



Broadcast Radio Ltd

# SMARTSIGN LITE

User Manual v1.0.8.0

## CONTENTS

Introduction .....	3
What's In the Box.....	3
Getting Started.....	3
Configuration .....	4
Application Settings .....	5
Adding Tiles .....	6
Analogue Clock .....	7
Digital Clock .....	7
Date and Time.....	8
Images.....	8
Text .....	8
Smart Display .....	10
Configuring SmartSign Lite to receive External Triggers .....	12
Setting Up SmartSign To Work With The Broadcast Radio Hardware Service .....	13
Installing & Configuring The Hardware Service.....	14
Setting Up The First SmartSign .....	19
The Theory.....	19
Configuring The SmartSign In The Studio To Broadcast Its Hardware Status.....	20
Configuring Other SmartSigns To Subscribe To Your Virtual Hardware .....	22
Shutting Down .....	24
Advanced Configuration with the Windows IoT Device Portal .....	25
Connecting to your device .....	25
Username and Password .....	26
Updating the App.....	27
Network Configuration .....	27
Adding Images for use in Tiles .....	28

## INTRODUCTION

Thanks for your interest in or purchase of SmartSign Lite from Broadcast Radio. SmartSign Lite is a simple studio tally and display system designed to run on a Raspberry Pi Mini Computer. It is available in many configurations on the [Broadcast Radio store](#). It can be used in your station reception to convey messages to visitors or in the studio to show the on-air team information they need.

More information including Tutorial Videos are available at [www.broadcastradio.com](http://www.broadcastradio.com).

## WHAT'S IN THE BOX

Dependent on the SmartSign package you purchased the box will contain:

- A Raspberry Pi pre-fitted in case OR a Raspberry Pi pre-fitted with a 7" Touchscreen and Case
- 5v Power Supply
- 16GB SD Card preinstalled with SmartSign Lite and fitted in the device
- Quick Connection Ribbon Cable for Hardware GPIO
- Separate monitor and HDMI Cable (if purchased)

## GETTING STARTED

1. If using a separate monitor, connect this using a standard HDMI cable.
2. If installing on your network, connect this now using a standard ethernet cable. (Note that SmartSign requires a network connection to synchronise time and talk to the Broadcast Radio Hardware Service)
3. To assist with configuring the SmartSign software, we recommend plugging in a USB Keyboard and Mouse temporarily.
4. Finally, connect the 5v PSU to the micro USB socket.

Your device should now start up Windows IoT and will load the SmartSign Lite demo layout automatically.

Congratulations, your SmartSign Lite is now ready to be customised to suit your requirements.

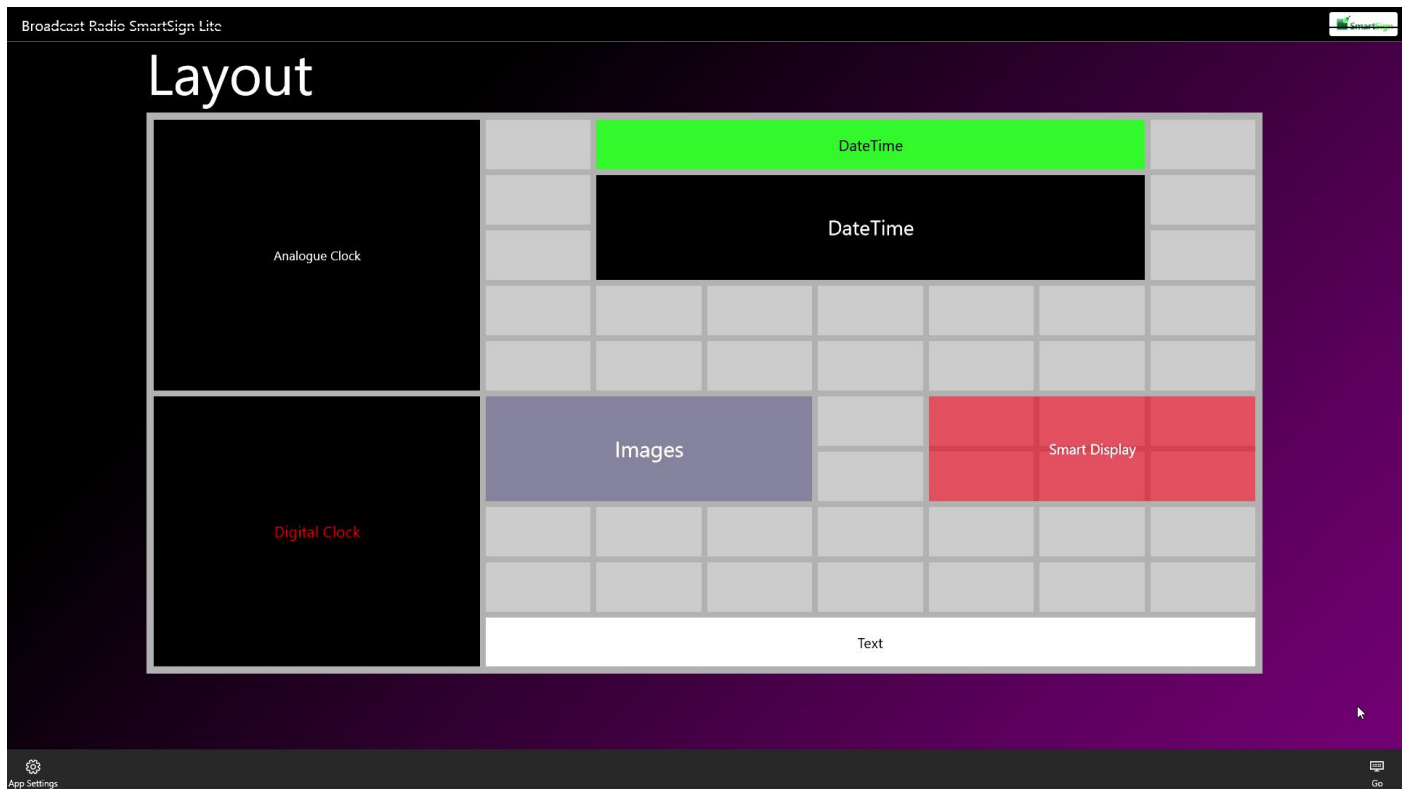
Notes: Once you have connected to a network with DHCP enabled your SmartSign Lite device should acquire a network address automatically when starting up, however, if you need to manually set a static address you will need to access the Windows IoT Device Portal on the SmartSign device. Details on the procedure are in the [Advanced Configuration Chapter](#) of this Guide.

If you wish to use your SmartSign Lite device to display hardware events, such as Mic Live or Phone Ringing events, you will need to use the included ribbon cable to connect to the GPIO header in the device.

## CONFIGURATION

Once you see the demo layout you can move your mouse to the bottom of the screen and left click on the grey bar to show the taskbar. You can then choose to either **Close** SmartSign Lite or go to the **Layout** screen.

SmartSign Lite runs on a grid system enabling you to design a layout to suit the needs you have at your station. The size of the grid can be controlled in **App Settings** found in the bottom left hand corner of the Layout Screen.



From this screen you can double click on any of the existing tiles to edit them. To see your working layout click on the bottom right **Go** button.

## APPLICATION SETTINGS

The Application Settings screen can be found from the main screen by going to the Layout screen and then selecting **App Settings** in the bottom left hand corner.

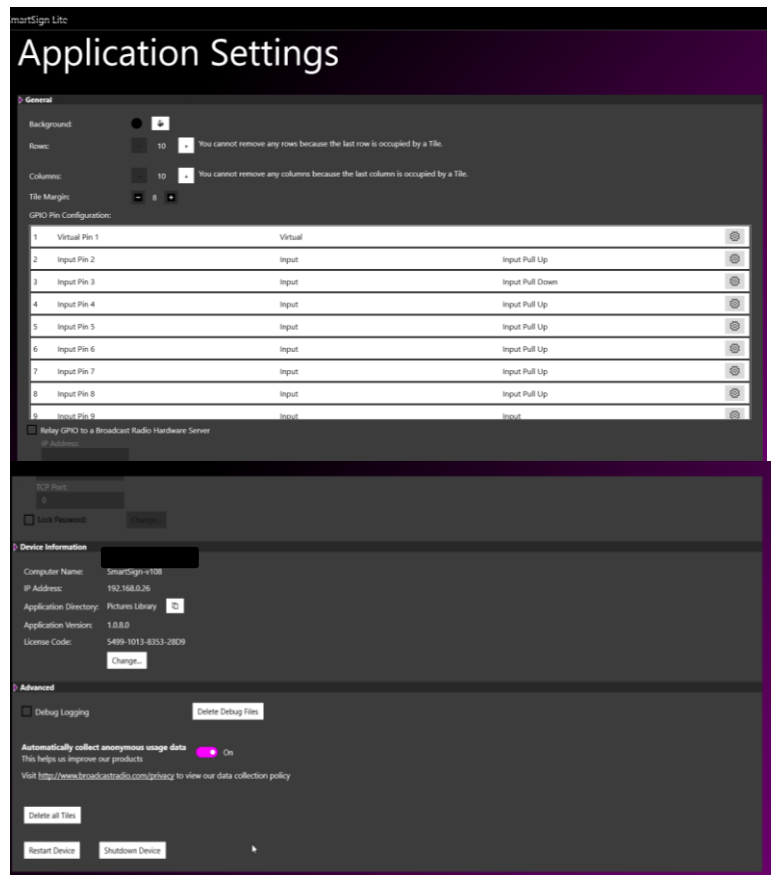
This enables you to change the background of the whole tile grid and change the size of each individual tile available and the margin around each tile.

This is also where you set your GPIO pins to trigger a smart display notification.

You can also choose to use the Broadcast Radio Hardware Service in which case you enter the IP address of the machine providing the service.

