

Raspberry Pi 3 Model B and JMRI with WiFi Access Point for Pi-SPROG One and Pi-SPROG Nano 2016-11-25-raspian-jessie build

© SPROG DCC December 31st 2016

These instructions describe the process of setting up a Raspberry Pi 3 Model B as a WiFi 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 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 during setup and if you require network (e.g. internet) access whilst using the R-Pi as an Access Point.

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>

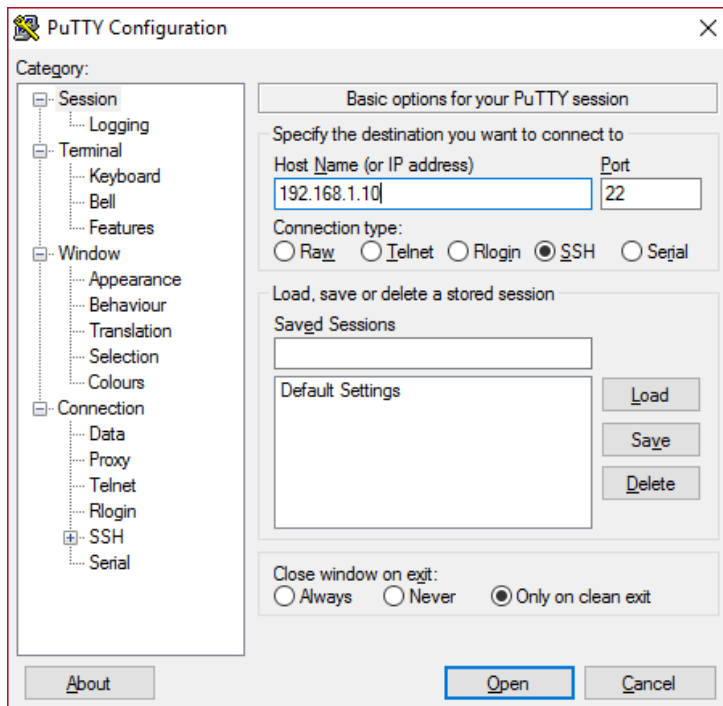
Create a file named ssh in the boot partition to enable SSH. With a windows host do this by viewing the contents of the SD card in Windows Explorer, right click, New Text Document and change the name to ssh. This will enable SSH and allow us to proceed to the next step.

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

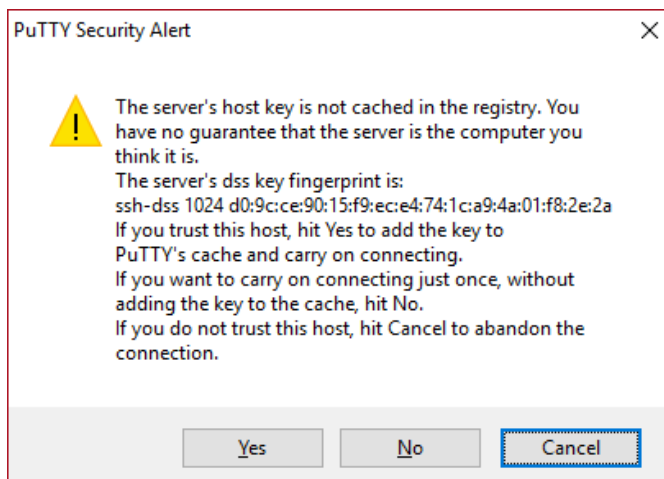
Connecting With SSH

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.

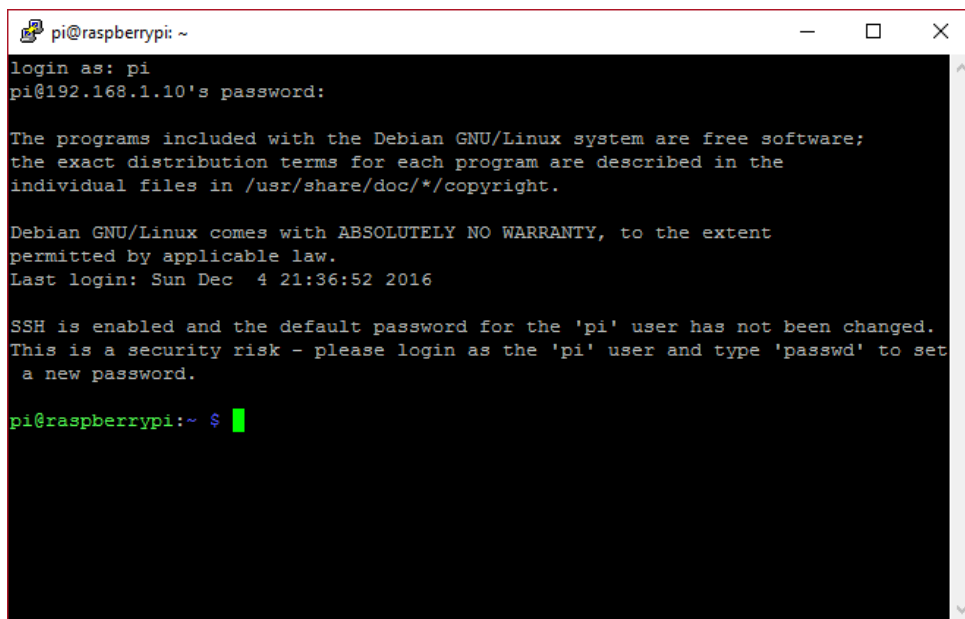
Use the IP address in putty to connect to the R-Pi using SSH.



