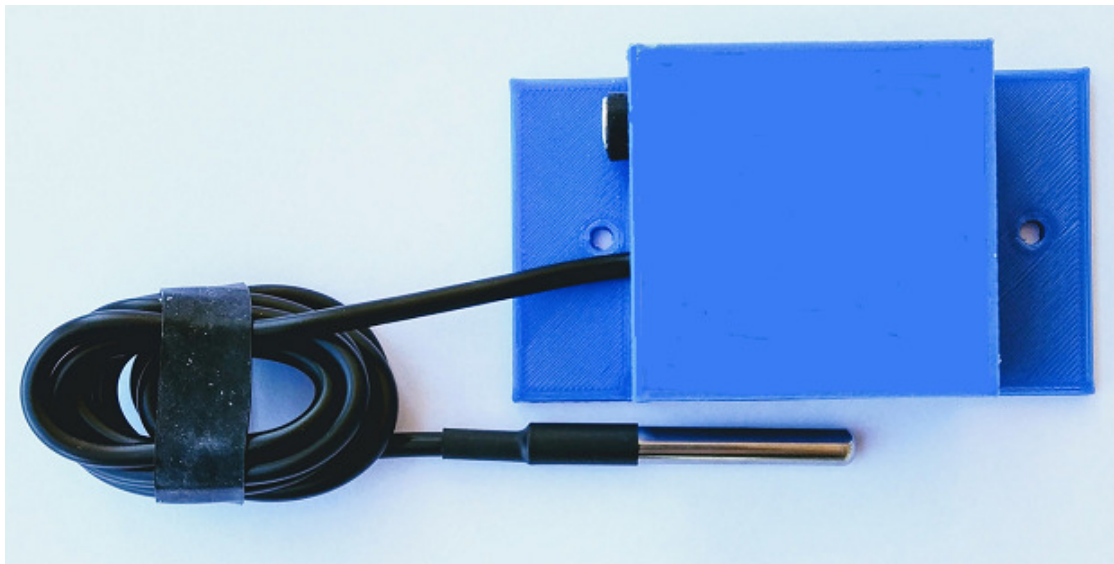




Tasmota-Plus Smart Gauge - Temperature



SG-TEMP User Guide

V20210330

Latest Version of this document available at:

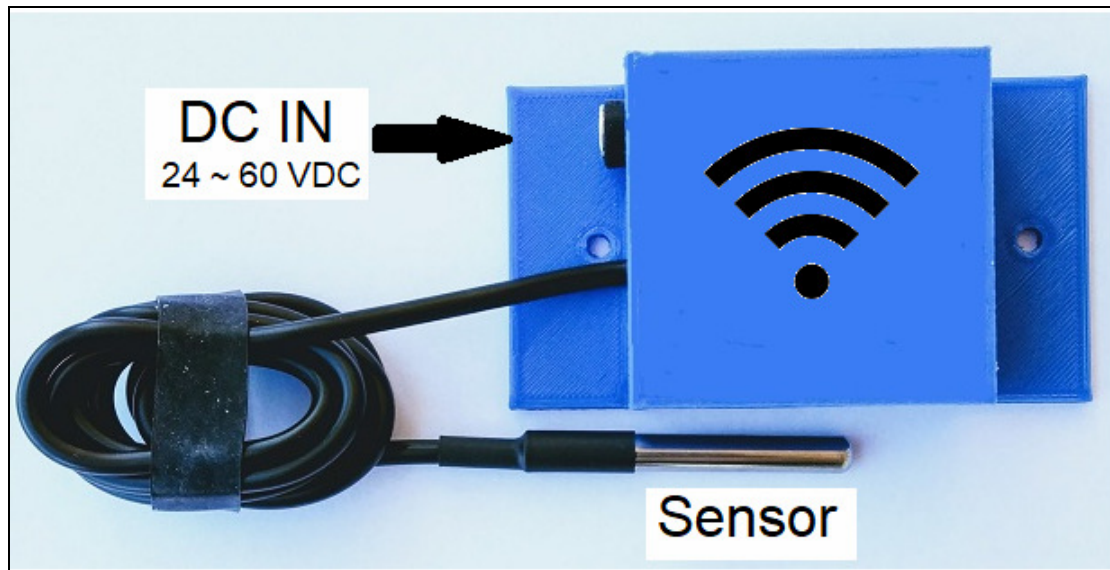
<https://github.com/UBWH/ubwh.github.io/blob/master/assets/UserGuides>

Table of Contents

Introduction	3
Hardware	4
Requirements	4
Web Browser Interface	5
Power Options	6
Near a Power Point (GPO)	6
Far from a Power Point (GPO)	7
Getting Started	8
Enabling the SG-TEMP Sensor	9
Relay Control based on Temperature	10
http:// Command Interface	12
Centralised Monitoring & Control	13
WebGUI Interfaces	14
openHAB Channel Definition (Example)	16
openHAB Sample History Plot	17
Specifications	18

Introduction

The SG-TEMP is a Temperature sensor with a smart WiFi interface.