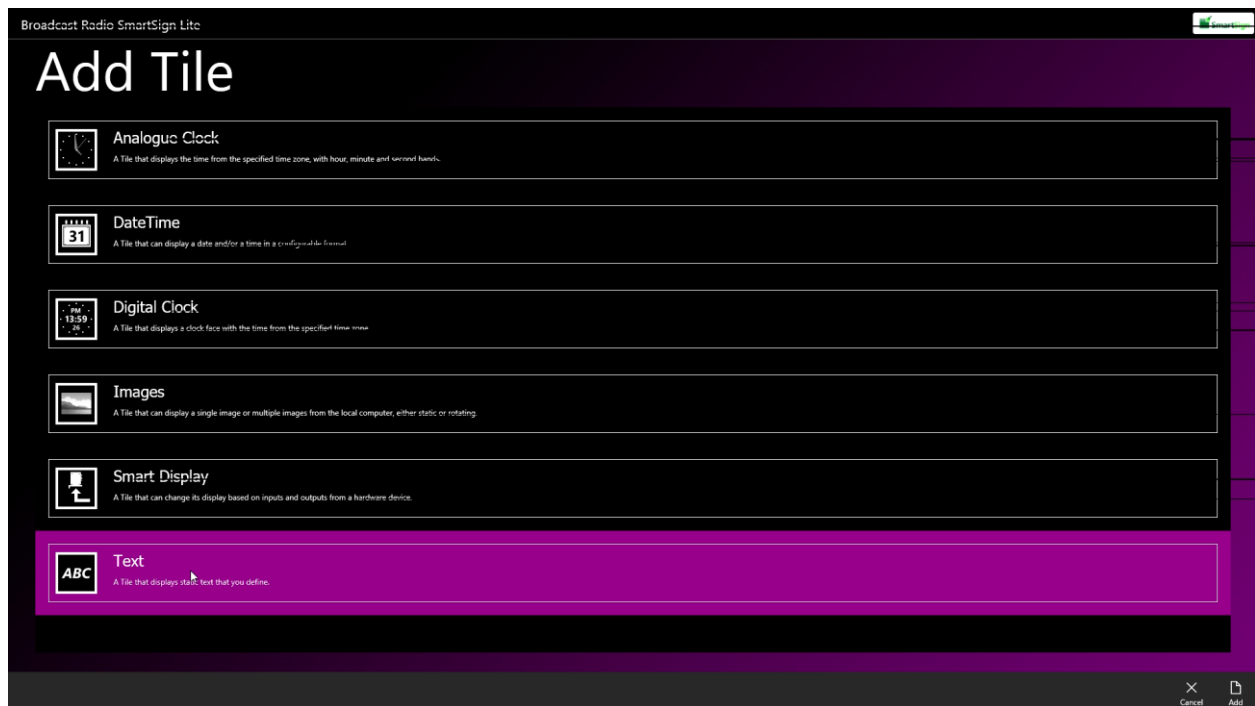
In Device Information you can see your computer name and IP address and applications directory and version number. This is also where you may need to enter your license code.

In the Advanced area there is the ability to turn on and delete debug logging and the option to turn off or on sending data back to Broadcast Radio. You can delete all the tiles on the demo layout and Shutdown or Restart the device.



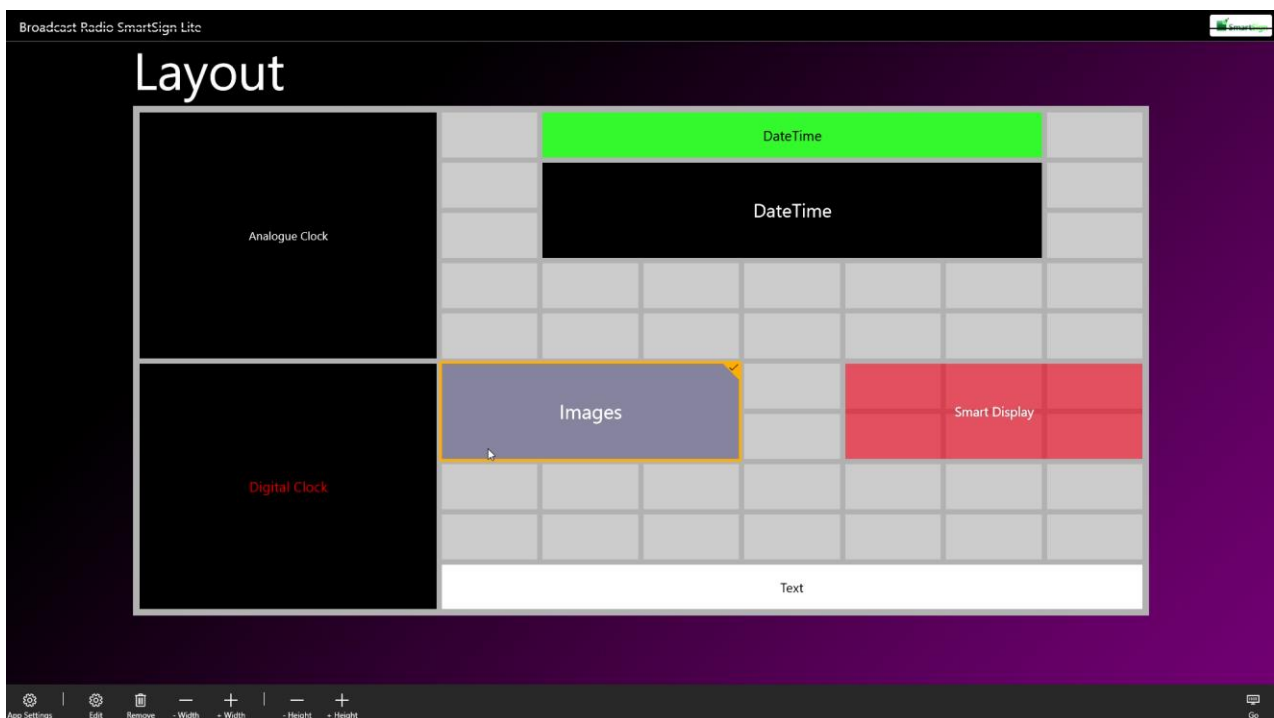
## ADDING TILES

Left click on an empty tile and choose Add to bring up the list of available tiles.



Double click on your choice of tile or left click and choose Add from the bottom right.

An options window will come up for whichever type has been chosen. Each tile type has a header, footer and appearance option which we will look at in more detail below. Once you are happy with your choices click on OK. Your tile will be labelled and coloured as you have selected.



To alter the size or position of your tile. Left click on it to highlight it in yellow (see above) and use the Height and Width buttons to change the size. To move the whole tile left click and drag it into the new position on the tile wall.

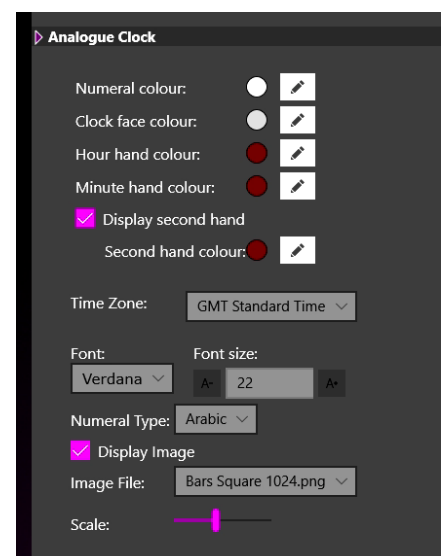


If you wish to edit your tile settings click on 'Edit' to get to the Settings Screen.

Every tile has a Header enabling you to add some text and a background colour to the Top of the tile. A Footer enabling you to do the same but at the bottom of the tile and an Appearance option. This allow you to choose a shape for the tile along with a background colour. If the shape allows you can also change the curve of the corners of the shape to suit. Above the footer options are tile specific settings which are described below.

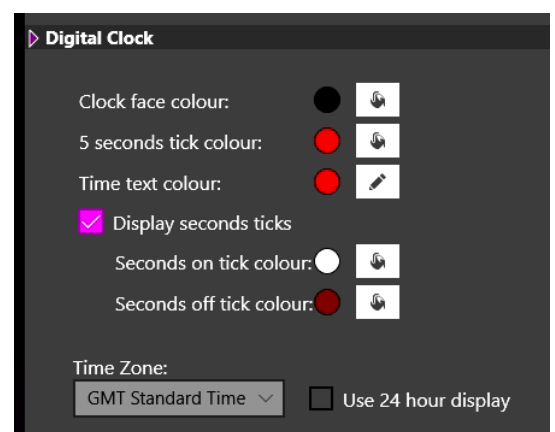
## ANALOGUE CLOCK

Most of these are self-explanatory. You have the option to display a second hand and you can choose a Time Zone and Fonts for the numbers and the Numeral Type. In our settings we have chosen to show an image in the middle of the clock and chosen a png file. There are details of how to add pictures in a later [chapter](#).



## DIGITAL CLOCK

In the options for the digital clock you can change the colour of each element of the clock. You have the option again to show the seconds and you can choose to use a 24 hour display if desired. You could choose to use a different timezone and use the header or footer to inform users as to the place or timezone used.



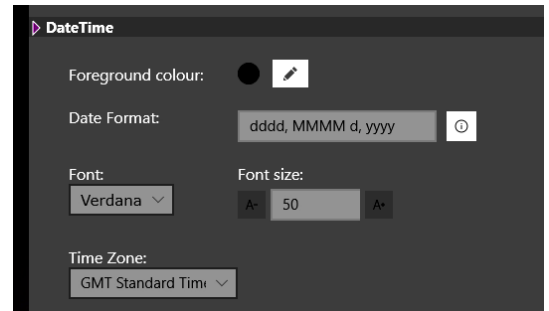
## DATE AND TIME

Here you can set the way in which the Date or Time is displayed. In our example in this guide we are using the Tile twice in our layout. One tile shows the date and the other tile shows the

d	The day of the month, from 1 through 31.
dd	The day of the month, from 01 through 31.
ddd	The abbreviated name of the day of the week.
dddd	The full name of the day of the week.
h	The hour, using a 12-hour clock from 1 to 12.
hh	The hour, using a 12-hour clock from 01 to 12.
H	The hour, using a 24-hour clock from 0 to 23.
HH	The hour, using a 24-hour clock from 00 to 23.
m	The minute, from 0 through 59.
mm	The minute, from 00 through 59.
M	The month, from 1 through 12.
MM	The month, from 01 through 12.
MMM	The abbreviated name of the month.
MMMM	The full name of the month.
s	The second, from 0 through 59.
ss	The second, from 00 through 59.
t	The first character of the AM/PM designator.
tt	The AM/PM designator.
y	The year, from 0 to 99.
yy	The year, from 00 to 99.
yyyy	The year as a four-digit number.
yyyyy	The year as a five-digit number.

Close

Time. By clicking on the button to the right of the Date Format box you can see the syntax needed to alter how the information is displayed.

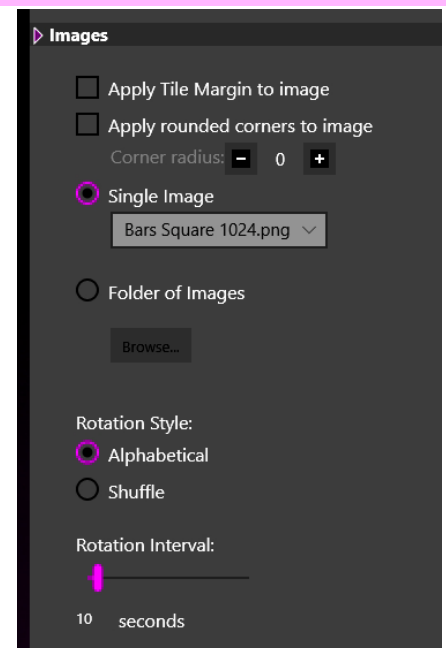


## IMAGES

This allows you to add images onto the SmartSign Lite screen. Normally used for Station Logos. See the [Advanced configuration chapter](#) for how to get images onto the device.

