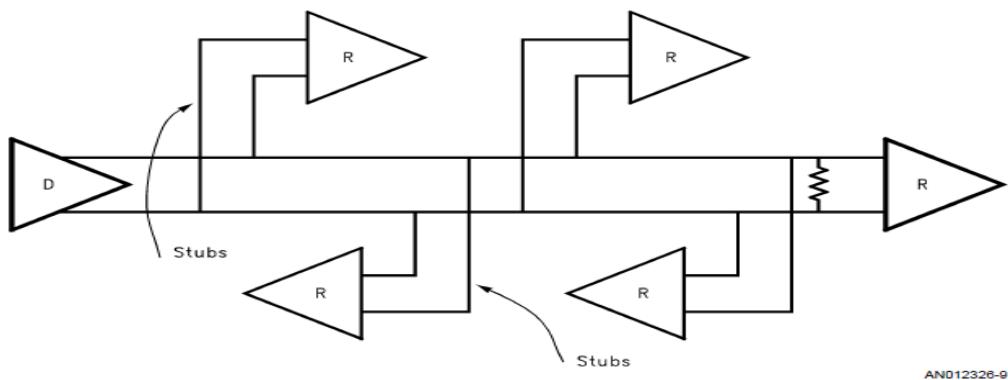


Figure 65: Multi-Drop Configuration Using LVDS

25 Signal Quality Across Cable

There are different types of determining the signal quality of the transmission media. Some of them are jitter, bit error rate, ratio of rise time, eye pattern, and unit interval.

26 conclusion

usb stands for Universal serial bus and was invented and standardized in 1995. the first motivation is to allow the peripheral automatically configured when plugged in your own

computer known as (plug and play) with out need to restart your computer.

usb has grown and become the most known interface for many industrial embedded application.and the different companies after that move the direction of saving power and reducing consumption by providing usb system sleep mode.

High speed signaling require fine specifications like in the PCB design the trace impedance should be matched, the trace should not cross over a split plane or a void if inevitable stitching capacitors is used, Vias is used to connect planes for a uniform impedance. spacing between the traces should be considered, especially if the trace carry a clock signal. Stubs should be avoided or back drilled to reduce EMI and ESD. protocol analyzer provide an efficient way to capture the traffic on the bus. Each control type has transaction steps for communication between the host and the device. Enumeration is a process follows attaching the device to begin the communication. Descriptors are data structures used to store host informations.

LVDS is a great technology that was developed in 1994 and it is able of transferring data at high rates, which helped with the advance of technology.

It has got a variety of industrial standards like the LVDS, LVPECL, CML, M-LVDS, and the B-LVDS. It also come with different configurations like the point-to-point configuration which has only one driver and one receiver, and the Bi-Directional which has 2 drivers and 2 receivers, and the Multi-Drop configuration which got only one driver and multiple receivers (drop points) that can be more than 10 receivers.

The Failsafe feature of this technology enables it to predict any future bugs and fix them by having the input at the receiver high at all time (in case of any faulty conditions).

The device signaling levels is way too lower than any family of technologies like the PECL one and its termination is so simple that it uses only a 100Ω resistor and that has a role in the rejection of common mode signals

References

- [1] Johnson, Howard W., and Martin Graham. High-speed Signal Propagation: Advanced Black Magic.Upper Saddle River, NJ: Prentice Hall/PTR, 2003.
- [2] Hall, Stephen H., and Howard L. Heck. Advanced Signal Integrity for High-speed Digital Designs.Hoboken, N.J.: Wiley , 2009.
- [3] Heck, Howard. USB 3.1 Electrical Design. USB 3.1 Developer Days, 2014.
- [4] Stephen C. Thierauf. High-Speed Circuit Board Signal Integrity. ISBN-13: 978-1580531313.
- [5] Texas Instruments DisplayPort Design Guide
- [6] Texas Instruments HDMI Design Guide
- [7] <https://docs.microsoft.com/en-us/windows-hardware/drivers/usbcon/standard-usb-descriptors>
- [8] <https://docs.microsoft.com/en-us/windows-hardware/drivers/usbcon/usb-endpoints-and-their-pipes>
- [9] <https://docs.microsoft.com/en-us/windows-hardware/drivers/usbcon/usb-bulk-and-interrupt-transfer>
- [10] <https://docs.microsoft.com/en-us/windows-hardware/drivers/usbcon/usb-control-transfer>
- [11] <https://docs.microsoft.com/en-us/windows-hardware/drivers/usbcon/how-to-send-a-usb-interrupt-transfer-uwp-app->
- [12] <https://techcommunity.microsoft.com/t5/microsoft-usb-blog/how-does-usb-stack-enumerate-a-device/ba-p/270685>