Click Yes if presented with a security alert.



When prompted enter the default user name 'pi' and password 'raspberrypi'



```
pi@raspberrypi: ~
login as: pi
pi@192.168.1.10's password:

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.
Last login: Sun Dec  4 21:36:52 2016

SSH is enabled and the default password for the 'pi' user has not been changed.
This is a security risk - please login as the 'pi' user and type 'passwd' to set
a new password.

pi@raspberrypi:~ $
```

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.

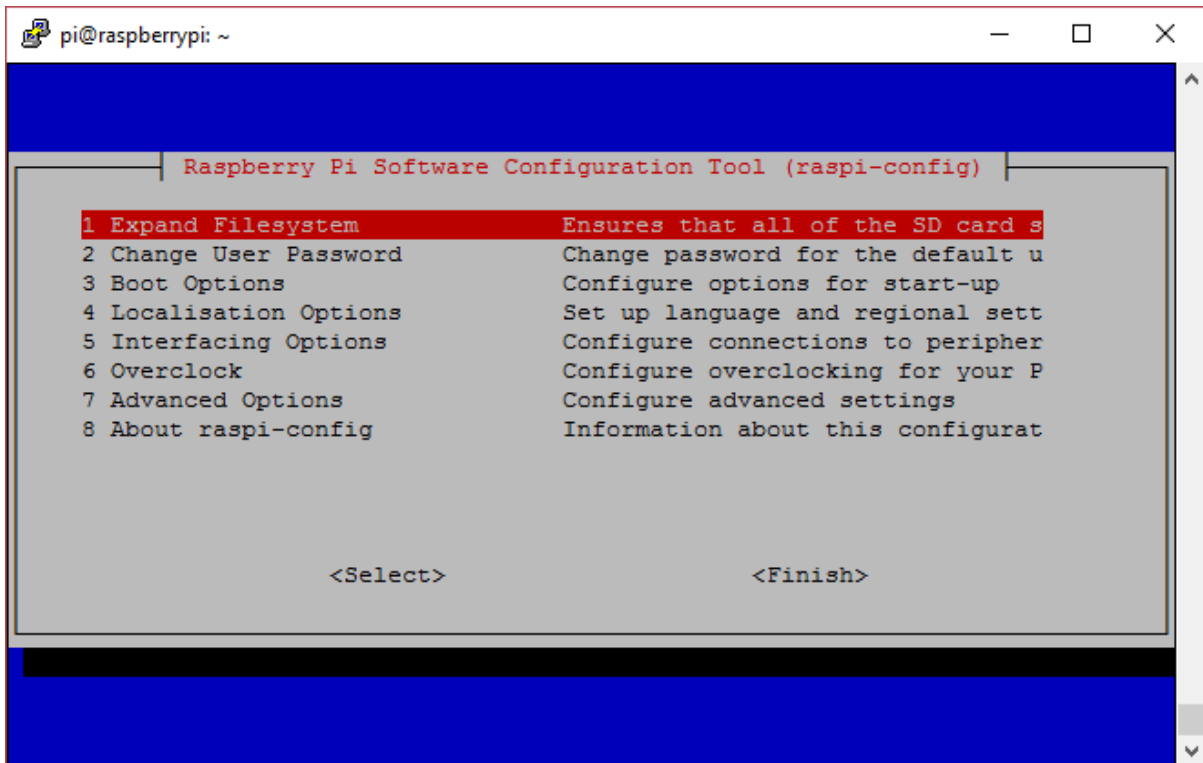
sudo reboot

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

Enable VNC

To allow remote access to the 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, see

https://support.realvnc.com/knowledgebase/article/View/523?_ga=1.23202815.450528412.1480880479 for the possibilities.

sudo nano /boot/config.txt

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

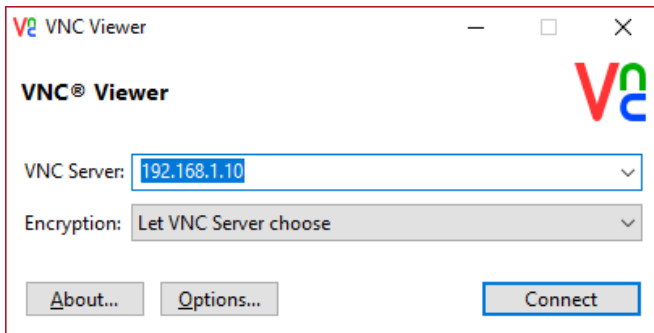
<https://www.raspberrypi.org/documentation/configuration/config-txt.md> for other options.