From the configuration screen you can choose to show a single image or you can select a folder of images which will rotate either alphabetically based on filename or shuffled in a random order. You can choose the interval in which these images change by clicking and dragging the pink slider.

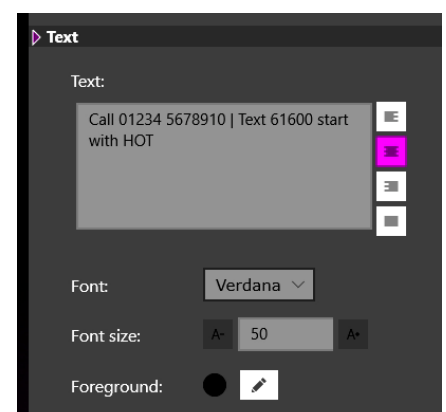
There is also the ability to add a margin around the image and to round the corners of the image also should you wish.



## TEXT

An obvious tile!

Allows you to add text to the screen. You can change the alignment with the buttons to the right of the text box. You can also change the font and size plus the colour of the Foreground.







This advanced tile enables you to configure what are Text tiles to change or trigger based on events around your studio.

This may be phone calls coming in, someone at the door, mics faded up or On Air control being given to the studio but could also be configured to act as a studio source selector or mic cough button using a small touchscreen.

Whatever you use them for there is a basic setup for every scenario. Choose 'Smart Display' as your tile type and you will see the configuration screen with the normal header/footer and appearance options plus the tile specific option area (see right).

In this guide we are using it to change the ON AIR Text Tile to a different colour when the studio is put to air using an analogue studio switcher which supplies a closed contact when selected.

Here it is selected in the appearance to be Red background and the text ON AIR will appear White as shown in the foreground.

The Display Change Type **1** has 3 options:-

On only while Pin is high – the tile will change to the options selected below when the circuit is completed (once control is given to the studio)

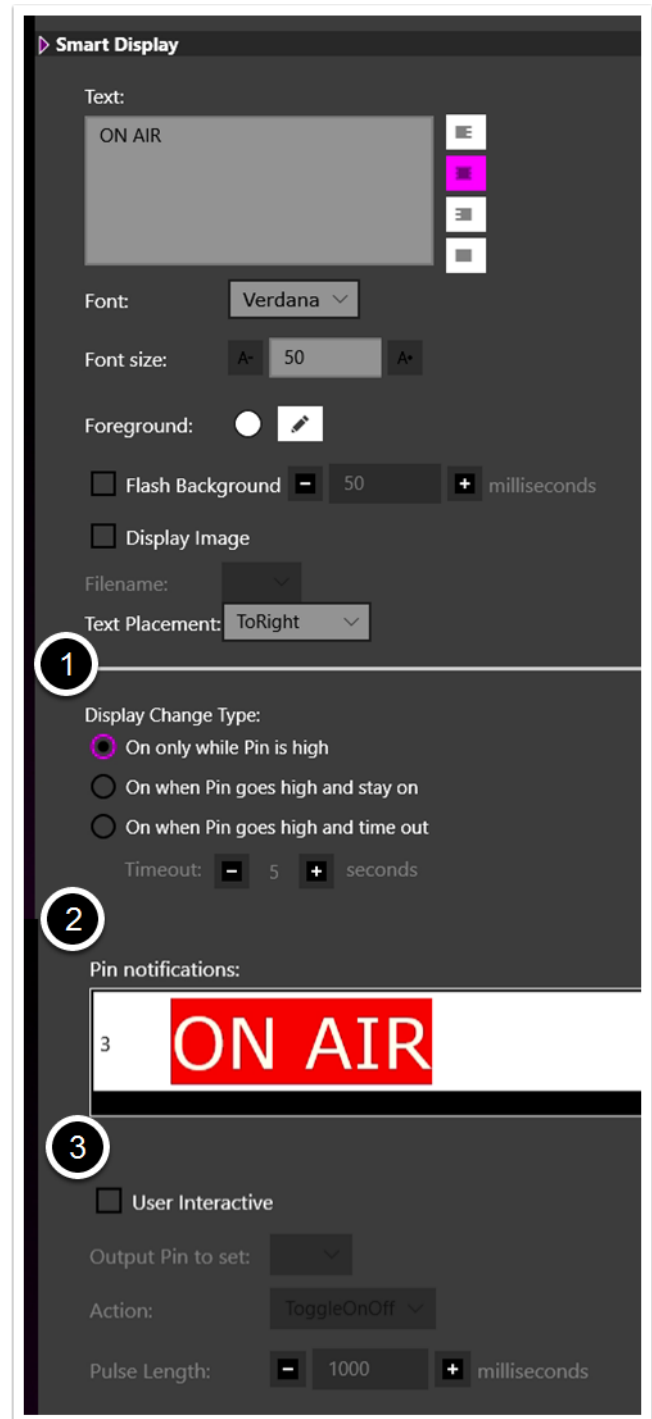
On when Pin goes high and stay on – tile will change to the options selected below when the circuit is completed however if studio control is taken away it will remain on.

On when Pin goes high and time out – similar to above except it will revert to the original tile after a set duration which you set in the Timeout settings below. Useful for a door entry system.

In Pin Notifications **2** you can select add pin to the right of the white box and choose the appropriate pin (see next chapter about the gpio pins available).

For our example we are using Pin 3 in Pull Up Mode. The mode Pull Up or Down is set in the Application Settings screen.

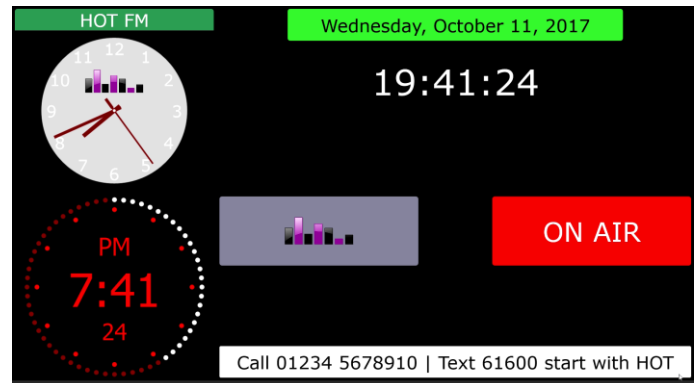
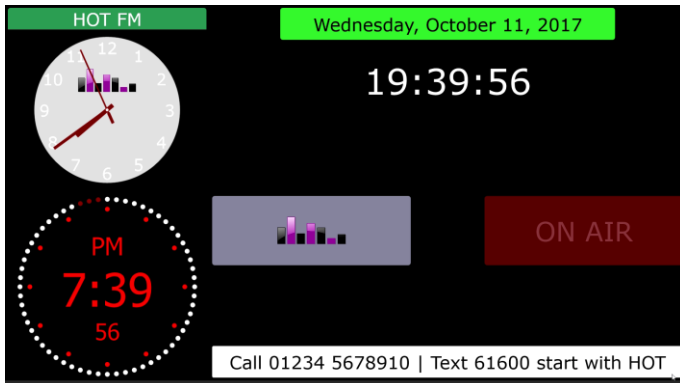
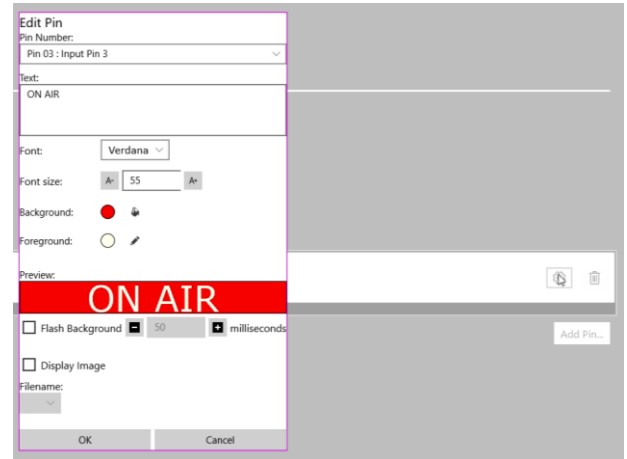
The User Interactive **3** section is used alongside a touchscreen device or could be used with a mouse. This could be used for a Mic Cough Button or to take the studio to air simply by pressing on the ON AIR tile in SmartSign Lite. You would set this to output Pin to set: 3 and the Action to be toggle On/Off.



Once a Pin notification has been added you will see a config dialog box for that notification. You can also get to this screen by clicking on the clog icon on the right side of the white box.

In this example we are choosing what the tile will look like when the studio is actually ON AIR. We are making the background brighter red and the text white. You can have the notification be an image instead of just text if required.

Now when the GPIO/PIN 3 circuit is completed the tile will change to the settings chosen on the right.



## CONFIGURING SMARTSIGN LITE TO RECEIVE EXTERNAL TRIGGERS

The Raspberry Pi has a GPIO (General Purpose Input Output) header fitted onboard and using the included ribbon cable can be connected to a variety of interfaces using a 'closed circuit' or 5v connection which is used by many 'Mic Live' systems, telephone flashing lights and mixers.

In these photos and in this example we are using GPIO3 (known as Pin 3 in SmartSign Lite) and a Ground pin.

GPIO header pins 02 and 03 are used by the touchscreen interface, so will not be available on the version of SmartSign Lite in the Touchscreen unit.

Virtual GPIOs can also be used via the Broadcast Radio Hardware Service (see separate documentation).



## Raspberry Pi 3 GPIO Header

Pin#	NAME		NAME	Pin#
01	3.3v DC Power		DC Power 5v	02
03	GPIO02 (SDA1 , I <sup>2</sup> C)		DC Power 5v	04
05	GPIO03 (SCL1 , I <sup>2</sup> C)		Ground	06
07	GPIO04 (GPIO_GCLK)		(TXD0) GPIO14	08
09	Ground		(RXD0) GPIO15	10
11	GPIO17 (GPIO_GEN0)		(GPIO_GEN1) GPIO18	12
13	GPIO27 (GPIO_GEN2)		Ground	14
15	GPIO22 (GPIO_GEN3)		(GPIO_GEN4) GPIO23	16
17	3.3v DC Power		(GPIO_GEN5) GPIO24	18
19	GPIO10 (SPI_MOSI)		Ground	20
21	GPIO09 (SPI_MISO)		(GPIO_GEN6) GPIO25	22
23	GPIO11 (SPI_CLK)		(SPI_CE0_N) GPIO08	24
25	Ground		(SPI_CE1_N) GPIO07	26
27	ID_SD (I <sup>2</sup> C ID EEPROM)		(I <sup>2</sup> C ID EEPROM) ID_SC	28
29	GPIO05		Ground	30
31	GPIO06		GPIO12	32
33	GPIO13		Ground	34
35	GPIO19		GPIO16	36
37	GPIO26		GPIO20	38
39	Ground		GPIO21	40

GPIO voltage levels are **3.3V** and are **not 5V tolerant**. There is no over-voltage protection on the board. If you are interested in interfacing to a mixer which provides a voltage relay you will need to use an external board with buffers, level conversion and analog I/O rather than directly onto the main board. More here:- <http://www.mosaic-industries.com/embedded-systems/microcontroller-projects/raspberry-pi/gpio-pin-electrical-specifications#rpi-gpio-input-voltage-and-output-current-limitations>

We strongly recommend that a closure to ground be used on inputs. Smartsign Lite inputs can be set to high or low so a closure to zero (ground) can be used to trigger a Smart Tile event. If you need to use a positive voltage to indicate a status change then follow the advice above or use an external board to control voltage / amps. You could also use a 'downstream' relay board to turn a voltage into a closure to ground. Contact Broadcast Radio support if you need more information.