Features include:

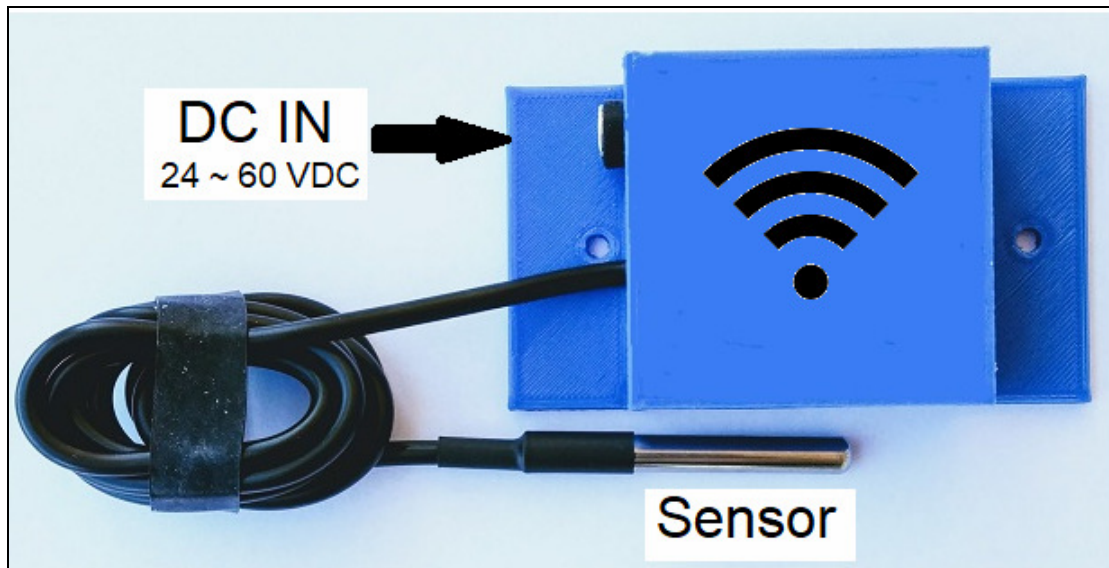
Temperature monitoring	Real-time temperatures can be viewed locally, or from anywhere in the world, accessible by any web browser.
Temperature logging	When combined with an openHAB/MQTT ¹ server, current and past temperatures are accessible locally, or from anywhere in the world, accessible by any web browser.
Flexible DC power supply	This device is powered by un-regulated DC. (Power supply not included) Voltages between 24 and 60 VDC can be used.
Relay	A SPST ² relay (16 A / 240 V) is included, which can be temperature controlled (ON/OFF). For example: a Fan or Heater could be turned on/off as the temperature passes user-defined set-points.
External switch monitoring	A user-supplied, external switch can be connected and monitored; i.e. Open or Closed.

Table 1 – Available Features

¹ <https://openhab.org> & <https://mqtt.org>

² Single Pole, Single Throw

Hardware



The **SG-TEMP** comes pre-assembled and tested. It consists of:

- a waterproof temperature sensor,
- a waterproof cable, and
- a wall-mount enclosure with built-in SS-1CHPro³ smart WiFi interface.

Requirements

The SS-1CHPro requires:

- **Initial Setup**
 - A device with a Web Browser & WiFi interface, located close to the SG-TEMP. A smart-phone, or tablet will usually be sufficient.
- **Operation**
 - A WiFi Access Point (AP) connected to the local LAN⁴, within the WiFi Range⁵ of the SG-TEMP.
 - A DHCP⁶ server on the LAN.
- **Ongoing Management**
 - Any device with a Web browser and connected to the same LAN as the SG-TEMP.

³ <https://ubwh.com.au/SS-1CHPro>

⁴ Local Area Network. See https://en.wikipedia.org/wiki/Local_area_network

⁵ See Specifications, page 18

⁶ Dynamic Host Configuration Protocol: See https://en.wikipedia.org/wiki/Dynamic_Host_Configuration_Protocol

