Extron

Understanding USB in Professional AV Environments

ABSTRACT

Like all technology, Universal Serial Bus - USB has evolved over time. Originally developed to minimize the number of peripheral ports on a PC, USB is now a comprehensive set of interface specifications for high speed wired communication between a wide range of devices, with or without a computer. USB is increasingly being used in AV systems for much more than basic keyboard and mouse connections. It is used for connecting storage devices, extending and distributing audio and video signals, and even for transmitting power. With increased bandwidth, a smaller connector format, and support from major device manufacturers, USB seems primed to act as the single connector for laptops and other portable systems in the future. AV professionals will benefit from understanding how USB works, including its advantages and limitations, in order to optimize its use in a Pro AV environment.

white paper

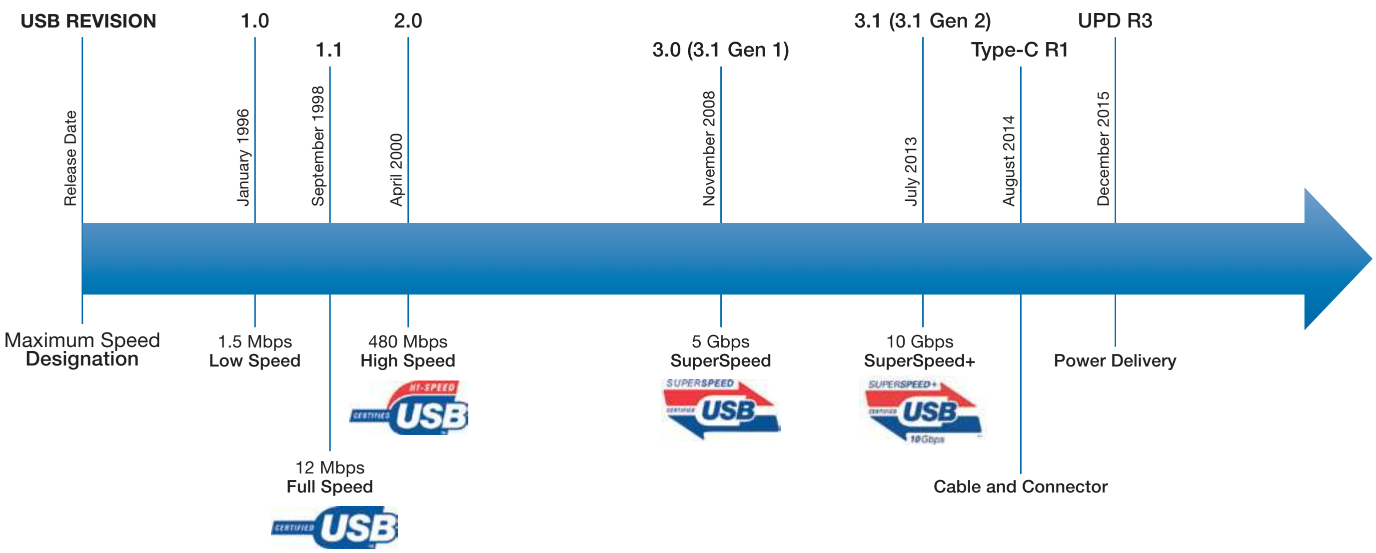
Table of Contents

HISTORY

USB – Universal Serial Bus – was originally developed in the mid 1990s as a cable, connector, and communications protocol standard for local connections between computers and electronic devices. It replaced serial, parallel, and other ports used by computing devices to connect peripheral equipment such as keyboards, mice, printers, etc. Compaq, Digital Equipment Corporation, IBM, Intel, Microsoft, NEC, and Northern Telecom developed USB and the first USB components were produced in 1995.

The USB 1.0 specification was released in January 1996 and defined Low Speed (1.5 Mbps) and Full Speed (12 Mbps) data rates. Over the years, the USB specification has been updated multiple times to improve performance. With revision 1.1 (1998), USB became widely accepted in the computing industry and began replacing legacy ports on new personal computers.

Figure 1: USB specifications timeline



First Introduced	Performance	Max Speed	Uses	Attributes
USB 1.0	Low Speed	1.5 Mbps	Keyboard, Mouse	Low Cost; Easy to Use and Connect/Disconnect; Many Peripherals
USB 1.1	Full Speed	12 Mbps	Audio, Microphone	Low Cost; Easy to Use and Connect/Disconnect; Many Peripherals; Guaranteed Bandwidth and Latency
USB 2.0	High Speed	480 Mbps	Video, Storage, Imaging	Easy to Use and Connect/Disconnect; Many Peripherals; Guaranteed Bandwidth and Latency; High Bandwidth
USB 3.0	SuperSpeed	5 Gbps	Video, Storage, Imaging	Easy to Use and Connect/Disconnect; Guaranteed Bandwidth and Latency; High Bandwidth
USB 3.1	SuperSpeed+	10 Gbps	Video, Storage, Imaging	Easy to Use and Connect/Disconnect; Guaranteed Bandwidth and Latency; High Bandwidth

Table 1: USB maximum speed comparison

USB 2.0 was released in April 2000 and added a High Speed (480 Mbps) data rate to the existing bus. This was followed in 2008 by USB 3.0 with its SuperSpeed (5 Gbps) data rate and in July 2013 with USB 3.1 and SuperSpeed+ (10 Gbps). The release of USB 3.1 also brought a name change: USB 3.0 was deprecated and became USB 3.1 Gen 1 and what is commonly thought of as USB 3.1 is actually USB 3.1 Gen 2.

The USB specification is maintained by the USB Implementers Forum (www.usb.org). The current revision of USB is backwards compatible with all previous revisions of USB. This will be discussed in more detail later.

Type-C Development

A separate open standard was released in August 2014 as USB Type-C revision 1.0. It uses a reversible 24-pin USB Type-C connector, or USB-C, that replaces the USB connectors and cables defined in the USB 2.0 and 3.0 specifications. This connector is often thought to be used exclusively for the USB 3.1 Gen 2 specification, but can be used for other protocols such as superMHL and Thunderbolt 3 as well. The Type-C connector provides additional pins for “alternate” modes of operation that use other transmission protocols passed down the cable. These alternate modes of operation are often associated with DisplayPort signaling, but can be used for HDMI, MHL – Mobile High-Definition Link, used primarily on mobile phones – or other protocols in the future. When used in this way, it is a USB-C alternate mode DisplayPort, USB-C alternate mode HDMI, or USB-C alternate mode MHL connector.

