

USB connectivity issues

These issues are not necessarily related to a specific OS, but are general issues that can be encountered.

Delay recognising USB device

During system startup or when a device is hot plugged, there will be a delay of between 5 and 10 seconds before the device is discovered and ready to use, as explained [here](#).

The UPDD Console shows the state of each UPDD device. The Console is periodically refreshed when on display.

Therefore, when a USB device is plugged in, it can take a short period of time to be discovered and start working and for the device status to be updated in the Console dialog.

USB stall issue

This can occur during the initialisation of the device or when attempting to read data from the device as described [here](#).

USB device not discovered

Sometimes the device can be listed in a macOS system but not seen by the driver as described [here](#).

USB device open errors

A problem opening a connection to a USB device typically shows up in the UPDD log as follows:

```
2021/01/13-02:28:03: TRC: VID:0xafa PID:0x3e8 BCD:0x200 is supported by this UPDD package
```

```
2021/01/13-02:28:03: TRC: USB location bus:0001-port:0001-0003
```

```
2021/01/13-02:28:03: ERR: USB error: (open) Operation not supported or unimplemented on this platform
```

USB connectivity issues

We've only ever seen that message when a USB extender or a filter driver is installed, and the open function is not supported.

This indicates a driver in the stack that does not support open, i.e a different device is present. Typically, this is a filter driver for a usb extender.

USB Device read errors

For HID compliant USB devices, the driver attempts to read information from the HID Descriptor. This can sometime fail.

The debug log will show that the driver has accessed and interacted with the device but the HID descriptor read failed as seen in the debug log:

```
2019/06/20-18:12:26: TRC: Configuration has 1 interfaces
```

```
2019/06/20-18:12:26: ERR: USB error: (get serial number) Other error
```

```
2019/06/20-18:12:26: ERR: USB error: (get product) Other error
```

```
2019/06/20-18:12:26: ERR: USB error: (get manufacturer) Other error
```

```
2019/06/20-18:12:26: TRC: Found interface 0 class 3
```

```
2019/06/20-18:12:26: TRC: Interface 0 is supported
```

```
2019/06/20-18:12:26: ERR: USB error: (claiming interface) No such device (it may have been disconnected)
```

or

```
2019/08/29-08:49:02: ERR: About to throw exception at line #969 in  
file ../../06_00/updd/upddmessagedispatcher.cpp Device not found
```

```
2019/08/29-08:49:02: ERR: About to throw exception at line #149 in file  
../../06_00/updd/upddmessagedispatcher.h Device not found
```

.

.

```
2019/08/29-08:50:04: DBG: USB warning: (get config active descriptor; failed will try index zero) No
```

USB connectivity issues

such device (it may have been disconnected)

2019/08/29-08:50:04: ERR: USB error: (get config descriptor) No such device (it may have been disconnected)

We also attempt to read the HID Report Descriptor and we use some information contained within to automatically configure the device.

Further, we read the HIDRD (even in some cases when it's not actually used) as this is one of the methods we use to determine whether a device is connected.

We have had many reports of USB connection issues purely related to a 'bad' USB cable and replacement cables have worked.

At other times, these failures have been seen when using USB extenders or incompatible USB hubs, such as reported with a [Sabrent USB3 hub](#). It has also been reported that some touchscreen failed with USB 3.1 gen2 hubs but were OK with a standard USB 3 hub.

Directly connecting the USB device with a standard short USB cable normally resolves the issue and / or when connected directly via two USB-A/C adapters. Also when going through Apple's own [USB-C Digital AV Multiport Adapter](#) (which is reported as USB2 according to "ioreg -p IOUSB").

If a USB extender is in use and failing, it's possible that:

1. Not enough power to the extender.
2. Extender not fully USB compliant.
3. Extender is trying to handle HID and making incorrect assumptions
4. Extender is not compatible with the Operating System USB stack.

One possible solution is to supply a driver that does not access the USB Descriptor and carries the device's HID Report Descriptor in the device's UPDD settings.

To allow for this, we need to supply a driver with settings **disable_interface_prolog** and **hid.skip_hidrd_check** enabled and be supplied the HIDRD from a UPDD driver [diagnostics](#) when the device is working (e.g. directly connected).

USB connectivity issues

Touch-Base Support

<http://support.touch-base.com/Documentation/50607/USB-connectivity-issues>