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

DCC Signalling System Setup Guide

Version 11 – January 2026

This guide is intended to walk you through the initial setup of your DCC Signalling System, and provide some top tips for the getting the best out of the Raspberry Pi.

For information on how to use the DCC Signalling Application, refer to the separate Application Quickstart Guide that comes packaged with the application (Select Help => Docs from the Main Menu Bar once the application has been opened).

Table of Contents

Unpacking

The *DCC Signalling System* is supplied as a pre-assembled and pre-configured unit comprising:

- A Raspberry Pi 4 single-board computer, providing a “windows-like” user experience (and versions of all the usual applications you would expect to find on a personal computer, such as web-browser, email, office-type applications etc).
- A Pi-SPROG3 DCC programmer controller, providing the interface to the DCC ‘accessory bus’ for control of points and signals out on the layout.
- A GPIO terminal breakout 'HAT' (Hardware Attached on Top) providing an easy means of connecting in external track sensors to enable train detection.
- The pre-installed Model Railway Signalling application, which comes packaged with comprehensive user documentation and a number of example layout files.

The *DCC Signalling System* is supplied with:

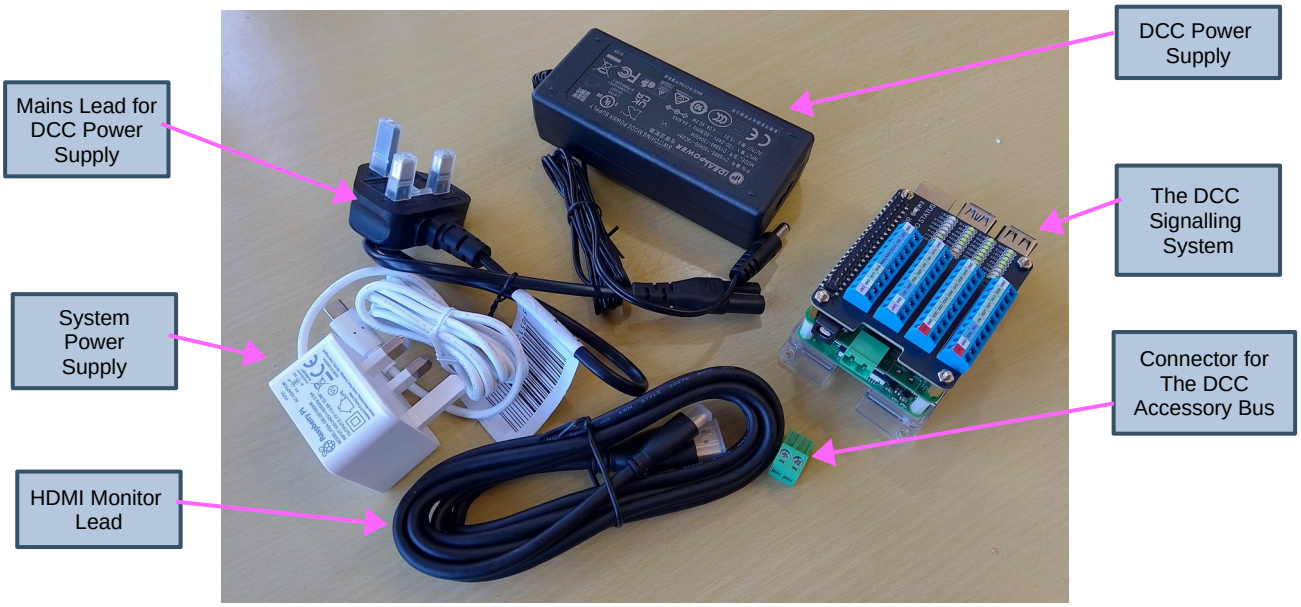
- A Raspberry-Pi USB-C UK power supply
- A DCC power supply (for powering the DCC accessory bus)
- A micro-HDMI to HDMI lead for connecting to your monitor
- A screw terminal connector for the DCC 'accessory bus” output

DCC Signalling System Version 1

Version 1 of the system is packaged in a clear plastic case and is supplied with the following items:

DCC Signalling System Version 2

Version 2 of the system comes without a case, but is instead fixed to a mounting plate to provide more options for fixing to your layout. It is supplied with the following items:



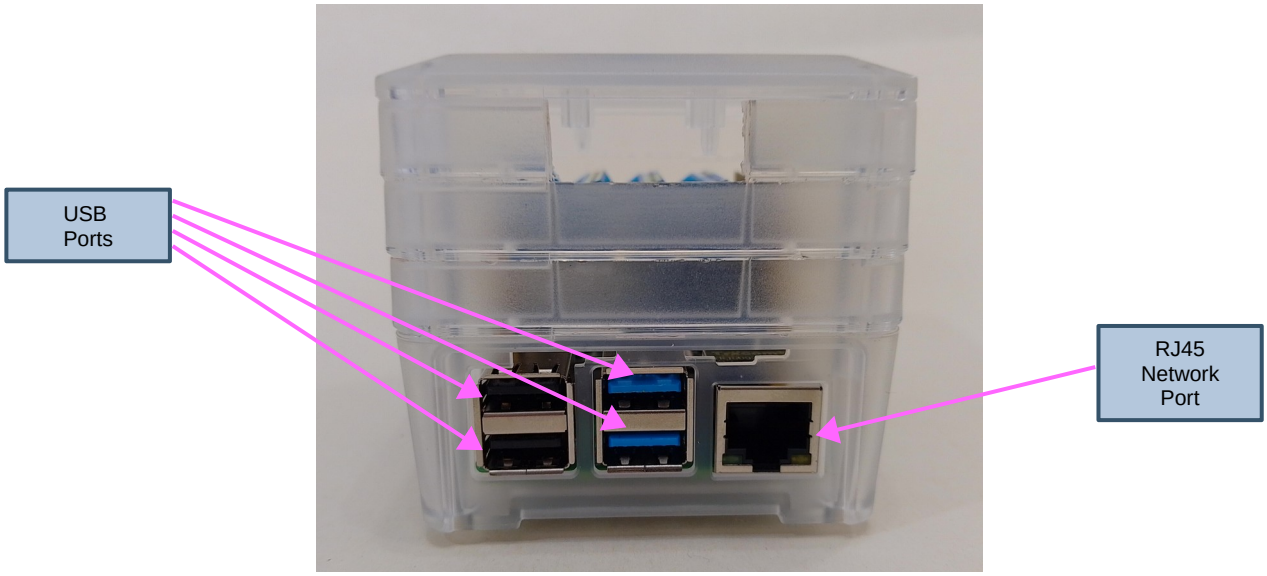
If required, Version 2 of the DCC Signalling System can be supplied, packaged in a case (similar to the Version 1 system), but additional lead times and charges may apply – Please enquire.

Before proceeding, inspect all of the components carefully for damage (especially the power supplies). 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

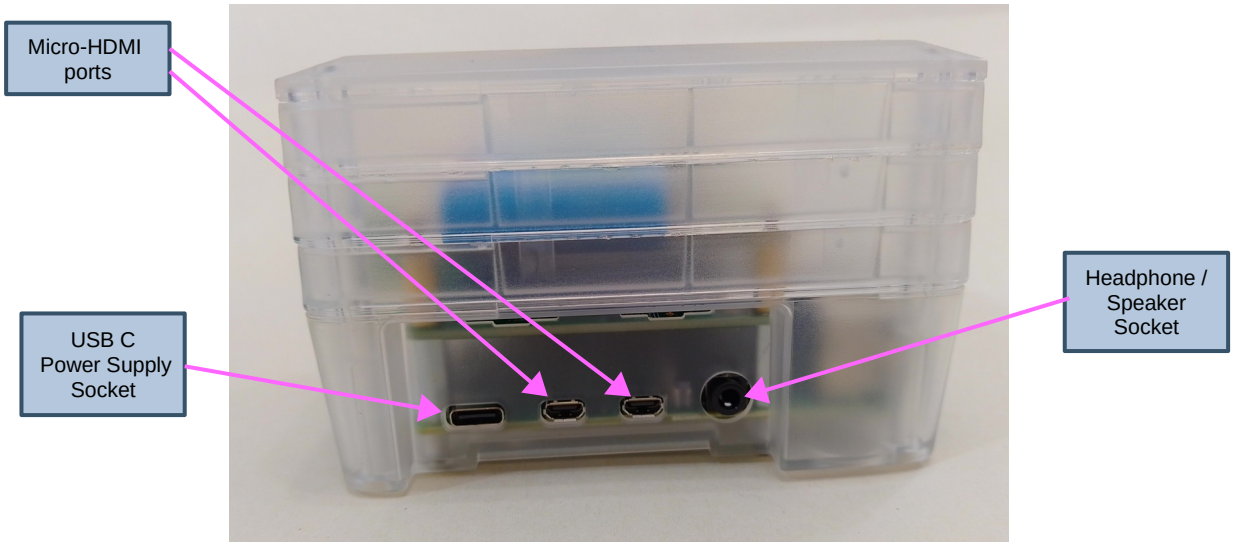
For the initial setup you will need a PC monitor (with a HDMI input port), a USB keyboard and a USB mouse (connecting to the DCC bus and track sensors is covered in later sections). Note that the layout of connectors is identical for the Version 2 system.

