

Adafruit's Raspberry Pi Lesson 5. Using a Console Cable

Created by Simon Monk



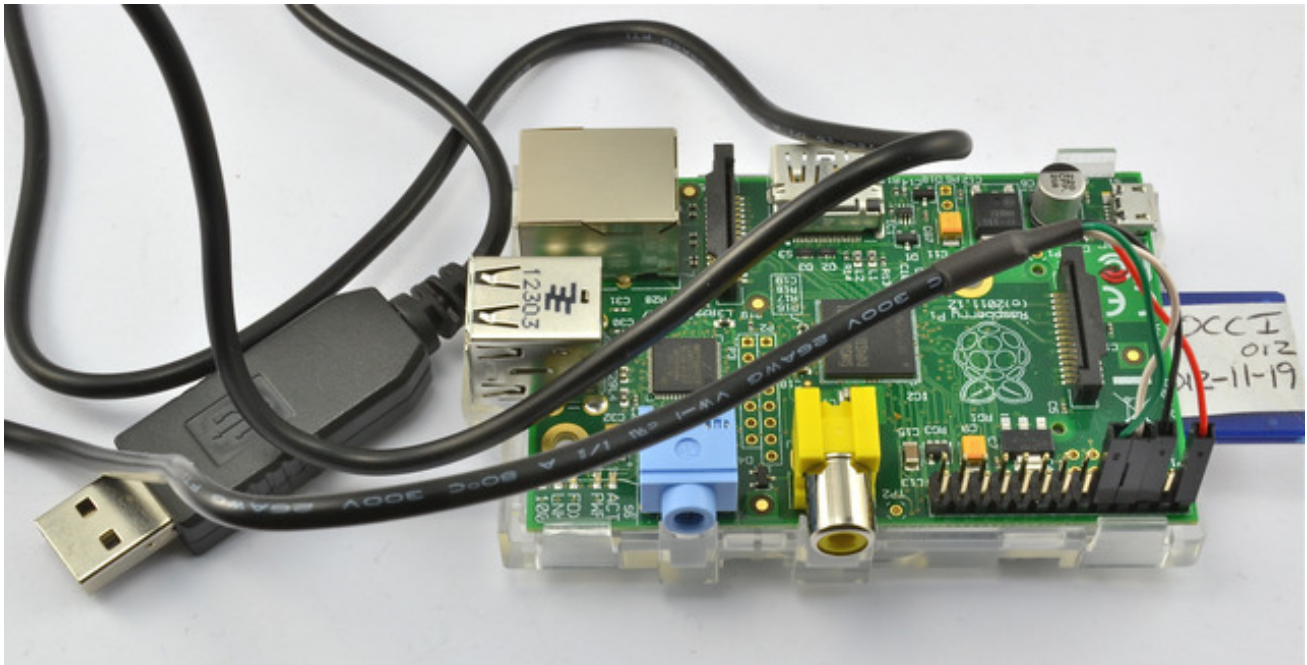
Last updated on 2014-09-15 12:00:13 PM EDT

Guide Contents

Guide Contents	2
Overview	3
You Will Need	4
Part	4
Software Installation (Mac)	6
Software Installation (Windows)	7
Software Installation (Linux)	12
Connect the Lead	13
Test & Configure	16

Overview

In this lesson you will learn how to remote control your Raspberry Pi with a console cable.



The great advantage of connecting this way is that it can even supply the power for your Pi and you do not need keyboard, mouse or display attached to the Pi to log into it.

You will need to install terminal emulation software (Putty) if you are using Windows and also USB drivers for the Console Lead. The Mac Terminal comes with its own software.

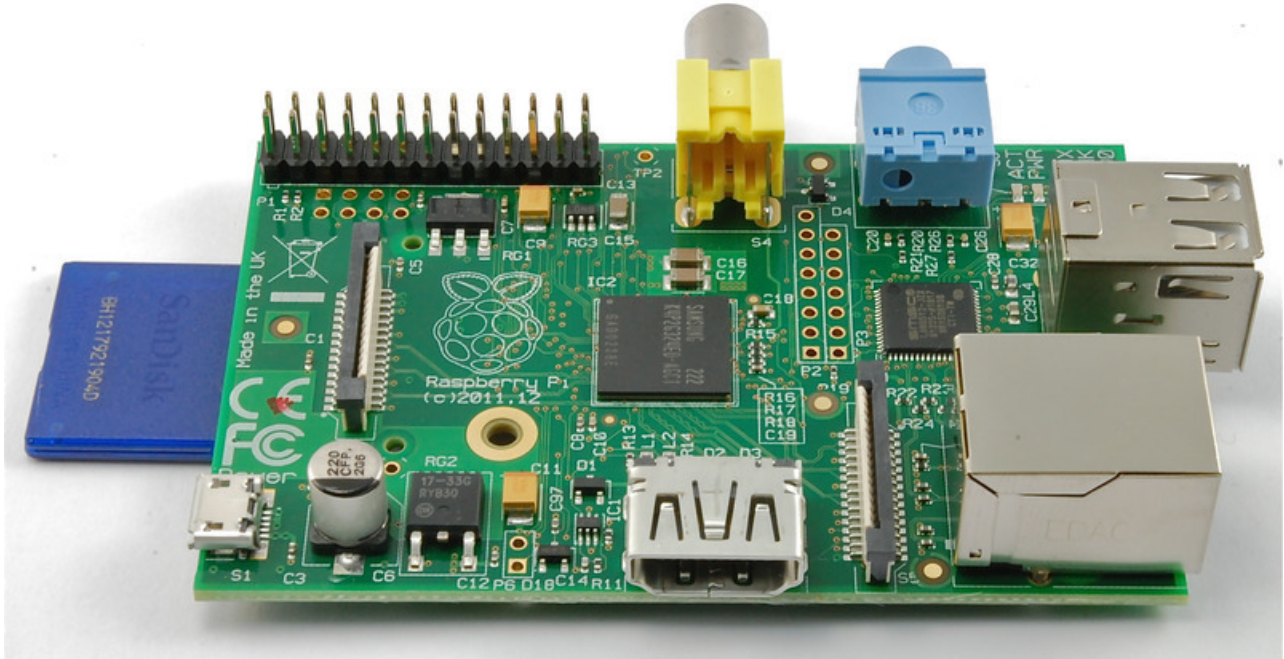
The Raspberry Pi uses its built-in serial port to allow devices to connect to its console and issue commands just as if you were logged in.

