Raspberry Pi 2 Model B and JMRI for Wired Connection to Router/Access Point for Pi-SPROG One and Pi-SPROG Nano 2016-11-25-raspian-jessie build

These instructions describe the process of setting up a Raspberry Pi 2 or 3 Model B to use a wired connection to a wireless router that will be used as an access point (AP) for connection by hand-held devices running WiThrottle, EngineDriver or other similar apps.

If you have downloaded or purchased the image on SD card from SPROG DCC then you do not need these instructions, other than for reference.

Some familiarity with the Raspberry Pi and Linux operating system is assumed, especially the use of the command line in a terminal and simple editors such as vi or nano.

Where we say "we like…" in the following instructions, we indicate how our SPROG DCC pre-built card images are made. These steps can be tailored by more advanced users to your own preference.

Hardware Requirements

Raspberry Pi 2 or 3 model B and suitable power supply for above

MicroSD memory card (8Gb recommended)

Pi-SPROG One and power supply

Host PC (Windows or Linux)

Wired network connection from the R-Pi to your router.

Software Requirements

SSH client installed on the host. These instructions assume the use of putty.exe http://www.chiark.greenend.org.uk/~sgtatham/putty/download.html on Windows.

VNC Viewer installed on the host, https://www.realvnc.com/download/viewer/

Create the SD Card Image

Start with the latest Raspbian Jesse image (2016-11-25 at the time of writing) installed on an SD card. See https://www.raspberrypi.org/documentation/installation/installing-images/README.md

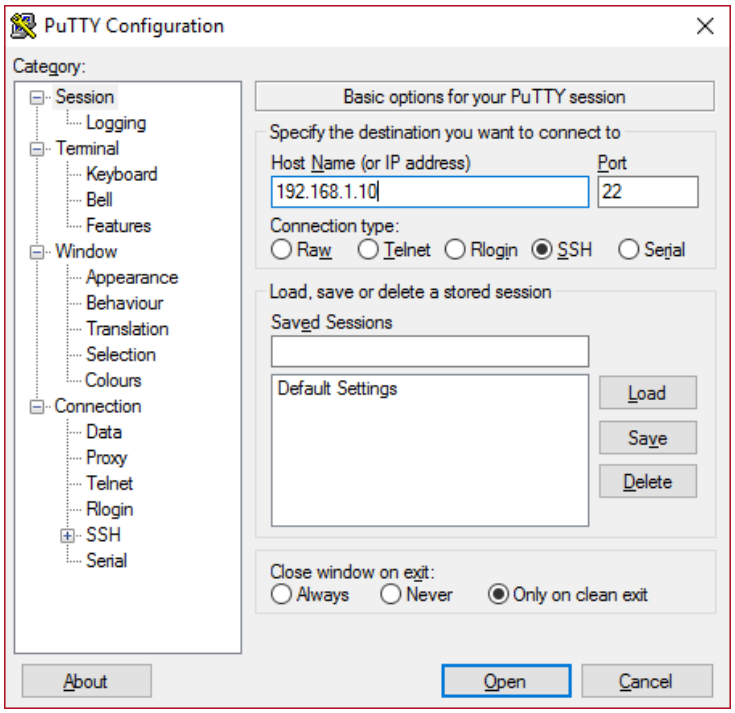
Create a file named ssh in the boot partition to enable SSH. With a windows host do this by viewing the contents of the SD card in Windows Explorer, right click, New Text Document and change the name to ssh. This will enable SSH and allow us to proceed to the next step.

Put the SD card in the R-Pi's SD card slot and connect power to the R-Pi.

