https://www.model-railway-signalling.co.uk/

DCC Signalling System – Remote Sensor Node

Version 5.0 – June 2025

This guide is intended to walk you through the initial setup of your DCC Signalling System *Remote Sensor Node* and provide some top tips for the getting the best out of the system.

The *Remote Sensor Node* runs exactly the same software as the DCC signalling System, with a default configuration to publish GPIO sensor events to the rest of your signalling network.

In theory, you could run a full layout file (representing another signal box or signalling area) on the *Remote Sensor Node*, although performance will degrade as layout complexity increases.

For further information on integrating and using remote GPIO sensors with your layout, refer to the separate Application Quickstart Guide and Application Networking Guide, both available for download from https://www.model-railway-signalling.co.uk/.

Table of Contents

Unpacking

The *Remote Sensor Node* is supplied as a pre-assembled and pre-configured unit comprising:

- A Raspberry Pi-zero-W single-board computer which, despite its tiny size, provides a “windows-like” user experience, albeit without the performance of the Raspberry Pi-4.
- A GPIO terminal breakout 'HAT' (Hardware Attached on Top), providing an easy means of connecting in external track sensors to the unit.
- The DCC Signalling application (installed and configured to open and run the default *Remote Sensor Node* layout configuration file at start-up, and pdf copies of all DCC Signalling System user guides (including this document).

The *Remote Sensor Node* is supplied with:

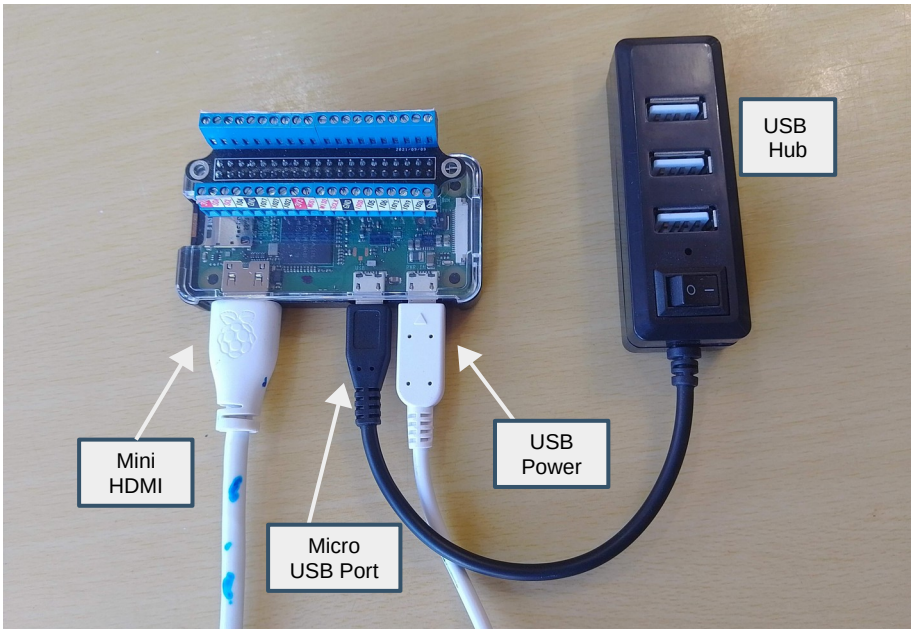
- A Raspberry-Pi micro-USB UK power supply.
- A mini-HDMI to HDMI lead for connecting to a monitor (for configuration).
- A USB mini-hub to enable connection of keyboard, mouse (for configuration).

Before proceeding, inspect all of the components carefully for damage (especially the power supply). If you suspect damage then contact DCC Model Railway Signalling immediately to arrange an exchange (see https://www.model-railway-signalling.co.uk/ for contact details).

Initial setup

For the initial setup of the *Remote Sensor Node* (specific to your layout) you will need a PC monitor (with a HDMI input port), a USB keyboard and a USB mouse.

1) First connect the USB mini-hub to the micro-USB port on the Raspberry-Pi.
2) The USB keyboard and mouse can then be connected into the hub, together with any other USB devices you might need (such as memory sticks for backing up the configuration).
3) Connect the HDMI lead into the mini-HDMI port and the other end to your monitor.
4) Connect the power supply to the micro-USB power supply socket.