Web Browser Interface

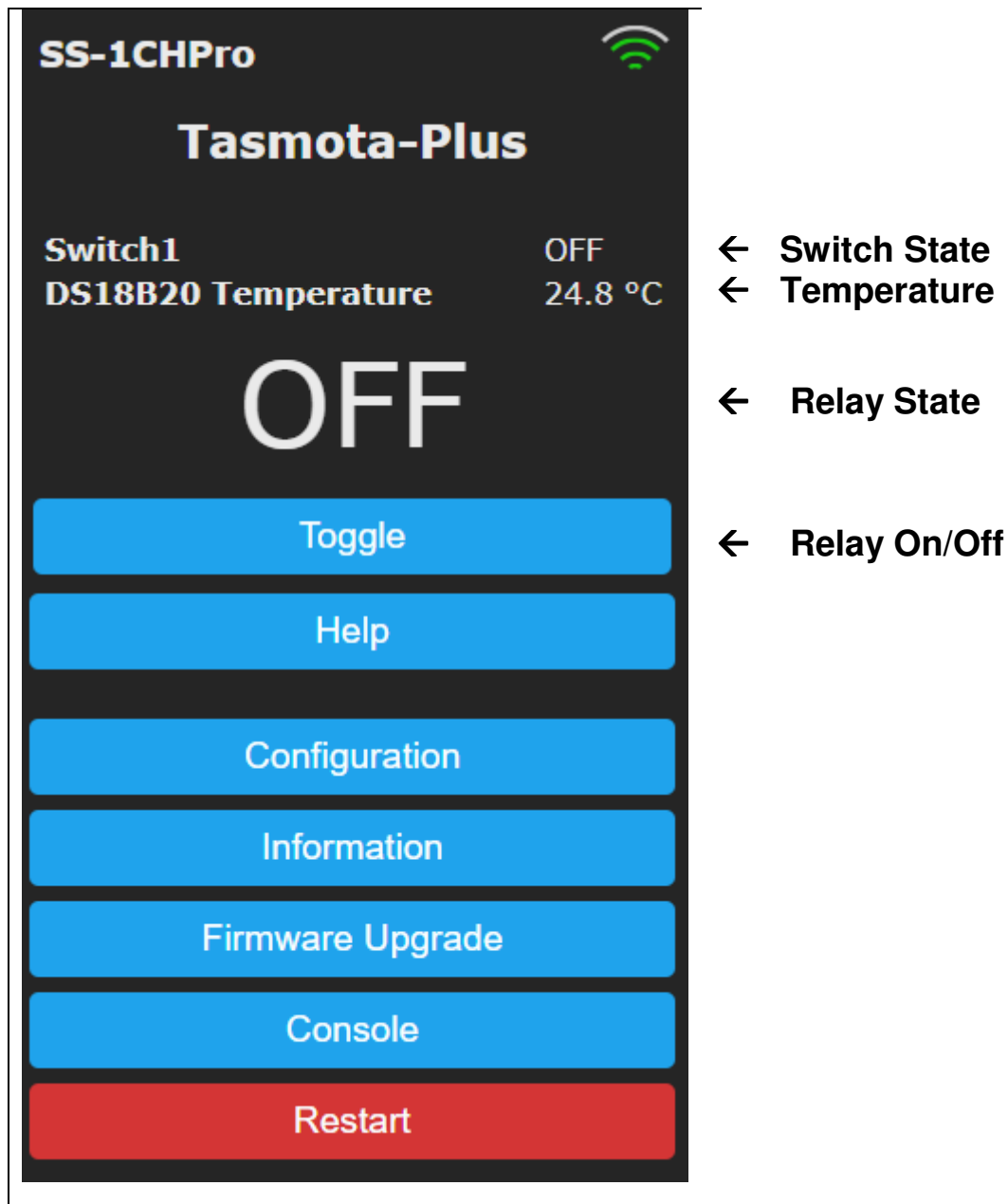


Figure 1 - The Tasmota Web Interface is available from any Web browser

Simply use any web browser to open the web page

`http://<device.ip.address>/`

See documentation here: <https://tasmota.github.io/docs/WebUI/>

Power Options

The DC-IN port is a common 2.1mm x 10 mm , centre-positive jack.

There are two common installation scenarios:

Near a Power Point (GPO)

In this case, simply use any DC power supply between 24 and 60V that has a suitable DC plug.