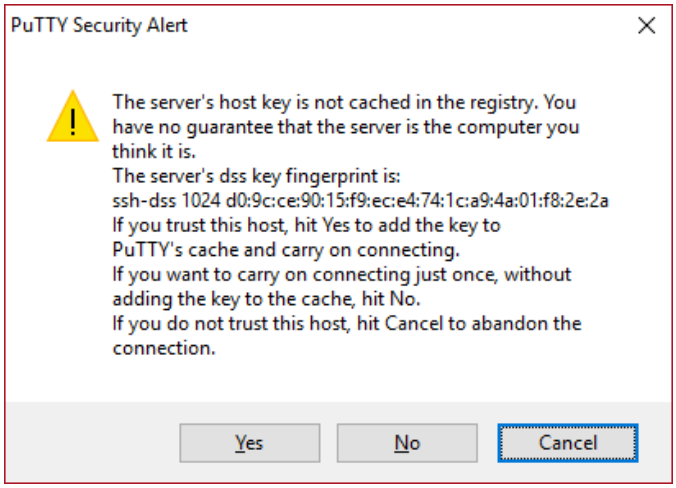
Connecting With SSH

Determine the IP address assigned to the R-Pi by your router. We recommend using your router setup to assign a static IP address to the R-Pi. You will need to refer to your particular router's instructions for this.

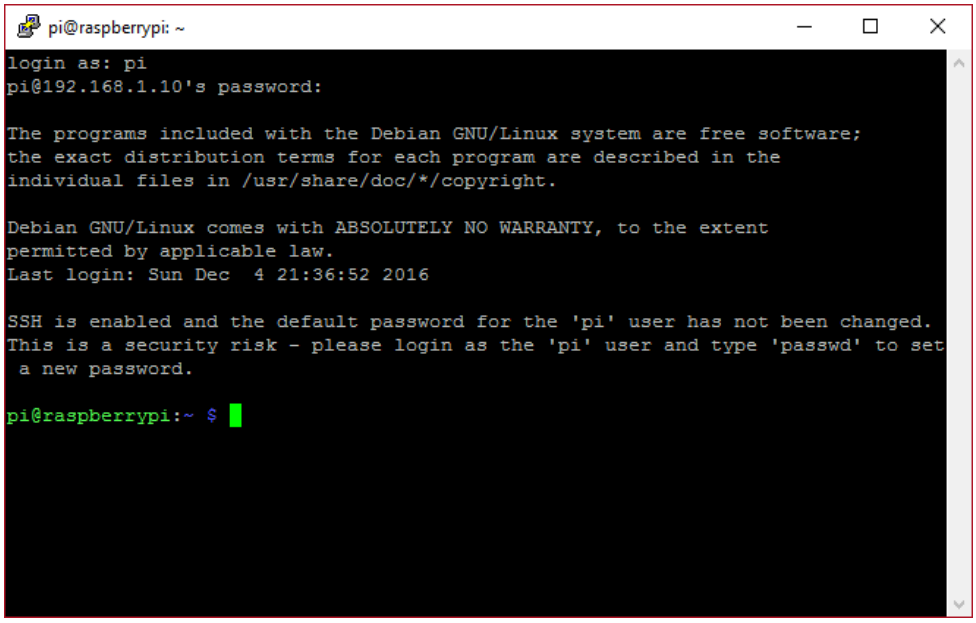
Use the IP address in putty to connect to the R-Pi using SSH.



Click Yes if presented with a security alert.



When prompted enter the default user name ‘pi’ and password ‘raspberry’



Note the warning message “SSH is enabled ...”

We strongly recommend that you change the password before proceeding, especially if you intend making your R-Pi visible on the Internet.

`Bold Consolas font text` indicates text that is typed at the R-Pi command line in an SSH shell, or a desktop terminal window.

> **`Indented Bold Consolas font text`** indicates text that is entered in text editor.

Ensure you have the latest updates.

```
sudo apt-get update
```

and then

```
sudo apt-get dist-upgrade
```

The updates may take a little time. Answer yes if prompted to continue.

Reboot your R-Pi.