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

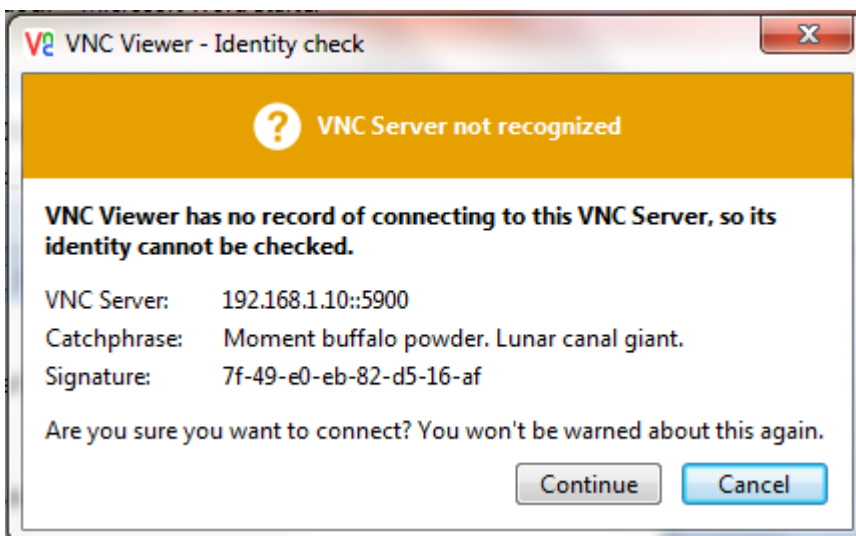
Reboot your R-Pi.

Connecting With VNC

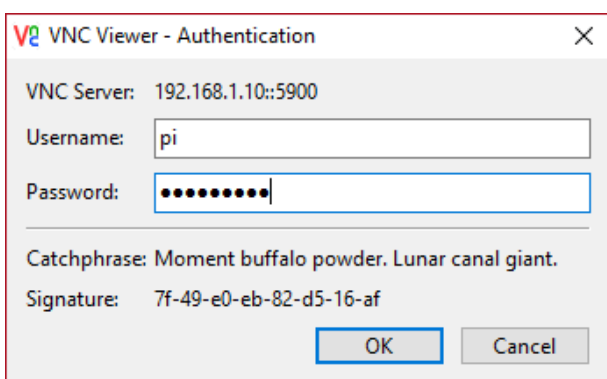
Start VNC viewer on the host PC. Enter the IP address and click Connect.



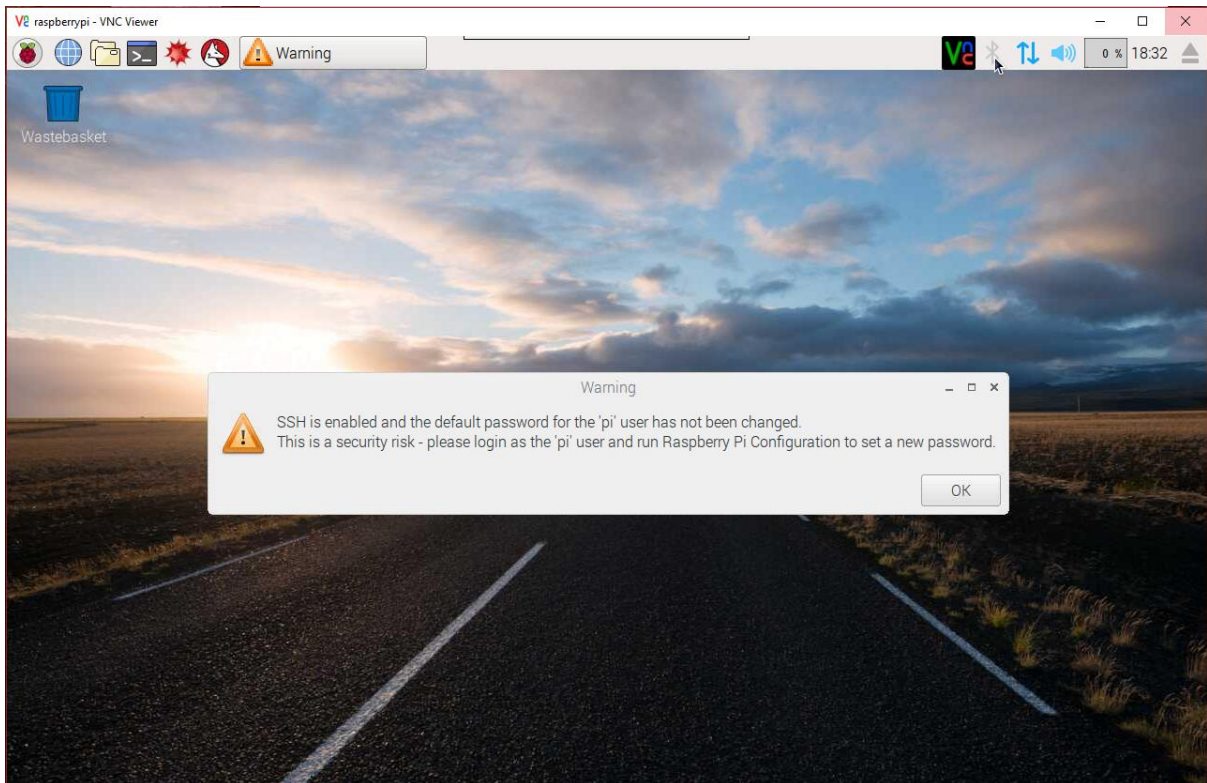
The first time you connect you will see an identity check