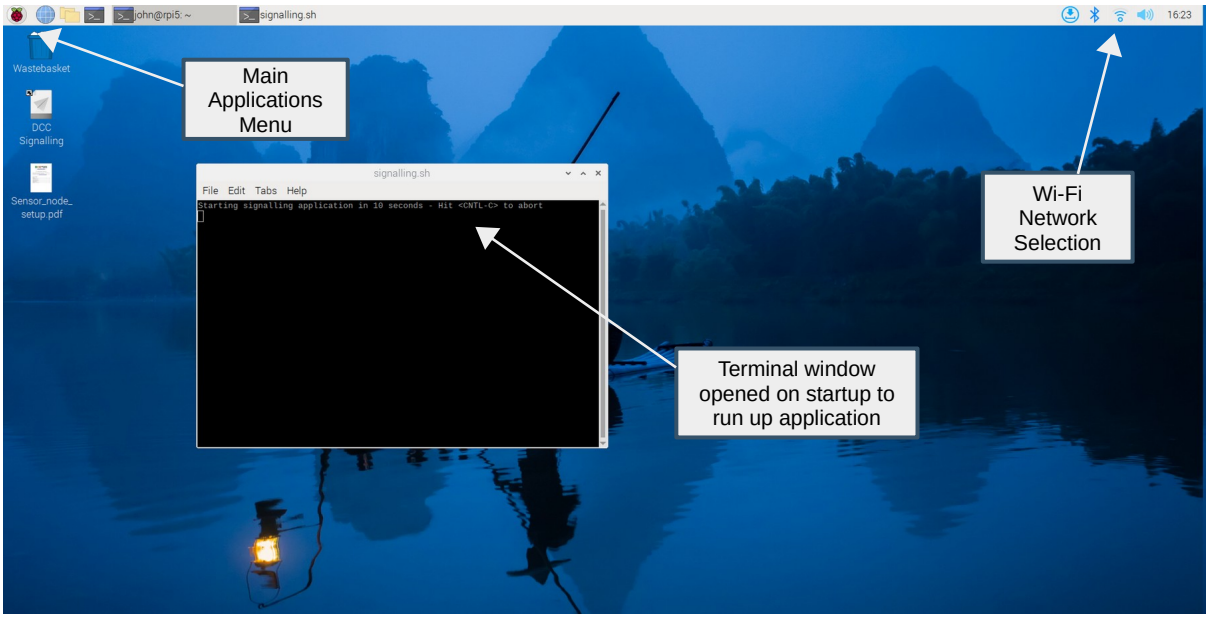


Only ever use the power supply that comes with the system (or another official Raspberry-Pi-zero micro-USB power supply). Also note that connections to the HDMI ports, the USB port and the Power Supply port are different from those on the DCC Signalling System, so always ensure you use the appropriate leads and power supply for each unit.