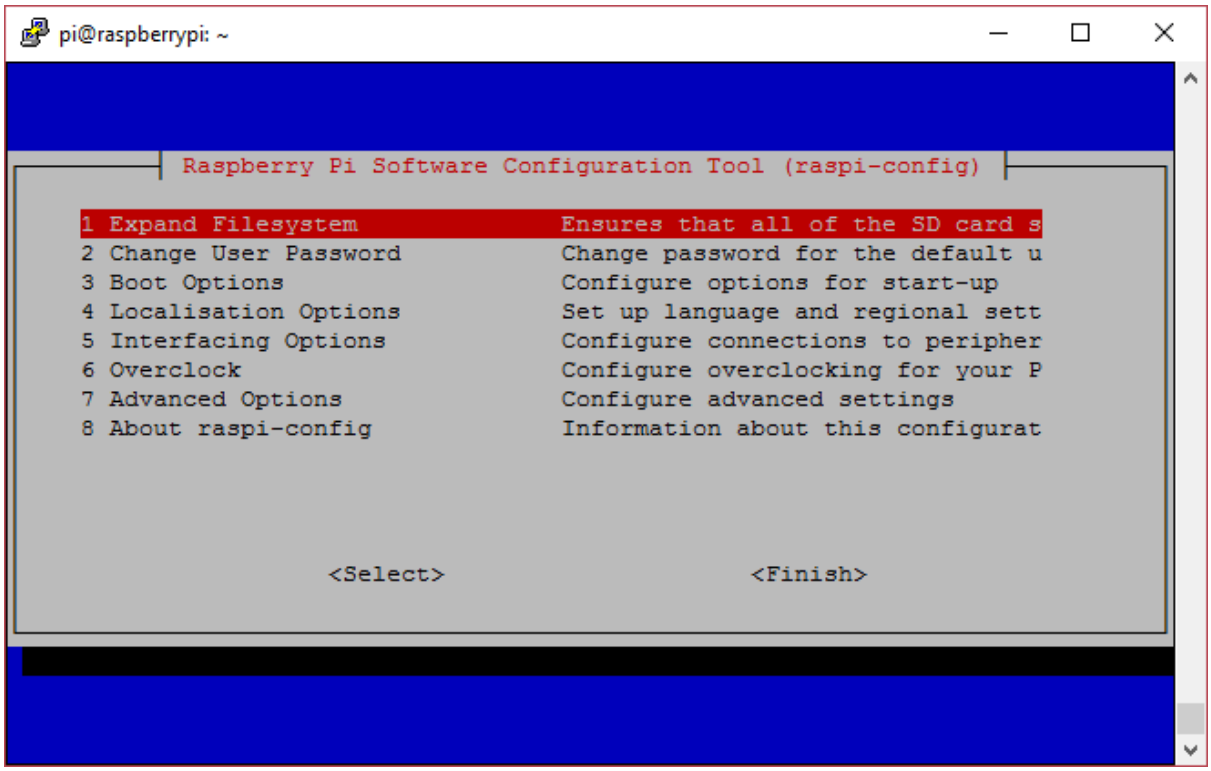
```
sudo reboot
```

Close the putty window and then reconnect (with putty) once the R-Pi has rebooted.

Enable VNC

To allow remote access to to virtual desktop, rather than the SSH command line, we need to enable VNC which is included with the Raspian distribution. Use the raspi-config utility.

sudo raspi-config



Use the cursor keys to select Interfacing Options and enable VNC.

Exit raspi-config.

Next we will change the screen resolution for the virtual desktop.

sudo nano /boot/config.txt

Scroll to the end of the file and add the following lines for 1280x800 resolution.

```
hdmi_force_hotplug=1
hdmi_ignore_edid=0xa5000080
hdmi_group=2
hdmi_mode=28
```

Save the file and quit the editor.

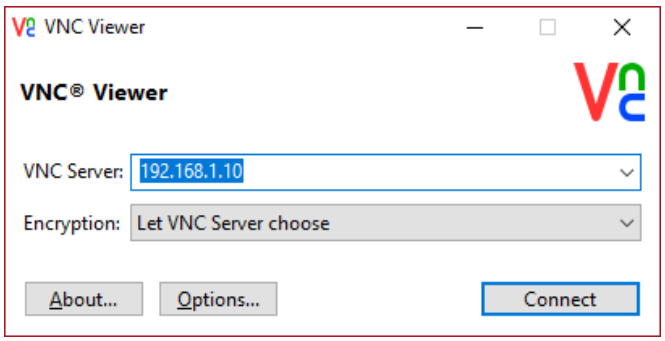
See https://support.realvnc.com/knowledgebase/article/View/523?_ga=1.23202815.450528412.1480880479 for the possibilities. See https://www.raspberrypi.org/documentation/configuration/config-txt.md for other options.