Example suitable power supply: <https://ubwh.com.au/PSU-24VDC-24W>

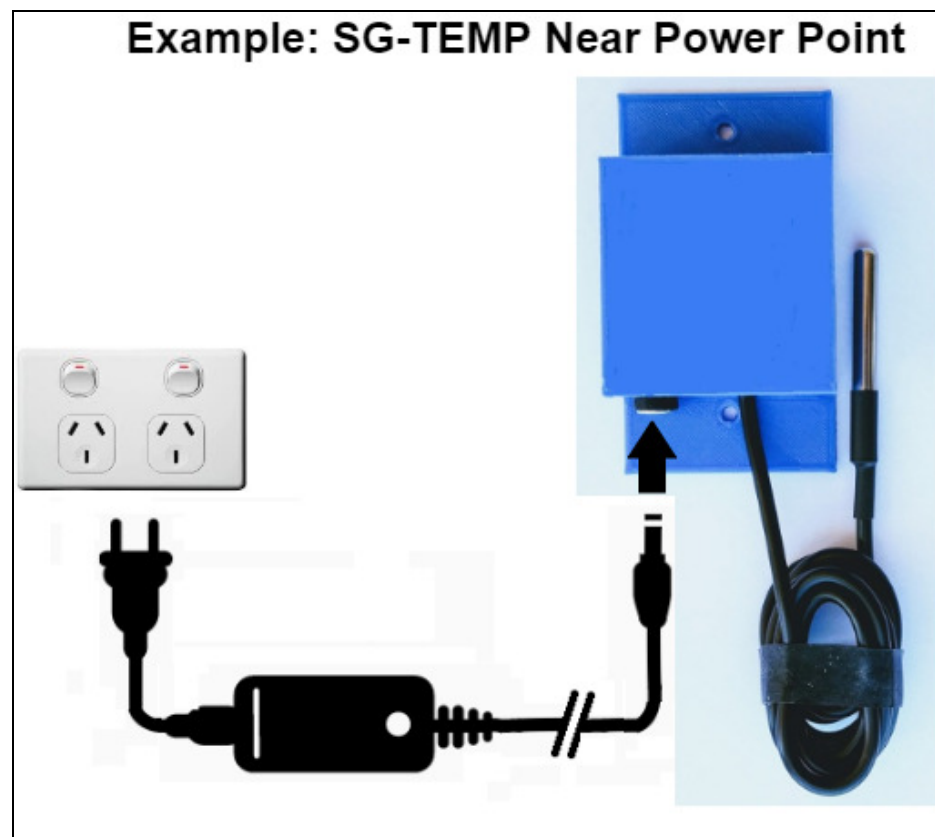


Figure 2 – Powering the SG-TEMP from a nearby power point

Far from a Power Point (GPO)

In this case, one recommendation is to use a common passive PoE⁷ power supply that provides either 24 or 48 VDC, with widely available LAN⁸ cable.

One end of the LAN cable can be terminated with either a:

- **DC Plug**, soldered onto the conductors carrying the PoE voltage, or
- **RJ45 LAN plug**, with a PoE Splitter⁹ to break out the DC to a 2.1mm plug.

Example suitable power supplies:

<https://ubwh.com.au/POE-24-12W>

<https://ubwh.com.au/POE-48-24W>

<https://ubwh.com.au/WI-PS306GF-UPS-V2>



Figure 3 – Powering the SG-TEMP over distance with passive PoE

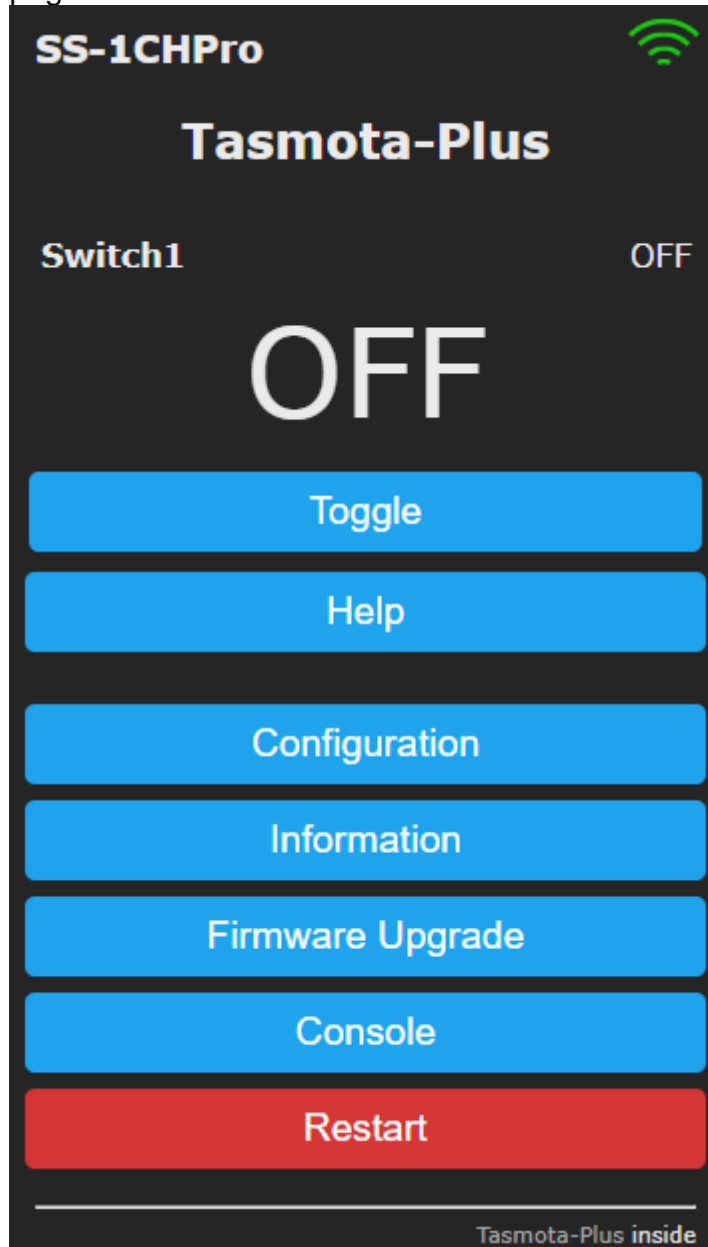
⁷ Power-Over-Ethernet

⁸ e.g. CAT5e

⁹ e.g. <https://ubwh.com.au/POE-P-SPLIT>

Getting Started

1. Power the SG-TEMP.
2. Follow the instructions in the **SS-1CHPro User Guide**¹⁰ until you see this page.



¹⁰ <https://tinyurl.com/28nybtzx>

Enabling the SG-TEMP Sensor

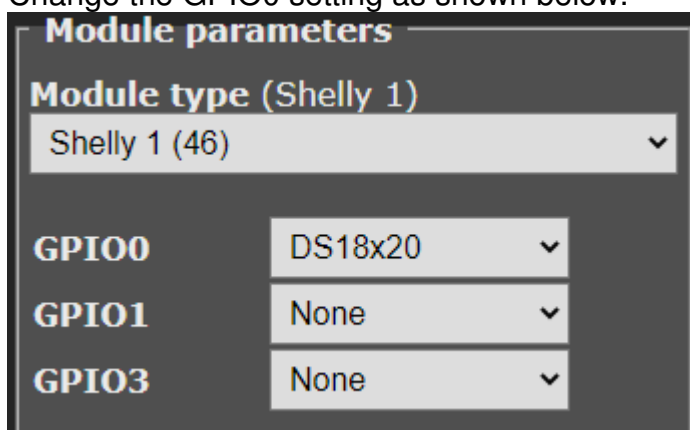
1. From the SS-1CHPro main page, click



then



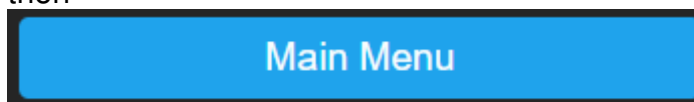
2. Change the GPIO0 setting as shown below.



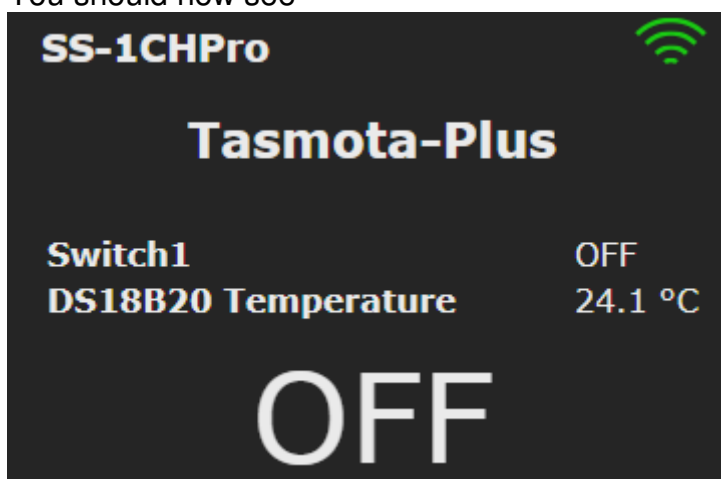
3. Click



then



4. You should now see



Relay Control based on Temperature

Using the Tasmota Rules¹¹ feature, the relay can be controlled by temperature.

This is done by setting two values:

- Relay ON set-point temperature
- Relay OFF set-point temperature

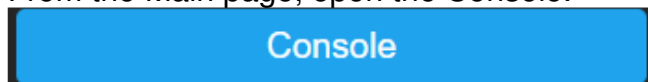
Note: Do not set these two values the same, as the relay may switch ON/OFF rapidly.

In the steps below you will define 3 *Rules*. These rules remain active after a reboot or power cycle.

Hint: To avoid typing mistakes, it is recommended to Copy (from the PDF file of this document) and Paste (to the Console input line).

Step 1 – Common Configuration

From the Main page, open the Console.



Enter these 2 commands, followed by the [Enter] key on your keyboard.

```
RULE2 ON system#boot DO BACKLOG VAR3 %VAR1%; VAR4 %VAR2%  
ENDON  
BACKLOG RULE1 ON; RULE2 ON; RULE3 ON
```

Continue to next page.

¹¹ <https://tasmota.github.io/docs/Rules/>

Two cases are given below. Follow Step 2a or Step 2b:

Step-2a – Relay ON when Temperature is too HIGH

Enter these 2 commands, followed by the [Enter] key on your keyboard.

<code>RULE3 ON DS18B20#temperature>%VAR4% DO BACKLOG POWER1 ON; VAR4 1000; VAR3 %VAR1% ENDON</code>
<code>RULE3 + ON DS18B20#temperature<%VAR3% DO BACKLOG POWER1 OFF; VAR3 -1000; VAR4 %VAR2% ENDON</code>

Step-2b – Relay ON when Temperature is too LOW

Enter these 2 commands, followed by the [Enter] key on your keyboard.

<code>RULE3 ON DS18B20#temperature>%VAR4% DO BACKLOG POWER1 OFF; VAR4 1000; VAR3 %VAR1% ENDON</code>
<code>RULE3 + ON DS18B20#temperature<%VAR3% DO BACKLOG POWER1 ON; VAR3 -1000; VAR4 %VAR2% ENDON</code>

Step 3 – Changing the Set-point Temperatures

The set-point temperatures are set in RULE1.

<p>Hint: VAR2 must be higher than VAR1 e.g. VAR1 21 VAR2 22 e.g. VAR1 -6 VAR2 -5</p>
--

To change the set-point temperatures to (e.g.) 21 & 22, enter these 2 commands, followed by the [Enter] key on your keyboard.

<code>RULE1 ON system#init DO BACKLOG VAR1 21; VAR2 22 ENDON</code>
<code>RESTART 1</code>

http:// Command Interface



Simple commands as below will (e.g.) turn the Relay ON.