Enter the username and password

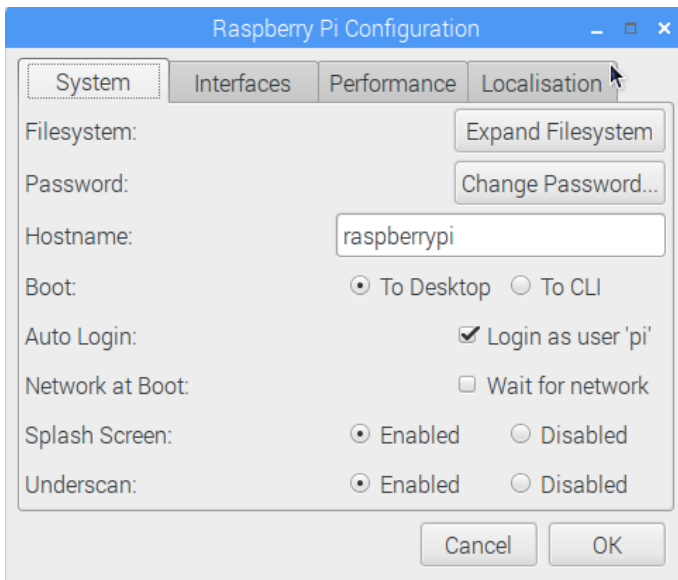


If you have not changed the default password, you will see a warning message each time you connect



R-Pi Configuration

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



Click Expand Filesystem

Change the Hostname (optional, useful if you have more than one R-Pi). We use sprog-pi3 for our images.

Click OK, then click No to postpone the reboot.

UART setup

We need to swap the UARTs so that serial0 appears on GPIO instead of being used for BlueTooth. This is done by adding an overlay to config.txt. We need to be root to edit this file so open a terminal and type

```
sudo nano /boot/config.txt
```

Scroll to the end of the file and add the lines

```
dtoverlay=pi3-miniuart-bt-overlay  
enable_uart=1
```

Save the file and quit the editor.

Edit the command line to prevent the Kernel using the UART at startup.

```
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 -> ttyS0  
lrwxrwxrwx 1 root root 5 Dec 13 13:34 /dev/serial1 -> 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.

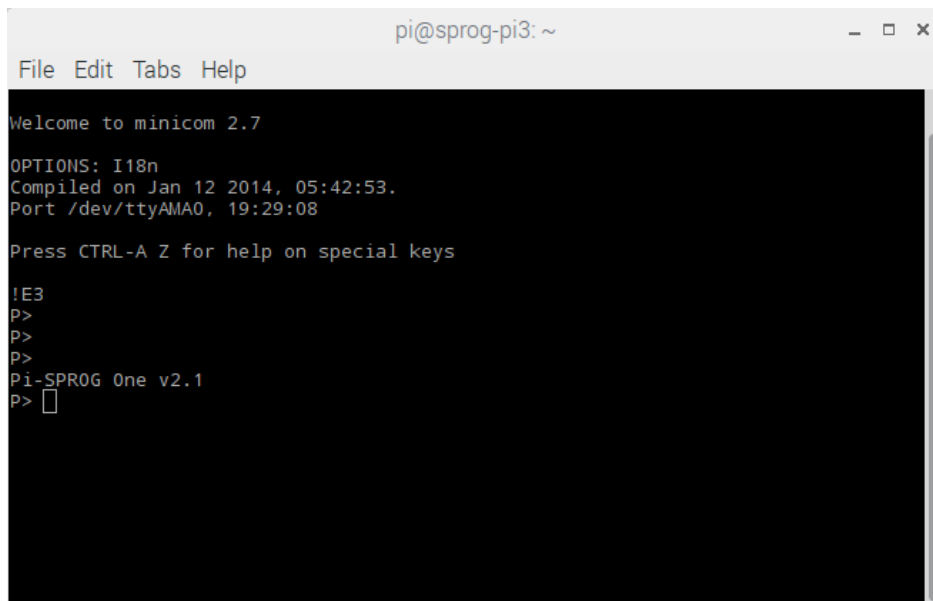
Connect the SPROG-Pi and it's power supply.

Start your R-Pi.

You can communicate with the Pi-SPROG using Minicom:

```
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.



```
pi@sprog-pi3: ~
File Edit Tabs Help
Welcome to minicom 2.7
OPTIONS: I18n
Compiled on Jan 12 2014, 05:42:53.
Port /dev/ttyAMA0, 19:29:08
Press CTRL-A Z for help on special keys
!E3
P>
P>
P>
P>
Pi-SPROG One v2.1
P> █
```

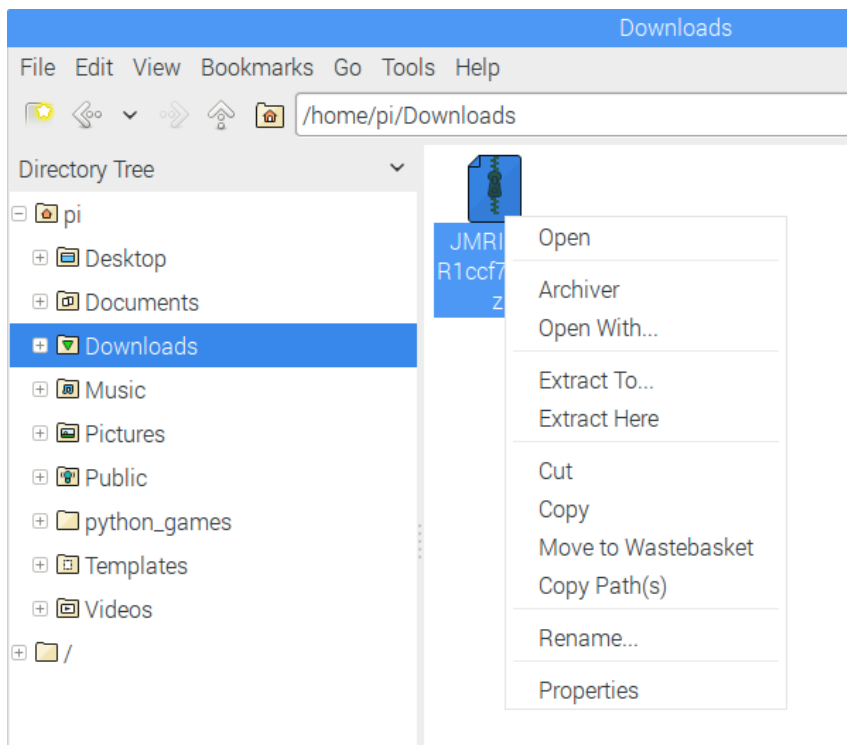
To exit minicom type control-A X then select yes.

You only need to use minicom for diagnostics such as checking that the 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

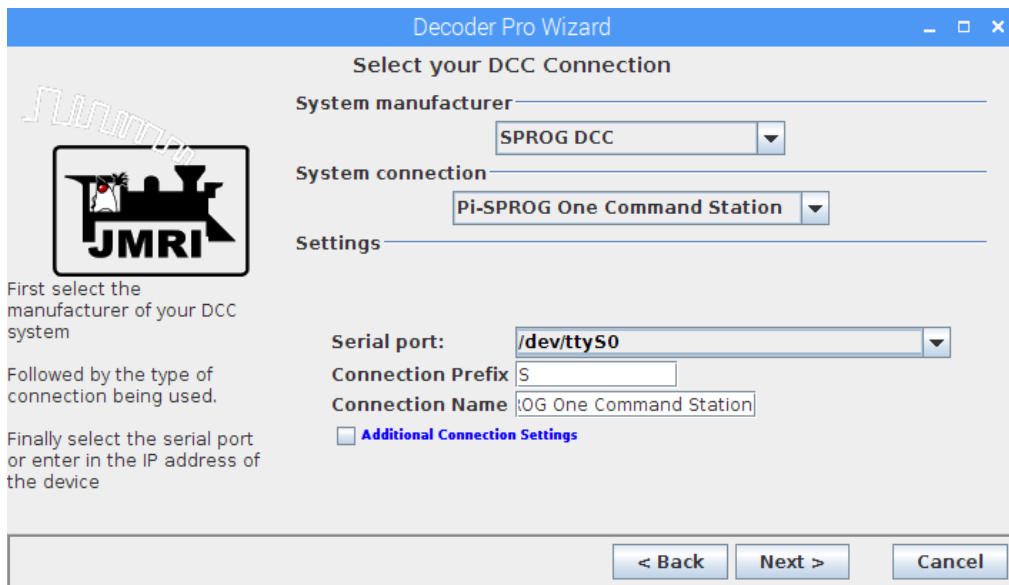
```
sudo nano /home/pi/Desktop/DecoderPro.desktop
```

```
[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;
```