Reboot your R-Pi.

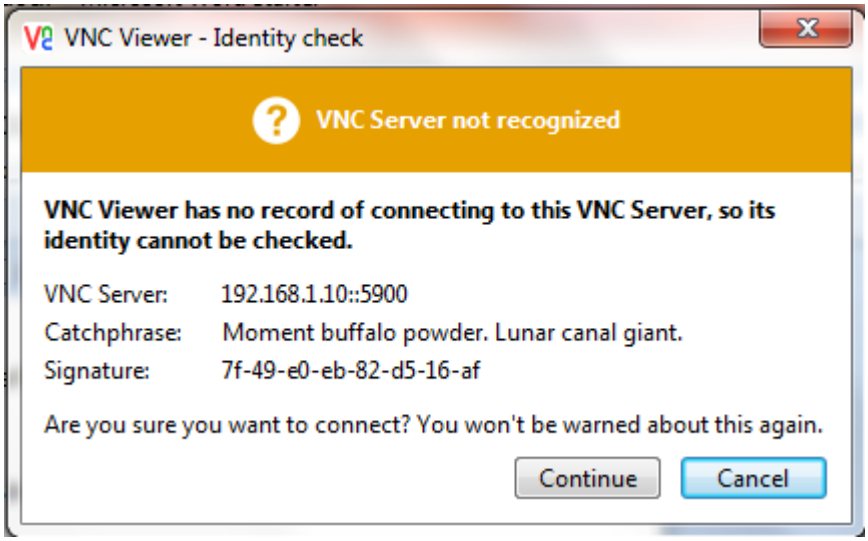
sudo reboot

Connecting With VNC

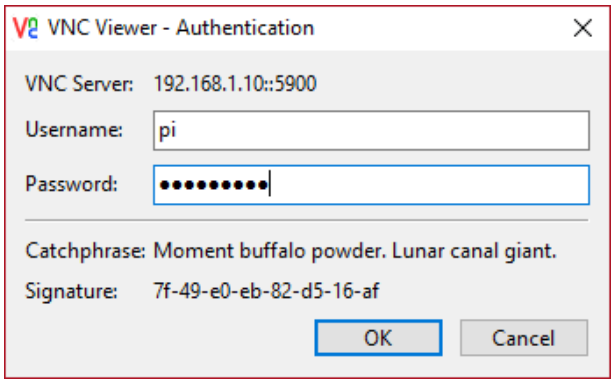
Start VNC viewer on the host PC. Enter the IP address and click Connect.