HOW IT WORKS

The serial bus in a USB 1.x/2.x system consists of one twisted wire pair using differential signaling. The bus is controlled by the USB host controller which is built into the motherboard of the PC or other host system. On a USB bus, there can be only one host or host controller, but many USB devices providing functions or services to the host. The host controller manages all of the devices on the bus and initiates all data transfers on the bus which is continuously polled by the host controller. Since only one host is allowed on a USB bus, it is not possible to connect two USB hosts or computers together on the same bus to share peripherals. There are exceptions to this rule, but they require unique cables and software drivers to allow two hosts to communicate with each other.

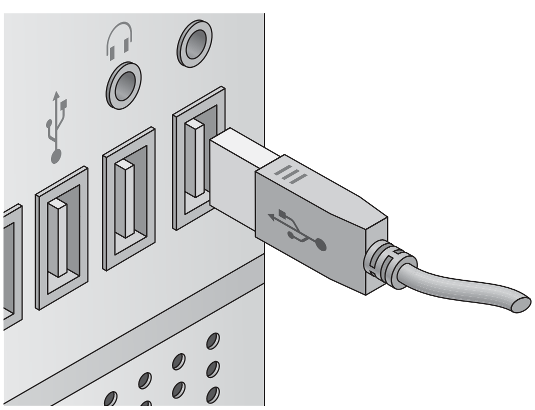


Figure 2: A USB peripheral device, such as a keyboard or mouse, connecting to a host computer

USB devices can be subdivided into hub devices (hubs) that allow additional connection points to the bus and peripheral devices (peripherals). Peripheral devices are entities such as keyboards, mice, printers, etc. that provide capabilities to the bus. Up to 127 devices can be attached to a single USB bus. This limit is determined by the USB

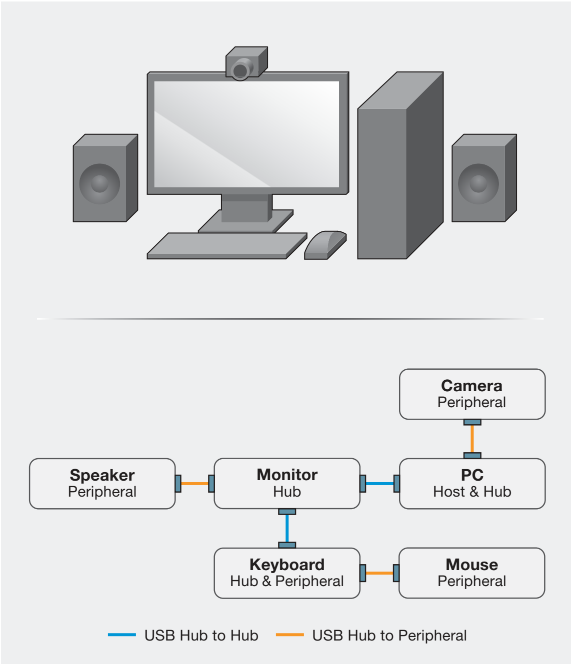


Figure 3: A typical computer system's USB connections; the lower diagram shows hub and peripheral connections.

address field which is seven bits long and therefore provides 128 possibilities (the zero address is reserved). Keep in mind that the 127 devices include both hub devices and peripheral devices. There are further limits on the number of hub devices on a USB bus that will be discussed later.

Since it is a bus, the signal passes down to every device connected to an enabled port, but only the addressed device accepts the data signal. Devices can respond back to a direct request from the host via regular polling by the host. Ports are enabled and disabled on a hub as devices are connected and removed.

When a device connects to the bus, the host determines whether a device supports Low Speed or Full Speed data rates. High Speed devices always connect as Full Speed devices and then negotiate the bus speed up.

Pipes

Pipes are logical connections created on the bus that allow data to move between the host and connected devices. USB uses two types of pipes for transmitting data: message pipes and stream pipes. Stream data has no specific USB format and allows raw data to be transferred between the host and device; message data uses a specific format to send control and message information between the host and device.

To manage transfers efficiently, each pipe can support one of four types of data transfers:

– Control transfers
– Bulk data transfers
– Interrupt data transfers
– Isochronous data transfers

As the name implies, control transfers are data transfers that control or configure a device. These are bidirectional message pipes that are created in pairs with one incoming pipe and one outgoing pipe. The remaining data transfer pipes are stream pipes, which are unidirectonal. Bulk data transfers are variable bandwidth data with error detection that are used for devices like printers, scanners, and flash drives. They typically have higher sustained throughput than other transfer modes, but with more unpredictable latency. Interrupt transfers are time dependent and typically have a limited amount of acceptable latency. Isochronous transfers have a guaranteed bandwidth and predictable latency. They work well for delivering audio, video, and other time-sensitive

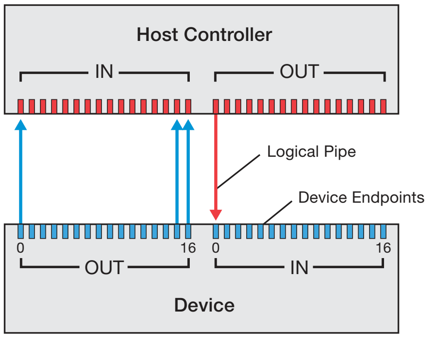


Figure 4: The USB host controller to device virtual communications paths; note that the Default Control Pipe is at endpoint zero.

high bandwidth data. Error free delivery is not guaranteed for isochronous transfers nor can they be sent at Low Speed data rates.

Most USB transactions begin with a token packet sent from the host controller to a device on the bus. The packet describes the type and direction of transfer that should occur. These packets are sent on a regular basis. Once the device decodes the token packet, it then either sends or receives a data packet to/from the host controller, as directed. The receiving device sends an acknowledgment packet to complete the transaction.

A USB endpoint is either the source or the destination of data. Endpoints exist at the end of a USB communication path. A device can have up to 16 IN (device to host) and 16 OUT (host to device) endpoints, including the Default Control Pipe described below.

Device Startup

As soon as a device is powered on and connected to the bus, it must go through an enumeration process to be assigned an address on the bus. It also needs the Default Control Pipe at endpoint zero set up, so the device can communicate with the host controller. The host gathers information from the device through the Default Control Pipe and from that information loads the appropriate driver. The software driver is loaded to configure the device so it can communicate on the bus and set up additional pipes that allow data transfers between the device and the host controller.

In the simplest form, a device such as a keyboard or mouse connects directly to a USB port on a PC along with other devices, such as a printer. The host controller opens pipes between the host controller and the device to communicate back and forth. As devices are added or removed from the bus, the host controller is responsible for managing the bus, communicating with the devices, and assigning or removing an address for them.