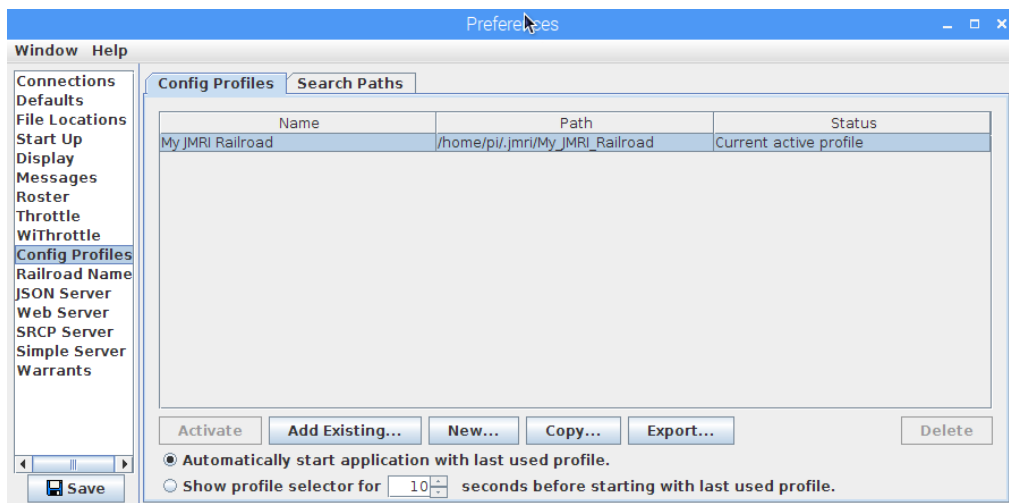
Save the file and quit the editor.

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/ttyS0 as the Serial port.

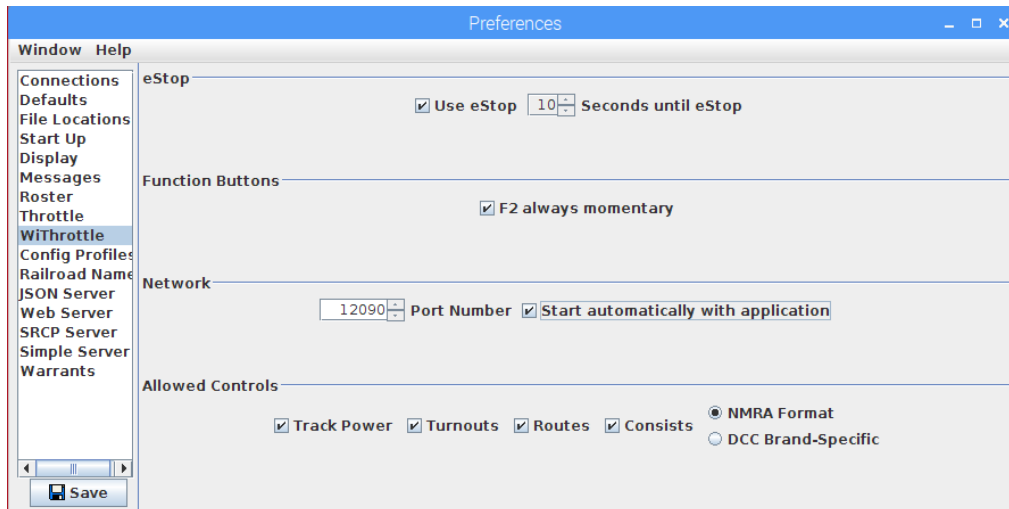


Once DecoderPro is running edit the preferences, config profiles and select used last profile.



Start Withrottle server automatically

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



Save the preferences and restart DecoderPro. The WiThrottle server should start up.

Exit DecoderPro.

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.

WiFi Access Point (AP) Setup

For reference, these instructions in this section are based on <https://frillip.com/using-your-raspberry-pi-3-as-a-wifi-access-point-with-hostapd/> with our own customization.

Download and install DNS and AP software packages:

```
sudo apt-get install dnsmasq hostapd
```

Answer yes if prompted to continue.

Apply Static IP Address to the R-Pi

Edit the interface settings to set a static IP address for your R-Pi wireless network. First, prevent dhcpd managing the WiFi network. Add a line to then end of the configuration file:

```
sudo nano /etc/dhcpd.conf
```

```
denyinterfaces wlan0
```

Write the file and quit the editor.

Apply the static IP address:

```
sudo nano /etc/network/interfaces
```

Edit the wlan0 section so that it looks like:

```
allow-hotplug wlan0  
iface wlan0 inet static  
post-up iw dev wlan0 set power_save off  
address 192.168.6.1  
netmask 255.255.255.0  
network 192.168.6.0  
broadcast 192.168.6.255  
# wpa-conf /etc/wpa_supplicant/wpa_supplicant.conf
```

Write the file and quit the editor.

This will also disable the WiFi power saving which can be too aggressive causing connections to be lost to wireless throttles.

Restart dhcpcd

```
sudo service dhcpcd restart
```

Reload the configuration for wlan0

```
sudo ifdown wlan0; sudo ifup wlan0
```

Configure the AP

Configure hostapd by creating a configuration file with your editor. The ssid can be the same as the hostname setup above.

Note the driver name is "en-el-eight-zero-two-one-one" .

You may wish to use a different wireless channel depending on which channels are less congested in your situation.

