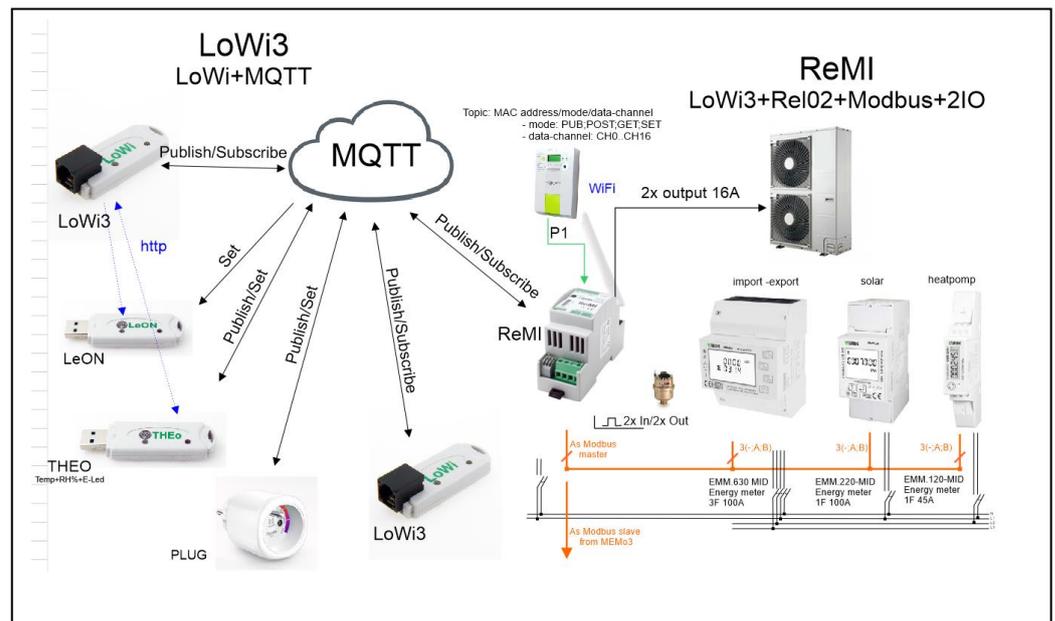


LOWI3, REMI, MEMO3 MQTT V1.4

MQTT to connect 2-WIRE loggers.



Objective

In this short guide we explain how to set the option http and especially MQTT in both the LoWi, LeON, THEo, PLUG, ReMI and MEMo3 module. For the link to external applications, we briefly discuss how MQTT Topics and the pay-loads are structured.

1. LOWI3 CONFIGURATION
2. HTTP CONNECTIONS
3. MQTT CONNECTIONS
4. ATTACHMENTS
5. TOPICS for OEM applications
6. Example link to the Android APP 'IoT MQTT'

1. LOWi3 (ReMI) CONFIGURATION:

Click on 'LOWi' at the top and you will enter the setup

The screenshot shows the LoWi3 configuration interface. At the top, there are tabs for 'TABEL', 'PRINT', '2-WIRE', and 'LoWi'. The 'LoWi' tab is active. The interface is divided into several sections:

- NETWORK:** telenet-DCA6F, IP: 192.168.0.125, PORT: 8082, RSSI: -78dBm, MAC: f4:cf:a2:59:f0:de, METER ID: ---
- LoWi Login:** LoWi: LOWi, PASS: *****
- HTTP Domain:** HTTP Domain: [empty], DIR: [empty]
- HTTP Login:** HTTP Login: [empty], PASS: [empty]
- UPLOAD INTERVAL:** 24 h, MQTT ENABLE:
- TAAL:** NL, NTP: , TIME ZONE: (GMT+01:00) Brussels, Copenhagen, Madrid, Paris
- UPGRADING:** v20.9.30.4 STOP, SAVE, EXIT
- COST:** (ct): ELEC_H: 25, ELEC_L: 15, GAS: 70, WATER: 520
- WARNING IMPORT MAX:** 1000, WARNING EXPORT MIN: 0
- MQTT:** MQTT Broker: mqtt.fespi.io, MQTT Login: [empty], MQTT Pass: [empty]