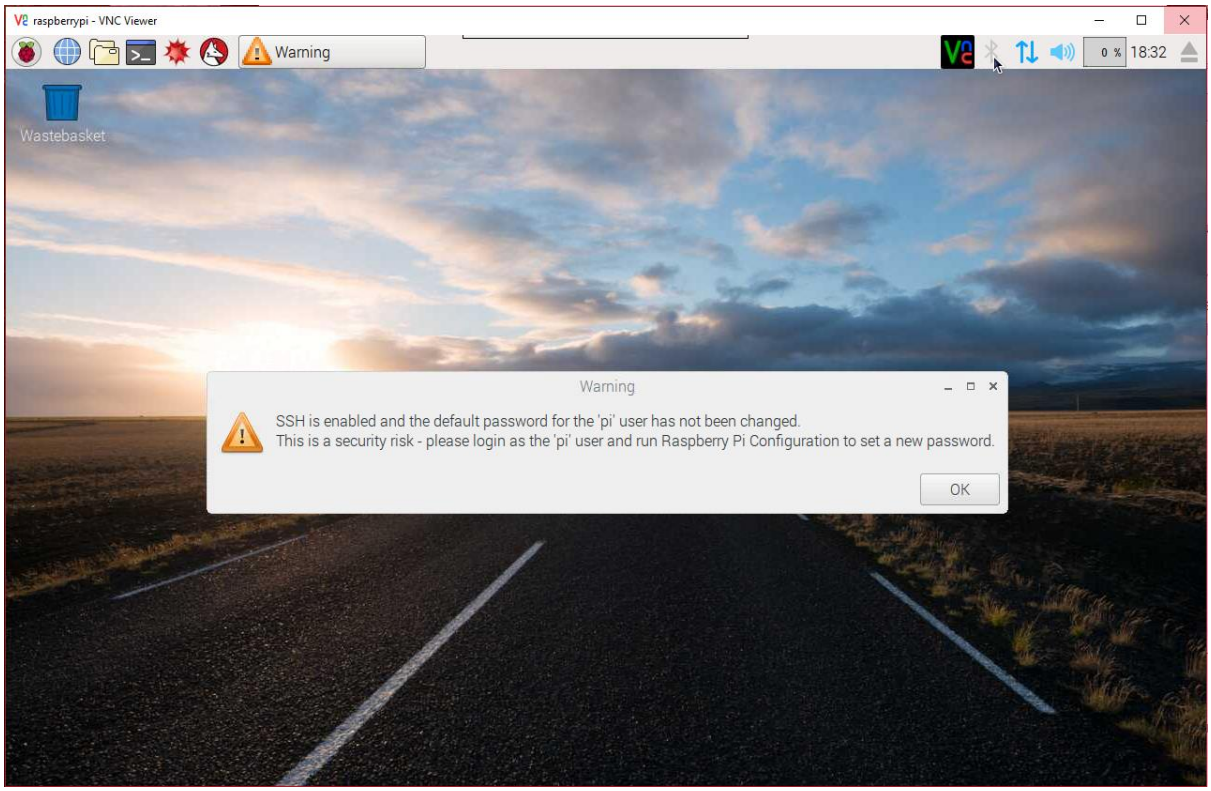
The first time you connect you will see an identity check



Enter the username and password

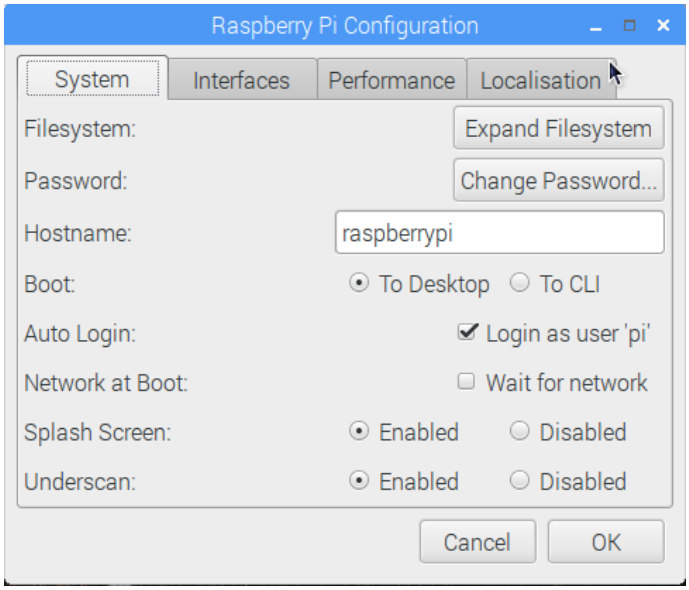


If you have not changed the default password, you will see a warning message each time you connect



R-Pi Configuration

Click the Raspberry logo and select Preferences -> Raspberry Pi Configuration



Change the Hostname (optional, useful if you have more than one R-Pi). We use sprog-pi for our images.

Click OK, then click No to postpone the reboot.

UART setup

We need to edit the command line to prevent the Kernel using the UART at startup. Open a terminal and type

```
sudo nano /boot/cmdline.txt
```

Remove 'console=serial0,115200' noting that this file must contain only one line. Be careful not to split the line when editing it.

Save the file and quit the editor.

Reboot your R-Pi. Close the VNC window.

Reconnect using VNC once the R-Pi has rebooted.

