



Measurably better value

## Controlling USB Devices that use an FTDI USB to Serial Chip from Linux

### Application Note

#### Products:

HA1600A  
TF900 Series  
TG2000 Series  
QL Series I

This application note describes how to configure a Linux computer to recognize the USB to Serial adapter embedded in some Aim-TTi products.

## Introduction

Several Aim-TTi products use a USB to Serial adapter chip made by FTDI internally in order to implement the USB interface on the back panel.

Usually if you plug an FTDI USB to Serial adapter into a Linux computer it is detected and works right away. This is because the Linux Kernel recognizes it and already has drivers for it. With TTI products that use the FTDI chip internally, the issue is that they use a different USB Vendor ID (VID) and Product ID (PID) and Linux doesn't know what driver to load. To get round this, we need to tell Linux to associate these with the FTDI driver.

## Getting the VID and PID

The first step is to find the Vendor ID (VID) and Product ID (PID) of the connected device:

Plug in the device.

In the terminal, enter:

```
lsusb
```

This should list all your USB devices and you should get something like

```
DELL3431:~$ lsusb
Bus 002 Device 001: ID 1d6b:0003 Linux Foundation 3.0 root hub
Bus 001 Device 008: ID 103e:0406 TTI TG2000 DDS Generator
Bus 001 Device 002: ID 0416:a0f8 Winbond Corp. Drevo keyboard
Bus 001 Device 001: ID 1d6b:0002 Linux Foundation 2.0 root hub
DELL3431:~$
```

In this list we can see that we have a TTI TG2000 DDS Function Generator connected on USB bus 001, device 008. The ID given here is the VID and PID separated by a colon:

The TTI Vendor ID is 103e (hex)

The TG2000 Product ID is 0406 (hex)

## Telling Linux what driver to use

In order to allow Linux to recognize that we want to use the FTDI drivers with this, we need to first make sure they are loaded:

Unplug the USB device

At the terminal:

```
modprobe ftdi_sio
```

This loads the FTDI Serial I/O kernel module.

Next, using the VID and PID from above:

```
echo 103e 0406 >/sys/bus/usb-serial/drivers/ftdi_sio/new_id
```

Here 'new\_id' is literal and not a placeholder. This tells the Linux kernel to associate the vendor ID 103e (TTi) and product ID 0406 (TG2000) with the ftdi\_sio driver so that it is recognized when any device with this VID and PID combination is plugged in. You may have to do this as root if you get a permission denied error.

Reconnect the Device

## Getting the Serial Port Details

Again, in the terminal, enter

```
dmesg | tail
```

This gets the last bit of the diagnostic messages from the kernel log and you should get something like:

```
DELL3431:~$ dmesg | tail
[185244.564262] usb 1-9: USB disconnect, device number 8
[185273.219047] usb 1-9: new full-speed USB device number 9 using xhci_hcd
[185273.373144] usb 1-9: New USB device found, idVendor=103e, idProduct=0406,
bcdDevice= 4.00
[185273.373164] usb 1-9: New USB device strings: Mfr=1, Product=2, SerialNumber=3
[185273.373171] usb 1-9: Product: TG2000 DDS Generator
[185273.373177] usb 1-9: Manufacturer: TTI
[185273.373182] usb 1-9: SerialNumber: G4825VQM
[185273.377409] ftdi_sio 1-9:1.0: FTDI USB Serial Device converter detected
[185273.377523] usb 1-9: Detected FT232BM
[185273.377997] usb 1-9: FTDI USB Serial Device converter now attached to ttyUSB0
DELL3431:~$
```

We can see that the device has now been connected to USB bus 1 as device 9 and is recognized as an FTDI USB Serial Device of type FT232BM and is attached to the system as ttyUSB0. To connect to it with our serial control software we now need to use /dev/ttyUSB0 as the serial port identifier and it should be available to use.

## Checking it works

To test it, try sending

```
*idn?
```

This queries the device for its ID and it should reply with the manufacturer, model, serial number and firmware versions.

If you get permission denied errors, make sure that you are member of the "dialout" group and that read and write permissions are set up appropriately on the port itself.

If you get no response, check the device manual:

- Check the serial port settings, particularly the baud rate.

- Check that the device is set to use the USB port for remote control