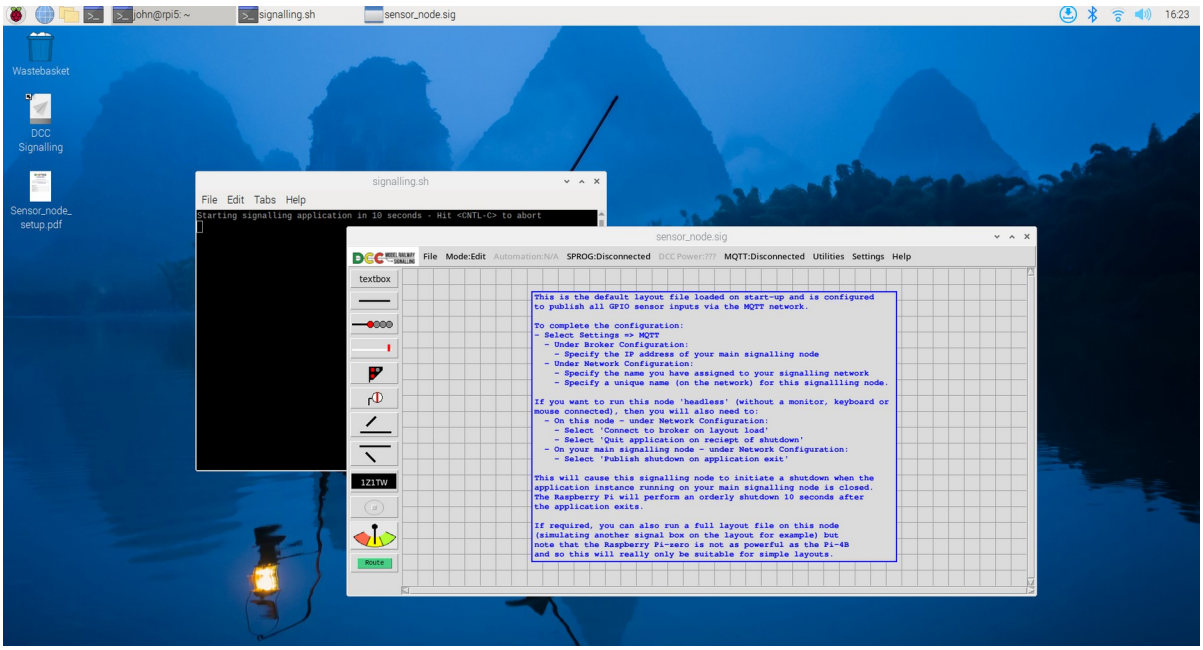
Powering up the system

Once everything is connected, apply power to the system and wait for it to boot into the desktop.

The *Remote Sensor Node* is configured to run-up the DCC Signalling application on start-up (with a 5 second delay to allow the application startup to be aborted if required). To abort the application startup click in the Terminal Window to select it and press **<CNTL-C>:**



If the start up sequence is not aborted , the application will open and load the default layout configuration file (‘/home/signalbox/Signalling/Files/sensor_node.sig’) :



Similarly, when you close the DCC Signalling application, the system will power down after a 5 second delay (with the same opportunity to abort the shutdown by pressing **<CNTL-C>**).

Connecting to your Wi-fi network

To connect to your home Wi-Fi network:

1. Click on the Network icon on the top right of the screen.
2. Select your Wi-Fi network from the list of available networks.
3. Enter your wireless password and click OK to connect.

Changing the default password

Remote Sensor Nodes are shipped with a default password of 'password1234' and are configured to boot-up into the desktop without requiring this to be entered, To change the default password:

1. Click on the “Raspberry Pi” icon in the top left hand corner of the screen to bring up the Main Applications Menu.
2. Select **Preferences** and then **Raspberry-Pi Configuration.**
3. On the System tab, select **Change Password**, enter and confirm the new password.

As the *Remote Sensor Node* relies on auto-login to open the application at start up, you should therefore never set ‘auto login’ (on the same configuration tab) to **Disabled** unless you always intend to run with a monitor, keyboard and mouse connected.

Powering off the system – Important!

Like all computers, you should never just remove power without shutting down as it may corrupt the underlying operating system (rendering the system unusable). To power-off the system:

1. Click on the “Raspberry Pi” icon in the top left hand corner of the screen to bring up the Main Applications Menu.
2. Select **Shutdown** from the drop-down list to bring up the Shutdown Menu.
3. Select **Shutdown** from the pop-up menu and wait for the system to shut down.
4. After a few seconds, the power can then be removed from the system.

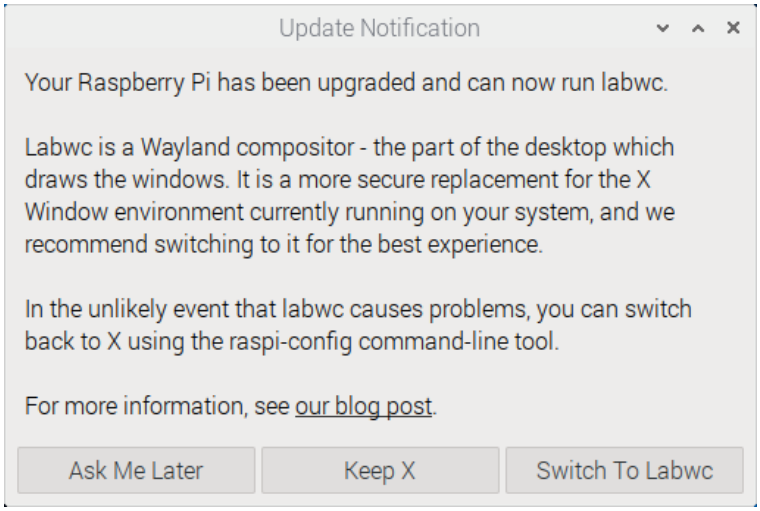
Note that the Remote Sensor Node can be configured to automatically shut down when you exit the DCC signalling application running on the main unit – this is covered later in the documentation. Once configured to do this, the Signalling node can be used entirely in ‘headless’ mode (without monitor, keyboard or mouse connected).