Cable Structure

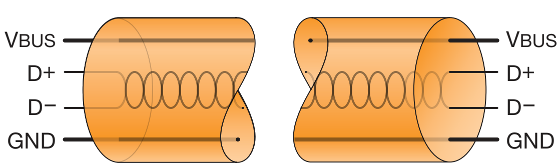


Figure 5: Original USB cable design with power, ground, and differential bus

The first USB cable, now referred to as a USB 2.0 cable, contained four wires: two wires for +5 V power (VBUS) and ground (GND) plus two wires for the bus (Data+ and Data–), a twisted pair that uses differential signaling. This bus wiring scheme was used from USB 1.0 through USB 2.0 with the bus supporting three data rates: Low Speed, Full Speed, and High Speed. For efficient communication under USB 2.0, transfers for Low Speed and Full Speed peripherals occur at High Speed between the host controller and hubs and then at their respective slower speed to peripheral devices.

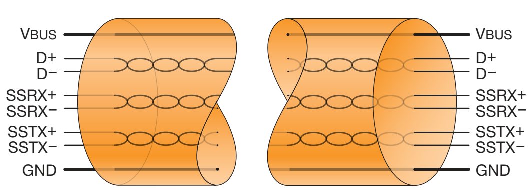


Figure 6: USB 3.x cable design with SuperSpeed duplex differential bus added to original power, ground, and differential bus

USB 3.0 cables added five more wires, consisting of two twisted pairs and a ground wire, to create the SuperSpeed high speed bus. One reason for the USB 3.0 bus speed increase over USB 2.0 is that it added duplex communication capabilities using two differential signal lanes for the SuperSpeed bus. In comparison, the USB 2.0 bus uses a single differential signal that provides only half duplex operation. USB 3.1 left the

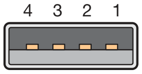
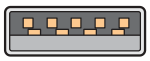
USB 2.0 TYPE A		USB 3.0 TYPE A	
			
PIN	**FUNCTION**	**PIN**	**FUNCTION**
1	+5 VBUS	1	+5 VBUS
2	Data-	2	Data-
3	Data+	3	Data+
4	GND	4	GND
5		5	SSRX-
6		6	SSRX+
7		7	GND_DRAIN
8		8	SSTX-
9		9	SSTX+

Table 2: USB Type-A connector pinouts

cable unchanged and, with more efficient encoding, doubled the bus speed to create the SuperSpeed+ bus. However, not all USB 3.0 cables are constructed to support the faster bus.

USB 3.0 and 3.1 (3.x) are backwards compatible to USB 2.0 because the USB 3.x 9-conductor cable is really a dual-bus system that runs both a USB 2.0 and USB 3.x bus in parallel within the same cable. Devices do not communicate directly with each other on the buses; all communication is initiated by and sent through the host controller. This allows the host controller to manage all communications on both buses simultaneously.

A USB 3.1 peripheral device must support SuperSpeed or SuperSpeed+ and at least one of the USB 2.0 speeds for backwards compatibility; however, the peripheral cannot operate on both buses at the same time. The USB 2.0 host controller works by polling the bus constantly for all transfers while the SuperSpeed/SuperSpeed+ protocol uses asynchronous notifications to communicate on the bus.

CONNECTORS

There are several types of USB connectors. Determining their differences can be a bit confusing. It all comes down to the power source and power sink (destination).

The two original USB connectors are not-so-creatively named Type A and Type B. Type A receptacles are downstream-facing connectors, meaning that power is being supplied from the Type A receptacle on the device. Think of all the PCs that contain host controllers which are supplying power to peripheral devices; they have flat rectangular Type A connectors. Type B connectors are almost square in shape and are found on downstream devices that receive power. These are seen on peripherals like printers or scanners that receive power from the host controller.

Detachable cables are typically Type A plug to Type B plug. This connector scheme was put in place to prevent plugging together two Type A upstream connectors. Doing so might short the two power supplies in the hosts or cause other problems.

Figure 7: USB 2.0 cable showing connection types and power flow

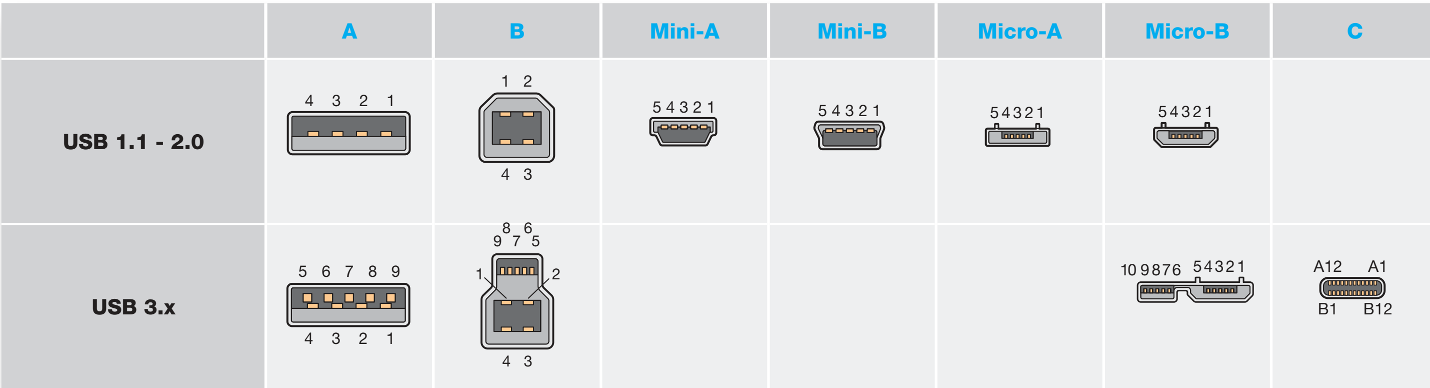


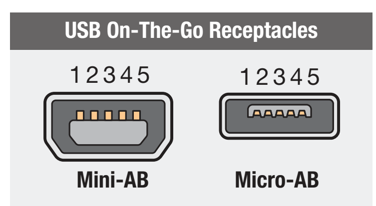
Table 3: USB cable connector pinouts, types, and their USB revision origin; note that USB 1.1-2.0 connectors can plug into their USB 3.x equivalent receptacle to support legacy devices.

