What's The Difference Between USB 2.0 and 3.0 Hubs?

Want to see more content like this?
Visit electronicdesign.com

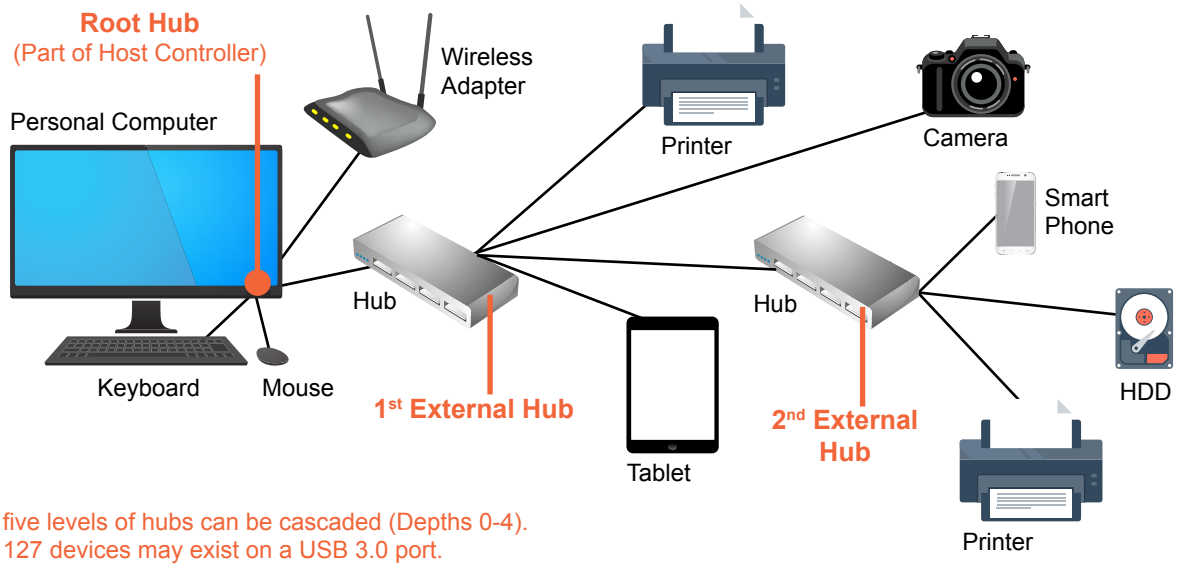
BROUGHT TO YOU BY:
electronic design

Market acceptance of USB 3.0 has been increasing steadily because of its many advantages over USB 2.0, including far higher speed (up to 5-Gbit/s raw bandwidth), higher power availability (up to 900 mA per port), and better power management through more levels of power reduction when maximum power isn't needed. These benefits are achieved while maintaining functional and mechanical compatibility with USB 2.0 devices, hubs, and host ports. But what can USB 3.0 users do when they need more USB 3.0 ports than their PC or docking station provides? External USB 3.0 hubs are the answer.