Updating the operating system

Like Windows (and other operating systems), the *Remote Sensor Node* should be regularly updated to keep it in the best condition but, unlike Windows, there is no automatic update facility and you therefore have to update the system manually. Do not fear, this is not complicated:

1. A ‘download’ icon will appear on the right hand side of the taskbar when system updates are available to download and install .
2. Click on the icon to either view the updates or download/install them.
3. Re-boot the system for the updates to take effect

Important – If you ever see the following message (or similar) pop up during the install process then always select ‘Keep X’. If you switch to Labwc then performance of the signalling application will be significantly compromised:



Updating the signalling application

The *Remote Sensor Node* runs exactly the same software as the DCC signalling System and you are always encouraged to use the latest version to benefit from any new features and bug fixes.

Release 5.2.0 and above of the application includes an application upgrade utility to check for and install the latest version of the software. This can be accessed from the Main Menubar (**Utilities => Application Upgrade**). For previous releases:

1. Check the latest published version at: https://pypi.org/project/model-railway-signals/.
2. Check the currently installed version by opening the application and selecting **Help** and then **About** from the main menubar at the top of the application window.
3. If there is a newer version available then you can upgrade by opening a Terminal window (double click on the Terminal icon top left corner of the screen next to the “Raspberry Pi” icon) and typing in the following command:

   - sudo pip install --upgrade --break-system-packages model-railway-signals

   Note the two hyphens in front of each command line argument.

In the unlikely event you ever need to ‘roll back’ the version then this can be achieved as follows:

1. sudo pip3 uninstall model-railway-signals
2. sudo pip3 install model-railway-signals==x.y.z

   where x.y.z is the application version to install (e.g. 4.9.0)

Backing up your files

Files on your system can be easily backed up (or copied between computers) using USB memory sticks. When a USB stick is plugged into the system, a pop-up window will appear asking if you want to open the device in a file manager window. Once open, files can then be copied as required.

Warning – one major difference to Windows is that you should never just unplug a USB stick as this may corrupt the device. Memory sticks should first be 'ejected' by clicking on the 'eject' icon (that will appear at the top of the screen whenever a USB stick is inserted). Once ejected, the USB stick can be safely unplugged.

Backing up your system

As with all computers, it is good practice to make regular backups of the entire system for disaster recovery purposes (so you can restore it in the unlikely event of fatal system crashes).

To back up your system we would recommend that you use 32GB Micro-SD cards (which are relatively inexpensive and widely available). In the event of a failure, the internal Micro-SD card (running the system) can then simply be swapped out for the backup Micro-SD card, to get you back up and running in the shortest possible time.

To perform a backup, you will also need a USB Micro-SD reader/writer (not supplied with the system but readily available at minimum cost).

To perform the backup:

1. Close all open applications.
2. Plug the backup Micro-SD card into the USB adapter and plug the USB adapter into one of the USB ports on the mini-hub supplied with the system.
3. Click on the “Raspberry Pi” icon in the top left hand corner of the screen to bring up the Main Applications Menu.
4. Select **Accessories** and then **SD Card Copier** to open the copier application.
5. For the 'copy from device', select the internal Micro-SD – this will normally appear at the top of the drop-down list as 'Y016U (/dev/mmcblk0)' or similar.
6. For the 'copy to device', select the new Micro-SD card.
7. Perform the backup – and then keep the backup in a safe place.

In the unlikely event you ever need to restore the system, contact DCC Model Railway Signalling (https://www.model-railway-signalling.co.uk/) for advice.