### SETTING UP SMARTSIGN TO WORK WITH THE BROADCAST RADIO HARDWARE SERVICE

The Broadcast Radio Hardware Service is a software application used to relay hardware events to our fifth generation software products, using your existing IT network.

In the context of SmartSign Lite, this is most commonly used to allow a physical hardware input on one SmartSign to trigger events on multiple other SmartSigns on your network (for example a second SmartSign outside the studio that warns when the microphones are 'live').

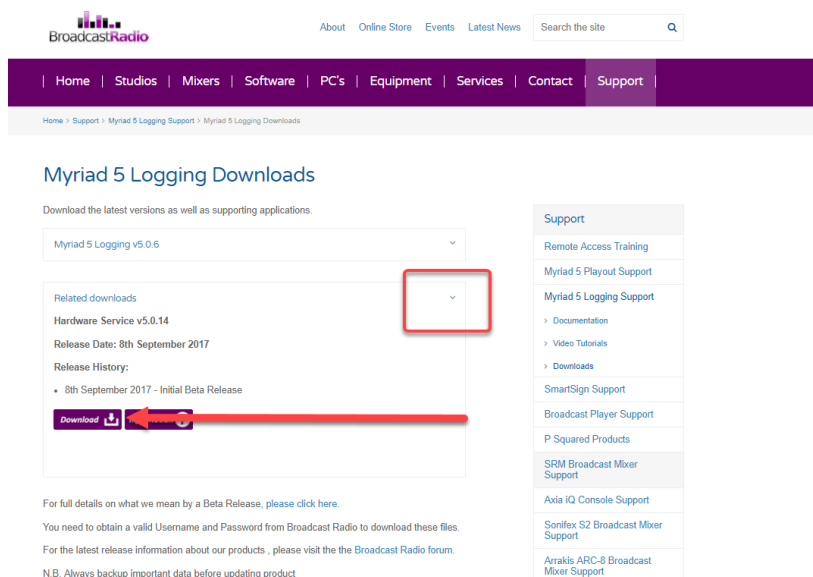
The Broadcast Radio Hardware Service is compatible with the following software.

- Myriad 5 Playout
- Myriad 5 Logging
- SmartSign Lite

In order to use this facility you need to download and install the Broadcast Radio Hardware Service which you can obtain for free from our website. You will also need to install it on a PC that is on the same network as the SmartSigns that need to use it.

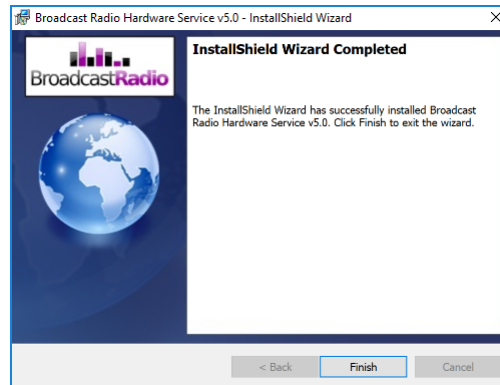
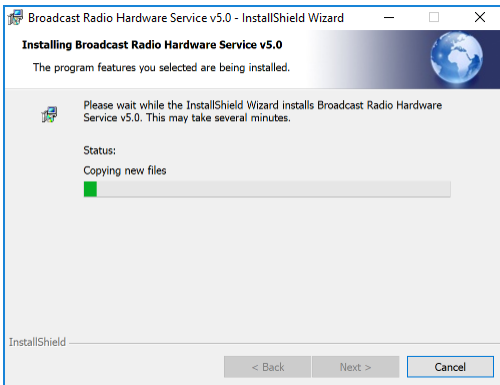
## INSTALLING & CONFIGURING THE HARDWARE SERVICE

Visit: <http://www.broadcastradio.com/support/myriad-5-logging-support/myriad-5-logging-downloads/>

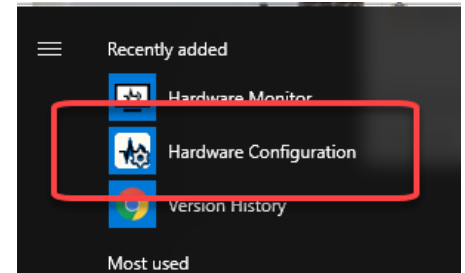


Expand the 'Related Downloads' option and then click on the **Download** button. The software will then download to your usual downloads folder.

Once the download is complete, double click on the file to begin the installation.

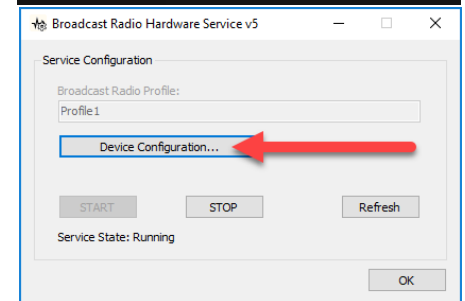


Next, press the Windows Start button on your keyboard and locate the **Hardware Configuration** option which should be listed under **Broadcast Radio Tools** (and in the Recently Added list).



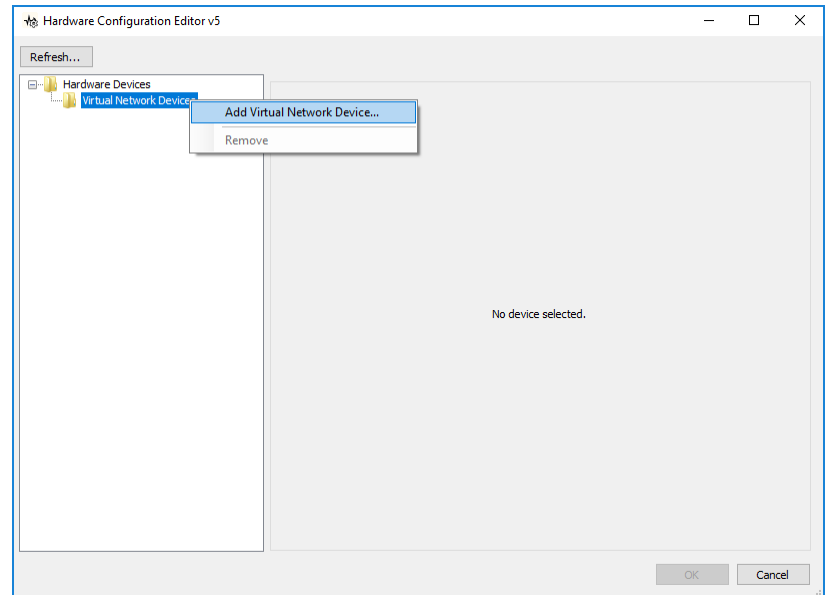
As the software starts, you may be asked to allow the software to configure a hardware device by Windows (depending on your user account access), if you are, just click 'yes'.

This window shows you the status of the Hardware Service and allows you to configure the devices. In this case you want to click on the **Device Configuration** button.

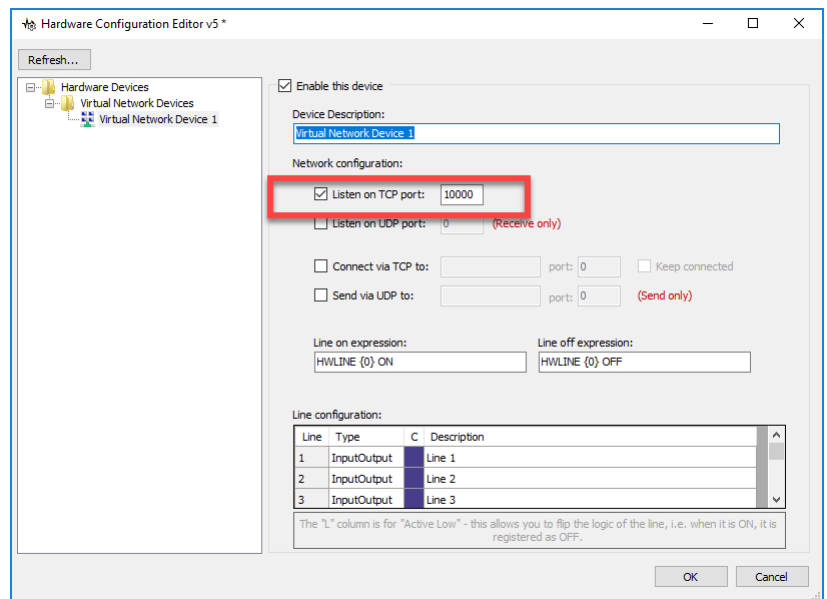


Next you will see an empty screen with 'No Device Selected' in the centre because we have not setup a device yet. Obviously if you have previously setup a device (physical or virtual) then it will be displayed here.

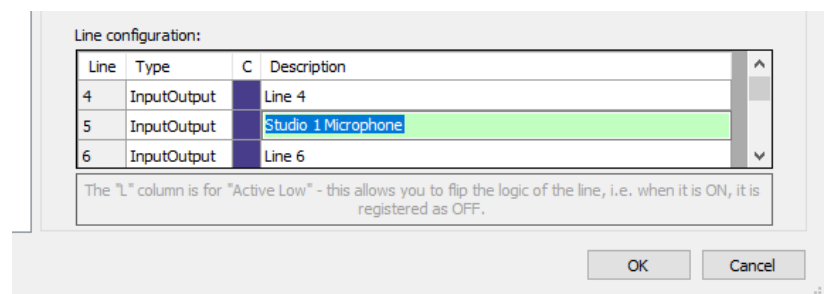
Right Click on the **Virtual Network Devices** folder and select **Add Virtual Network Device**.



This will create a new Virtual Network Device with the default settings. The settings are covered in separate documentation but in most cases you can leave the defaults in place. You will need to make note of the **Listen on TCP port** number which is set to **10000** by default. You will need that number when we come to configure the SmartSigns.



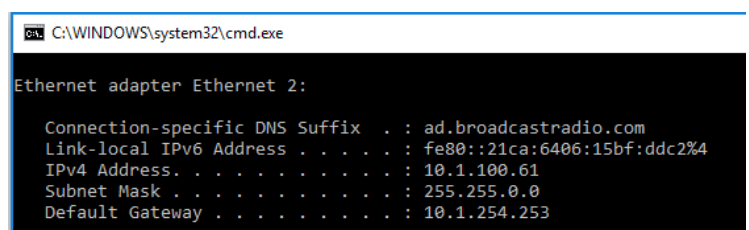
You can change the hardware line descriptions if you like to make testing easier. In this example we are going to use line 5 for the 'mic live' indicator in Studio 1 so I have altered the description for line 5.



