Raspberry Pi 3 Model B and JMRI with WiFi Access Point for Pi-SPROG One and Pi-SPROG Nano

SPROG DCC July 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 to your own preference.

It assumed that the user is logged in as pi and the home directory is /home/pi, as is the case for the unmodified Raspbian Jesse image. Some steps will need to be modified if this is not the case.

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

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

Hardware

Raspberry Pi 3 model B

R-Pi power supply for above

MicroSD memory card (8Gb recommended)

Pi-SPROG One

Power supply for Pi-SPROG One

For one time setup you will need a HDMI monitor (or R-Pi LCD) and USB keyboard and mouse connected to the R-Pi. You can leave these connected for future operation, or install the TightVNC server for remote connections.

You will need a wired network connection to your router during setup and if you require network (e.g. internet) access whilst using the Pi as an Access Point.

Pre-requisites

Start with the latest Raspbian Jesse image (2016-05-27 at the time of writing) installed on an SD card and boot the R-Pi from it https://www.raspberrypi.org/documentation/installation/installing-images/README.md

Connect your R-Pi to your router and ensure you have internet access

```
ping www.bbc.co.uk
```

Type control-C to quit ping.

Determine the IP address assigned to the R-Pi by your router (hover the mouse over the networking icon in the desktop menu bar). 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.

Ensure you have the latest updates. In a terminal enter:

```
sudo apt-get update
sudo apt-get upgrade
```

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

Reboot your R-Pi.

Install TightVNC Server (Optional)

Installing TightVNC on your R-Pi allows you to connect remotely to the R-Pi from a PC, etc. You can see, and interact with, the R-Pi desktop without requiring a monitor or LCD screen, keyboard or mouse to be connected. This connection requires the wired network connection to be connected to your router.

To use TightVNC you must also install a VNC viewer on your PC. We recommend the free VNC viewer from REALVNC https://www.realvnc.com/download/viewer/. Note: you are installing the viewer on your PC, do not download the R-Pi version!

Download and install TightVNC on your R-Pi, in a terminal:

```
sudo apt-get install tightvncserver
```

Answer yes when prompted to continue.

Run the server:

```
/usr/bin/tightvncserver
```

Set a password, we use "sprog-pi". Whatever you enter will be truncated to 8 characters. Enter the password again when prompted to verify. Answer no when prompted to enter a view-only password.

The password can be changed in the future using the 'vncpasswd' command.

Using your favoured editor, create /etc/systemd/system/tightvncserver.service with the following contects (do not type the line numbers, they are shown for reference only). You must use 'sudo' to create this file as root. Enter these lines exactly as shown, taking car over the capitalization of certain words.

sudo nano /etc/systemd/system/tightvncserver.service

```
[Unit]
Description=TightVNC remote desktop server
After=sshd.service
[Service]
Type=forking
PAMName=login
ExecStart=/usr/bin/tightvncserver -depth 24 -geometry 1280x800 :1
User=pi
[Install]
WantedBy=multi-user.target
```

Save the file and quit the editor.

Now change the ownership of the file you created, so that it is owned by root:

sudo chown root:root /etc/systemd/system/tightvncserver.service

Change the access permissions:

sudo chmod 755 /etc/systemd/system/tightvncserver.service

Reboot your R-Pi.

Test the installation by manually starting the service:

sudo systemctl start tightvncserver.service

Errors are usually due to not copying the above file contents exactly.

On your PC, start the VNC viewer and verify that you can connect to the R-Pi desktop. Enter the R-Pis IP address and port number 5901, e.g. 192.168.1.10::5901

Enable TightVNC to run automatically at startup:

sudo systemctl enable tightvncserver.service

We like to edit .vnc/xstartup to get a nice cursor.

sudo nano /home/pi/.vnc/xstartup

Change the xsetroot parameters to

xsetroot -solid grey -cursor_name left_ptr

Save the file and quit the editor.

Reboot your R-Pi and you should be able to connect to it with the VNC viewer.

You can now disconnect the monitor, mouse and keyboard.

From now on the instructions are mostly the same whether you use VNC Viewer or a physiocally connected display, etc..

R-Pi Configuration

Run the R-Pi configuration tool

```
sudo raspi-config
```

Hostname

We like to change the R-Pi hostname to “sprog-pi3”

Select “Advanced Options”

Select “Hostname” and type the new name

Select OK

File System

We like to expand the root file system to use all of the available space on the SD card.

Select “Expand file system”

Disable Console Desktop GUI

We prefer to disable the console desktop GUI as we do not use HDMI monitor, LCD screen, mouse nor keyboard attached to the R-Pi. Doing this ensures DecoderPro will be able to start up on a desktop viewed remotely using VNC

If you are not using VNC to connect to your R-Pi, or you wish to retain the console desktop GUI, then you can skip this step, but DecoderPro will fail to open on a VNC desktop.

Select “Boot options”

Select “Console” or “Console Autologin”

Select “OK”

You can start a console GUI at any time by logging in and running 'startx'.

When you have completed the configuration, select “Finish” and reboot.

UART setup

Swap UARTs so that serial0 appears on GPIO instead of being used for BlueTooth. Add the overlay to

```
sudo vi /boot/config.txt
```

```
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.

Check that the UARTs are mapped correctly:

```
ls –l /dev/ser*
```

The output should be something like

lrwxrwxrwx 1 root root 5 Jul 13 13:34 /dev/serial0 -> ttyS0
lrwxrwxrwx 1 root root 5 Jul 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 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-SPROIG 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. 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.

Save the prefences.

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.

Run DecoderPro at Startup

In a terminal open the session autostart

sudo vi /home/pi/.config/lxsession/LXDE-pi/autostart

Add the following line

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

Save the file and quit the editor.

Note: If you have left the R-Pi console desktop enabled then DecoderPro will start on the console desktop. If you then connect remotely buy VNC then DecoderPro will attempot to start again but the connection will fail as the serial port is already in use.

WiFi Access Point (AP) Setup

The 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 dhcpcd managing the WiFi network. Add a line to then end of the configuration file:

```
sudo vi /etc/dhcpcd.conf

      denyinterfaces wlan0
```

Write the file and quit the editor.

Apply the static IP address:

```
sudo vi /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
```

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 vi /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
```

You may get some errors but wlan0 should be enabled and you should be able to see (but not connect to the new WiFi network "sprog-pi3", or whatever SSID you used, above.

Tell hostapd where to look for the config file when it starts up on boot;

```
sudo vi /etc/default/hostapd
```

Uncomment the DAEMON_CONF and enter the path to .conf file you just created.

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

Configure dnsmasq

Save the old configuration file and create new one

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

```
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"
```

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 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.