In the next lesson, we will look at another way of doing much the same thing but over a local network using something called SSH.

You Will Need

To follow this lesson, you will need:

Part



Raspberry Pi



Adafruit Console Lead

This cable has 3.3V logic levels so its safe to use, but not all cables are 3.3V! Only use the Adafruit console cable for this tutorial to avoid damaging your Pi's GPIO pins

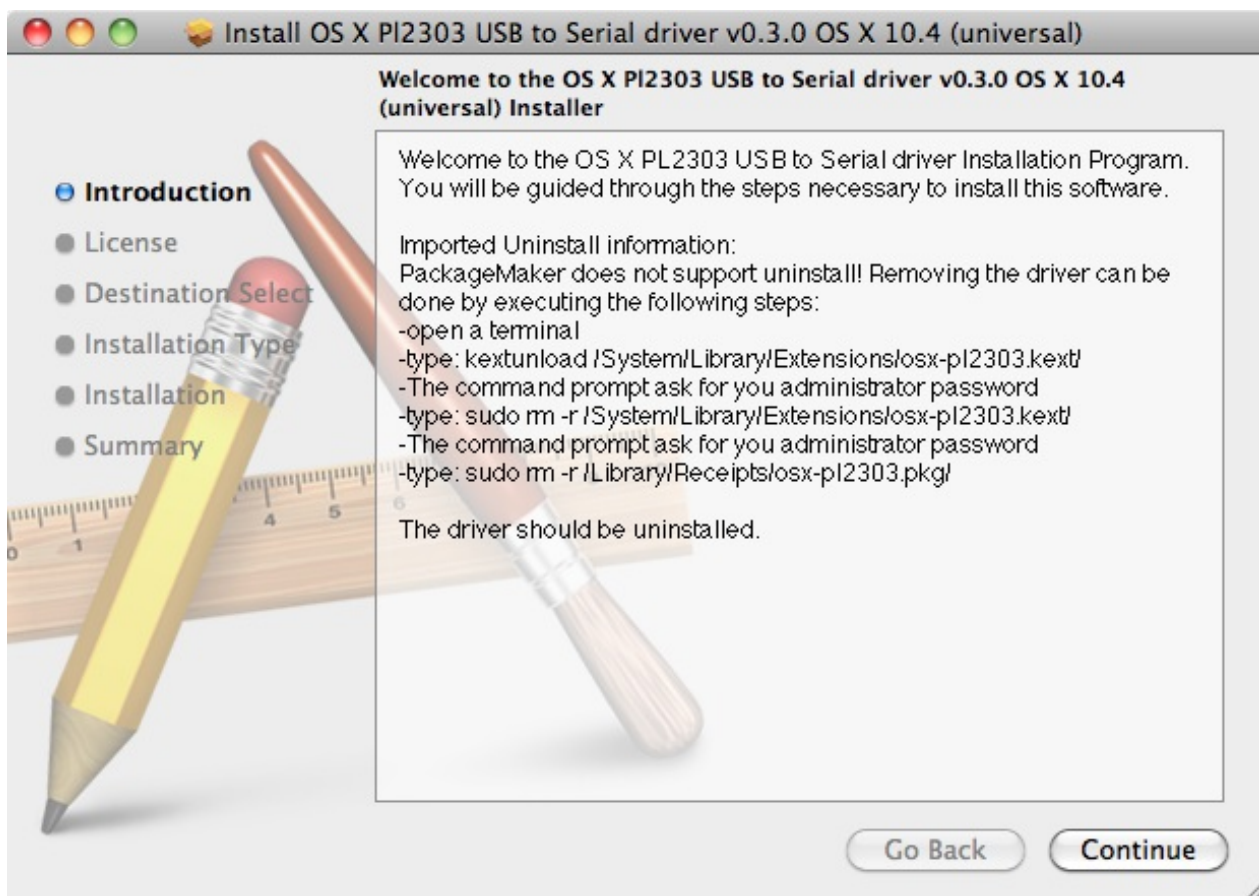
Software Installation (Mac)

OSX includes terminal emulation software that you can use from the command line, so we only need to install the USB drivers for the cable.

Check out the most recent Mac Driver from prolific (<http://adafru.it/dYM>)

If using Lion or Mountain Lion or later OS X, try this driver here <http://changux.co/osx-installer-to-pl2303-serial-usb-on-osx-lio> (<http://adafru.it/aWR>)

Both downloads are standard Mac installers. Accept all defaults when prompted.



