

## USB controller support

UPDD driver supports many devices as listed [here](#) in a variety of ways. If your device is not listed then we need to identify some USB specific information on your system to help identify the device.

### Identifying USB Vendor and product id

To identify a device or add support for a new touch device as a minimum we need to know the device's vendor and product id. Select the appropriate link here to [Windows](#), [Linux](#), [Mac](#) to see how to view this information when the device is connected to a computer. Having identified the Vendor and Product id then if the device is known to us and already configured in our driver we can send a trial driver for testing. Where it is a new touch device we can send a test driver, that may work if it is from a known touch manufacturer already supported with similar touch devices, or which can be used to capture incoming touch data for further analysis.

### Touch Data capture

Where a trial or test driver does not work try capturing incoming touch data as described [here](#).

Should you not be able to capture any touch data then read on.....

### USB interface and endpoint settings

It is possible that the test driver will not capture any touch data. Touch data is sent to the driver on a specific USB interface and this interface can vary between touch devices. With an unknown controller we will default this interface 0. Further each interface has a number of endpoints and by default we will set this to endpoint 1. Only if the data is being delivered on this interface and endpoint will the data be captured. If it is not then we need further information from the system that will show us the correct interface values to set as described [here](#). We can then configure the correct interface and endpoint and send another driver to test.

### Multi-touch USB request

Once we have the correct vendor and product id along with the correct interface and end point we should be able to capture the data. However, although a touch device may offer multi-touch functionality it may operate in single touch mode unless it receives notification (a multi-touch USB Feature Request command) from the driver indicating that it supports multi-touch input. This command is sent from a multi-touch aware HID driver (e.g. Windows 7 / 8 ) when the device's HID descriptor indicates it can support more than 1 touch. It is for this reason that many multi-touch devices are advertised as Window 7 / 8 compatibility with no mention of other OS. Some multi-touch controllers will output multi-touch data at all times irrespective of the receipt of this request and therefore it is not required.

If the touch controller is multi-touch but the UPDD captured data (under XP, Mac or Linux - traditionally non MT OS) is only in single touch format then we will request that you capture further information on a Windows 7 / 8 device using [USBlyzer](#) as the device, if HID compatible, will run in multi-touch mode in these OS.

## USB controller support

USBlyzer will capture both the multi-touch USB Feature Request command as sent from the MT HID driver and the touch data sent in MT format. The USB FR command is always 3 bytes but varies between controllers (and is based on the controller's HID descriptor). This same command will then be issued from our driver to ensure the touch device operates in multi-touch modes, irrespective of OS in use.

InstantKB.NET

<http://kb.touch-base.com/KnowledgebaseArticle50018.aspx>