You may choose your own passphrase (password), but be sure to remember it.

```
sudo nano /etc/hostapd/hostapd.conf
```

```
interface=wlan0  
driver=nl80211  
ssid=sprog-pi3  
hw_mode=g  
channel=6
```

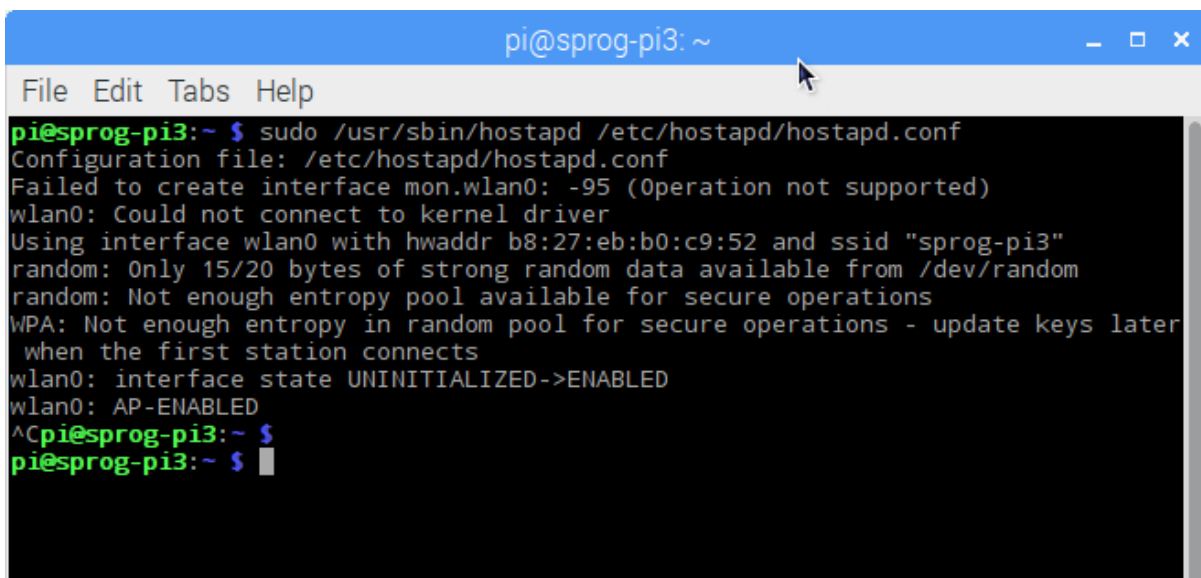
```
ieee80211n=1
wmm_enabled=1
ht_capab=[HT40][SHORT-GI-20][DSSS_CCK-40]
macaddr_acl=0
auth_algs=3
ignore_broadcast_ssid=0
wpa=3
wpa_key_mgmt=WPA-PSK
wpa_passphrase=pi-sprog
wpa_pairwise=TKIP
rsn_pairwise=CCMP
eap_reauth_period=360000000
```

Write the file and quit the editor.

Test hostapd

```
sudo /usr/sbin/hostapd /etc/hostapd/hostapd.conf
```

The error relating to mon.wlan0 can be ignored. You should be able to see (but not connect to the new WiFi network “sprog-pi3”, or whatever SSID you used, above.



```
pi@sprog-pi3: ~
File Edit Tabs Help
pi@sprog-pi3:~ $ sudo /usr/sbin/hostapd /etc/hostapd/hostapd.conf
Configuration file: /etc/hostapd/hostapd.conf
Failed to create interface mon.wlan0: -95 (Operation not supported)
wlan0: Could not connect to kernel driver
Using interface wlan0 with hwaddr b8:27:eb:b0:c9:52 and ssid "sprog-pi3"
random: Only 15/20 bytes of strong random data available from /dev/random
random: Not enough entropy pool available for secure operations
WPA: Not enough entropy in random pool for secure operations - update keys later
when the first station connects
wlan0: interface state UNINITIALIZED->ENABLED
wlan0: AP-ENABLED
^Cpi@sprog-pi3:~ $
pi@sprog-pi3:~ $
```