Note: %20 in a URL = Space character

From Web Browser

`http://<device.ip.address>/cm?cmnd=power1%20on`

From Windows or Linux command/terminal window

`curl http://<device.ip.address>/cm?cmnd=power1%20on`

From a Windows Batch file (*.bat file)

`curl http://<device.ip.address>/cm?cmnd=power1%%20on`

Note: need double % characters if in a batch file

From a PHP script (*.php file)

```
file_get_contents(  
    'http://<device.ip.address>/cm?cmnd=power1%20on');
```

You can retrieve the **Switch and Temperature Status** as below.

`http://<device.ip.address>/cm?cmnd=status%208`

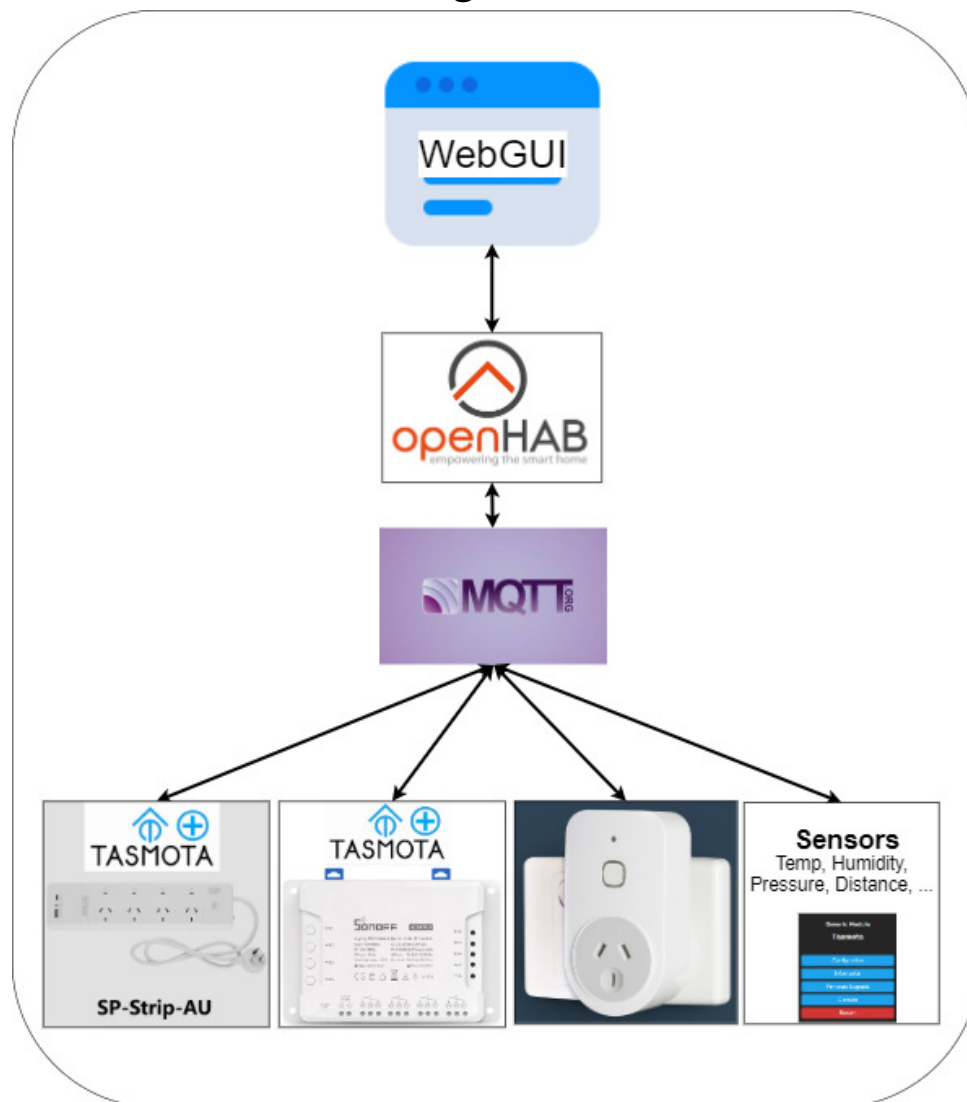
A typical JSON response looks like:

```
{  
    "StatusSNS": {  
        "Time": "2021-01-18T02:06:53",  
        "Switch1": "OFF",  
        "DS18B20": {  
            "Id": "011937A99651",  
            "Temperature": 24.4  
        },  
        "TempUnit": "C"  
    }  
}
```

More information:

<https://tasmota.github.io/docs/Commands/#management>

Centralised Monitoring & Control



While this device can operate 100% stand-alone, it can also be monitored and controlled, along with multiple other devices, from a single management platform.

One popular management platform is **openHAB**¹².

In simple terms:

- MQTT compatible devices (e.g. Tasmota) connect to an **MQTT Broker**¹³.
Status information sent TO the MQTT broker.
Commands received FROM the MQTT broker.