Color coding, although not required as part of the USB specification, is used on USB receptacles to help users determine which revision of USB the host or hub supports. Black and white connectors are traditionally used to indicate USB 2.0 (and earlier) support. Blue and teal blue connectors represent SuperSpeed and SuperSpeed+ bus support, respectively. Yellow, orange, and red connectors are power-only ports found on many devices. Plugging a USB cable into any color USB port will not cause damage, but it may not work as expected. USB power will be discussed in greater detail later.

Connector Color	Connector Type	Common Use
Black	A and B	Low & Full Speed
White	A and B	Low, Full & High Speed
Blue	A and B	SuperSpeed
Teal	A and B	SuperSpeed+
Yellow	A	USB & Sleep-and-Charge
Orange	A	USB & Sleep-and-Charge
Red	A	USB & High Power Charging

Table 4: Typical USB connector color coding. Note that color coding is recommended, but is not required.

The Mini-A, Mini-B, Micro-A, and Micro-B connectors were created for use on smaller peripherals where a "Standard" receptacle was too bulky. The Mini connector was developed first and then followed by the Micro connector, which is slightly slimmer, for use on even smaller peripheral devices like smartphones. These connectors also added a fifth pin to the 4-pin standard Type A and Type B connectors. This fifth pin works as an ID pin for USB Mini-AB and Micro-AB receptacles – OTG mini or OTG micro plugs. USB On-The-Go (OTG) products can act as either a host or a peripheral, depending on which function they are performing. They accomplish this by having only one USB connector, type AB, on the product. Then, depending on which end of the USB cable is plugged into the device, the ID pin determines which function (host or peripheral) the device performs.



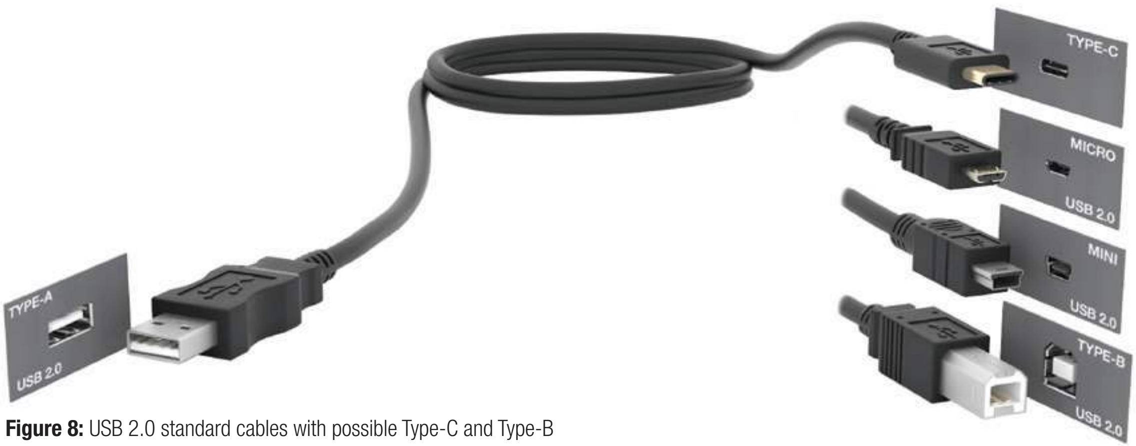


Figure 8: USB 2.0 standard cables with possible Type-C and Type-B connectors shown on right

Cable Types

All detachable cables must support at least Full Speed or High Speed mode; Low Speed cables are usually hardwired to the device. An engineering change notice in October 2000, part of the USB 2.0 standard, created the USB Mini-A and Mini-B connectors for small form factor devices such as digital cameras and smartphones. The Mini-A connector is no longer used while the Mini-B connector can still be found on many devices where space is not a concern. The Mini connector has been replaced by the USB Micro-A and Micro-B connectors which were introduced in 2007. The Micro connector is easier to integrate into thinner electronic devices since it is about half the thickness of the Mini connector.

Figure 9: USB 3.x standard cables with possible Type-C and Type-B connectors shown on right

Detachable cables have a Type A connector (Standard, Mini, or Micro) on one end and a Type B or vendor-specific connector on the other end for attaching hosts/hubs to peripherals. These cables can be found in USB 2.0 versions (4-wire) and USB 3.x versions (9-wire). The common cables are USB 2.0 Type-A to Type-B Standard, Mini, or Micro connector and USB 3.0 Type-A to Type-B Standard or Micro connector. These USB 3.0 Type-B Standard and Micro connectors are larger than their USB 2.0 counterparts to accommodate the additional pins required for SuperSpeed bus support.

Figure 10: USB 2.0 Micro-A to Micro-B connector

The only other common USB 2.0 cable has a Micro-A and a Micro-B connector. It is used with OTG devices that may act as a host. In this case, the Micro-B connector end of the cable is connected to the peripheral.

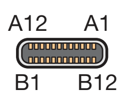
USB Type-C

One other USB connector that is becoming common is the USB-C 24-pin reversible connector. It eliminates the need to have a unique connector on hosts and peripherals in order to prevent cross-wiring power; a single Type-C connector can be used on both hosts and peripherals using a Type-C to Type-C cable. For compatibility with earlier USB devices, a USB-C cable can have a Type-C connector on one end and a Type-B Standard, Mini, or Micro USB 2.0 (4-wire) or Type-B Standard or Micro USB 3.x (9-wire) plug on the other end of the cable.

Alternate Mode

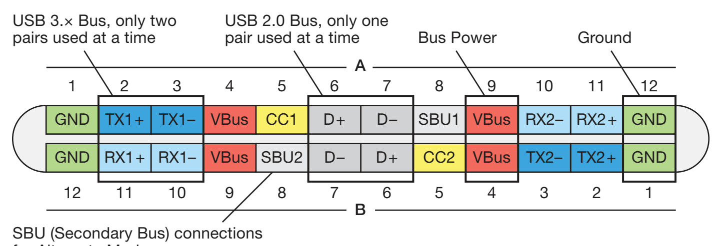
Type-C receptacles are designed for at least 10,000 connections/disconnections and are small enough to use on mobile devices. The connector interface can be used for more than USB. The 24-pin Type-C cable has four wire pairs for data use in addition to a wire pair dedicated to the USB 2.0 bus. With this design, it can use half of the

