Connecting the Afterburner to Android   
IoT MQTT Panel

A close up of a sign

Description automatically generated

There are several MQTT clients available via the Google Store for Android devices.  
I have the greatest success using the “IoT MQTT Panel” app.

# Configuring the MQTT Broker.

The first thing for any MQTT client is to configure the settings to connect to the MQTT broker that the Afterburner is publishing to (and subscribed).

You need to edit the Connections panel to be able to attach to the MQTT broker.  
Note that several fields here are unique to your particular situation as everybody’s broker may be completely different / independent.  
  
Note that more than one Afterburner CAN connect to the one broker.  
**BUT**  
**Each Afterburner must use a unique topic prefix to ensure the correct device is being monitored & controlled.**If using a public broker it is essential you create a unique name to ensure you are talking to your own device.This will be covered shortly.

A screenshot of a cell phone

Description automatically generated

# Creating a “Device”

Inside the Connection configuration, there is a “Add Device” button next to the Device List.

Clicking upon that allows you define the topic prefix you need to use to your Afterburner.

## Device name

The **Device name** is what appears on you control panel you are creating, just leave that as Afterburner (or any other name you like)

## Topic Prefix

The next entry is the all important topic prefix that allows you to distinguish your particular Afterburner you wish to control from any others also present on the broker.  
By default, as shipped, the Afterburner has the topic prefix “Afterburner” defined.  
**It is strongly recommended that this prefix be made unique when you configure the MQTT configuration on the Afterburner.**

## Lock Panels

As you wish to build a new control panel, you must leave Lock Panels unchecked for now.  
Lock Panels allows you to remove the ability to alter the field’s parameters when showing the control panel, obviously a good idea once you have it working as you desire.

## Set as connection home

Set as connection home allows this device (control panel) to be what is automatically opened when you start the broker connection

## Enable device status

There are two aspects to the MQTT connection.   
1/ your client connecting to the broker,  
2/ The Afterburner connecting to the broker.

Enable device status allows you to know whether the Afterburner is actually “online” and communicating with the broker.

Enter the fields as shown as those values are hard coded into the Afterburner, which also defines the Last Will and Testament to the broker as “offline” upon the initial connection should the connection be lost unexpectedly.

### Payload sync request and Auto sync on connect

Is used to ask the Afterburner to resend its current state upon the client attaching to the broker.  
The general topics are not retained by the broker, so this ensures everything is up to date.

# A screenshot of a cell phone Description automatically generated

# Building the Control Panel

You can design your panel however you like, but here I will describe how this panel was created:

A screenshot of a cell phone

Description automatically generated

Note that in the descriptions I present the complete topic that is published or subscribed.  
The topicPrefix/ was previously discussed – ie must be unique etc.

This just leaves the fundamental suffix of the full topic to be defined for each panel.   
NICE feature.

A close up of text on a white background

Description automatically generatedAt the top left is the “Power” switch.

Add a “Switch” panel then complete the details as shown here.

### Panel name

Is used to annotate the control on the final control panel.

### Topic

Is the MQTT topic published to the broker to request the heater to be turned on or off: topicPrefix/cmd/Run.

### Subscribe Topic

Is used to obtain the current run state of the heater by subscribing to the broker using topicPrefix/sts/Run.  
According to the payload value received here, the switch will have the appropriate graphic shown.

A hand icon is used is when the heater is on.   
It’s coloured red to indicate pressing the switch will make the heater stop.  
The naming seems contrary but makes sense when you consider the actual state.  
Pressing the switch will cause ‘0’ to be the payload when publishing topicPrefix/cmd/Run – thus requesting the heater stops.

A green “On finger” is shown when the heater is off.   
Pressing the switch will cause ‘1’ to be the payload when publishing topicPrefix/cmd/Run – thus requesting the heater starts.

A screenshot of a cell phone

Description automatically generatedThe current run status of the heater is a simple “Text Log” panel.

### Panel Name

Is the Annotation shown on the panel, in this instance “Status”

### Topic

Is the topic that the Text Log panel subscribes to, in this instance topicPrefix/sts/RunString

A screenshot of a cell phone

Description automatically generatedThe current ambient temperature reported by the After burner is presented as a “Gauge” panel.

### Panel name

Is the Annotation shown on the panel, in this instance “Ambient Temperature”

### Topic

Is the topic that is subscribed to, in this instance topicPrefix/sts/TempCurrent

### Payload min/max

Define the extents of the displayed gauge. 0-40 gives a nice nominal just past vertical.

### Unit

It is not that easy to distinguish, or show Fahrenheit using these simple MQTT clients. It only makes sense to set the units as °C

### Color sectors

Defines the colour of the gauge, according to the payload’s value. The zoning can be altered if desired, but as shown here is just a simple 1/3 2/3 of full scale.

A screenshot of a cell phone

Description automatically generatedThe current body temperature reported by the After burner is presented as a “Gauge” panel.

### Panel name

Is the Annotation shown on the panel, in this instance “Heater Temperature”

### Topic

Is the topic that is subscribed to, in this instance topicPrefix/sts/TempBody

### Payload min/max

Define the extents of the displayed gauge. 0-240

### Unit

It is not that easy to distinguish, or show Fahrenheit using these simple MQTT clients. It only makes sense to set the units as °C

### Color sectors

Defines the colour of the gauge, according to the payload’s value. The zoning can be altered if desired.  
120 corresponds with all 6 heat bars on the OEM LCD

A screenshot of text

Description automatically generatedThe desired temperature is presented as a “Slider” panel

### Panel name

Is the Annotation shown on the panel, in this instance “Set Demand”

### Topic

is the topic published to when the slider is moved by the user, in this instance topicPrefix/cmd/TempDesired

### Subscribe Topic

Is the topic the Afterburner publishes to report its current set point, in this instance topicPrefix/sts/TempDesired

### Payload limits

match the default embedded values the heater uses.

A screenshot of a cell phone

Description automatically generatedThe current fan RPM is a simple “Text Log” panel.

### Panel Name

Is the Annotation shown on the panel, in this instance “Fan”

### Topic

Is the topic that the Text Log panel subscribes to, in this instance topicPrefix/sts/FanRPM

A screenshot of a cell phone

Description automatically generatedThe current pump delivery rate is a simple “Text Log” panel.

### Panel Name

Is the Annotation shown on the panel, in this instance “Pump”

### Topic

Is the topic that the Text Log panel subscribes to, in this instance topicPrefix/sts/PumpActual

A screenshot of a cell phone

Description automatically generatedThe current error status of the heater is a simple “Text Log” panel.

### Panel Name

Is the Annotation shown on the panel, in this instance “Error”

### Topic

Is the topic that the Text Log subscribes to, in this instance topicPrefix/sts/ErrorString

# Panel scaling

When you just add the panels, they will typically be full width.  
You can adjust them afterwards to be a proportion of the width, and you can also adjust the order about.

# Locking the panels

Once happy with the layout, you should lock the panels to avoid accidentally clicking on the 3 dots, starting the edit process of each panel.

Revisit the device configuration and check the Lock Panels checkbox.