¹² <https://www.openhab.org/> (Freeware, Open source)

¹³ <https://mqtt.org/> (Freeware, Open source)

- **openHAB** also connects to the MQTT broker.
Status information received FROM the MQTT broker.
Commands sent TO the MQTT broker.
- Users interact via web pages (WebGUI)

WebGUI Interfaces

openHAB supports a number of User Interfaces (UIs). Each UI is highly customisable.

The images below show example visualisations.

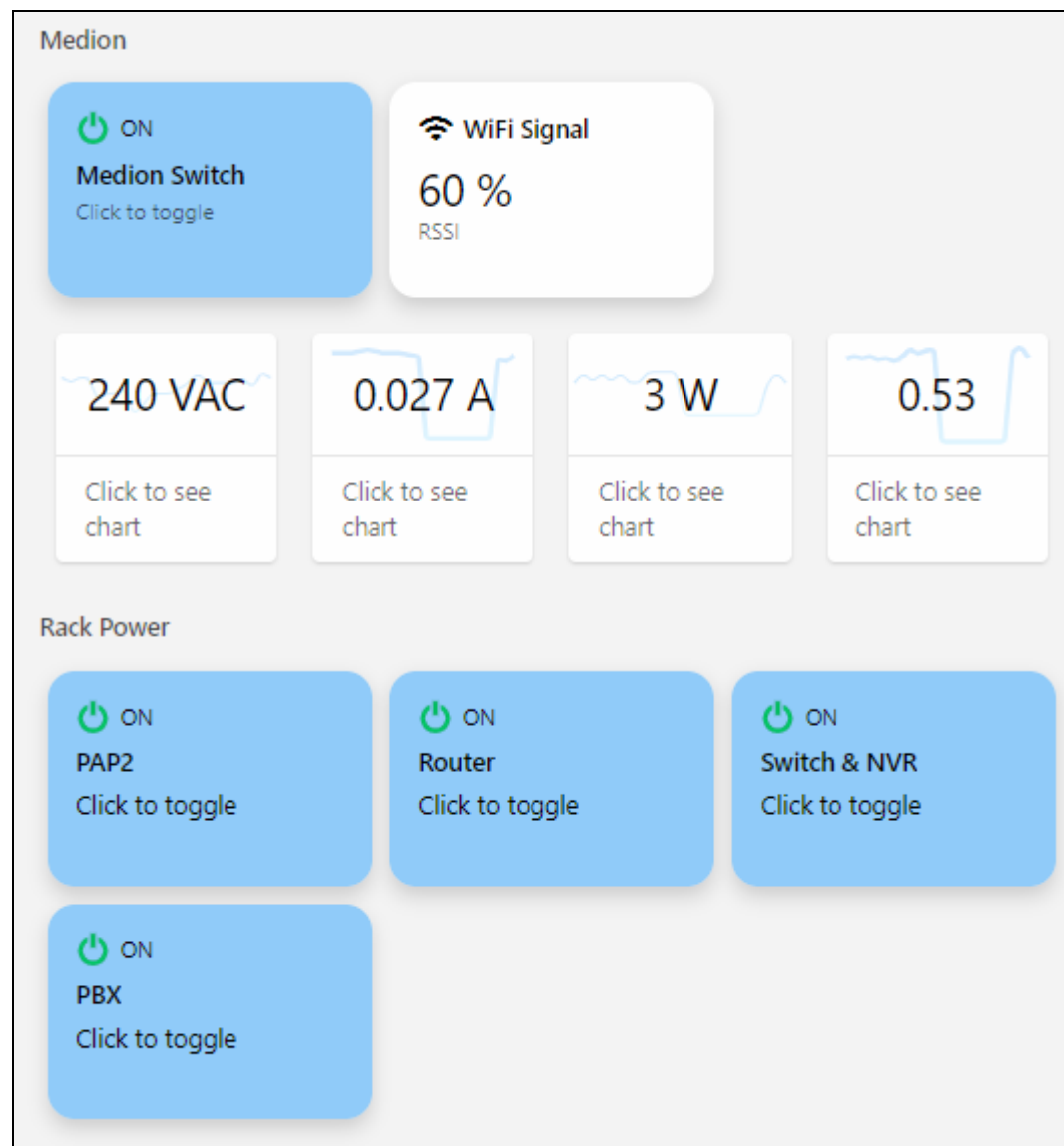


Figure 4 - Classic UI. Mobile friendly.



Figure 5 - Basic UI. Mobile friendly.



Figure 6 - Panel UI. Ideal for touch screens.

openHAB Channel Definition (Example)

Below is shown the channel definition for an SG-TEMP correctly integrated into openHAB.