USB TYPE-C RECEPTACLE

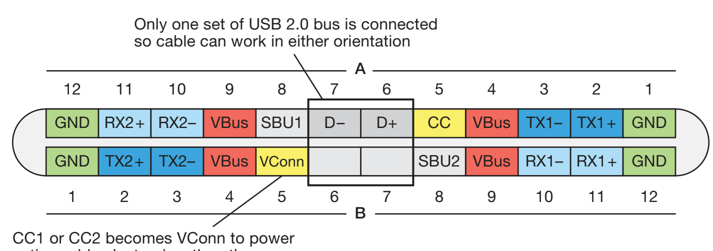


PIN	FUNCTION	PIN	FUNCTION
A1	GND	B1	GND
A2	SSTX1+	B2	SSTX2+
A3	SSTX1-	B3	SSTX2-
A4	+5 VBUS	B4	+5 VBUS
A5	CC1	B5	CC2
A6	Data1+	B6	Data2+
A7	Data1-	B7	Data2-
A8	SBU1	B8	SBU2
A9	+5 VBUS	B9	+5 VBUS
A10	SSRX2-	B10	SSRX1-
A11	SSRX2+	B11	SSRX1+
A12	GND	B12	GND

Table 5: USB Type-C 24-pin pinout



Type-C Receptacle Layout



Type-C Plug Layout

Figure 11: USB 3.x standard cables with possible Type-C and Type-B connectors shown on right. Note, only USB 2.0 speeds are supported on legacy devices.

four data lanes for Alternate Mode audio and video data with the other half used for a USB 3.x SuperSpeed or SuperSpeed+ bus. Similarly, all four data lanes could be used for Alternate Mode AV data and the USB 2.0 data pins used to notify the host, using the USB Billboard Device Class, that the device supports an alternate mode of operation.

The alternate modes could be DisplayPort – with two or four lanes devoted to DisplayPort signaling – or SuperMHL, HDMI, or some other signaling in the future. With the combination of data transfer and communications capability along with power delivery and AV, the USB-C connector is becoming the single connector on many computers.

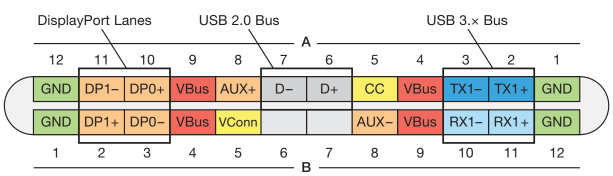


Figure 12: DisplayPort Alternate Mode Plug
(2) DisplayPort Lanes and (1) USB 3.1 and (1) USB 2.0

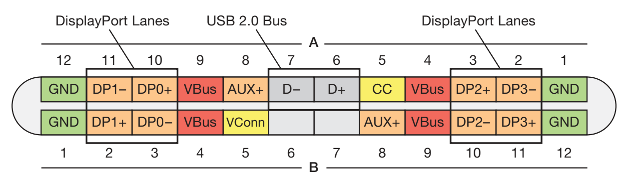


Figure 13: DisplayPort Alternate Mode Plug
(4) DisplayPort Lanes and (1) USB 2.0

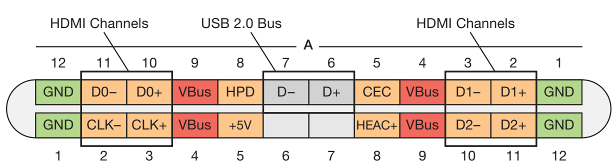


Figure 14: HDMI Alternate Mode Plug
(4) HDMI channels and (1) USB 2.0

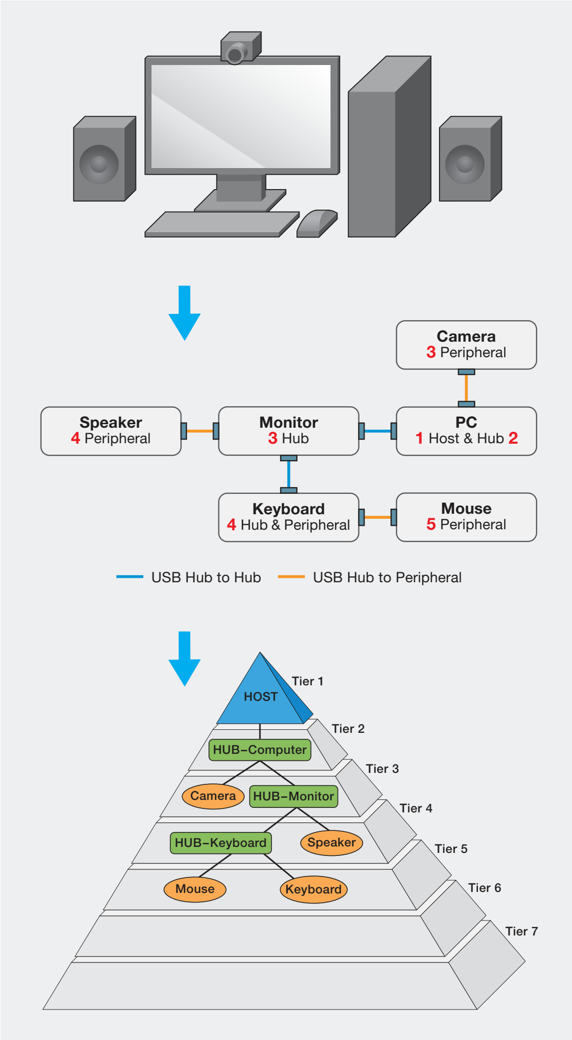


Figure 15: A simple USB topology showing an interconnection diagram and topology tree

USB TOPOLOGY

USB devices are interconnected in a tiered star topology with the host controller being the top tier (Tier 1). This tiered topology prevents multiple or circular attachments to the bus. While it is easy to understand how a few peripherals connect directly to the host controller, it is more difficult to understand the limits on the number of devices that can be daisy-chained to the bus.

USB cable length is limited by the speed of electrical signals. The time required for the host controller to poll the bus and electrical signals to pass from the furthest peripheral to the host controller limits hub expansion. The tiered star topology has a maximum of seven tiers of communication, with the host controller at Tier 1 and the furthest peripheral device at Tier 7. That leaves five tiers in between that can be occupied by hubs daisy-chained from one to another to regenerate the USB signals. This five cascading hub limit often creates problems when designing USB distribution or extension systems since it is not always clear where the hubs are located.

Hubs

A USB hub is a special kind of USB device with one input port sharing bandwidth among all devices connected to multiple output ports. It consists of a:

– Hub repeater

– Transaction translator

– Hub controller

The hub repeater connects the upstream facing port and multiple downstream facing ports running at the same speed. When a USB 2.0 upstream facing port is connected to a Full Speed device, the High Speed capability on the downstream facing ports is disabled and they must operate only at Low/Full Speed. When the upstream facing port is connected to a High Speed device, the High Speed downstream facing ports connect through the repeater. For any Low Speed or Full Speed downstream facing ports, a transaction translator is used to convert between the different port speeds. The hub controller provides status and control information and communicates to the host controller for the hub device.