Click on **OK** to close the window, you will be asked to confirm you wish to overwrite the previous settings. Once you have done that, you can click on **OK** on the Service Configuration window to close it. At this stage it will disappear but don't worry, the Hardware Service runs as a **Service** which means that it is running in the background all the time but you will not be able to see it working unless you run the **Hardware Monitor** application.

Now the Hardware Service is running and configured, you need to find out one more thing before you can configure your SmartSigns, you need to find out the IP address of the computer you have just installed onto. The quickest way is:

1. Windows Key + R (this opens the Run box)
2. Type in CMD and press the <Enter> key on your keyboard (this opens a command line window).
3. Type in **ipconfig** and press <Enter> (this will list your IP address).
4. Write down the IP Address on the **IPv4** line. This will be in the format xxx.xxx.xxx.xxx (in the example above it is 10.1.100.61).
5. Type **Exit** and press <Enter> to close the window.



You now have everything ready to setup SmartSign Lite to work with the Hardware Service.

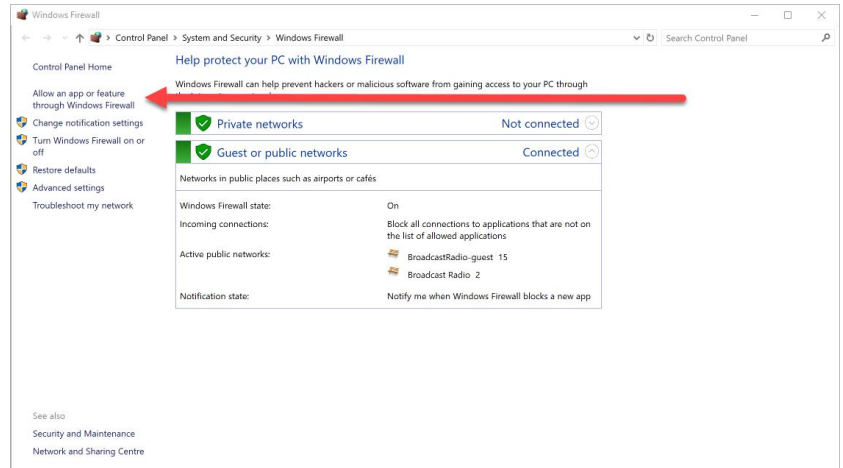




## Important: You Need To Update Your Windows Firewall

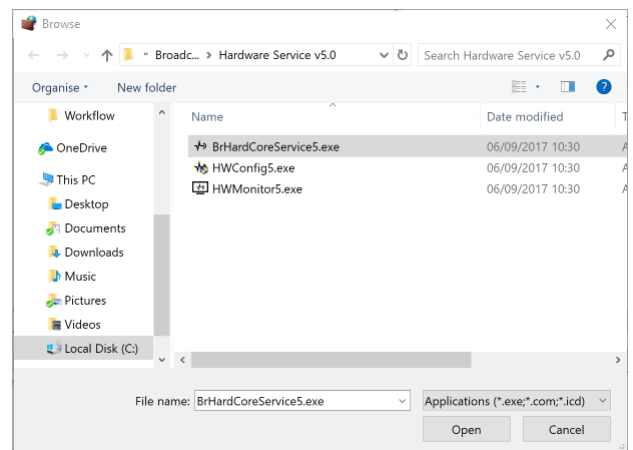
**IMPORTANT:** Whilst not technically part of the Hardware Service installation process, by default, your **Windows Firewall** will block your Hardware Service from communicating with the SmartSigns. You need to add in an 'exception' to allow the service to safely send and receive data from other connecting systems.

1. Press the Windows key on your keyboard and type in Firewall. You should see an option called **Windows Firewall**, run that option.
2. Click on the **Allow an app or feature through Windows Firewall** option on the left hand list.
3. Click on the **Change Settings** button (top right) and then the **Allow Another App** button (lower right).



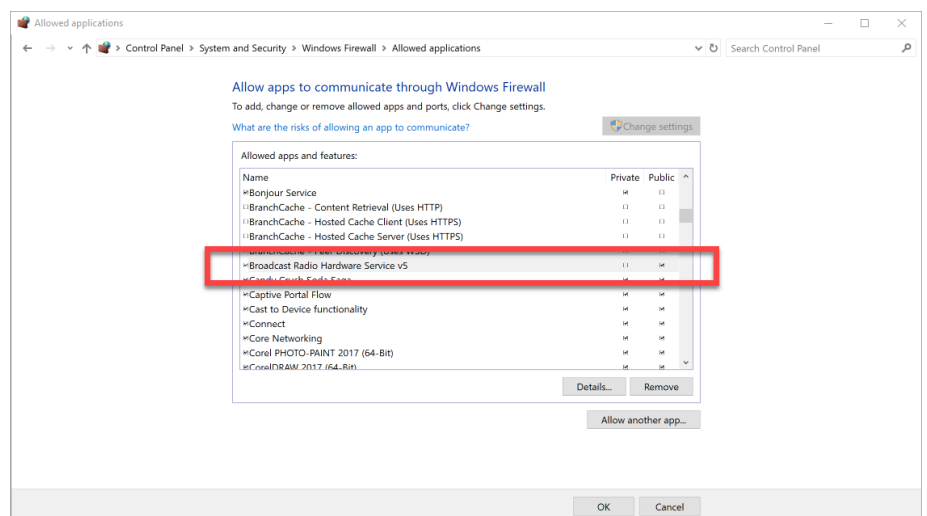
4. You now need to browse to the location where the Hardware Service is installed and select **BrHardCoreService5.exe**. The default location is:

C:\Program Files (x86)\Broadcast Radio\Hardware Service v5.0\BrHardCoreService5.exe



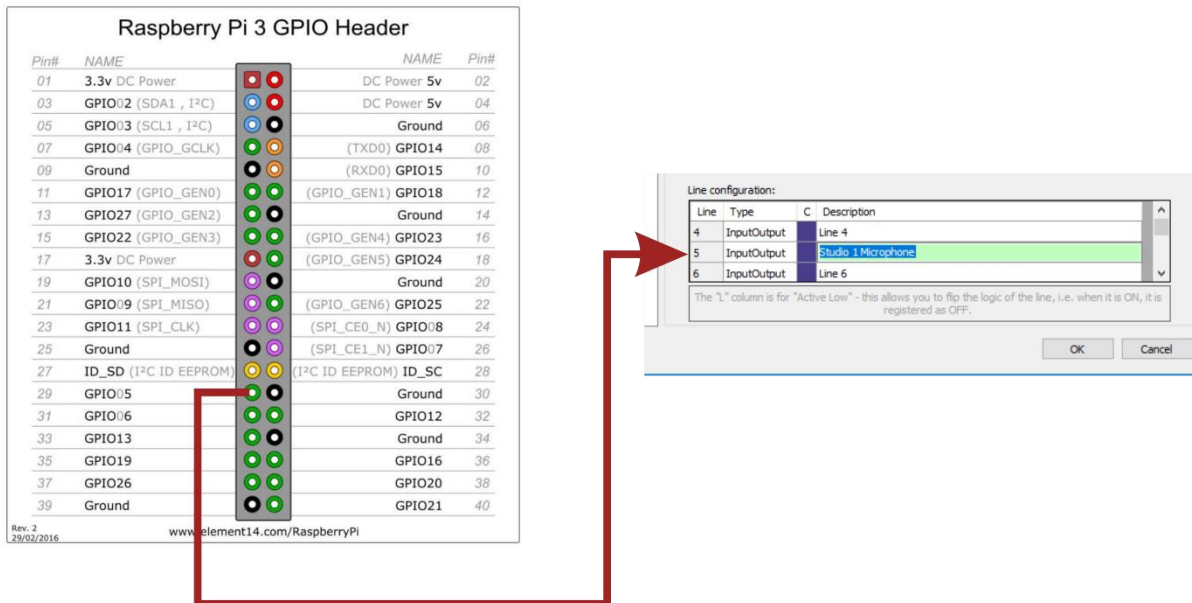
5. Now click on the **Add** button to add the app to the Firewall exceptions list. You should see it listed in the list of allowed applications.

Your Hardware Service should now be able to communicate with the other Broadcast Radio software and systems on your network.



## THE THEORY

The Hardware Service creates 100 virtual hardware lines that can be used for inputs or outputs. The first 27 of these lines are also directly linked to the corresponding GPIO inputs / outputs on the Raspberry Pi such that Virtual Line 5 in the Hardware Service is intrinsically linked to GPIO 5 on the Raspberry Pi (keep in mind this is not the same thing as pin 5 as the inputs are not mapped directly on Raspberry Pi's, in fact GPIO Line 5 on a Pi is actually physical pin 29 on the connector!)



The diagram above shows how GPIO Line 5 on the Pi (physical pin 29) is directly linked to Virtual Line 5 in the hardware service, so if you trigger an input on GPIO Line 5 on the Pi, Virtual Line 5 will automatically reflect that in the Hardware Service.

In practice, GPIO Lines 1-4 on the Pi are reserved for board use so GPIO 5 should be considered the first usable GPIO.

To understand how this works in the real world, we use the example of a 'Mic Live' indicator in a studio that needs to be duplicated to a second SmartSign located outside the studio to prevent people opening the door when the microphone is on.

Within the studio it is a very simple setup. The 'Mic Live' logic generated by the mixing desk whenever the microphone channel is 'on' is connected to GPIO Line 5 on the Raspberry Pi (usually this is actually a closure between Line 5 (pin 29) and a ground pin (there is conveniently a ground pin on pin 30 directly beside pin 29)). So when the microphone channel is 'opened' on the mixer, the mixing desk logic output creates a 'closure' between GPIO Line 5 (pin 29) and ground (pin 30).

You then configure your SmartSign to have a Smart Display Tile that changes 'state' whenever GPIO Line 5 is triggered.



When you turn on the microphone, the Smart Display Tile on the SmartSign will change accordingly.

So far this is nothing new but we now want to have the same thing happen on a second SmartSign outside the studio but we want to use a virtual hardware line to trigger the Smart Display Tile. Here are the steps we need to take:

1. Setup the first (in studio) SmartSign to 'broadcast' its GPIO Hardware Events to the Hardware Service.
2. On the second (outside studio) SmartSign, setup the corresponding GPIO line as an **Output** (the reason why is explained later).
3. Setup a Smart Display Tile on the second SmartSign to react to the corresponding GPIO Line as if it were a normal physical input.

Let's get started.

---

#### CONFIGURING THE SMARTSIGN IN THE STUDIO TO BROADCAST ITS HARDWARE STATUS

As already discussed, the hardware service has 100 virtual lines and lines 5 to 27 are directly mapped to the corresponding physical GPIO Lines on any Pi that is connected to the service.