Check that the UARTs are mapped correctly, open a terminal and type

```
ls –l /dev/ser*
```

The output should be something like

lrwxrwxrwx 1 root root 5 Dec 13 13:34 /dev/serial0 -> ttyAMA0

Install a terminal Emulator for SPROG Command Line Access

We like to do this for testing purposes, but it is entirely optional.

```
sudo apt-get install minicom
```

Answer yes if prompted to continue.

Shutdown your R-Pi.

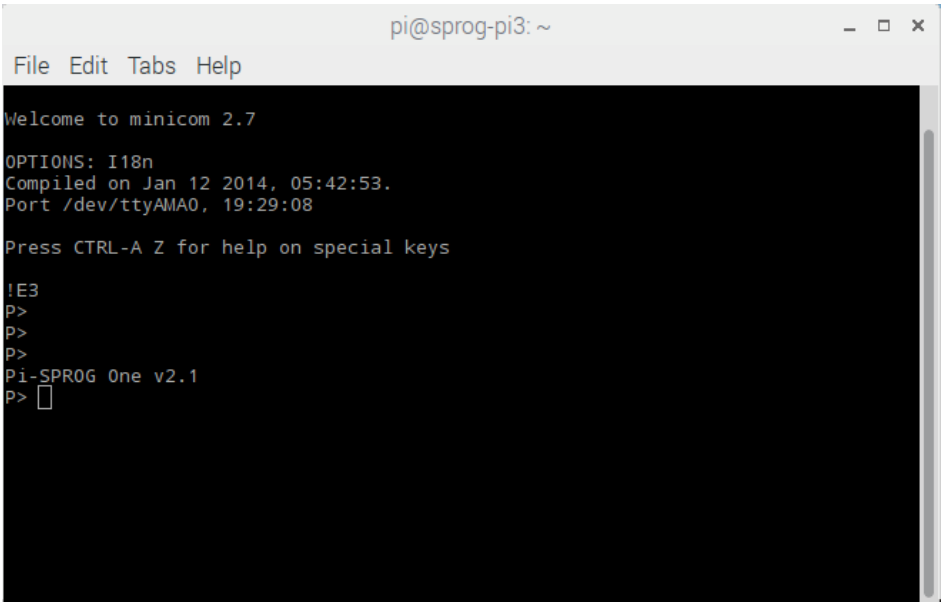
Connect the SPROG-Pi and its power supply.

Start your R-Pi.

You can communicate with the Pi-SPROG using Mincom:

```
minicom --device=/dev/serial0 --baud=115200
```

As an example, type carriage return a couple of times to get the SPROG prompt 'P> ' then type a ? followed by carriage return to see the Pi-SPROG version.



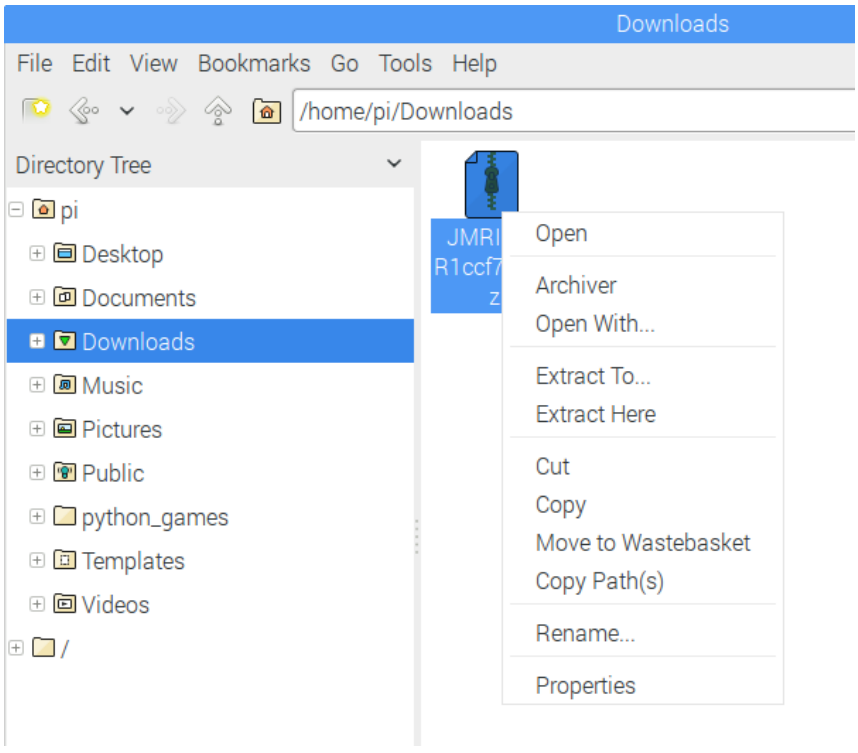
To exit minicom type control-A X then select yes.