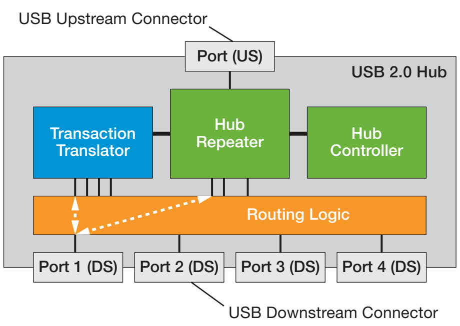


Figure 16: A simple USB 2.0 hub flow diagram

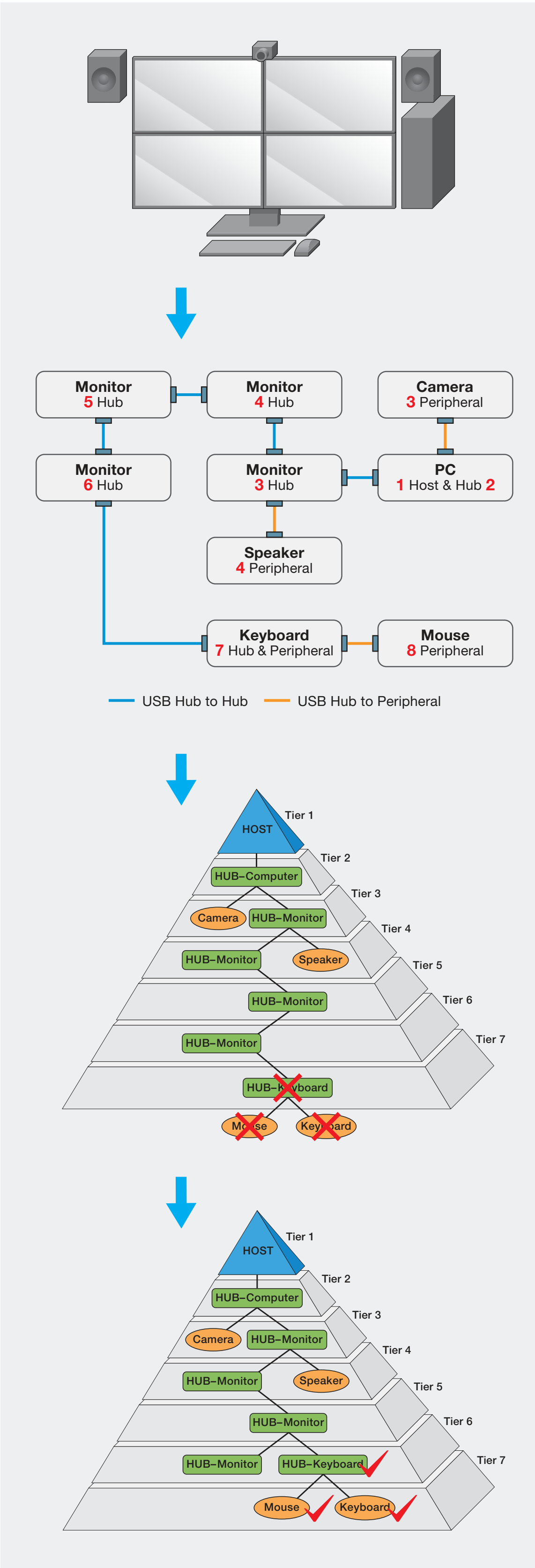


Figure 17: A complex USB topology showing how connecting USB devices one way can exceed the cascading hub limit. The bottom diagram shows the fix.

For backwards compatibility, USB 3.x simultaneously operates both a USB 2.0 bus and a USB 3.x bus for devices that are either USB 1.x/2.0 or USB 2.0/3.x capable. Similarly, a USB 3.x hub operates two hubs within an enclosure: a USB 2.0 hub and a USB 3.x SuperSpeed/SuperSpeed+ hub. Both hubs operate concurrently on each port to allow lower speed downstream devices to communicate with higher speed upstream devices.

Compound Devices

A USB device can be either a compound device or a composite device. A compound USB device has peripheral(s) and a hub enclosed in one package – e.g., a keyboard and trackpad in one enclosure with one USB cable. A composite USB device has multiple functions within the device, but has only one address – e.g., a webcam with video and audio functions in one device. Compound devices that include a hub within a single device, such as a touch display or interactive whiteboard, make it difficult to keep track of the total number of hubs in a USB topology.

Cascading Hub Limit

It is important to calculate the total number of hubs when you build a USB extension or distribution system. Closely review the entire system to ensure that the USB bus will support all of the peripherals that are connected to the bus and that the bus does not exceed the five cascading hub limit. To determine the total number of cascading hubs in a USB topology, count all of the hubs between the host controller and the farthest peripheral. This includes all of the hubs located inside any computer connected to the bus, plus all hubs enclosed within USB compound devices. The USB Hub Usage document lists the hubs used within Extron's USB products.

A computer's USB host controller has a root hub as part of the host controller, but many computers that have USB ports on the front and rear of the unit include another hub within the computer that is connected to the root hub. Also, if a computer has a built-in hub and a compound USB peripheral (e.g., a touch display), two hubs of the five cascading hub limit are already in use. In this case, only three more hubs can be daisy-chained to this system before the five cascading hub limit is reached.

Extenders

USB extenders are becoming more common and allow USB signals to be extended point-to-point over twisted pair cabling for hundreds of feet or over category or optical cabling via networks for thousands of feet. Early versions work by utilizing a timing provision in the USB specification that requires a USB peripheral to answer a host request within a defined time period. When a peripheral is too busy, it can send a

Cascading Hub Limit, Don't Exceed it!

