Endpoints Technical Note

Cambrionix
CHARGE • CONNECT • MANAGE

Do you find devices are not visible on your USB port or do they appear and disappear intermittently? If you connect too many devices to most computers' USB ports, you may be hitting what is known as the endpoint limit.

If you experience the endpoint limit, you are likely to see an error indicating you have "run out of USB resources" on your host computer, you may not see any error message at all, and (any of) your USB devices may randomly fail to operate correctly or become intermittent.

The problem with the limited number of endpoints provided by host controllers is that most USB devices, especially mobile phones and tablets, enumerate as more than one endpoint. Please be aware that all USB devices, including USB hubs, require endpoints, so when setting up your work-flow, it is essential to bear this limitation in mind.

A limitation on USB connections to consider is that the maximum number of possible USB devices that can be connected to a root hub is 127. So you will never be able to have more than 127 devices connected to one host system.

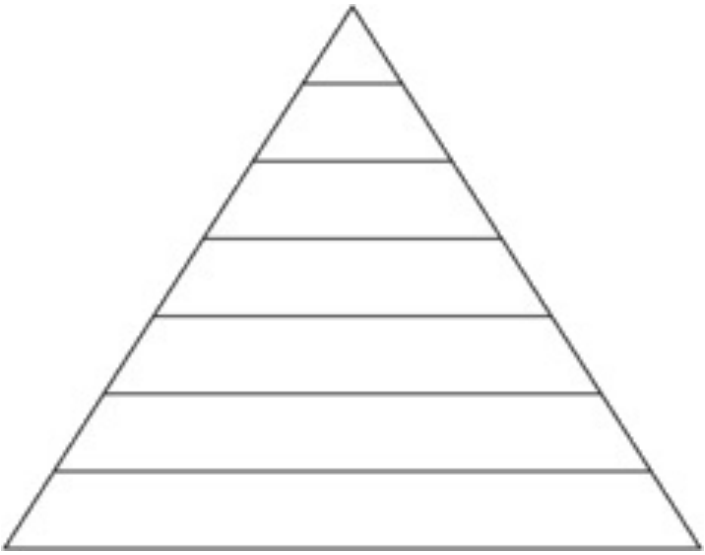
Hub Depth

USB depth is a result of the use of multiple hubs. Each hub is its USB device and the start of a new level of USB devices. The host and root hub reside on the first tier; connecting to a hub adds another level. The level number for any USB device is the number of non-root hubs in the connection chain plus 2. The USB specification is limited to a maximum of 7 tiers. Therefore, there is a hard limit of 5 daisy-chained hubs.

Most host systems have just one USB controller integrated into the host system. Many host systems use hub chips internally to expand the number of available USB connectors. So, the USB connector on your host machine is already one or even two tiers deep in the overall USB tree. This will affect the ability to daisy-chain external hubs. Remember that many low-cost USB expansion cards add an internal hub.

Another thing to consider with hub depth is the internal architecture of the USB hub(s) you are daisy-chaining together. Some ports are connected directly to the initial internal USB hub, and others may be connected to a further internal USB hub, limiting the number of other hubs that can be connected. We would advise viewing the Internal hub architecture from the product user manuals to decide which port to chain hubs to maximise the potential and avoid hub depth limits.

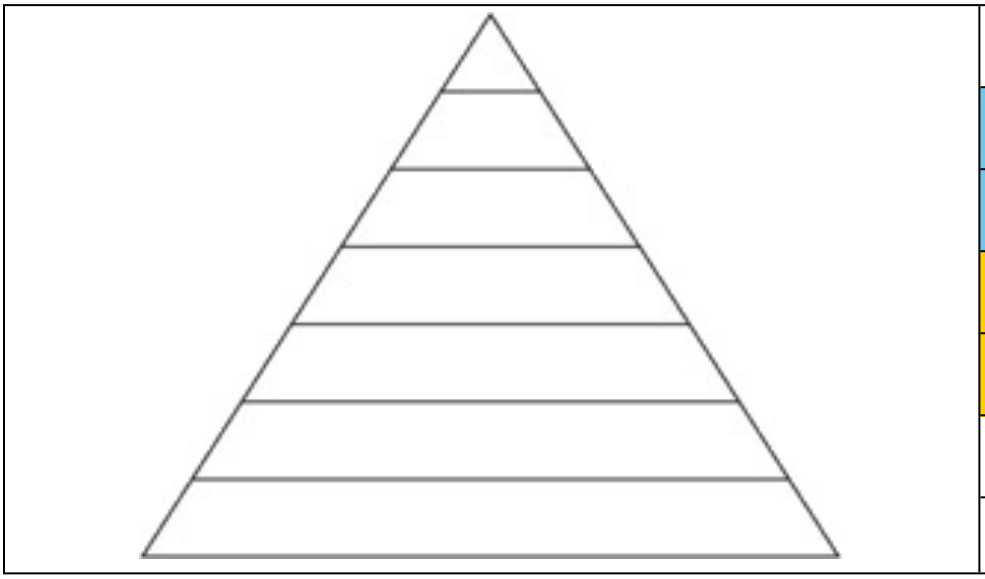
www.cambrionix.com/product-user-manuals



Tier 1 (USB Host controller)
Tier 2 (USB Hub or Device)
Tier 3 (USB Hub or Device)
Tier 4 (USB Hub or Device)
Tier 5 (USB Hub or Device)
Tier 6 (USB Hub or Device)
Tier 7 (USB Device)