Tier Topology And Data Paths

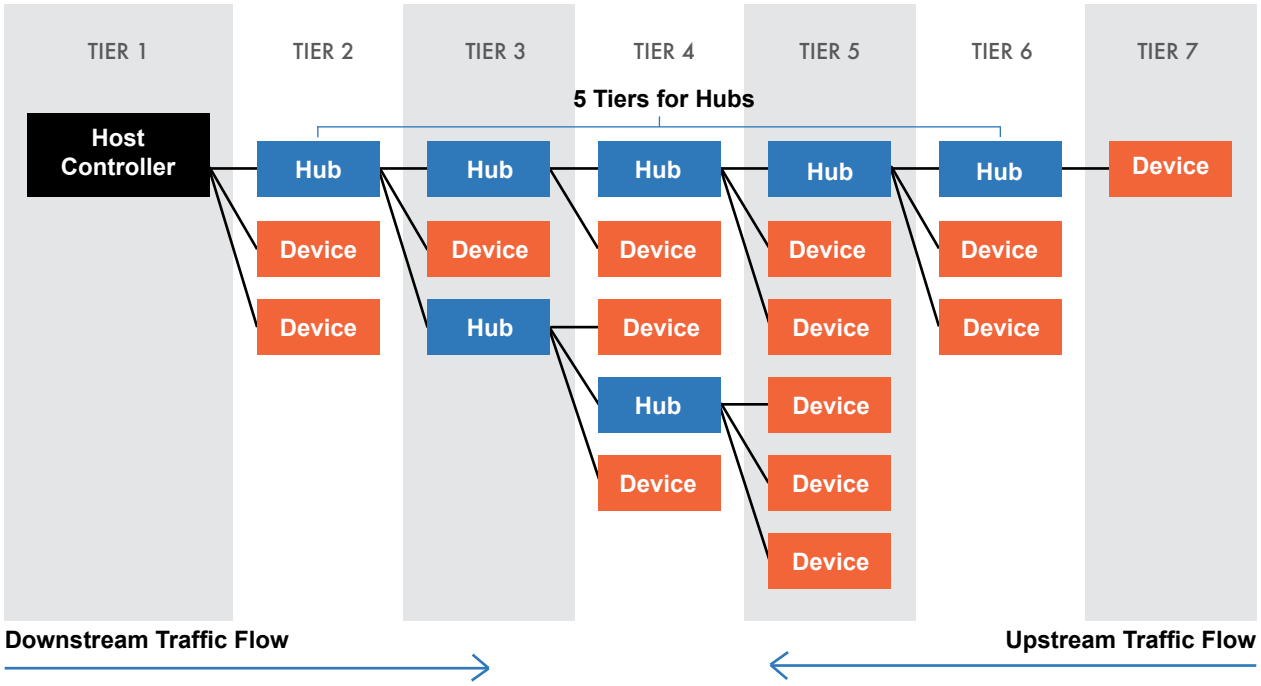
A typical host PC might have two USB 2.0 ports and two USB 3.0 ports. The USB 2.0 ports might be used for a USB keyboard and USB mouse, but the user may have far more than two additional USB devices to connect to the PC at the same time, and many of those might be capable of USB 3.0 operation. Connecting two hubs into the topology enables support for all of the additional USB devices, and there may even be some USB ports still available for more devices. To obtain the speed and power benefits of USB 3.0, the hubs and interconnecting cables will all also need to be USB 3.0 compliant, including USB 2.0 support for USB 2.0 devices.

Figure 1. A typical PC may have USB 2.0 and USB 3.0 connections to USB devices via an external USB 3.0 hub.



Up to five levels of hubs can be cascaded (Depths 0-4).
Up to 127 devices may exist on a USB 3.0 port.

Figure 2. The USB 3.0 system architecture topology supports up to five hub tiers.



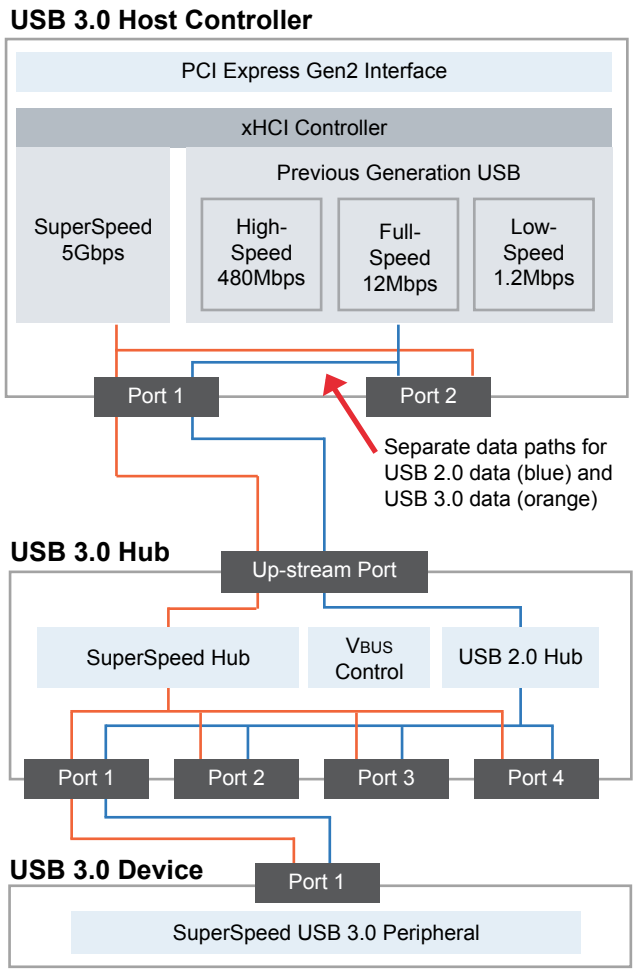
With the second hub connected to the first, USB 2.0 and USB 3.0 allow up to five levels of hubs to be cascaded together (Fig. 1). There are typically four downstream ports on each hub, but other numbers of ports on a hub are possible as well. The total bandwidth of the downstream ports together can't be greater than the bandwidth available on the upstream port. The ports on the PC are known as "root ports," and the root ports are designated as "tier 1." Tiers 2 through 6 represent additional levels made possible by hubs, and tier 7 is the final level of devices supported by a hub at Tier 6.