1) Connect your keyboard and mouse to the USB ports (there are 4 USB ports in total – any of which can be used for the keyboard and mouse).



2) Connect the power supply to the USB power supply socket

3) Connect the HDMI lead into the left hand Micro HDMI port and connect the other end to your monitor (the right hand socket can be used for driving a second monitor if required, but you will need another Micro-HDMI to HDMI lead for this).



Only ever use the power supply that comes with the system (or another official Raspberry-Pi USB-C power supply) – use of non compatible power supplies may damage the system.

Powering up the system

Once everything is connected, apply power to the system and wait for it to boot up into the desktop. A desktop shortcut is provided for the signalling application.

Note that your desktop wallpaper may be different.

Connecting to your home WiFi 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

All systems are shipped with a default password of 'password1234' but 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.
4. To prompt for the password at boot time set the Auto login to **Disabled**.

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:

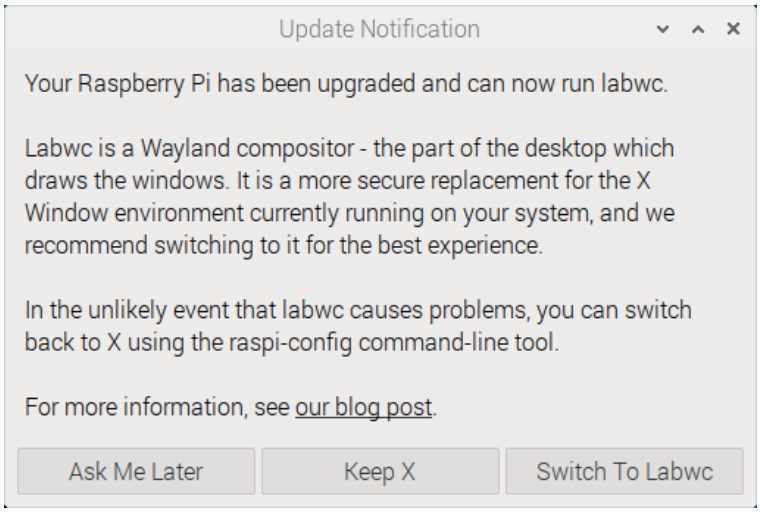
1. Click on the “Raspberry Pi” icon in the top left hand corner of the screen to bring up the Main Applications Menu.
2. Select **Shutdown** from the drop-down list to bring up the Shutdown Menu.
3. Select **Shutdown** from the pop-up menu and wait for the system to shut down.
4. After a few seconds,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

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 signalling application is subject to ongoing development, 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**).

If upgrading from Release 5.1.0 or below you should upgrade as follows:

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 pip3 install --upgrade model-railway-signals
4. If you get the error message “error: externally-managed-environment“ then try the following:
    - sudo pip3 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 files on 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, t 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 (so you can connect the Micro-SD card to any one of the USB ports). 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.
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

Just double click on the desktop icon to open the application.