Example

A simple example of hub depth can be seen below. Using a SuperSync-15 and chaining them together this is how the depth would look. Alternating blue and yellow for each physical SS15 unit.



Tier 1 (USB Host controller)
Tier 2 (SS15 Host port)
Tier 3 (SS15 Internal Hub)
Tier 4 (SS15 Host Port or USB Device)
Tier 5 (SS15 Internal Hub)
Tier 6 (USB Device)
Tier 7 (Not used)

From this you can see we can connect 2 hubs in a daisy chain fashion and still connect USB devices to both hubs, if we were to connect a 3rd hub the hub would connect but you wouldn't be able to connect any USB devices as you will have run out of tiers.

What you can do though to maximise hubs without hitting hub depth limits is to connect multiple hubs to the first hub in the tier and as such any subsequent hubs will not push beyond the maximum depth. So good practice on USB hubs would be to use one 'parent' hub and multiple 'child' hubs..

USB Endpoints

Each USB device can define up to 32 endpoints (16 inputs and 16 outputs. One must be a control endpoint) but often uses much less; for example, an iPhone 11 will use nine endpoints. USB hubs will also use a control endpoint. Every USB controller will have a layer or endpoint limitation.

USB 3.0 and USB 2.0 are in a similar USB architecture. They also share the same endpoints on controllers. USB 3.0 devices may use both the USB 3.0 and 2.0 buses. This will reduce the number of devices which can be attached to a single host controller. You can disable USB 3.0 from the BIOS, or an even more straightforward method would be to use USB 2.0 cables to connect your devices and, as such, would not use the USB 3.0 lanes.

Some examples of devices and their endpoint usage can be found below, this information is as accurate as possible, but values may vary depending on different manufacturer variations.

Device	Maximum endpoints used
Ingenico Inc. Moby/8500	8
iPad (10th Gen)	11
iPad (2nd Gen)	8
iPad Mini (6th Gen)	11
iPad Pro (3rd Gen)	8
iPhone 5s	8

Device	Maximum endpoints used
iPhone 10/11/12	9
iPhone 13/15/16	11
Samsung S20	11
Samsung S23 Ultra	9
Samsung S5e	9
Samsung Galaxy S8	7
Google Pixel 4	4
Google Pixel 6	3
Google Pixel 9	4

USB hubs will also use endpoints, below is a table of Cambrionix hubs and the endpoints that they use.

Hub	Maximum endpoints used
OEM-PDSync-C4	6
OEM-U8S	8
OEM-U16S-Spade	8
PDSync-C4	6
PowerPad-8S	8
PowerPad-15S	10
SuperSync15	16
ThunderSync3-16	17
ThunderSync3-C10	11

Endpoint Memory

Each endpoint will use an amount of memory. Typically a USB 2.0 endpoint will use up 4Kb, and a USB 3.0 endpoint will use up 8Kb. Each USB host controller will have a limit on how much memory it has and how much can be used at any one time. This is another limit when connecting USB hub(s) to your host system.

We have made a table below detailing host controllers and the memory we have recorded that they have available, this information is as accurate as possible. Still, values may vary depending on different manufacturer variations.

Host controller	Maximum endpoint memory (Kb)
Intel(R) USB 3.20 eXtensible Host Controller - 1.20	940
Cambrionix Thunderbolt internal controller	1012

Example

For example if I was looking to get the maximum number of iPad Pro 3rd gen I can connect using the SuperSync15 using USB 3.0 and the Intel(R) USB 3.20 eXtensible Host Controller - 1.20. I would need to use the following equation

```
Endpoints x Endpoint memory = memory usage per device(8 x 8 = 64)

Memory available - hub memory usage = (940 - 8 = 932)

Memory available / device memory = possible devices(932 / 64 = 14.56)
```

So i will be able to connect 14 iPad Pro 3rd gen using USB 3.0, If I wanted to increase the devices i could use USB 2.0 instead.

```
Endpoints x Endpoint memory= memory usage per device(8 x 4 = 32)

Memory available - hub memory usage = (940 - 8 = 932)

Memory available / device memory = possible total devices(932 / 32 = 29.12)
```

Using USB 2.0 I will increase my device count to 29.

If i then wanted to increase devices even more I can then use multiple host controllers to maximise the devices able to connect all at once.