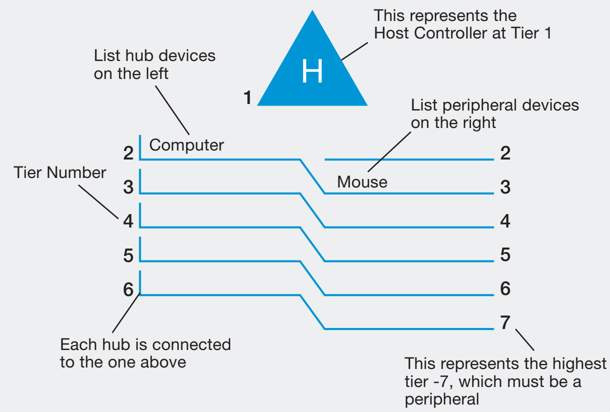


Figure 18: How to create a hand-sketched USB topology diagram

A simple topology diagram can be created to document a USB design and confirm that it meets the five cascading hub limit. In these diagrams, the USB Host Controller (Tier 1) is represented by the triangle. The five horizontal lines on the left beneath the triangle represent hubs at Tiers 2-6. The six horizontal lines on the right beneath the triangle represent peripheral devices at Tiers 2-7 that are connected to the Host Controller or a hub at Tiers 2-6. The diagonal lines indicate a connection from a hub on Tier 2-6 to a peripheral device on Tier 3-7. Add the hubs/peripherals for your specific USB topology to this diagram to determine whether your design conforms to the five cascading hub limit.

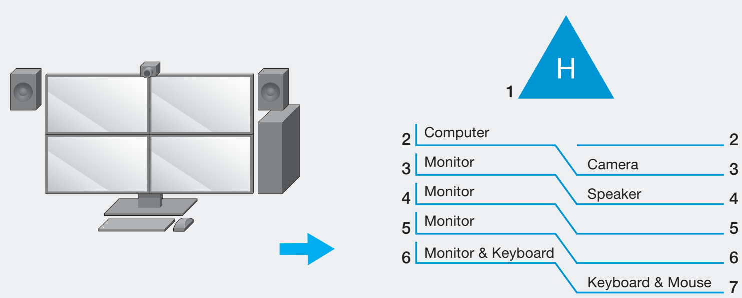


Figure 19: Creating a hand-sketched USB topology diagram of your proposed design will help you determine whether you have exceeded the five cascading hub limit.

NACK (negative acknowledgement) and the host controller will resend the same request one millisecond later. Using NACKs, the extender can "delay" the request from the host controller until the response can be returned from the peripheral through the extender transmitter/receiver pair.

Current extenders use some of the same techniques, but also add patented technology to manipulate signal timing of USB data to provide time for the extender transmitters and receivers to exchange the information over Ethernet, or other high speed protocol, to bridge the extender transmitter and receiver. Extenders usually require that at least one or two hubs of the five cascading hub limit be used within the transmitter for emulation and "keep alive" functions; for receivers containing a multiport USB hub,

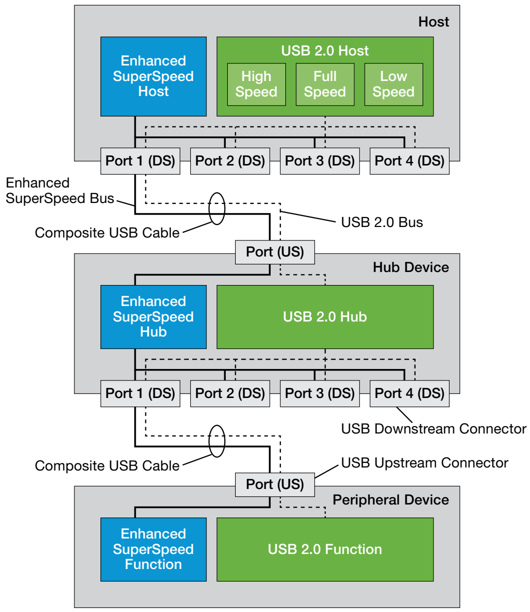


Figure 20: A USB 3.x system flow diagram showing a USB 2.0/3.0 peripheral and hub connected to a host

another hub will be used in the receiver. These hubs must be counted in the total hub count to ensure that the five cascading hub limit is not exceeded by the peripheral at the far end of the extender.

Switches

Sometimes host controllers need to switch between USB buses. Keyboard/Video/Mouse – KVM switches commonly deal with this requirement for Low Speed keyboard and mouse devices. However, USB switches can be used to switch between other speed devices as well. Switching can be a bit slow if the USB bus must have all devices on the bus enumerate each time a switch change occurs. Many USB switches use emulation functions on each of the switch's USB ports to indicate to the peripheral that a host is present even when it is not connected. Once the switch connects to that peripheral again, enumeration and USB driver loading are not needed, so total switching time is reduced.

DEVICE CLASSES

Device classes define the different types of peripherals that can be attached to a USB bus and the associated software drivers that must be loaded on the host computer to allow those devices to communicate with the host over the bus.

Each device class uses a common set of software drivers that are loaded as part of the enumeration process to bring the peripheral device online. By using device classes, unique drivers do not need to be loaded for each USB bus device. Instead, common drivers can be loaded by the operating system to communicate with different types of USB devices.

The USB-IF – USB Implementers Forum – created a driver specification for standard functions such as video and audio. UVC 1.5 – USB Video Class – was published in 2012 and is the current device class specification for video devices. Major computer operating systems such as Windows, Mac, and Linux include these drivers with the operating system. Cameras and other USB video devices that utilize these standard drivers do not require additional drivers to be loaded when they are first connected to the USB bus. Similarly, UAC 3.0 – USB Audio Class – is the current device class specification for audio devices. When you select a USB AV device, make sure it supports the UAC and UVC drivers. This will greatly simplify the process required to get them to work with your computer's operating system.

HID – Human Interface Device – is probably the most common device class; it includes keyboards and mice that typically communicate at Low Speed. USB ports can be restricted to connecting only HID devices. This increases security by preventing data transfers from storage and communications devices that could infect a computer with a virus through a USB port.

The HCI – Host Controller Interface – enables the USB host controller to communicate with the host controller driver software running on the PC. Four versions of HCI are defined as noted below:

– OHCI – Open Host Controller Interface
- supports USB 1.1 (Low Speed and Full Speed)

– UHCI – Universal Host Controller Interface
- supports USB 1.x (Low Speed and Full Speed)

– EHCI – Enhanced Host Controller Interface
- supports USB 2.0 (High Speed)
- hands off Low Speed and Full Speed to OHCI and UHCI

– xHCI – Extensible Host Controller Interface
- supports USB 3.0 (all USB speeds)

When specifying extenders and other USB products, it is important to confirm that they support the most recent HCI.