USB 3.0 hubs preserve this basic tier topology (Fig. 2) but add USB 3.0 support internally in addition to USB 2.0 support (Fig. 3). A complete USB 2.0 hub is encapsulated within the complete USB 3.0 + 2.0 hub, with simultaneous parallel data paths for USB 3.0 SuperSpeed traffic and USB 2.0 High Speed, Full Speed, or Low Speed traffic. There are physically separate pins in the connectors and separate wires in the USB 3.0 cable for USB 3.0 SuperSpeed traffic and USB 2.0 traffic. The USB 3.0 SuperSpeed path operates at a raw bit rate of 5.0 Gbits/s, while the USB 2.0 path operates at 480 Mbits/s (High Speed), 12 Mbits/s (Full Speed), or 1.5 Mbits/s (Low Speed). Within the hub, only the port power control logic is shared between the USB 3.0 path and the USB 2.0 path, since there is only one 5-V power path in either USB 2.0 or USB 3.0.

The additional pins and wires for USB 3.0 include SuperSpeed Transmit (SSTX+, SSTX–), SuperSpeed Receive (SSRX+, SSRX–), and an additional ground (GND). The additional pins are arranged mechanically so a USB 2.0 connector or cable can be used in place of a USB 3.0 connector or cable in nearly all cases to allow USB 2.0 data traffic (at USB 2.0 speeds) even if there is no available path for USB 3.0 SuperSpeed traffic. The main mechanical incompatibility arises when attempting to use a USB 3.0 cable for a USB 2.0 device, due to the physical size of the USB 3.0 Standard-B plug on a USB 3.0 cable. Conversely, a USB 2.0 cable can be used with USB 3.0 ports to allow USB 2.0 data flow at USB 2.0 speeds.

Likewise, USB 2.0 hubs can be used instead of USB 3.0 hubs, or vice versa, allowing USB 2.0 data flow at USB 2.0 speeds. The only way to achieve USB 3.0 SuperSpeed operation, however, is for

Figure 3. Separate data paths are used for USB 2.0 and USB 3.0 traffic.



the host port, the device, all intervening hubs, and all connecting cables to be designed for USB 3.0 operation, with an unbroken USB 3.0 pathway from host to device.

USB enumeration is the process of detecting, identifying, and loading the correct software drivers for a USB device.

During the enumeration process, the host and its driver automatically detect whether or not a USB 3.0 path exists to each device, and the driver configures the host controller to use the USB 2.0 path if a working USB 3.0 path is not found (or if the device doesn't support USB 3.0 SuperSpeed at all). Similarly, a USB 3.0 device uses its USB 2.0 pathway instead of USB 3.0 if the device was configured to do so during enumeration.