When a physical input connected on GPIO Line 5 is triggered on any SmartSign connected to the Hardware Service, the corresponding Virtual Line 5 will also trigger and all other SmartSigns that are 'subscribing' to the service will also see that trigger.

In practice that means you are safest assigning a GPIO / Virtual Line for each single trigger or event you want to use throughout your system eg:

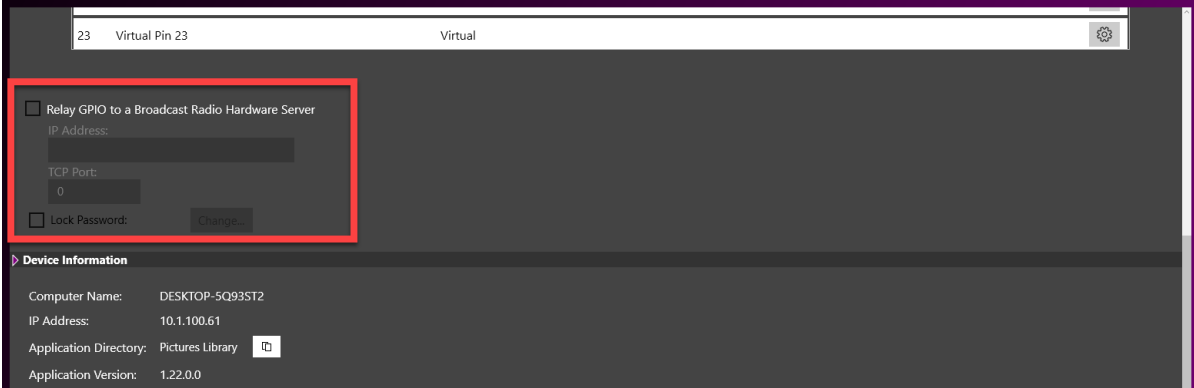
- Line 5 = Mic in studio1
- Line 6 = Mic in studio 2
- Line 8 = Studio 1 has on air control
- Line 9 = Studio 2 has on air control etc

That means only the SmartSign in Studio 1 is using physical GPIO Line 5 on it's Pi with the SmartSign in Studio 2 using a different line altogether.

In this example, we are assuming that we already have the SmartSign in studio 1 setup to receive a physical GPIO input on Line 5 and have a Smart Display Tile setup to change when it happens (mic live). Now we need to 'broadcast' this event via the Hardware Service.

On your SmartSign, go into **Layout** mode and click on the **App Settings** option located on the extreme left. Scroll down until you see the **Relay GPIO to a Broadcast Radio Hardware Server** option.

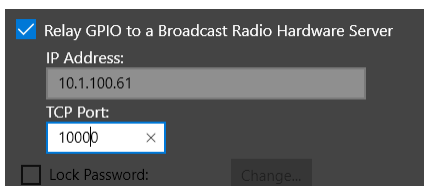
# Application Settings



Tick the box to enable and add the following information (which you noted when we setup the Hardware Service).

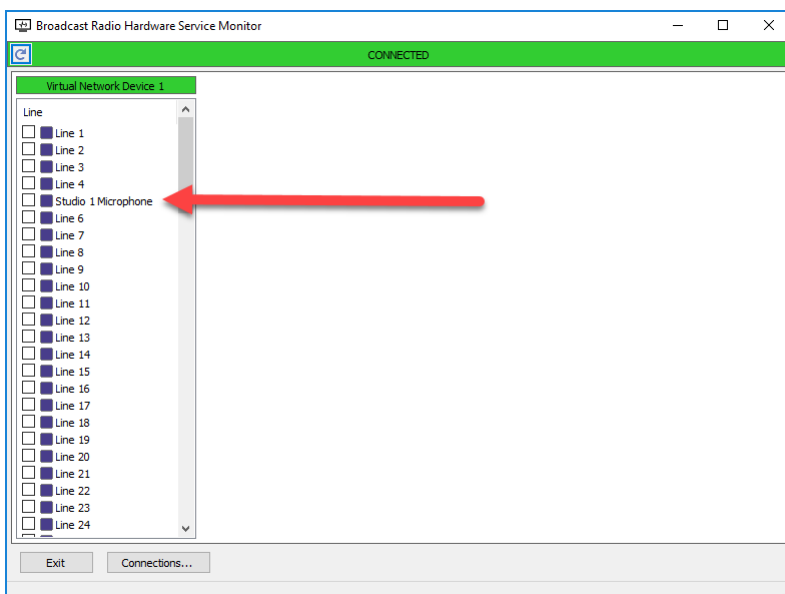
- IP Address of the PC running the Hardware Server/Service
- TCP Port used for the virtual device

In the example we looked at when setting up the Hardware Service they were 10.1.100.61 and 10000 respectively. Your IP address will be different but your default port should be 10000 unless you altered it when setting up the Virtual Network Device.



Click on the **Back** button at the bottom left of the **Applications Settings** screen to save your changes and return to the **Layout Page**. You can now click on **Go** to return to your SmartSign display.

*Tip: You can now run the Hardware Monitoring application on the PC that the service is running and you should be able to 'see' when the virtual hardware is triggered. Try running the Monitoring Application and then switching the microphone on and off in the studio, you should see the corresponding Virtual Line switching on and off.*



Now that we are 'broadcasting' the trigger via the Hardware Service, we need to setup the other SmartSign to react to the **Virtual Hardware Line**.

Assuming it is plugged into the same network as the Hardware Service, it is simply a case of setting the IP Address and Port for the Hardware Service (just like we did on the other SmartSign) and then set the GPIO Line we are interested in (5 in this case) to be an **Output**.

Why an Output I hear you say? Good question! There is a complex technical reason which involves instructing the internal chipset on the PI to ignore physical inputs and instead rely on virtualised ones provided in software but it is easier to just accept, for this to work you need to set the GPIO Line we are interested in to be an **Output**.

First thing is to add in the IP Address and Port for the Hardware Service using the same method as on the first SmartSign:

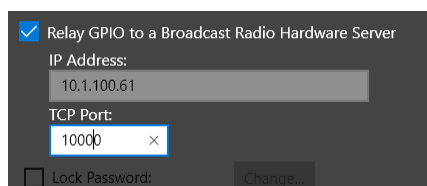
Go into **Layout** mode and click on the **App Settings** option located on the extreme left.

Scroll down until you see the **Relay GPIO to a Broadcast Radio Hardware Server** option.

Tick the box to enable and add the following information (which you noted when we setup the Hardware Service).

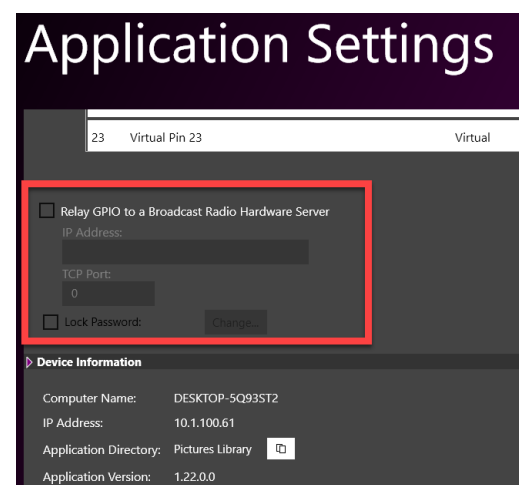
- IP Address of PC running the Hardware Server/Service
- TCP Port Used for the Virtual Device

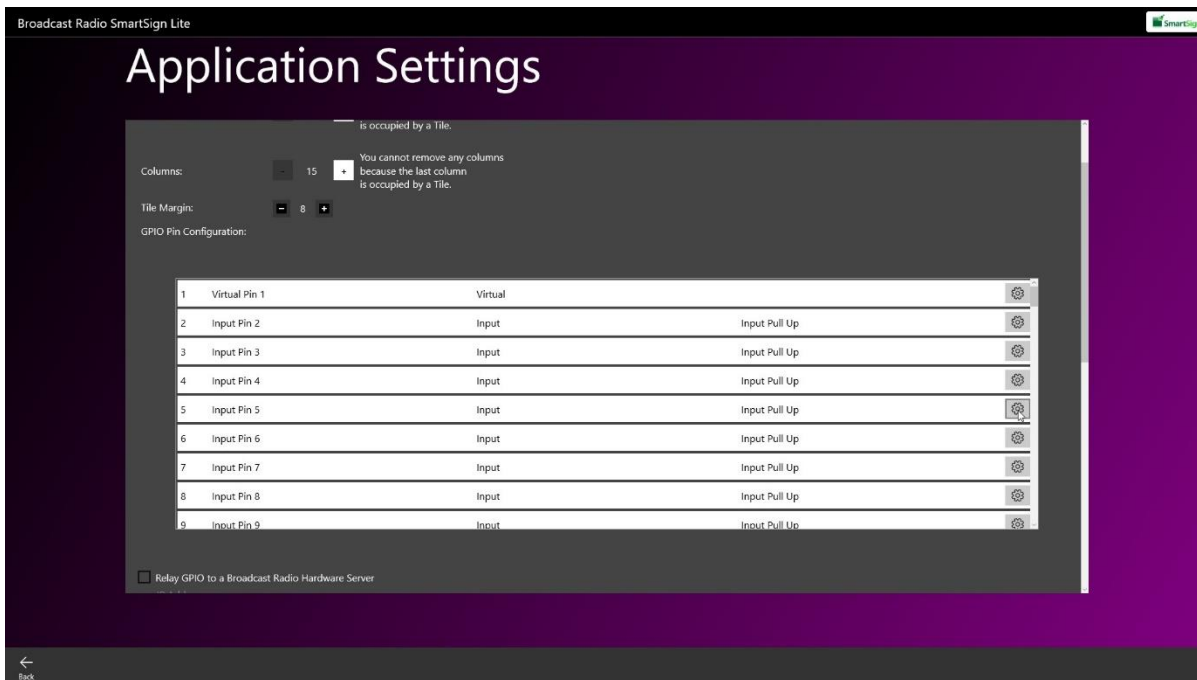
In the example we looked at when setting up the Hardware Service they were 10.1.100.61 and 10000 respectively. Your IP address will be different but your default port should be 10000 unless you altered it when setting up the Virtual Network Device.



But we also need to set GPIO Line 5 to be an **Output**.

Scroll up until you locate the **GPIO Pin Configuration** section.





Click on the  icon to edit the setting for the GPIO Line you are interested in (5 in this example).

On the config window.

1. Set the Description to something useful.
2. Change the **Type** to **Output**
3. Change the **Drive Mode** to **Output**
4. Click on **OK**

Configure Gpio Pin 5

Description:

Type:

Drive Mode:

Configures the GPIO pin in strong drive mode, with low impedance.

OK

Should now look something like this:

4	Input Pin 4	Input	Input Pull Up	
5	Studio 1 Mic	Output	Output	
6	Input Pin 6	Input	Input Pull Up	

Your SmartSign is now configured to 'subscribe' to the Virtual Hardware Line 5 and you can setup a **Smart Display Tile** to react to GPIO Line 5 changing just like it was a physically connected input.