Software Installation (Windows)

Download and install Putty from here: <http://www.putty.org/> (<http://adafru.it/aUb>)

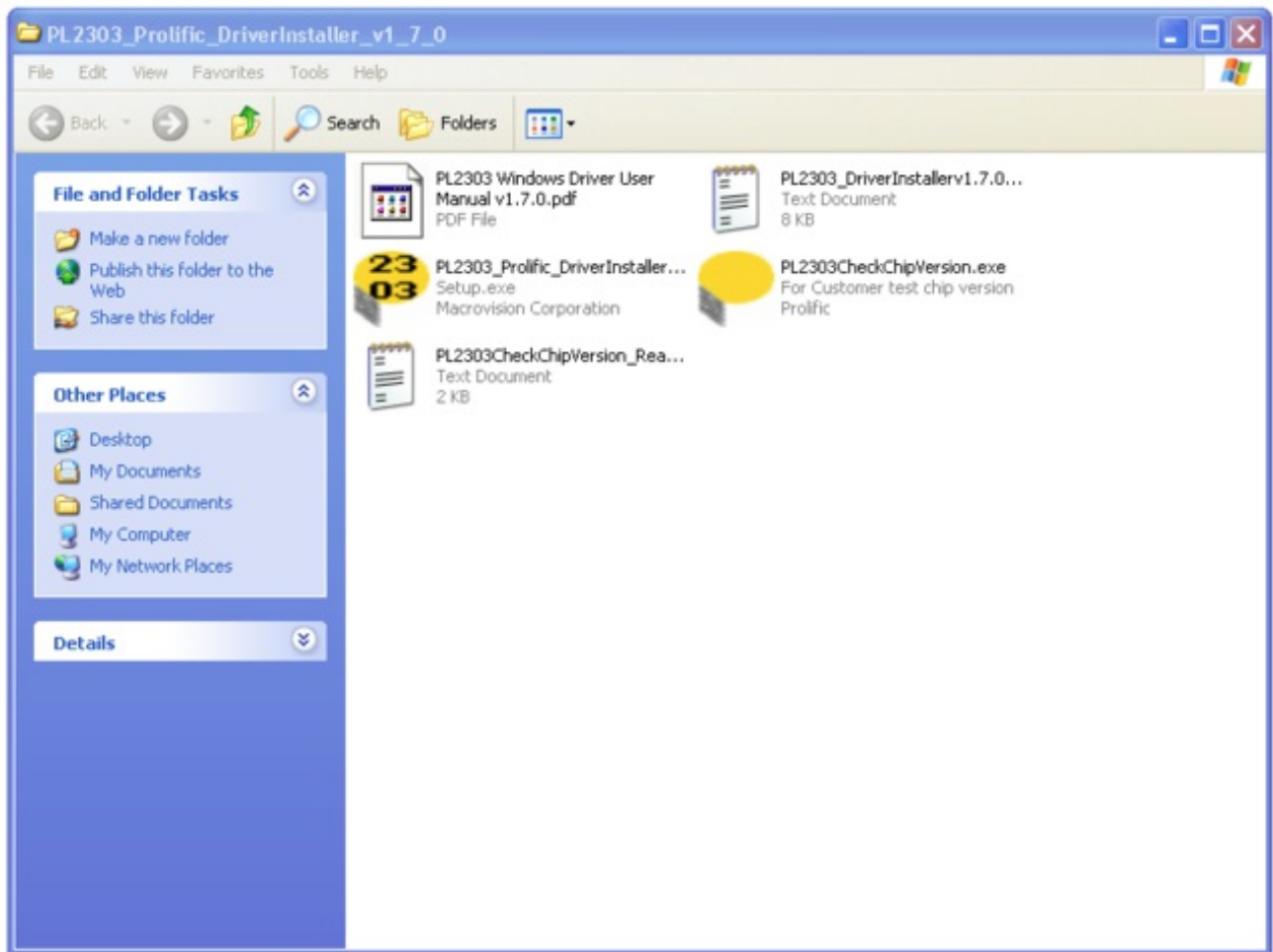
From the list of downloads select the binary called just "putty.exe" from the section "For Windows on Intel x86". This will prompt you to save the file. Save it onto the Desktop for now.

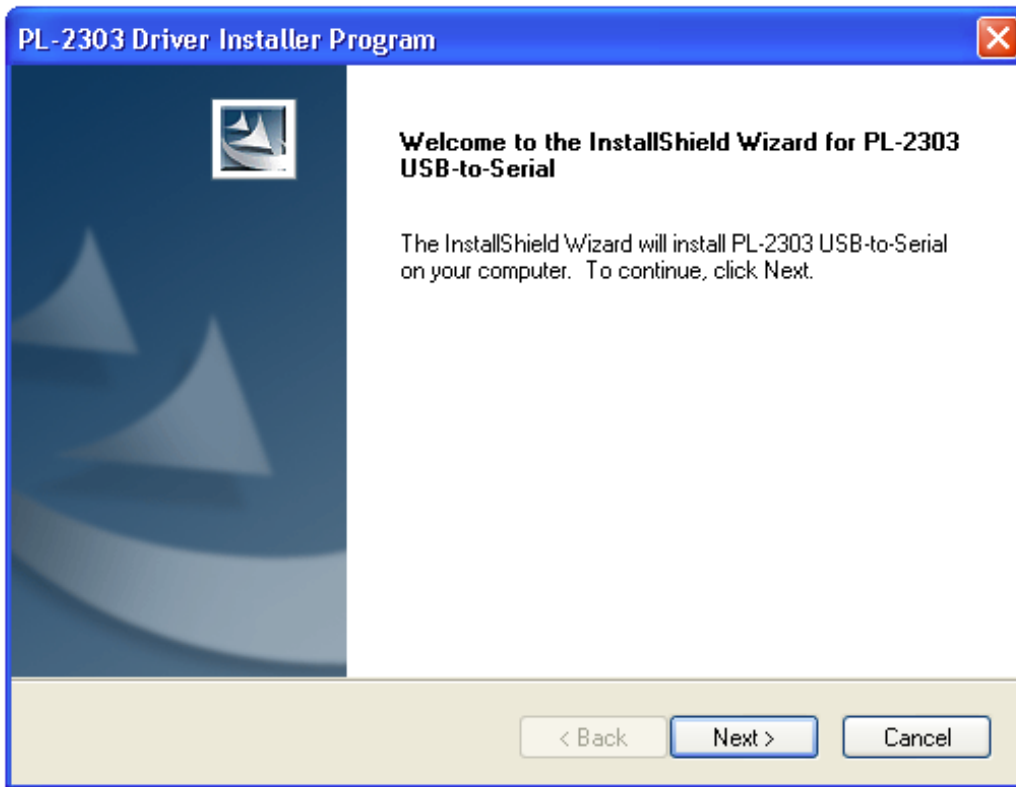
Note that this actually saves the Program itself not an installer. Simply double click putty.exe to run putty!