NETWORK – PASS – IP – PORT : is set via **CONFIG PORTAL** (see separate chapter)

LoWi Login – PASS:

default: login: LOWi and password: 20.1.1.1

This is your login and password to gain access when you are out of the house. But for this, a 'port-forwarding' must be done in your router! Within the local network you can easily adjust login password

HTTP Domain – DIR – HTTP Login – PASS: you can send all logs to an HTTP server for backup. The time is determined with **UPLOAD INTERVAL** (from 1 minute to 24h)

With '**TEST**' you can directly test the settings with the chosen HTTP server. See separate chapter for this application.

MQTT ENABLE: upon activation, one can now send or receive data via an 'MQTT BROKER' See separate chapter

LANGUAGE: choice of NL-FR-EN-DU

NTP: the internal clock is synchronized every hour with an external 'NTP server'. **If it turns green, there is a connection to the internet.**

TIME ZONE: this will be set according to your location.

UPGRADING: sometimes there are improvements or 'bugs'. By selecting 'START', the LOWi will check whether

there are updates and possibly perform them (time approx. 1 minute)

STYLE B/W: NEW!! the layout has 2 styles and can be set to your taste.

The screenshot shows the 2-WIRE configuration interface. At the top, there are tabs for 'TABEL', 'PRINT', '2-WIRE', and 'LoWi'. The 'LoWi' tab is active. The interface is divided into several sections:

- NETWORK:** telenet-DCA6F, IP: 192.168.0.88, PORT: 8090, RSSI: -62dBm, MAC: cc:50:63:e3:a5:a6, METER ID: 15AG1100000006
- LoWi Login:** LoWi: andre, PASS: ****
- HTTP Domain:** HTTP Domain: http://ps-10.danteck.nl, DIR: [empty]
- HTTP Login:** HTTP Login: [empty], PASS: [empty]
- UPLOAD INTERVAL:** 24 h, MQTT ENABLE:
- TAAL:** NL, NTP: , TIME ZONE: (GMT+01:00) Brussels, Copenhagen, Madrid, Paris
- UPGRADING:** v20.10.1.2 STOP, SAVE, EXIT
- COST:** (ct): ELEC_H: 25, ELEC_L: 15, GAS: 70, WATER: 520
- MQTT:** MQTT Broker: mqtt.fespi.io, MQTT Login: [empty], MQTT Pass: [empty]

SAVE: all changes are saved after confirmation with '**adminLOWi**' or '**adminREMI**'

The screenshot shows the LoWi3 configuration interface with a login dialog box open. The dialog box has the title '192.168.0.125:8082 meldt het volgende' and the text 'ENTER LOGIN'. The username 'adminLOWi' is entered in the input field. There are 'OK' and 'Annuleren' buttons.

EXIT: back to main screen (DASHBOARD)

COST: here you can enter (approximate) cost prices of your energy (electricity, gas, water)

WARNING IMPORT MAX: if you have installed a '**LEON**' or '**THEO**', it can give a 'flash warning' when your total consumption of the P1 meter exceeds this limit.

Flashing stops when consumption falls below 10% of this setting. (see more explanation for the module '**LEON**' or '**THEO**')

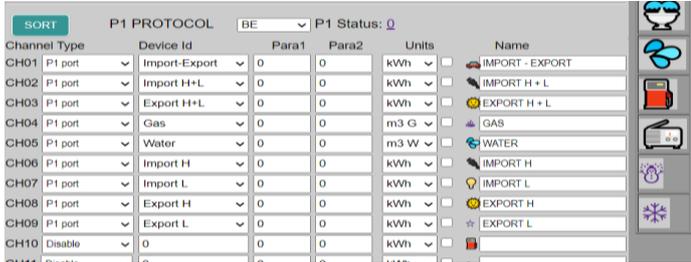
WARNING EXPORT MIN: if you have installed a '**LEON**' or '**THEO**', it can give a 'green flash warning' when you are going to export energy (solar panels) above this limit.