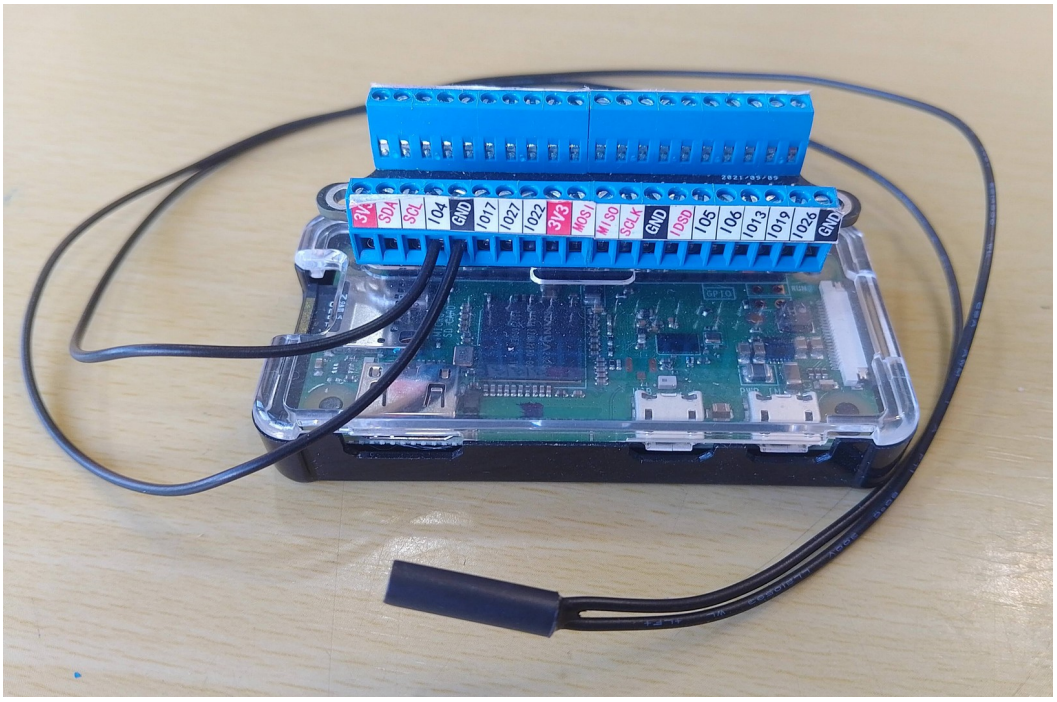
Running up the Signalling application

As previously mentioned, the *Remote Sensor Node* is configured to open the DCC Signalling application on power-up and load the default layout configuration file (‘sensor_node.sig’) unless aborted by the user (during the 5 second period from boot to application start).

Alternatively, you can just double click on the desktop icon to open the application.

Connecting external track sensors

The Raspberry Pi-zero provides a number of General Purpose Input / Output (GPIO) pins which can be connected to external track sensors (similar to those on the main DCC Signalling System). These are made available via the screw terminals on top of the unit. The following picture shows a magnetic track sensor connected between GPIO-4 and 0V DC:



Track sensors that provide a floating (electrically separated from all other devices) 'normally-open' (momentarily closed when triggered) output are recommended as these can be connected directly between the appropriate GPIO input terminal and one of the 0V DC terminals.

Other types of Track Sensors should <u>never</u> be connected to the unit or this could (almost certainly will) damage the system irreparably. In these cases, external opto-isolator circuits are recommended to protect the GPIO input pins.

Only a subset of the GPIO inputs are available for use by the signalling application (Open the application and select *Settings* and then *GPIO* to bring up a list of supported inputs). Never connect track sensors to the unsupported GPIO inputs or to the +5V terminals (red) or the +3.3V terminals (pink) or this could damage the system irreparably.

The connection into the screw terminals on top of the unit should be the <u>last</u> connection you make, and with the unit <u>powered off</u>. Always power off the unit and disconnect it from your layout for any subsequent layout wiring changes. As an cautionary tale, I managed to trash a couple of GPIO inputs whilst re-soldering a couple of connections under my layout. It turns out the soldering iron I was using wasn't particularly "ESD Safe" and must have introduced a spike large enough to cause damage to the Raspberry Pi – you have been warned!!!!

Further information and advice on the type, positioning and wiring of external track sensors, can be found in the main DCC Signalling System setup guide (included with the system or available to download from: https://www.model-railway-signalling.co.uk/downloads/)

Configuring the System

The *Remote Sensor Node* interfaces with the DCC Signalling System via your home WiFi network, so both the Remote Sensor Node and the DCC Signalling System will need to be connected to the network and then configured to be part of the same “Signalling Network”.