Add Input Pin

Pin Number:  
Pin 05 : Studio 1 Mic

Text:  
Mic Live

Font: Verdana

Font size: A- 58 A+

Background: ●

Foreground:

Preview:  

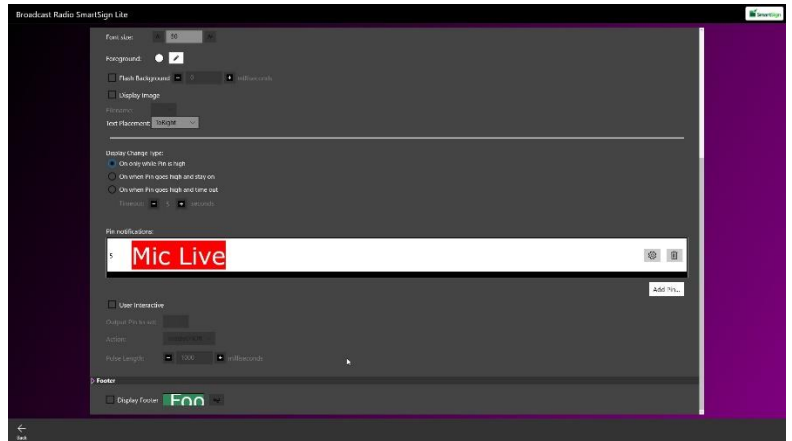
Mic Live

☐ Flash Background 0 milliseconds

☐ Display Image

Filename:

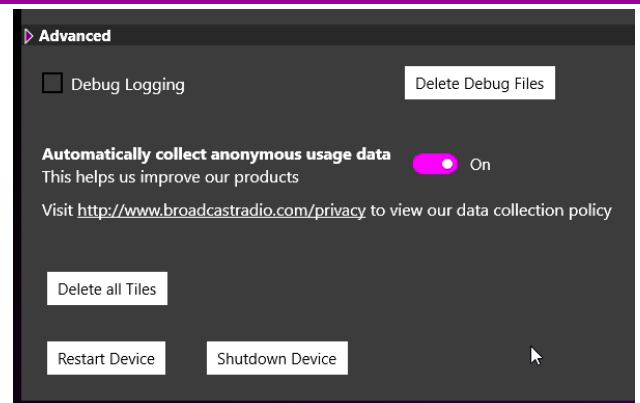
Add Cancel



## SHUTTING DOWN

It is important that SmartSign Lite is shut down correctly should you need to shut it down. This is done by going to the application settings window and scrolling down to the advanced area and clicking on **Shutdown Device** button.

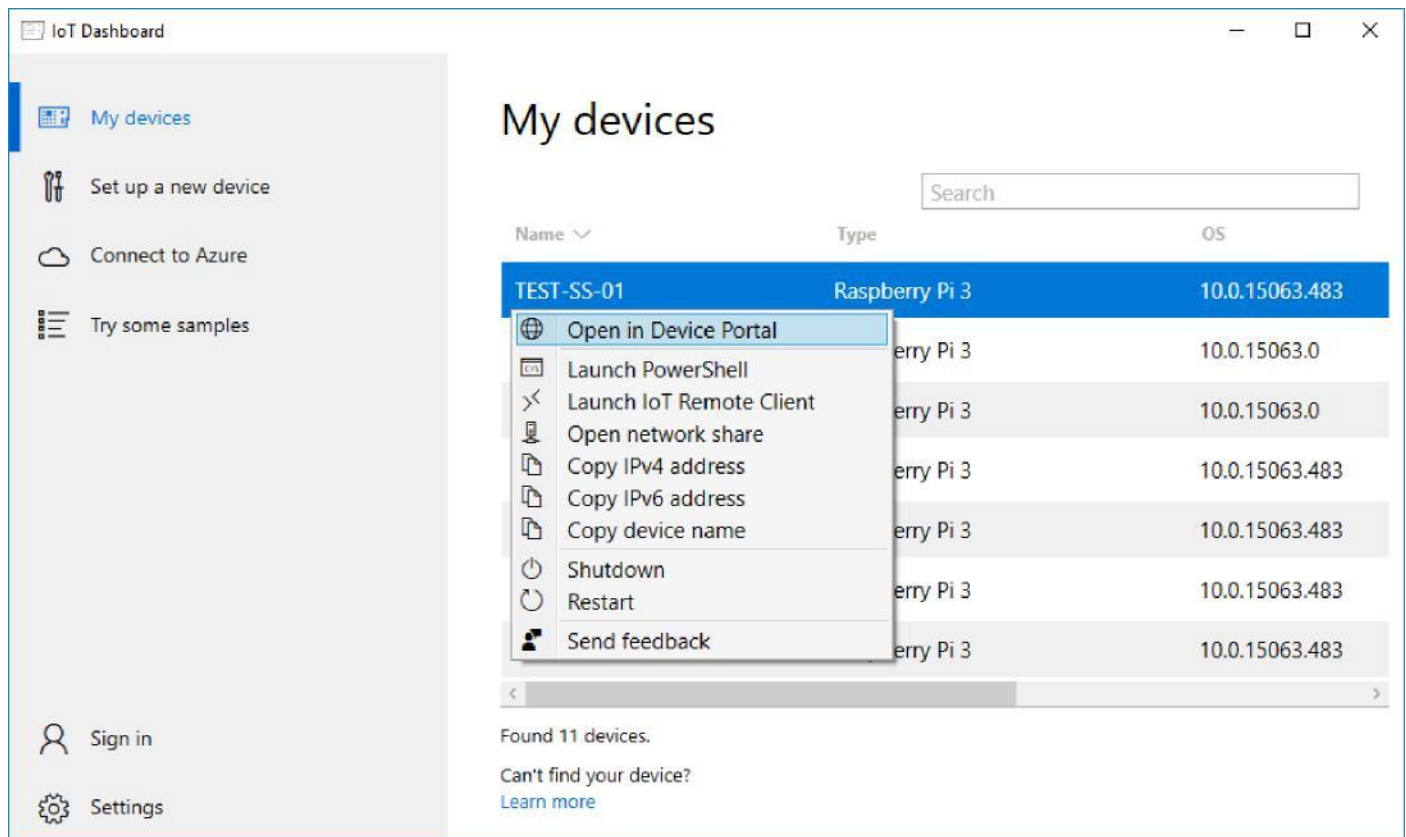
Wait until the screen goes completely black before unplugging the device.





## ADVANCED CONFIGURATION WITH THE WINDOWS IOT DEVICE PORTAL

SmartSign Lite runs on a version of Windows developed specifically for devices such as the Raspberry Pi called [Windows 10 IoT Core](#). To change system settings, such as network addresses, passwords or device name you will need to connect to the web interface called the [Windows Device Portal](#).



## CONNECTING TO YOUR DEVICE

The simplest method of connecting to Windows IoT, if you have a Windows 10 PC, is to install the [Windows 10 IoT Core Dashboard](#) software from Microsoft that can be found at <https://developer.microsoft.com/en-us/windows/iot/docs/iotdashboard>. This will automatically detect Windows IoT devices on the network and allow you to simply right click and connect to the Device Portal.

Alternatively, if you do not have Windows 10 or would like to manually connect to a Windows IoT you will need to know the IP Address. To find the address that your device has been assigned, look in the SmartSign Lite software application settings by clicking in the bottom right of the screen and selecting the [Layout](#) icon, then the [App Settings](#) icon on the bottom left. On this screen you should find details of the IP address that the device has been assigned.

Once you know this address, you can connect to the Windows IoT Device Portal using a web browser and typing the address and port into the address bar. The IP address will vary depending on your network, but the port should always be 8080. In your address bar type: <http://x.x.x.x:8080> replacing the x.x.x.x with the IP address of your device.

## USERNAME AND PASSWORD

Your SmartSign Lite Device has been preconfigured with a simple administrator logon, but this is not very secure and we would recommend that you change this password. However, if you do decide to change the administrator password, *please ensure that you take a note of this and keep it somewhere safe* as you will not be able to reset this easily if you forget it!

Username: Administrator

Password: admin

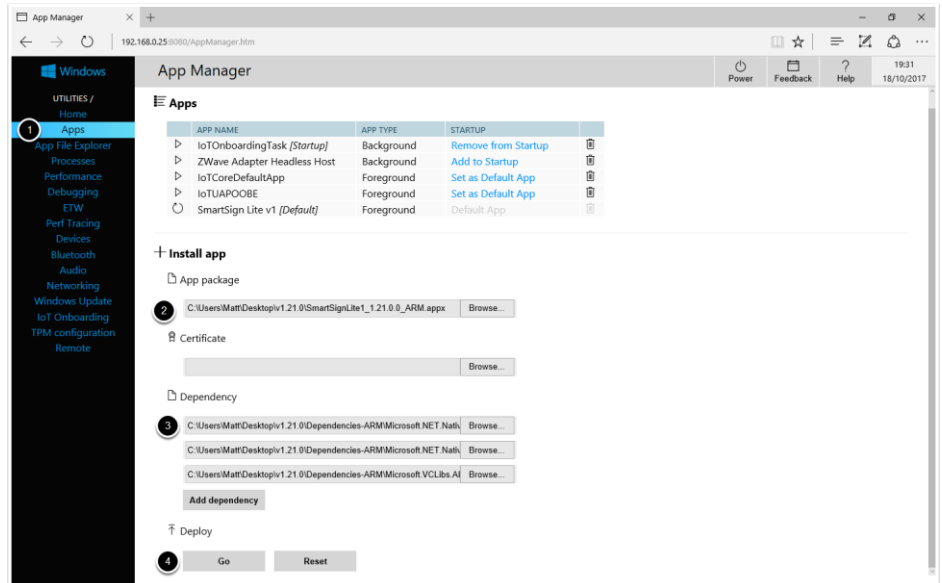
Once logged on, you can change the administrator password, Time Zone, or Display Settings from the [Device Settings](#) Tab.

The screenshot displays the 'Device Settings - Windows Device Portal' interface. The top navigation bar includes a hamburger menu, the title 'Device Settings - Windows Device Portal', the time '17:29' and date '13/06/2017', and links for 'Help', 'Feedback', and 'Power'. The left sidebar lists various settings categories: Device Settings (selected), Apps, Processes, Debug, Devices, Connectivity, TPM Configuration, Windows Update, Remote, Scratch, and Tutorial. The main content area features a header banner with an illustration of a person at a desk and a robot, along with the version 'v.10.0.15063.297' and the device identifier 'TEST-SS-01 Raspberry Pi 3'. Below the banner, the 'Device Settings' section is active, showing options to 'Change your device name' (with a 'New device name' input field and a 'Save' button) and 'Change your password' (with 'Old password', 'New password', and 'Confirm password' input fields, and 'Save' and 'Cancel' buttons). To the right, the 'Audio Control' section shows 'Audio Devices' with a 'Refresh' button and the current speaker 'Speakers (Raspberry Pi 2 audio)'. Below that, the 'Screenshot' section has a 'Capture Screenshot' button.