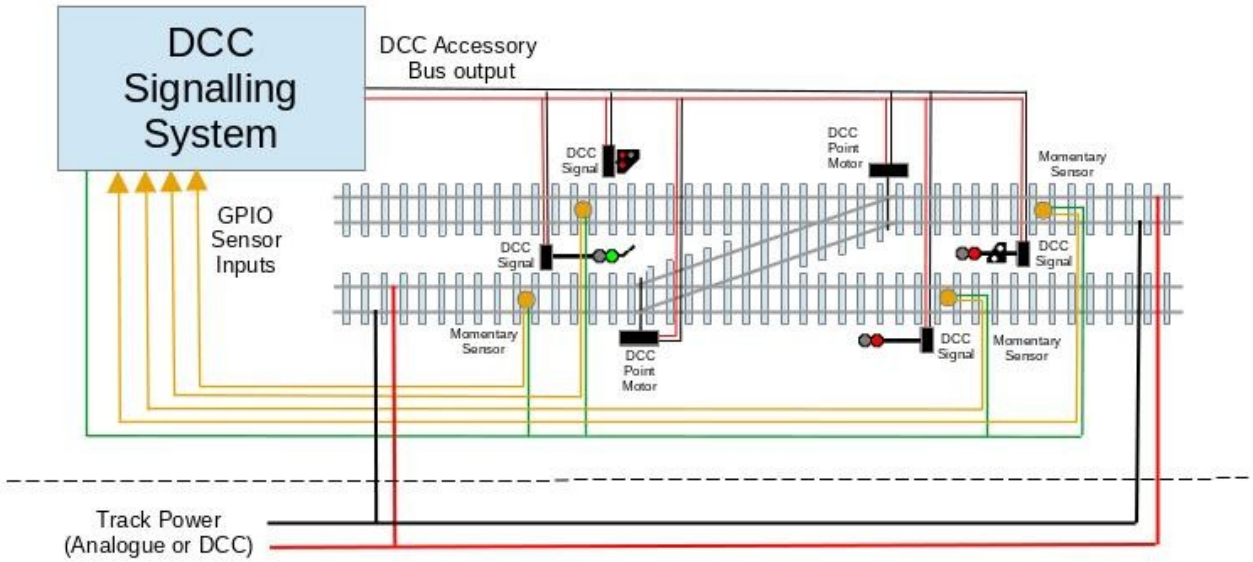
For more information on using the application, refer to the User Documentation packaged with the application (Select Help => Docs from the Main Menu Bar):

- Application Quickstart Guide - Provides an introduction to the Model Railway Signalling Application and the art of the possible in terms of the signalling configurations that can be achieved by working through a number of example layout configurations.
- Networking Guide – How to use the application’s built-in networking features to link systems together over your home WiFi network (e.g. if adding additional *Signalling Nodes* or *Remote Sensor Nodes*).

Connecting to your Layout

The DCC Signalling System uses a DCC 'accessory bus' for control of the signals and points out on the layout. This should be routed around the layout as required and **only** connected to the DCC-enabled signals and points on your layout (and/or appropriate DCC accessory decoders).

If train tracking and signal automation is required then momentary track sensors can be also be positioned on your layout (by each signal) and wired back to the General Purpose Input/Output terminals on top of the unit.

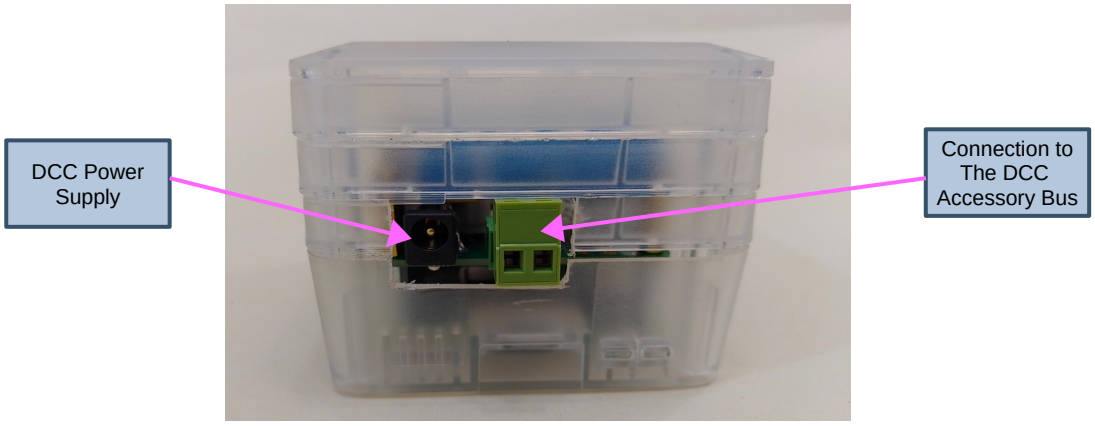


Connecting to the DCC accessory bus

The green connector, contains two screw terminals for connection to the DCC accessory bus. The other connection is for the DCC power supply.

Warning – Never connect the system to the DCC track bus (if you are already using DCC on your layout). The DCC accessory bus must be kept electrically isolated from the DCC track control bus, so the DCC Signalling system does not interfere with your command station.