Install PL2303 Drivers from here: http://www.prolific.com.tw/US/ShowProduct.aspx?p_id=225&pcid=41 (<http://adafru.it/aTV>)

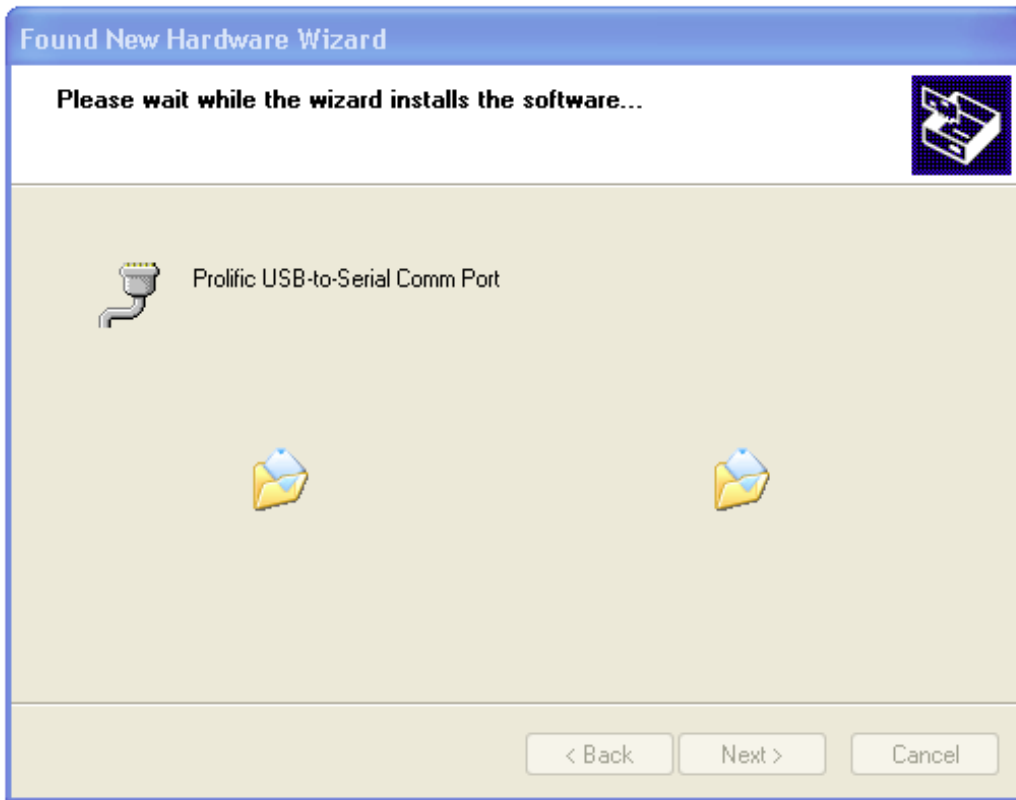
This will save a zip file called "PL2303_Prolific_DriverInstaller_v1_7_0.zip". Unzip this onto the desktop and within the folder run the installer called "PL2303_Prolific_DriverInstaller_v1.7.0.exe".

The current PL2303 console cables are not compatible with Windows 8!

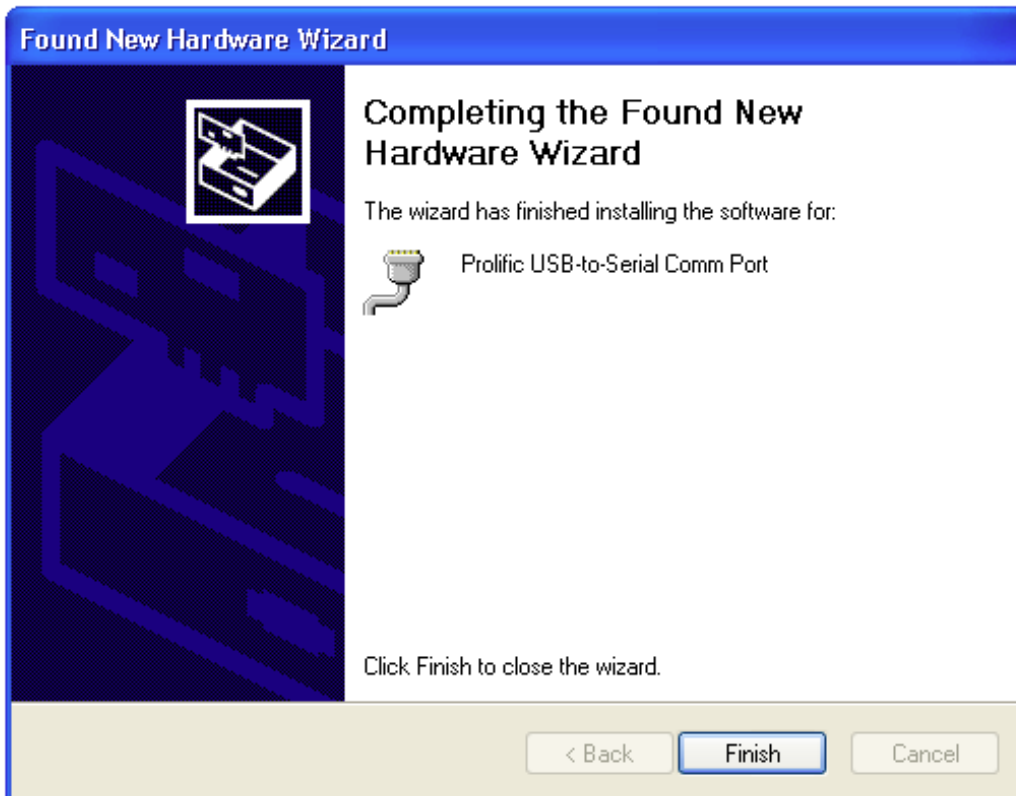




The driver is installed in such a way that when you later plug in the USB console lead, it will still launch the “Found New Hardware” wizard. If you allow the Wizard to search the Internet and install it should work.