## UPDATING THE APP

There may be occasional updates to the app released and these can be installed using the Windows Device Portal on a web browser.

- On a machine on the same network in your address bar type: <http://x.x.x.x:8080> replacing the x.x.x.x with the IP address of your device.
- Enter your username and password.
- Click on Apps ① on the left hand side.
- In the App package ② browse to the location of the .appx file.
- Add in any dependencies which came with the app update ③.
- Click on Go ④ to deploy the app to the Raspberry Pi.
- Once uploaded you will need to restart your Raspberry Pi using the Power button in the top right of the Window.

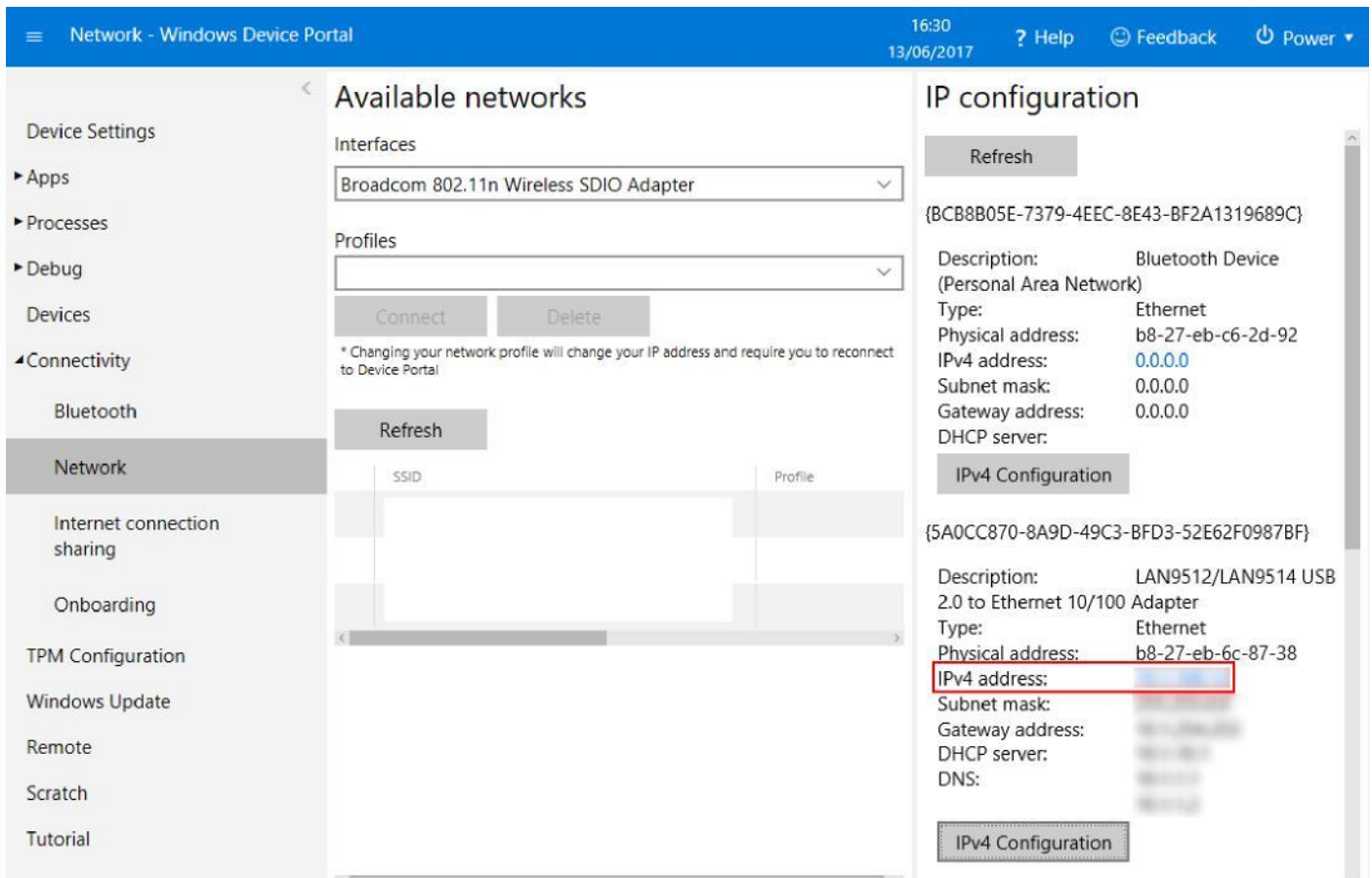


## NETWORK CONFIGURATION

If you need to manually configure an IP address for your wired network or to connect to a wireless network you can do this from the [Connectivity](#) menu of the Device Portal. (see Figure 3)

To configure a static IP address for the wired network, look in the list of devices on the righthand side for the one that currently has the IP address you are connected to the Device Portal. Click on the [IPv4 Configuration](#) button for that device and set the static details you need. Once these are set, the device will need to restart and you will need to reconnect to it on the new IP address to do any further work in the Device Portal.

*Please be careful when inputting these details as if you enter them incorrectly you will not be able to reconnect to the device!*



#### Notes:

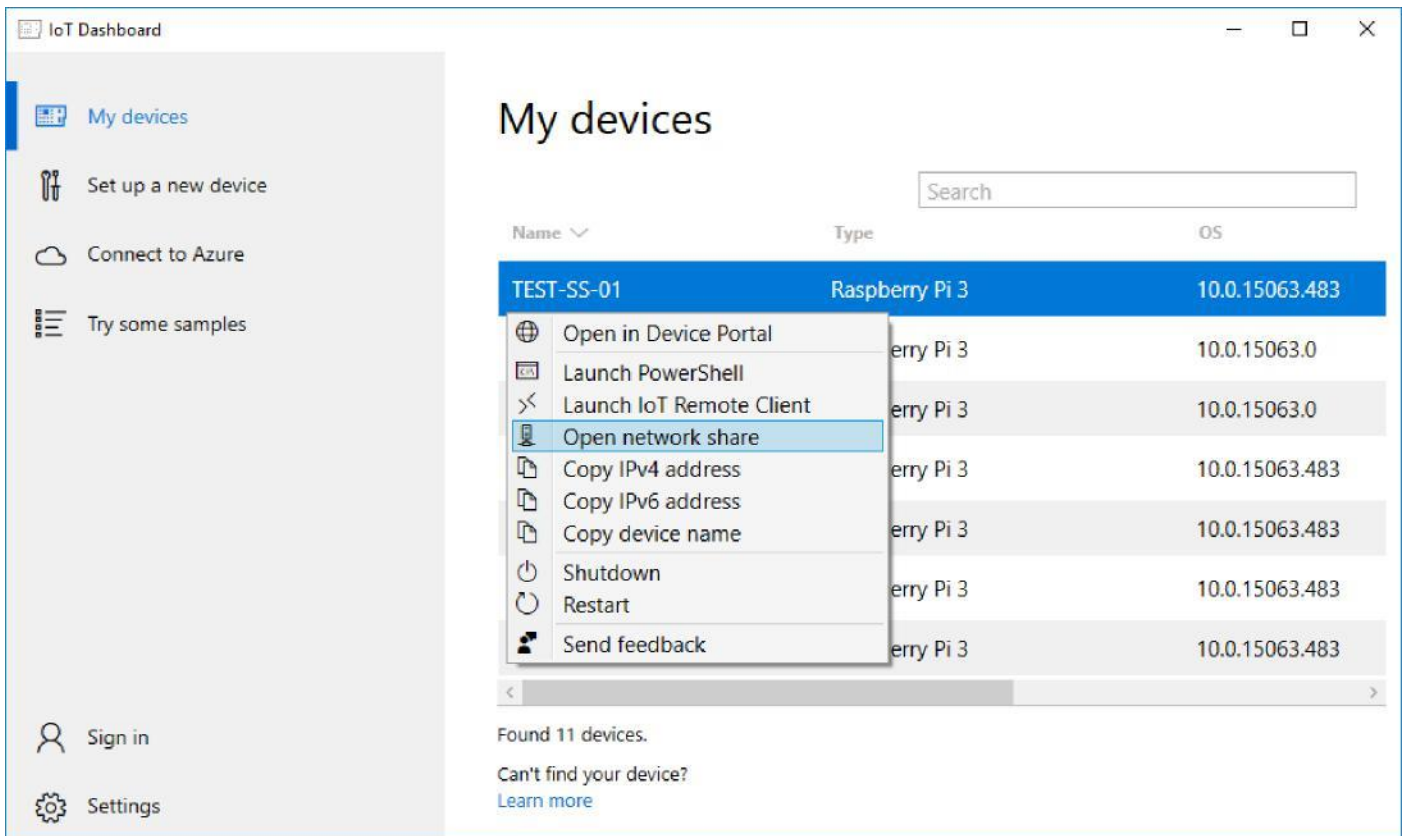
Broadcast Radio Strongly recommends you use a wired network to maintain a more reliable connection to SmartSign Lite, especially if using it in conjunction with the Broadcast Radio Hardware Service to pass GPIO signals to other Broadcast Radio software or devices!

#### ADDING IMAGES FOR USE IN TILES

You may want to add graphics or images to be used in SmartSign tiles for station logos or a rotating carousel of photographs to display.

Before you will be able to select picture in the tile configuration screen of SmartSign, you will need to copy them into the correct folder on the Device. Again, the simplest method is to use the [Windows 10 IoT Core Dashboard](#), right click on the device you would like to add pictures to and then choose the [Open Network Share](#) option.

As before, if you do not have Windows 10 on your PC, so cannot use the Dashboard, there is an alternative. By opening a Windows Explorer window and typing [\\x.x.x.x](#) into the address bar (replacing the x.x.x.x with the IP address of your device) you should get to the same place.



You will be asked for logon details to connect, so type in the Administrator username and password and you should be connected to the root of the C: drive on the device [\\x.x.x.x\\c\\$](\\x.x.x.x\\c$).

Browse to [\\x.x.x.x\\c\\$\\Data\\Users\\DefaultAccount\\Documents](\\x.x.x.x\\c$\\Data\\Users\\DefaultAccount\\Documents) and copy any picture files you would like to add to the rotating carousel list. To add files to be used on single logos or graphics, open the [\\x.x.x.x\\c\\$\\Data\\Users\\DefaultAccount\\Pictures\\Camera Roll](\\x.x.x.x\\c$\\Data\\Users\\DefaultAccount\\Pictures\\Camera Roll) folder and copy these here.

You can also use a web browser to navigate to your Raspberry Pi. Enter the ip address of your unit followed by :8080 and login using you username and password.

Click on the App File Explorer and navigate to the User Files \\ CameraRoll as in the image to the right.

Browse to the image you want to upload and click on Upload to send it to the machine.

Once the files have been copied across they should appear in the list of available pictures in the SmartSign Images Tile configuration screen.