It is mechanically possible to connect up to 1024 devices (45) at tier 7 if tiers 2 through 6 consist entirely of hubs with four downstream ports each. Unfortunately, the 8-bit device address used in USB limits USB topologies to a maximum of 255 devices. Data throughput considerations will usually limit the practical number of devices further, and there is also usually a limit in the host controller on the number of device "slots" (one "slot" per device) that the host controller can support.

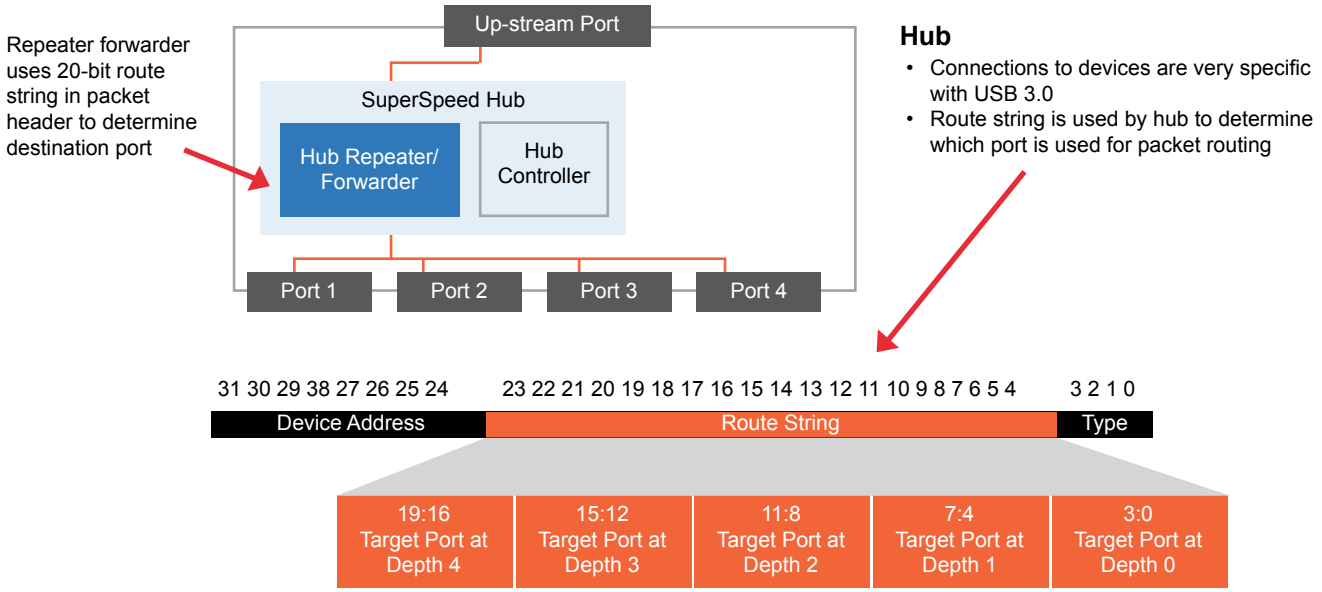
Point-To-Point Packet Routing

One major enhancement in USB 3.0 compared to USB 2.0 is the use of point-to-point packet routing from host to device, instead of the "broadcast-to-all-points" characteristic of USB 2.0. This reduces data traffic on USB 3.0 links that aren't involved in a given transaction and facilitates keeping unused links in a reduced power mode to conserve total system power. To enable point-to-point USB 3.0 packet routing, packets originating in the host contain a 20-bit "route string" field (Fig. 4). The route string consists of five 4-bit subfields signifying the port numbers on the hubs to which the packet should be routed.

Each hub is assigned a "depth" number from zero through four, and the hub uses the port number at its assigned depth to determine which of its downstream ports the packet should go to. Hub depth zero corresponds to tier 2 and so on up to depth four at tier 6. For example, a hub residing at a depth of 3 (tier 5) and assigned "depth 3" during enumeration will use the port number in the "depth 3" field of the route string to determine the intended downstream port for the packet. A port number of zero means the packet is targeted for the hub itself, not for any of the hub's downstream ports.