Full information can be found in the DCC Signalling System Networking Guide (included with the system or available to download from: https://www.model-railway-signalling.co.uk/downloads/)

The main steps are summarised below for convenience:

Establishing Network Connectivity

On the *Main DCC Signalling System*:

- Open the MQTT configuration window (**Settings => MQTT**) and select the **Network** tab.
- Check the Broker configuration is set to an Address of **127.0.0.1** and the Port is **1883** (this is the local broker installed on the DCC Signalling Network).
- For the Network Configuration, choose a **Network** name for your Signalling Network, and a unique **Node** name for the main DCC Signalling System
- **Apply** the configuration.
- Connect to the Broker, either by clicking the **Test Broker Connectivity** button or by clicking on the **MQTT** button on the Main Menubar and selecting **Connect.** If successful then the MQTT status (on the main menubar) should show as **Connected.**
- Find the local IP address of the main DCC Signalling system by selecting the **Status** tab of the MQTT configuration window. The IP address should be displayed with the ‘last seen’ time in green (if it doesn’t appear straight away, try clicking **Refresh Display**).

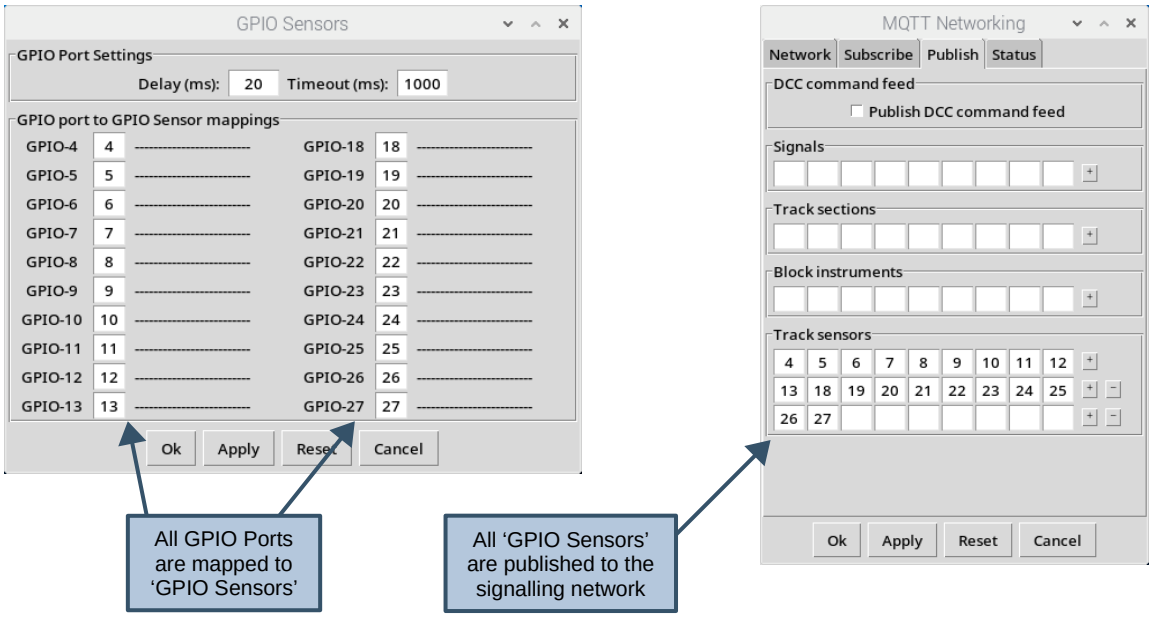
On the *Remote Sensor Node*:

- Open the MQTT configuration window (**Settings => MQTT**) and select the **Network** tab.
- Under Broker Configuration, change the set the Address to the Address of the main DCC Signalling node (identified in the previous steps). The Port should be set to **1883.**
- For the Network Configuration, enter the same **Network** name that you set for the DCC Signalling System, but then choose a unique **Node** name for the *Remote Sensor Node*.
- **Apply** the configuration.
- Connect to the Broker, either by clicking the **Test Broker Connectivity** button or by clicking on the **MQTT** button on the Main Menubar and selecting **Connect.** If successful then the MQTT status (on the main menubar) should show as **Connected.**

If successful then the **Status** tab of the MQTT configuration window on both systems should show both ‘Nodes’, with the ‘last seen’ time in green (try clicking **Refresh Display**).