When it has finished installing the driver, you should get this message:



Software Installation (Linux)

Linux Kernels 2.4.31 and above already have the PL2303 USB driver for the Console Lead built-in, so you should not need to install that.

Some distributions such as Ubuntu 12.10 do not include the "screen" command. Try running the command "screen" and if you get an error message, you can install it by typing the following command:

```
sudo apt-get install screen
```

Connect the Lead

The Console lead has four female connections that can be plugged directly onto the GPIO header of the Raspberry Pi.

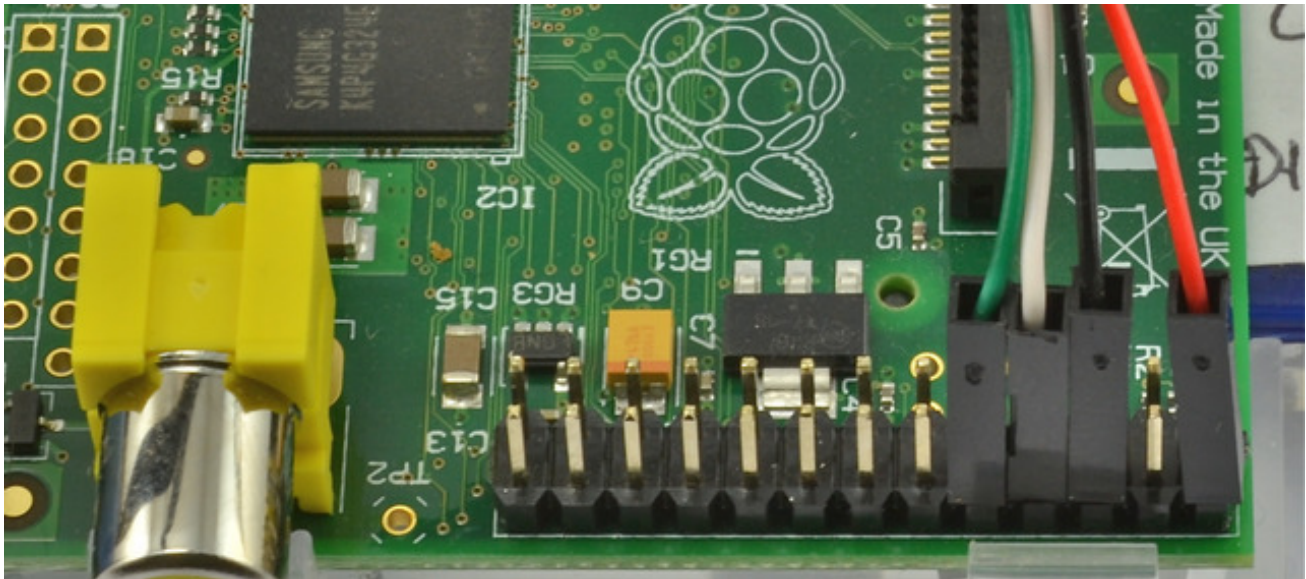
The Adafruit USB console cable has 3.3V logic, so its safe to use with your Pi.



For this experiment, the serial lead is going to power the Raspberry Pi. So, **DO NOT** attach the Pi's USB power adapter. If you would rather power the Pi from your USB power adapter then leave the Red lead from the Serial lead un attached.

The important thing here is to only power it from one source, the USB power adaptor or the Console Lead **BUT NOT BOTH**.

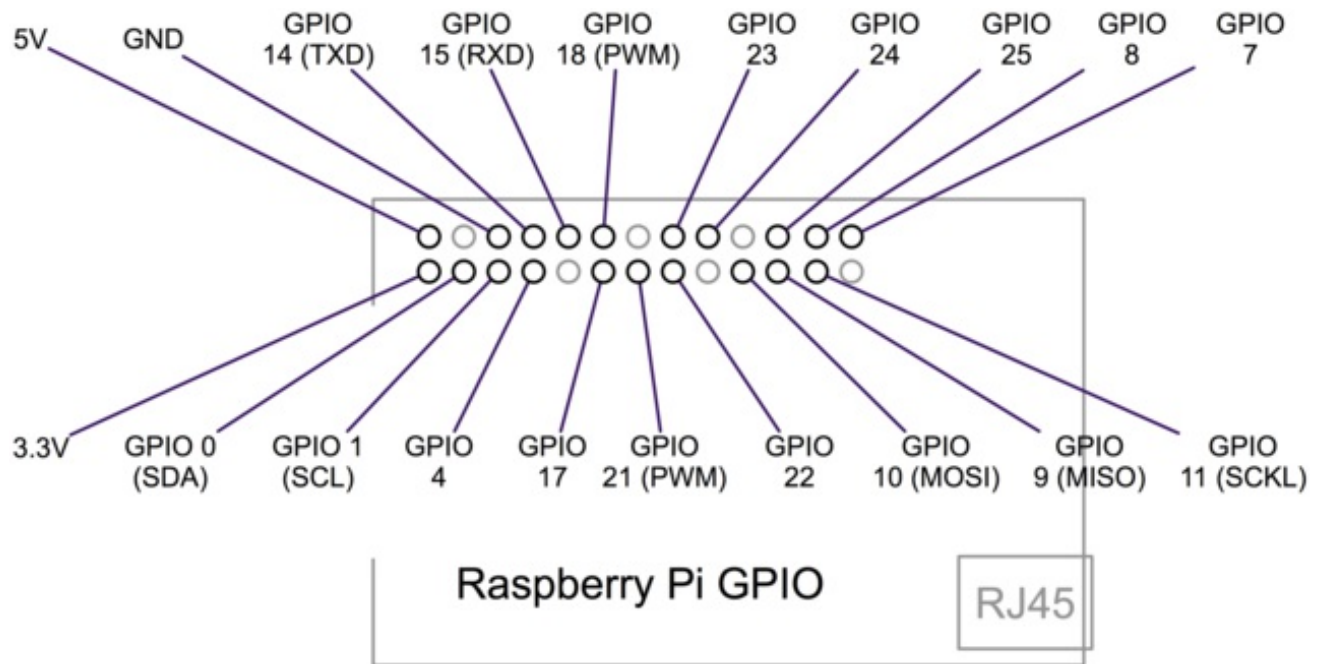
Attach the leads as shown below:



The connections are to the outside pin connections of the GPIO header. See Lesson 4, for more information about the header: <http://learn.adafruit.com/adafruits-raspberry-pi-lesson-4-gpio-setup/the-gpio-connector> (<http://adafru.it/aTW>)

- The red lead should be connected to 5V,
- The black lead to GND,
- The white lead to TXD.
- The green lead to RXD.

Check the diagram below if you are not sure about the connections. But note that the connect is being viewed upside down, so rotate the diagram by 180 degrees to compare it with the photograph.



Now plug in the USB lead and you will see the Pi's power LED should light.

Test & Configure

If you are using a Mac, then all you need to do is open a Terminal window and issue the command:

```
screen /dev/cu.PL2303-00001004 115200
```

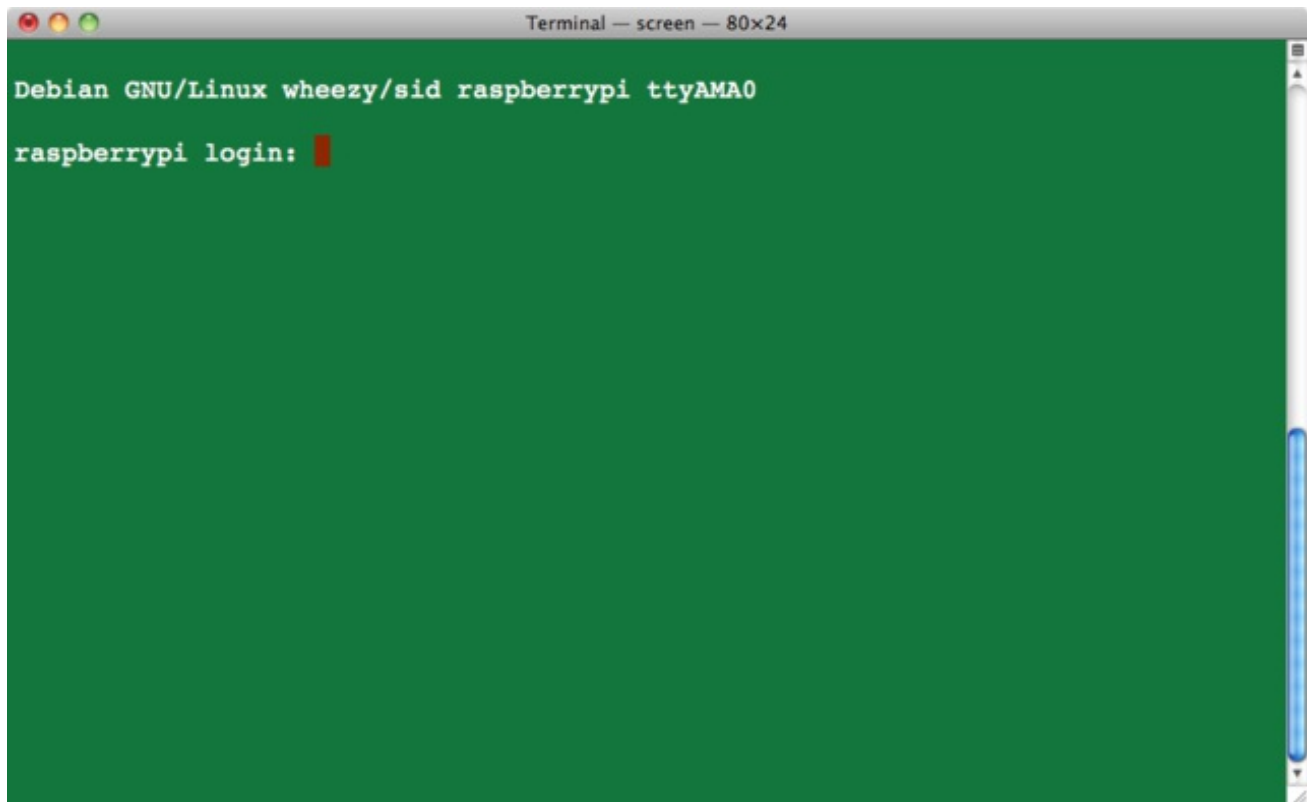
The device will have a slightly different name to mine. So, type the line above as far as "cu.PL" then press the TAB key to auto-complete to whatever your device is called, before adding 115200 (which is the baud rate) to the end.