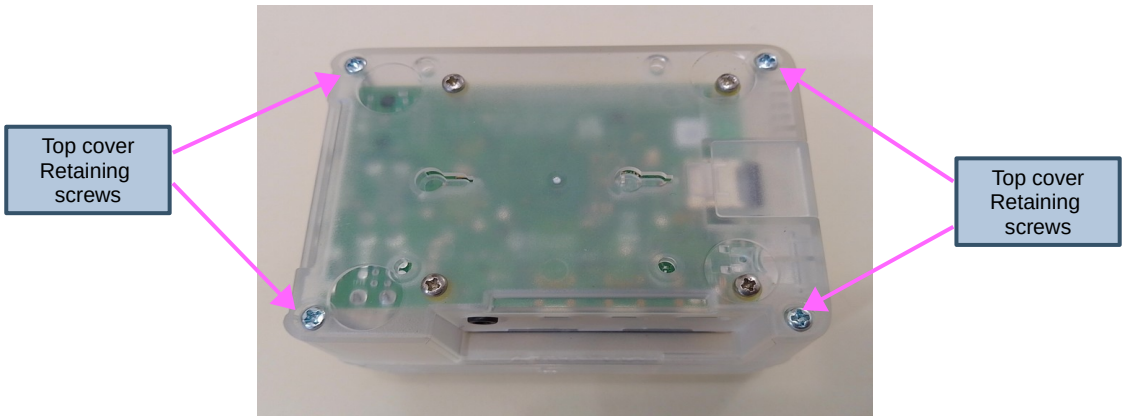
Warning - Only ever use the DCC power supply that comes with the system – use of non compatible power supplies may damage the system.



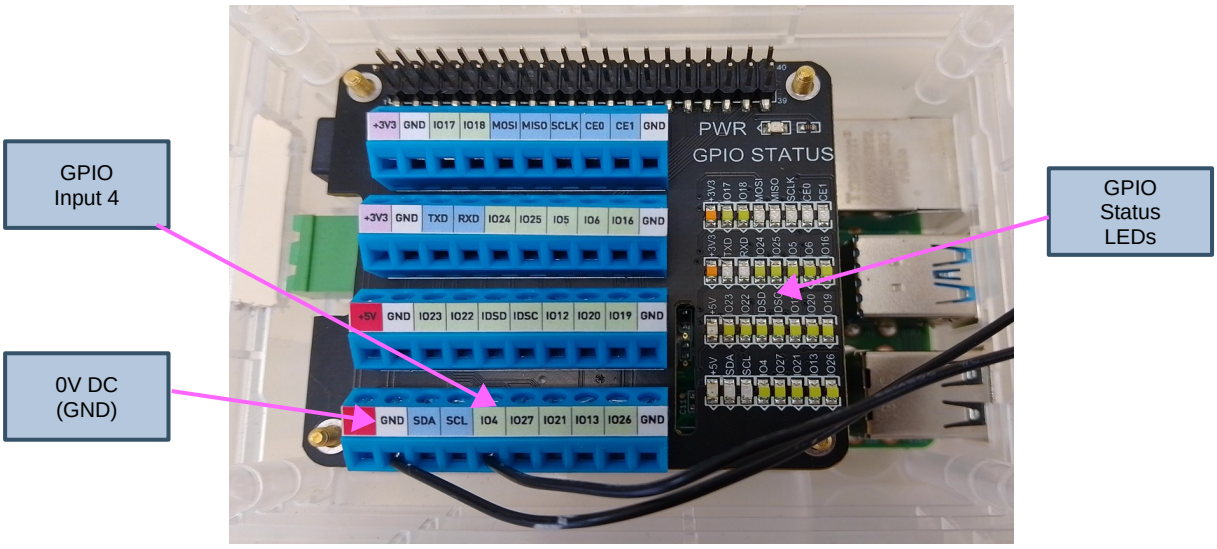
Connecting external track sensors

The Raspberry Pi supports a number of General Purpose Input / Output (GPIO) pins which can be connected to external track sensors, enabling the signalling application to provide a mimic diagram of track occupancy and to support automation of your signalling scheme.

To access the GPIO screw terminals on the Version 1 system, the top cover must first be removed by removing the 4 retaining screws on the base of the unit.