Upstream packet routing, from a device to the host, is always point-to-point inherently. The host is always the final destination for any packet transmitted by a device. Packets moving upstream are not broadcast to other devices or USB links. In USB, there is no mechanism for one device to transmit a packet to another device instead of transmitting it to the host. There is always one

Figure 4. USB 3.0 hub header packet routing provides very specific connections to devices.



host and one or more devices (if any data flow is occurring). Packet flow is from host to device or vice versa, never device-to-device.

Note that all the data and control paths exist in a USB 3.0 hub to support USB 2.0 data traffic flowing simultaneously with USB 3.0 SuperSpeed traffic. For example, the host controller may still be finishing a USB 2.0 transmission or packet reception at the same time that a USB 3.0 packet begins to flow from a USB 3.0 device through a USB 3.0 hub and finally to the USB 3.0 host controller. This is only possible with USB 3.0 hubs. USB 2.0 hubs don't have separate data paths to allow this kind of simultaneous data flow.

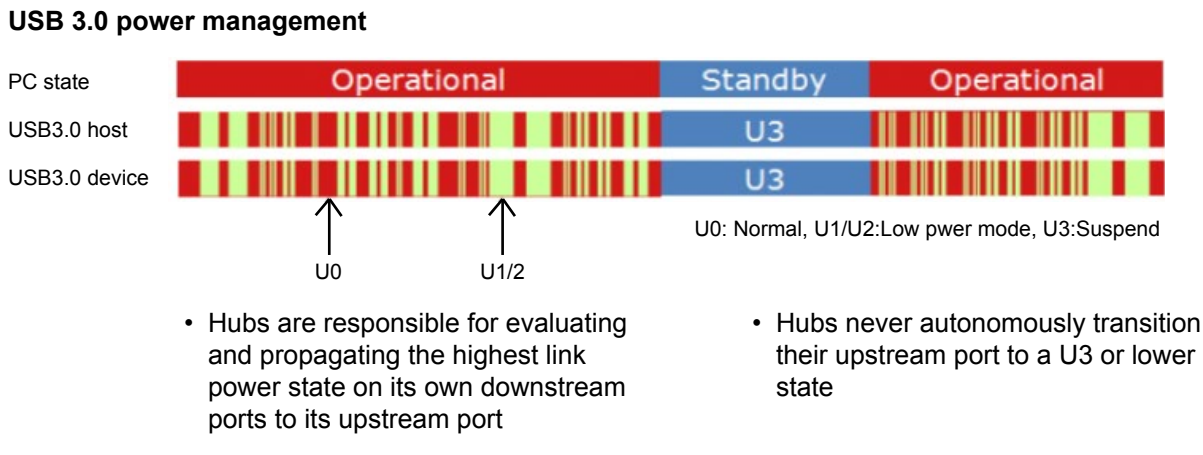
Data Buffering And Throughput

Another characteristic of USB 3.0 hubs is that they contain more data buffering than USB 2.0 hubs. USB 3.0 hubs store USB 3.0 SuperSpeed packets in a buffer and then retransmit them when there is an available time slot in the SuperSpeed data path. Unlike USB 2.0, the buffering in the USB 3.0 hub (and the host controller) allows a USB 3.0 SuperSpeed transfer to continue immediately with the next packet, without needing to wait for an acknowledgment of successful receipt of a previous packet. The acknowledgments can be combined into a single packet to acknowledge a group of several data packets. USB 3.0 hubs can do this completely independently of any simultaneous USB 2.0 data flow that may also be occurring.

It was mentioned earlier that a hub cannot increase the total bandwidth of all the downstream ports combined, compared to the bandwidth on the upstream port. As a very rough estimate of total bandwidth available on a host controller port, SuperSpeed uses a 5-Gbit/s raw bit rate, with 8b/10b encoding, which reduces the effective bandwidth for data to 4 Gbits/s (500 Mbytes/s) or less. Link protocol and packet framing reduce this estimated maximum still further, and any idle time between packets imposes still more effective data throughput reduction.