If you are using Linux then use the command:

```
sudo screen /dev/ttyUSB0 115200
```

To start communication with the Pi, press ENTER and you should see the login prompt from the Pi.

Here it is running on a Mac.

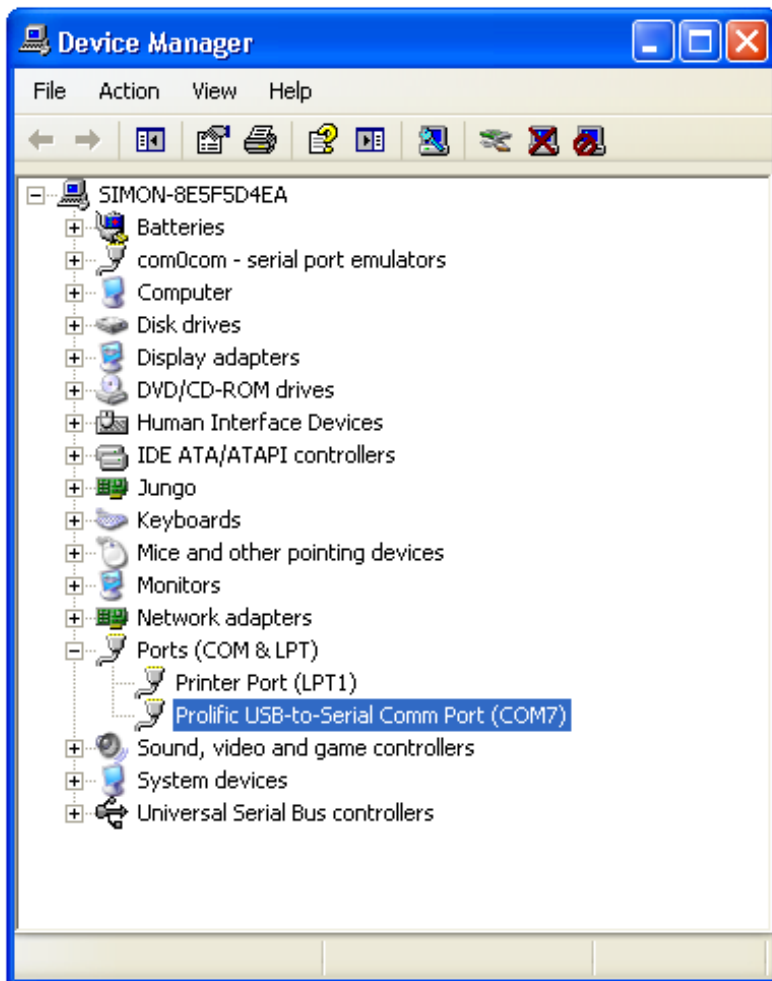


and here is what it looks like in Ubuntu.


```
simon@simon-UB: /dev
[ 2.889118] usb 1-1.1: New USB device strings: Mfr=0, Product=0, SerialNumber=0
[ 2.902423] smsc95xx v1.0.4
[ 2.974585] smsc95xx 1-1.1:1.0: eth0: register 'smc95xx' at usb-bcm2708_usb-1.1, smsc95xx USB 2.0 Ethernet, b8:27:eb:d8:9e:69
[ 5.317736] bcm2708_i2c bcm2708_i2c.0: BSC0 Controller at 0x20205000 (irq 79)
[ 5.392119] Adafruit Industries' Raspberry Pi PWM driver v1.0
[ 5.480127] bcm2708_spi bcm2708_spi.0: SPI Controller at 0x20204000 (irq 80)
[ 5.531192] bcm2708_i2c bcm2708_i2c.1: BSC1 Controller at 0x20804000 (irq 79)
[ 9.763013] EXT4-fs (mmcblk0p2): re-mounted. Opts: (null)
[ 10.116994] ### snd_bcm2835_alsa_probe c067cbf8 ##### PROBING FOR bcm2835 ALSA device (0):(1) #####
[ 10.134267] Creating card...
[ 10.138094] Creating device/chip ..
[ 10.143196] Adding controls ..
[ 10.147169] Registering card ....
[ 10.158814] bcm2835 ALSA CARD CREATED!
[ 10.165128] ### BCM2835 ALSA driver init OK ###
[ 10.260888] i2c /dev entries driver
◆◆
Debian GNU/Linux wheezy/sid raspberrypi ttyAMA0
raspberrypi login: █
```

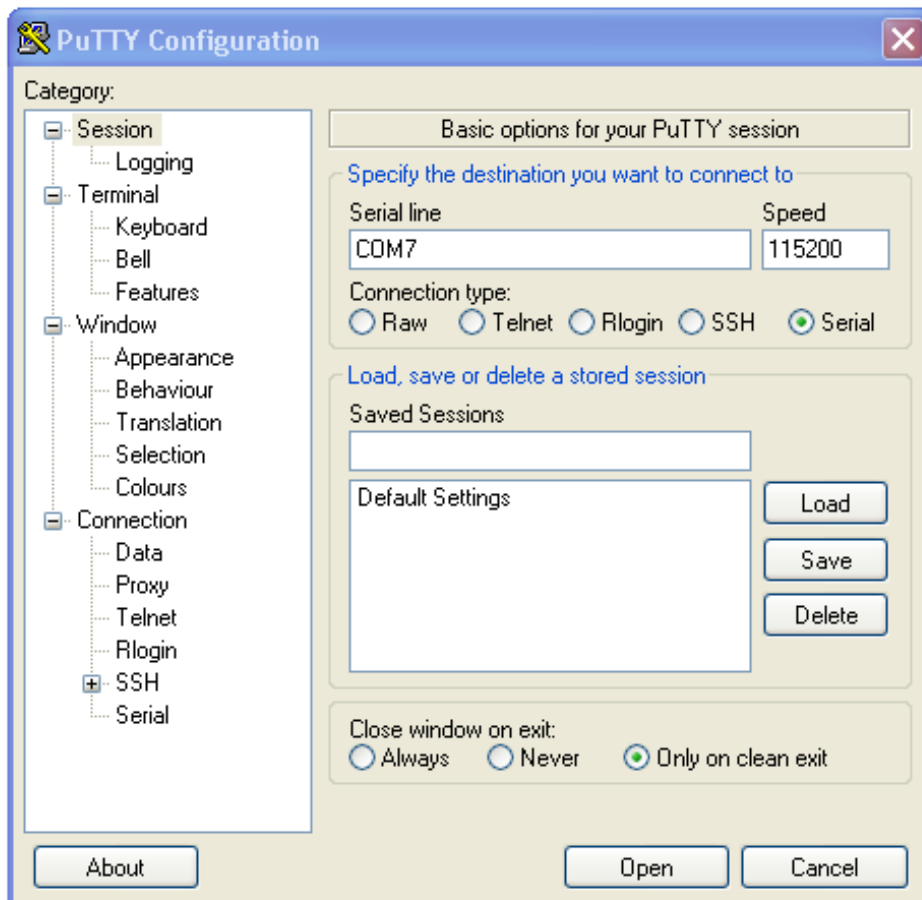
If you are using a PC, then before you start Putty, you need to know which com port is being used for the cable. You can find this by looking in the Ports section of the Windows Device Manager.