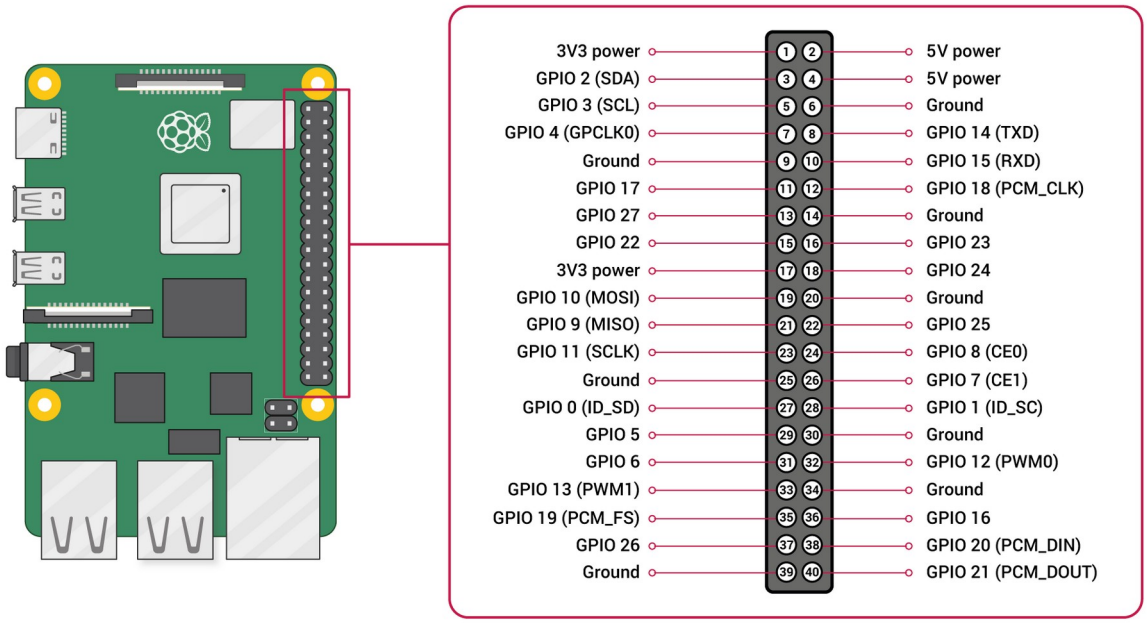


To trigger an event, the appropriate GPIO input pin needs to be momentarily connected to one of the 0V supply pins. These are all made available via screw terminals on the top of the unit.



Note that only a subset of GPIO ports are available for use by the application (open the application and select *Settings* and then *GPIO* to bring up a list of supported inputs). Also, not all available GPIO ports are explicitly identified on the GPIO header as some ports can support alternative functions (if the Raspberry Pi is configured to use these alternative functions).

To identify these pins, please refer to the diagram overleaf, which provides a translation between the basic GPIO port identifiers and the ‘alternative’ functions – for example, the CE0 and CE1 pins on the screw terminal header map to GPIO 8 and GPIO 7 respectively.



Type and positioning of external track sensors

The Signalling Application depends on track sensors that will generate a momentary event as trains pass a fixed point on the layout (rather than track sensors which will detect when a train is occupying a particular section of track).

The track sensors themselves should be positioned by each signal to trigger events when a train passes the signal. If you plan to use “approach control” for certain signals (refer to the application quickstart guide for further information) then additional sensors should be positioned such that they will trigger events as the train approaches the signal.

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 never 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.

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 last connection you make, and with the unit powered off. 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 some intermediate sensor connections. 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 – you have been warned!!!!

Wiring external track sensors

If you are planning to use floating, ‘normally-open’ track sensors connected directly into the unit, you should note that the GPIO inputs are low voltage (signal level) and susceptible to electrical interference which could result in false triggering. The maximum practical wiring distance for track sensors (without additional circuitry) will therefore depend on a number of factors specific to your layout and its environment (location of your layout, wiring runs, nearby equipment etc).

On my loft layout I’ve achieved sensor wiring runs of up to 5 meters, with the sensor wiring routed alongside the (analogue) track power wiring and alongside the 12VDC wiring for the remaining few Cobalt Analogue point motors I have yet to upgrade to DCC. This certainly isn’t best practice, but nevertheless, the DCC Concepts magnetic sensors I use on my layout seem to work reliably with a GPIO trigger period of 20ms (although if I set it to less than 10ms then I do see the occasional false trigger). Disclaimer – your situation may be better or worse.

When planning your sensor wiring, to mitigate the risk of interference I would recommend:

- Keep the track sensor wiring runs physically separated from other layout wiring runs (such as track power, whether analogue or digital) and any mains wiring that may be nearby
- Keep the track sensor wiring runs physically separated from potentially “noisy” equipment such as axillary power supplies, solenoid point motors, capacitor discharge units etc.

And if you do hit problems (false triggering), you could try:

- Increasing the ‘GPIO trigger delay’ parameter (Settings => GPIO)
- Re-wire the track sensors to switch 12VDC (or similar) and connect into the system via opto-isolator circuits (these are readily available from ebay and other suppliers).
- Contact us at DCC Model Railway Signalling and we’ll try to help

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).