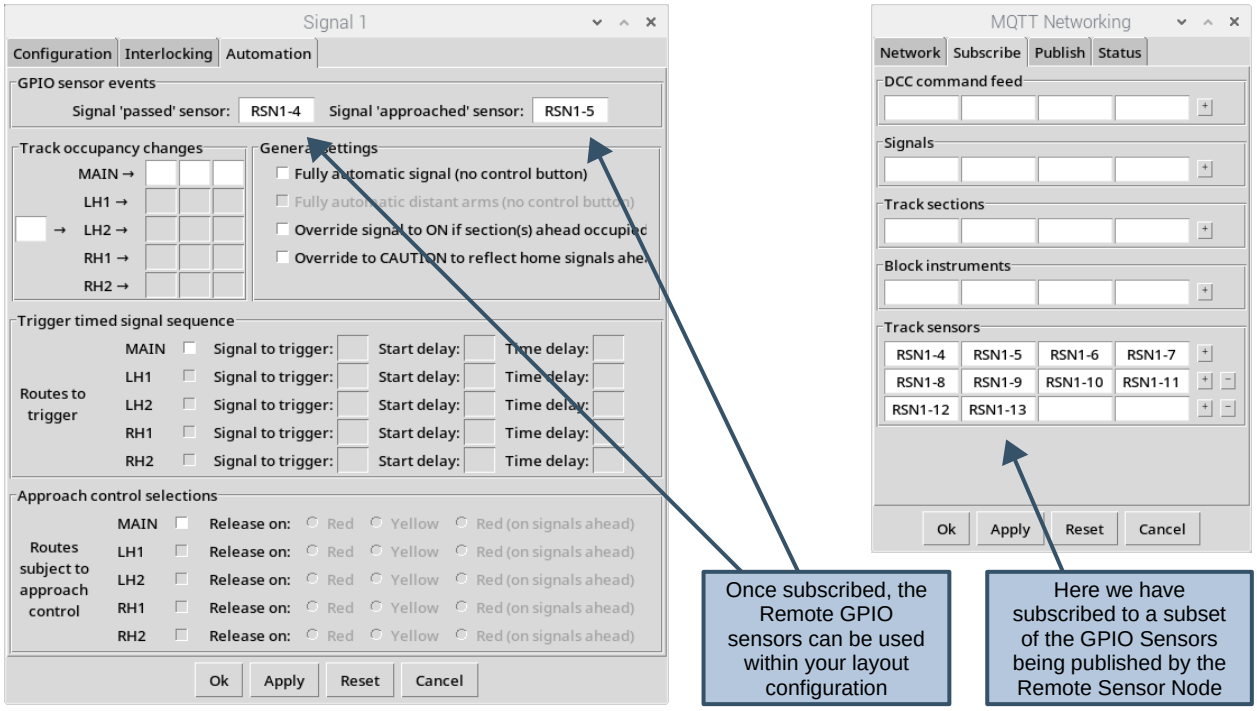
Using Remote GPIO Sensors

The default layout file for the *Remote Sensor Node* (loaded on power-up/application startup) assigns ‘GPIO sensors’ to each of the available GPIO Ports and then ‘publishes’ all events to the network:



To use these ‘Remote GPIO Sensors’ in your layout configuration, you first need to ‘subscribe’ to them (as ‘**nodename-itemid’**) in your layout configuration (running on the main DCC Signalling System). They can then be used to trigger signal or track sensors events on your schematic

The following example configuration assumes you have assigned a Node name of ‘**RSN1**’ for your *Remote Sensor Node* (in its Network Configuration).