The Device Manager is accessible from the Control Panel under System.



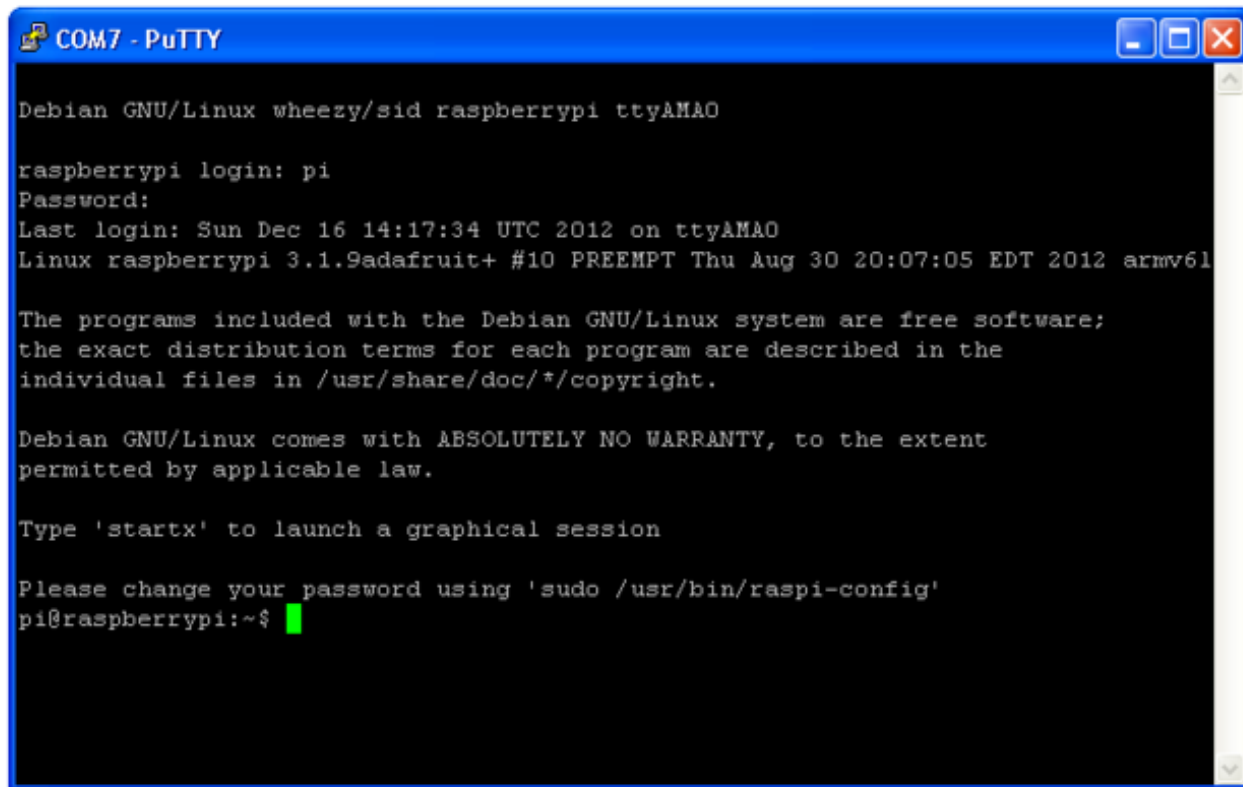
In this case it is COM7 that is in use.

Now start Putty and you will see a connection window.



Select a connection type of "Serial" from the radio buttons, then set the speed to 115200 and the serial line to COM7.

Finally click 'Open' to connect. Remember to press ENTER to start communications.



```
COM7 - PuTTY

Debian GNU/Linux wheezy/sid raspberrypi ttyAMA0

raspberrypi login: pi
Password:
Last login: Sun Dec 16 14:17:34 UTC 2012 on ttyAMA0
Linux raspberrypi 3.1.9adafruit+ #10 PREEMPT Thu Aug 30 20:07:05 EDT 2012 armv6l

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.

Type 'startx' to launch a graphical session

Please change your password using 'sudo /usr/bin/raspi-config'
pi@raspberrypi:~$
```

For a new installation of Occidentalis or Wheezy, the default username is “pi” and the default password is “raspberrypi”.

That's it! You are connected and can use the command line to navigate around your Pi.

In the next session we will look at using SSH as another means of connecting to your Pi over your local network.