Instead of 500 Mbytes/s per SuperSpeed link, the measured data throughput may be significantly less due to all these overheads. In particular, the host system and end device may not be able to keep up with the available bandwidth, resulting in added idle time between packets actually transmitted on the SuperSpeed link. And, remember that a four-port SuperSpeed hub effectively splits the bandwidth available on its upstream port into four branches, with each having only 25% of the upstream

Figure 5. USB 3.0 hubs are required to support U0, U1, U2 and U3 link power states on all ports, upstream and downstream.



bandwidth if all four ports are contending equally for the available upstream port bandwidth.

Furthermore, if the host controller is a bridge from a PCI Express bus to USB 3.0, then the bandwidth of the PCI Express interface will limit the bandwidth that the host controller can support on its USB 3.0 ports. If the PCI Express interface is "x1 Gen2," meaning one lane with a raw bit rate of 5 Gbits/s, then the host controller will be able to support only one USB 3.0 port operating at 5 Gbits/s. Two or more USB 3.0 ports supported by a single PCI Express Gen2 (x1) interface will suffer the same kind of bandwidth splitting arising in USB 3.0 hubs.

USB Power Management

The USB Implementers Forum, PCI SIG, and Intel have published several specifications pertaining to USB devices, hubs, and host controllers known as the USB 2.0 Specification (including engineering change notices, or ECNs), USB 3.0 Specification, PCI Express Base Specification and other related specifications, and the xHCI Specification. These specifications describe various "power states" for USB and PCI Express devices, including D0 through D3 for PCI Express devices, LPM (Link Power Management) L0 through L3 for USB 2.0, and U0 through U3 for USB 3.0.

The power states range from fully on and operational (D0, LPM-L0 and U0) to minimally powered (D3hot, L2, U3) or completely unpowered (D3cold, LPM-L3). The minimally powered or unpowered states have the lowest power consumption and the longest "latency" to return to a fully operational state (due to loss of "context" information), while intermediate power states have higher power consumption levels but shorter "resume" latencies, partly depending on whether or not their clocking is stopped and needs time to restart.

In general, a bus driver and/or higher-level driver running on the host CPU implements the overall power management strategy for the USB topology, such as when to put any part of the USB topology into a reduced power state and how deeply to reduce its power, depending on the resume latency that might be needed. For instance, a fully unpowered device generally will need a complete USB hardware reset (using specified USB signaling in USB 2.0, or "polling.LFPS" in USB 3.0) and re-initialization to become operational again. That is likely to be too time consuming for users who are trying to utilize their device and the system is only trying to prolong the useful operational battery time. In this case, software drivers can detect what level of usage is occurring and determine a suitable tradeoff between power savings and quick responsiveness as seen by the user.

The process of a device resuming to its fully operational state can be triggered either by the host software or by user activity, such as pressing

a key on a keyboard, moving and/or clicking a mouse, or receipt of new incoming data on a network connection.The ability of a USB device to support device-initiated resume depends on the host software putting the device into a properly “enabled” condition before sending the device into a reduced-power state.

USB 3.0 power management can save considerable power compared to USB 2.0. A system’s host controller may have only two power states, fully on or standby, but the USB 3.0 U1 and/orU2 power states may be utilized by either USB 3.0 host or devices and links that aren’t actually being used during time intervals when the system is still fully on (Fig. 5). USB 2.0 does not have intermediate options between fully on and standby unless LPM-L1 has been implemented, and even LPM-L1 does not provide as many power options to the host software as USB 3.0 U1/U2 can provide. As already noted, the point-to-point routing characteristic of USB 3.0 also allows greater flexibility in putting inactive or less active devices and links into reduced power states.

Summary

USB 3.0 is poised for rapidly increasing deployment in the market. Only a few short years ago, it was limited to a few types of USB peripheral devices and a few USB host controllers. Full USB-IF certification for USB 3.0 hubs just became available in December 2012, and the μPD720210 from Renesas Electronics became the first USB 3.0 hub to receive certification. USB 3.0 availability is expected to expand rapidly in the market at all levels: host controllers, peripheral devices, and now hubs as well.