You only need to use minicom for diagnostics such as checking that he Pi-SPROG is working. From now on all other communication with the Pi-SPROG will be handled by DecoderPro.

Download and Install JMRI

Using the R-Pi web browser, navigate to www.jmri.org/download/index.shtml and download your chosen JMRI version and extract the archive to /home/pi/ or your preferred location.

Open the Downloads folder in File Manager, right click the downloaded archive and Extract To…



Change the path to your chosen location.

If extracted to /home/pi then the path to the application will be /home/pi/JMRI which we use in the following instructions.

Create a desktop icon for DecoderPro in an editor

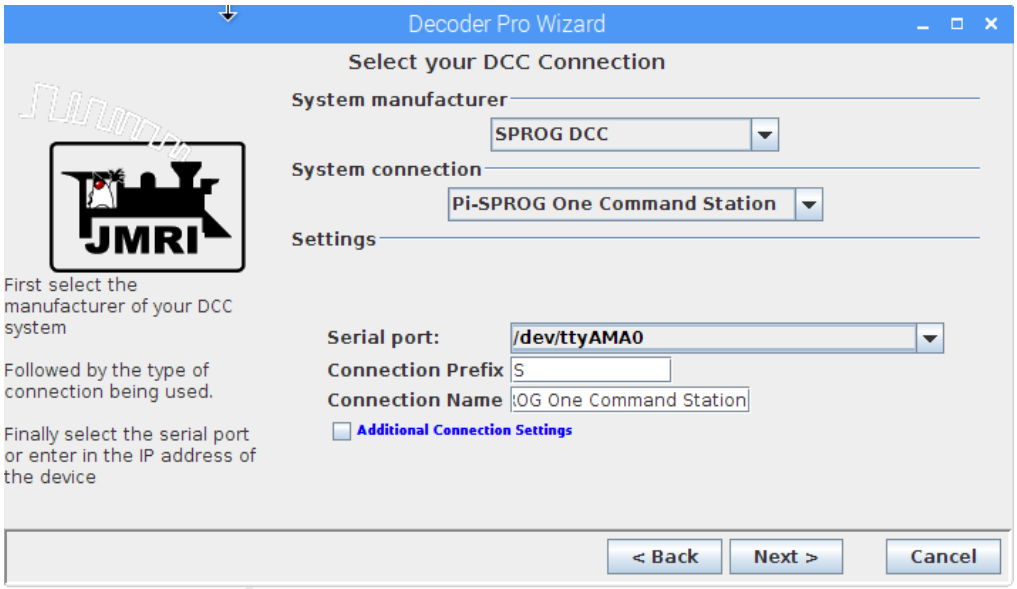
```
sudo nano /home/pi/Desktop/DecoderPro.desktop
```

```
[Desktop Entry]
Type=Application
Encoding=UTF-8
Name=DecoderPro
Comment=JMRI DecoderPro
Icon=/home/pi/JMRI/resources/dp3_48x48.gif
Exec=/home/pi/JMRI/DecoderPro
Terminal=false;
```