Base Class	Descriptor Usage	Description	Examples
00h	Device	Unspecified	Device class is unspecified, interface descriptors are used to determine needed drivers
01h	Interface	Audio	Speaker, microphone, sound card, MIDI
02h	Both	Communications and CDC Control	Modem, Ethernet adapter, Wi-Fi adapter, RS232 adapter. Used with class 0Ah (below)
03h	Interface	HID (Human Interface Device)	Keyboard, mouse, joystick
05h	Interface	Physical	Force feedback joystick
06h	Interface	Image	Webcam, scanner
07h	Interface	Printer	Laser printer, inkjet printer, CNC machine
08h	Interface	Mass storage	USB flash drive, memory card reader, digital audio player, digital camera, external drive
09h	Device	Hub	USB hub
0Ah	Interface	CDC-Data	Used with class 02h (above)
0Bh	Interface	Smart Card	USB smart card reader
0Dh	Interface	Content security	Fingerprint reader
0Eh	Interface	Video	Webcam
0Fh	Interface	Personal Healthcare	Pulse monitor
10h	Interface	Audio/Video Devices	Webcam, TV
11h	Device	Billboard Device Class	Describes USB Type-C alternate modes supported by device
12h	Interface	USB Type-C Bridge Class	
DCh	Both	Diagnostic Device	USB compliance testing device
E0h	Interface	Wireless Controller	Bluetooth adapter, Microsoft RNDIS
EFh	Both	Miscellaneous	ActiveSync device
FEh	Interface	Application Specific	IrDA Bridge, Test & Measurement Class (USBTMC), USB DFU (Device Firmware Upgrade)
FFh	Both	Vendor Specific	Indicates that a device needs vendor-specific drivers

Table 6: USB device classes

POWER

In addition to being a common connector for device communications, USB has become a convenient source of power for many devices since it provides power to computing devices without the need of a bulky AC power supply. It also can charge portable or mobile devices such as tablets and smartphones.

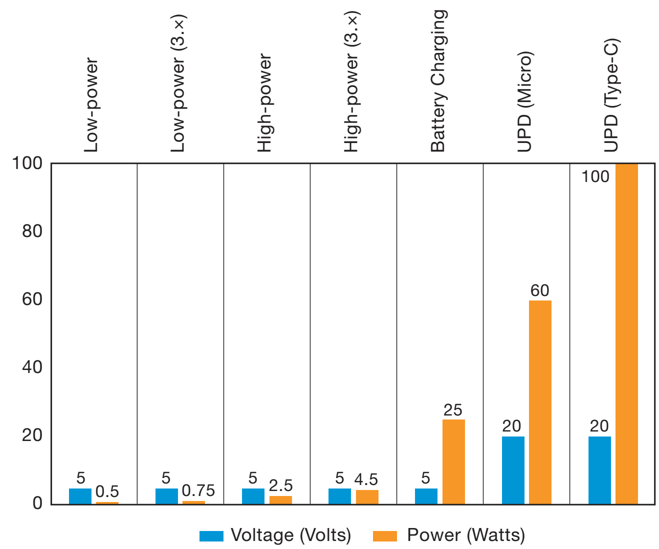


Table 7: USB power standards showing voltages and the maximum power draw

The +5 Volt line within the cable provides power to USB devices. The power wires in a USB cable can vary in size to accommodate power delivery for assorted cable lengths. There are limits to how much power a device can draw. According to the original USB 1.1 specification, the unit load is 100 mA; it increased in later USB revisions to 150 mA for SuperSpeed devices. All devices must start out drawing one unit load, but high-power devices can draw up to five unit loads (500 mA). USB 3.0 SuperSpeed devices are allowed a maximum of six unit loads (900 mA).

Hubs can be self-powered or bus-powered. With power limits as defined above, devices can draw power from the bus. For devices needing more power, they must be externally powered and not draw any power from the bus.

After the initial USB 2.0 specification was released, problems arose because many devices needed to provide power to more than just the attached peripheral. In 2009, a supplementary specification called Battery Charging 1.1 was introduced; this has since been updated to 1.2. These revisions define two types of USB charging ports (CDP and DCP). They are higher power and power-only USB ports. The DCP – dedicated charging port – is used only for charging and does not provide data signals. The CDP – charging downstream port – provides charging power, but also is a standard USB data port. These ports can supply up to 25 Watts (5 Amps @ 5 Volts) of power to a device and are different from the SDP – standard downstream port – that existed in earlier USB revisions.

Figure 21: Examples of a USB dedicated charging port – DCP on AV interface plates displaying a battery symbol

In 2012, the USB Power Delivery – UPD – specification was defined; it is currently at revision 2.0 and was released as part of the USB 3.1 specification. UPD allows the bus to deliver up to 100 Watts of power either to a host (e.g., a laptop for charging) or from a host (e.g., a video monitor). It uses the Type-C connector and defined power rules yielding four nominal voltages of 5 V, 9 V, 15 V, and 20 V to provide power ranging from 0.5 to 100 Watts. The USB devices negotiate the level of power to deliver over the USB cable's power wires. Above 60 Watts, an active Type-C cable with internal electronics is required. This confirms to the USB controller, before it enables the power, that the cable can carry it safely.

Sleep-and-charge ports are another USB port power option. These ports are known by various computer manufacturer trade names (e.g., PowerShare), but they all allow a computer's USB port to provide power for charging devices even when the computer is turned off. These ports are usually labeled or use a color coded yellow or red USB port to indicate their function – see Table 4.

CONCLUSION

Since the initial intent to standardize the numerous connections on early personal computers, USB has become a ubiquitous connector on electronic devices. Today, USB is used for those needs as well as to provide high speed data transfer for AV and storage devices. The bus also has found wide adoption as a power connector for computing and mobile devices and, with the USB-C connector, is likely to become the single standard connector on computing and mobile devices in the future. Understanding its distribution, connector, and power requirements are key to using it effectively in AV systems.

Notes

Notes

Every day, millions of people around the world have their experiences enhanced by Extron audiovisual signal processing, distribution, and control products. Our advanced technologies create better looking images and higher quality sound with systems that are easier to control and work more reliably. Our powerful asset management tools are helping technology professionals efficiently manage large numbers of audiovisual systems deployed throughout their enterprises and institutions. For more than 30 years, Extron has been a recognized leader in high-tech electronic manufacturing. Our world-class engineering and manufacturing facilities enable us to develop industry leading technologies for the AV industry. With over 40 offices around the globe, Extron provides dedicated, full-service support and training to industry professionals worldwide.

www.extron.com

WORLDWIDE SALES OFFICES

Anaheim • Raleigh • Silicon Valley • Dallas • New York • Washington, DC • Toronto • Mexico City • Paris • London • Frankfurt
Madrid • Stockholm • Amersfoort • Moscow • Dubai • Johannesburg • Tel Aviv • Sydney • Melbourne
New Delhi • Bangalore • Singapore • Seoul • Shanghai • Beijing • Hong Kong • Tokyo

www.extron.com