Configuring Headless Operation

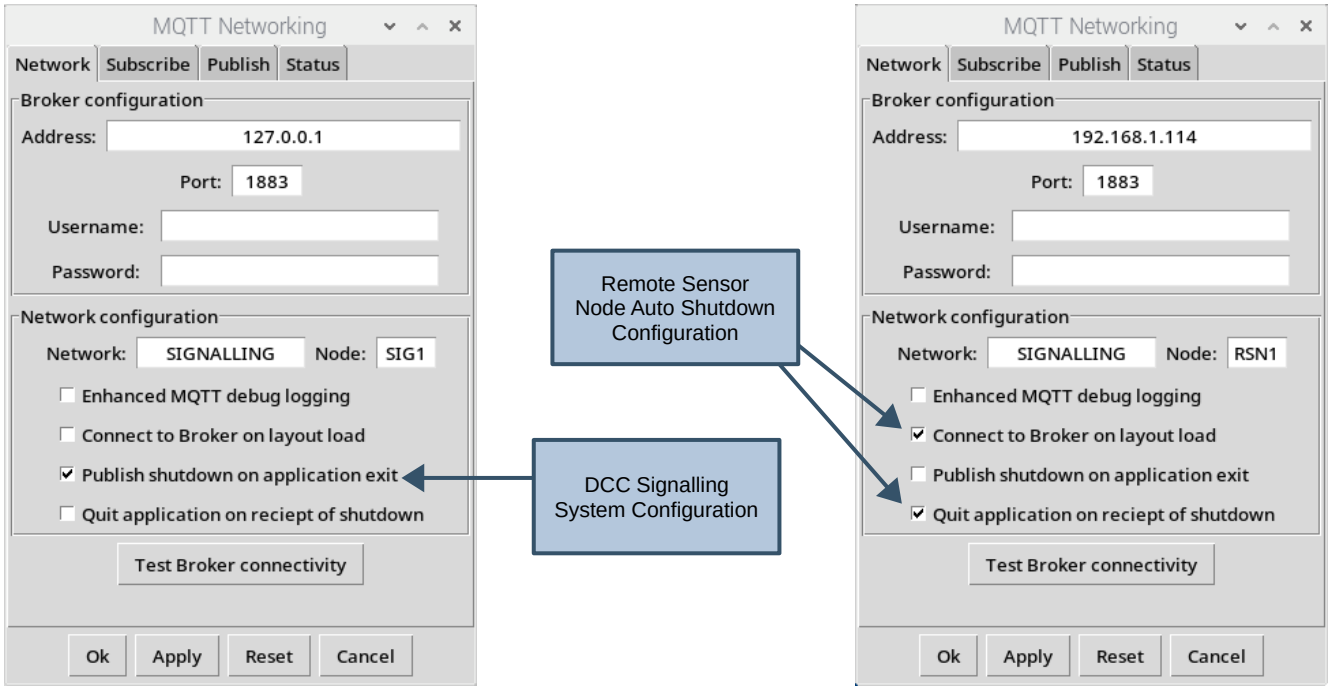
As mentioned in previous sections, the *Remote Sensor Node* will automatically start when power is applied and shutdown when the application exits. To complete the configuration for fully automatic operation we need to connect to the MQTT broker on layout load and set the application to shut down when it receives a shutdown signal from the main DCC Signalling System.

On the *Remote Sensor Node*:

- Open the MQTT configuration window (**Settings => MQTT**) and select the **Network** tab.
- Under Network Configuration select **Connect to broker on layout load**
- Under Network Configuration select **Quit application on receipt of shutdown**. This will cause the *Remote Sensor Node* to shut down on receipt of a network shutdown message.
- **Apply** the changes

On the main *DCC Signalling System*:

- Open the MQTT configuration window (**Settings => MQTT**) and select the **Network** tab.
- Under Network Configuration select **Publish shutdown on application exit**. This will cause the DCC Signalling System to issue the network shutdown message.
- **Apply** the changes



The configuration should then be tested to ensure the *Remote Sensor Node* starts-up and shuts-down correctly before disconnecting the keyboard, mouse and monitor to run it its ‘headless’ mode.

Useful Links

1. https://www.model-railway-signalling.co.uk/ - The homepage for the DCC Signalling System providing details of how you can about obtain further information and support.
2. https://pypi.org/project/model-railway-signals/ - the Python Package Index page for the signalling application, providing details of the latest published version.
3. https://signalbox.org/ - Comprehensive information on signal types and the 'Block System' and a vast library of signal box diagrams for you to draw inspiration.
4. https://en.wikipedia.org/wiki/UK_railway_signalling – A useful starting point for research to increase your knowledge of British railway signalling practice (its Wikipedia after all).
5. http://www.railway-technical.com/signalling/ - A section of the Railway Technical Website covering signalling. There are many great resources on these pages including:
    1. http://www.railway-technical.com/signalling/infopaper-6-basic-railway.pdf – A useful paper (downloadable pdf format) on basic railway signalling.
    2. http://www.railway-technical.com/signalling/british-signalling--what.pdf – A useful paper (downloadable PDF) on "What the driver sees".
6. https://www.sprog-dcc.co.uk/ - The SPROG DCC web site containing details of their other products (the DCC signalling system uses the Pi-SPROG3-V2 for its DCC interface).