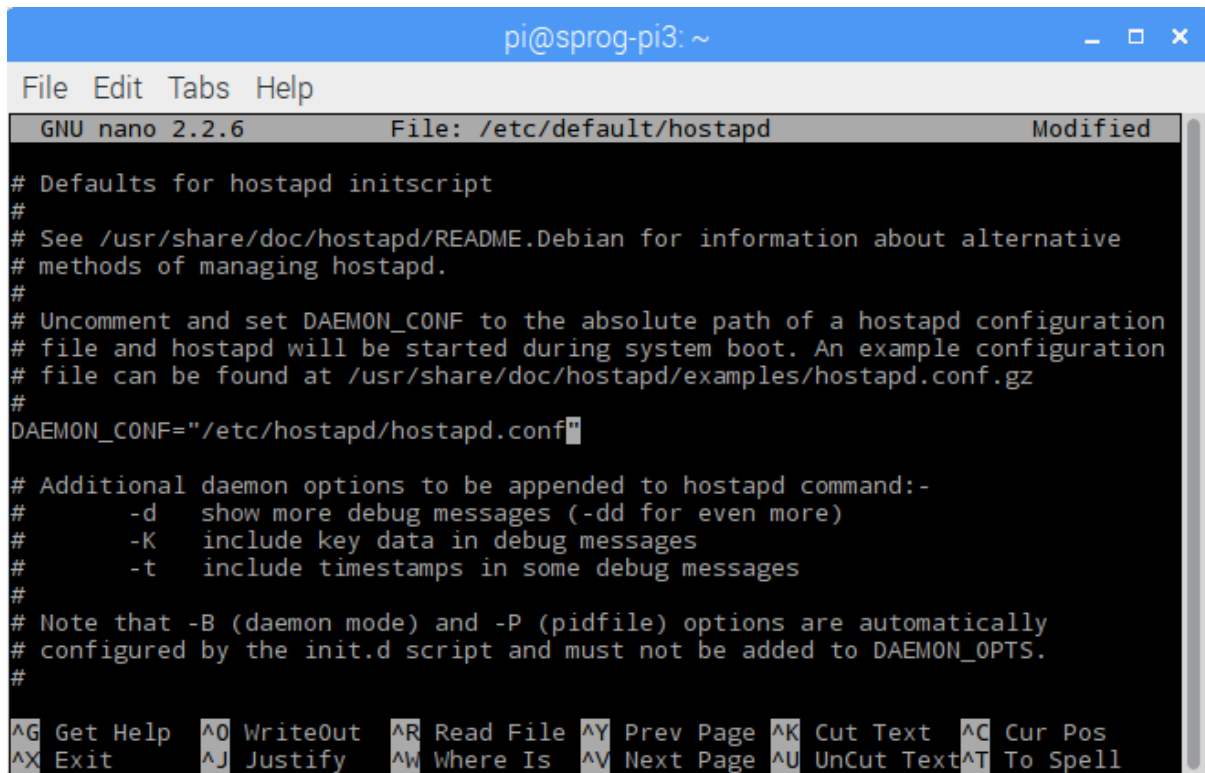
Type Ctrl-C to exit.

Next, tell hostapd where to look for the config file when it starts up on boot;

```
sudo nano /etc/default/hostapd
```

Uncomment (remove the leading # character) the DAEMON_CONF and enter the path to .conf file you just created.

```
DAEMON_CONF="/etc/hostapd/hostapd.conf"
```



```
pi@sprog-pi3: ~
File Edit Tabs Help
GNU nano 2.2.6 File: /etc/default/hostapd Modified
# Defaults for hostapd initscript
#
# See /usr/share/doc/hostapd/README.Debian for information about alternative
# methods of managing hostapd.
#
# Uncomment and set DAEMON_CONF to the absolute path of a hostapd configuration
# file and hostapd will be started during system boot. An example configuration
# file can be found at /usr/share/doc/hostapd/examples/hostapd.conf.gz
#
DAEMON_CONF="/etc/hostapd/hostapd.conf"
#
# Additional daemon options to be appended to hostapd command:-
#   -d show more debug messages (-dd for even more)
#   -K include key data in debug messages
#   -t include timestamps in some debug messages
#
# Note that -B (daemon mode) and -P (pidfile) options are automatically
# configured by the init.d script and must not be added to DAEMON_OPTS.
#
^G Get Help ^O WriteOut ^R Read File ^Y Prev Page ^K Cut Text ^C Cur Pos
^X Exit ^J Justify ^W Where Is ^V Next Page ^U UnCut Text ^T To Spell
```

Save the file and quit the editor.

Configure dnsmasq

Save the old configuration.

```
sudo mv /etc/dnsmasq.conf /etc/dnsmasq.conf.orig
```

Create a new one by entering the following in your editor.

```
sudo nano /etc/dnsmasq.conf
```

```
interface=wlan0
domain-needed
dhcp-range=192.168.6.50,192.168.6.99,255.255.255.0,12h
dhcp-option=252,"n"
```

Save the file and quit the editor.

The dhcp-range should be in the same subnet as the static IP assigned to the R-Pi. In our example the R-Pi was assigned the static IP address 192.168.6.1. The dhcp-range example here is in the same 192.168.6.x subnet and will assign up to 50 IP addresses to WiFi connected devices in the range 192.168.6.50 - 192.168.6.99.

Start the services

```
sudo service hostapd start
```

```
sudo service dnsmasq start
```

Reboot your R-Pi.

You should now be able to connect to the sprog-pi3 network and be assigned an IP address. You may receive a notification that there is no internet connection.

WiThrottle or EngineDriver apps should now be able to connect to the WiThrottle server in JMRI.

You may also connect to the Pi from, e.g., a laptop or tablet by selecting the sprog-pi3 network in the wireless network setup. Login with the WPA passphrase, entered above, as you would when connecting to your own wireless router. Run CNV viewer, or similar, on your laptop or tablet to see the R-Pi desktop.

The setup presented here does not allow wireless devices connected to the R-Pi to access the wired connection. This is intended as a security measure. More advanced users can perform further configuration (not covered here) to allow forwarding.