Values that will be different for each installation have been replaced with XXXXXX.

```
UID: mqtt:topic:XXXXXX
label: SG-TEMP
thingTypeUID: mqtt:topic
configuration:
  payloadNotAvailable: Offline
  availabilityTopic: tele/tasmota_XXXXXX/LWT
  payloadAvailable: Online
bridgeUID: mqtt:broker:30e6c58453
channels:
  - id: Temp
    channelTypeUID: mqtt:number
    label: Temp
    description: null
    configuration:
      stateTopic: tele/tasmota_XXXXXX/SENSOR
      transformationPattern: JSONPATH:$.DS18B20.Temperature
      unit: C
  - id: Relay
    channelTypeUID: mqtt:switch
    label: Relay
    description: ""
    configuration:
      commandTopic: cmnd/tasmota_XXXXXX/POWER
      stateTopic: stat/tasmota_XXXXXX/POWER
      off: OFF
      on: ON
  - id: Switch
    channelTypeUID: mqtt:switch
    label: Switch
    description: ""
    configuration:
      stateTopic: tele/tasmota_XXXXXX/SENSOR
      transformationPattern: JSONPATH:$.Switch1
```


openHAB Sample History Plot

The plot below shows an example history from an SG-TEMP sensor immersed in a water tank.

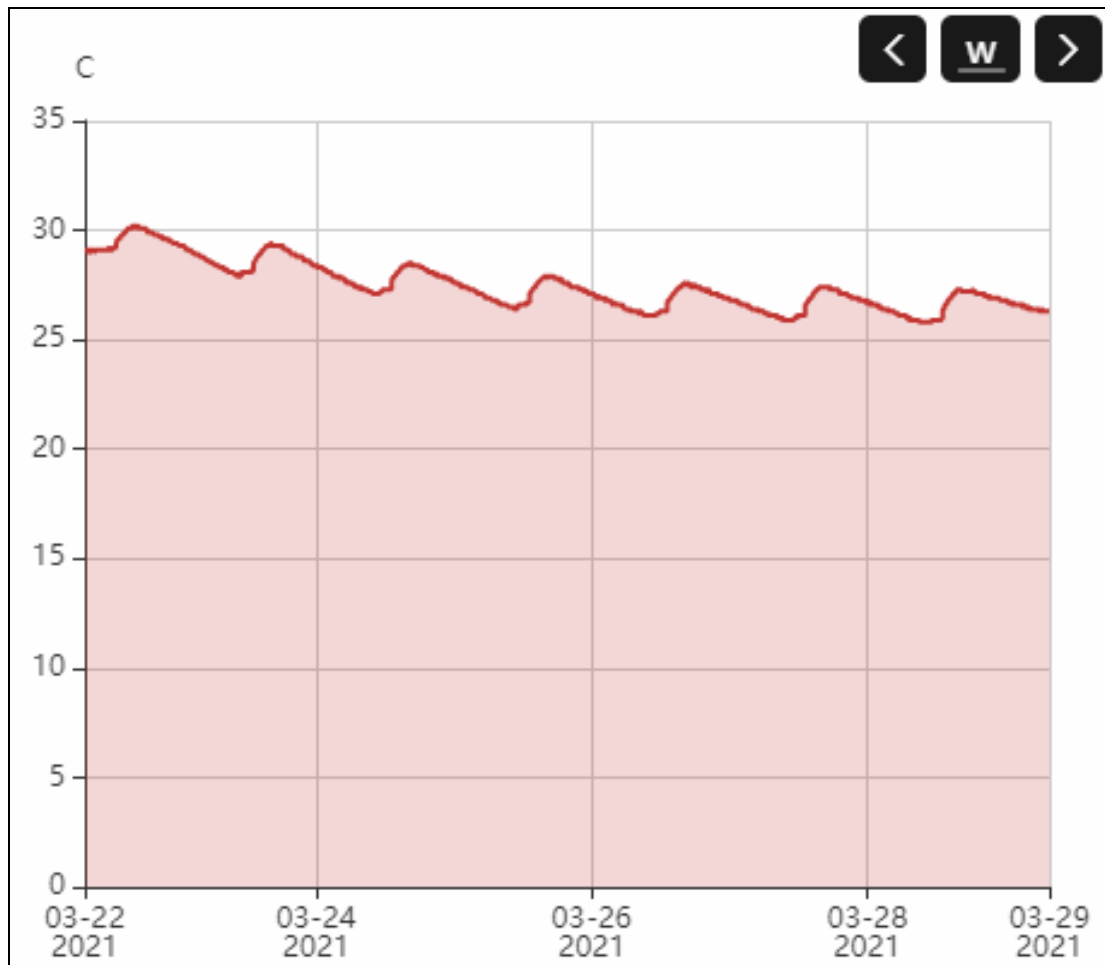


Figure 7 - Tank water temperature over 1 week

Specifications

Temperature sensor	Type: Range ± 0.5 °C Accuracy Waterproof Length Diameter	DS18B20 -55 °C to +125 °C -10 °C to +85 °C Yes (to length of cable) 50 mm 6 mm (6.5 mm over black jacket)
Cable	Length Diameter Waterproof	900 mm 3.7 mm Yes
Enclosure	Material Dimensions Weight Mounting holes	PLA 90 x 50 x 27 mm (W x L x H) 60 g (including SS-1CHPro) 2 x 3 mm dia (71 mm spacing)
Power supply	DC Only Connector Power consumption	24 to 60 V DC (unregulated) 2.1 x 10 mm , Centre positive < 1W (= 40 mA @ 24 V)
WiFi	Range Standards	20 m (Typical, no walls) 10 m (Typical, walls) 802.11b/g/n 2.4 GHz