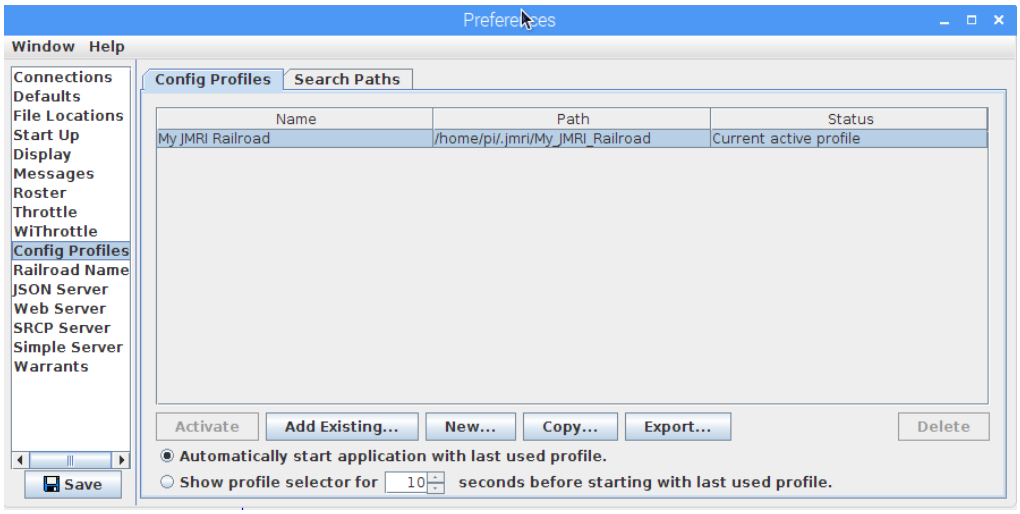
Save the file and quit the editor.

Start DecoderPro by double clicking the new icon.

Create a new profile by selecting SPROG DCC as the System and Pi-SPROG One Command Station as the System connection. Select /dev/ttyAMA0 as the Serial port.

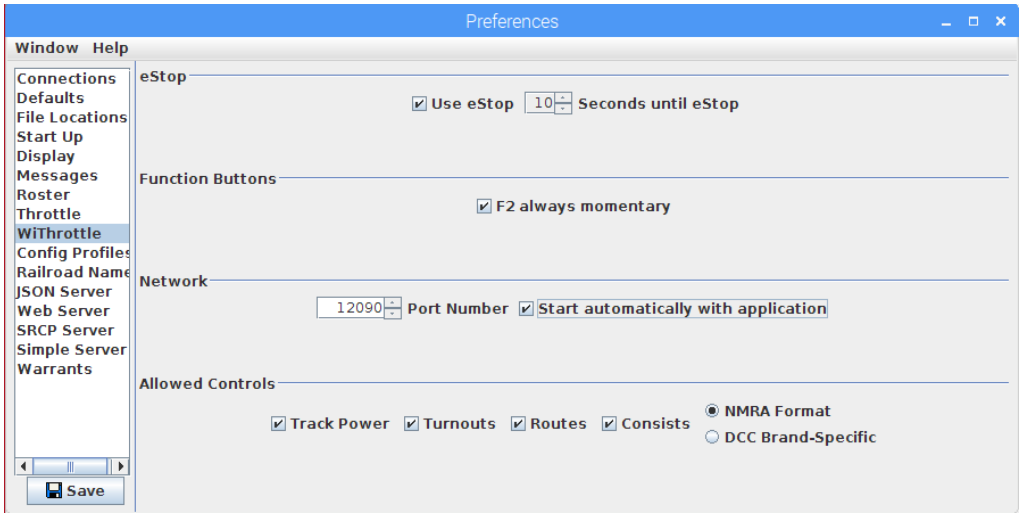


Once DecoderPro is running edit the preferences, config profiles and select “Automatically start application with last used last profile”.



Start Withrottle server automatically

Edit DecoderPro preferences, WiThrottle, Start automatically on port 12090.



Save the preferences and restart DecoderPro.

The WiThrottle server should start with DecoderPro and you should now be able to connect to it using e.g., WiThrottle or EngineDriver apps via your WiFi router.

Run DecoderPro at Startup

In a terminal open the session autostart file

```
sudo nano /home/pi/.config/lxsession/LXDE-pi/autostart
```

Add the following line, with the path to the location where you installed DecoderPro

```
@/home/pi/JMRI/DecoderPro
```

Save the file and quit the editor.

Reboot your R-Pi.

JMRI and WiThrottle server should start up automatically.

That’s it! You are now setup to drive your trains.