Flashing stops when the export is zero.

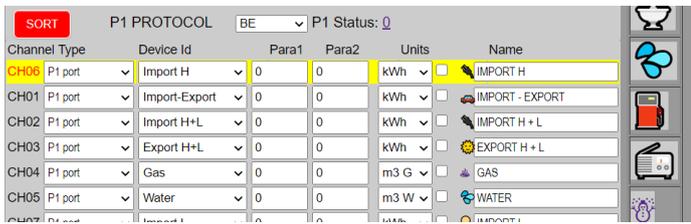
P1 PROTOCOL: this will be set according to your digital meter (Belgium = BE, NL 2.0-4.0-5.0)

CHANNELS: Up to 16 log channels.

Upon delivery, the first 9 channels are set to the P1 port with protocol 'BE'

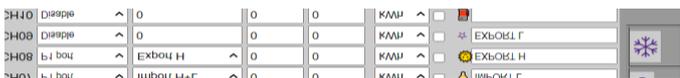


Click on 'SORT' to change the order if necessary: Left click on the desired channel, hold click, and shift to the desired position. This operation can be repeated until all channels are in the desired order.



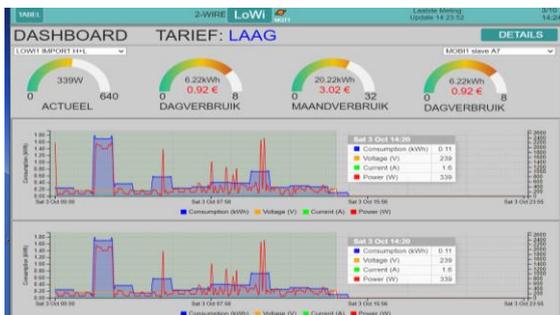
Then click 'SORT' back and 'SAVE' to keep this setting.

If you do not want to use all channels (e.g. only IMPORT H+L and GAS) you can disable the channels that are not desired by selecting 'Disable'.

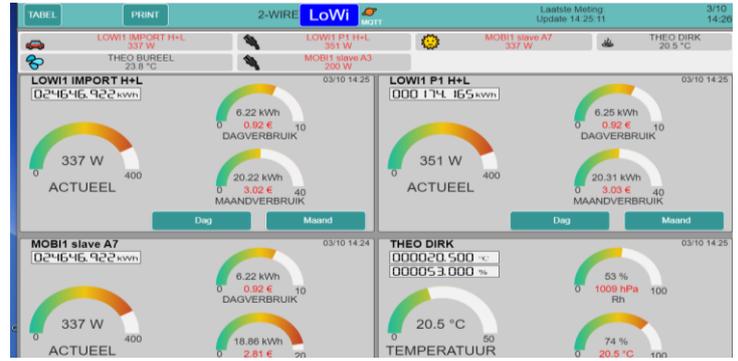


IMPORTANT! With 'Name' a name must be entered (min 3 characters, max 16 characters), otherwise this channel will NOT be shown or logged!

After 'SAVE' and 'EXIT', it may take a few seconds before the custom **DASHBOARD** becomes visible.



Press 'DETAILS' to view all selected channels. Clicking back on 'LOWI' is the 'DASHBOARD'.



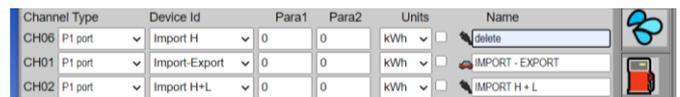
The P1 port configuration therefore does not require any special knowledge to configure it.

By default, all 9 channels of the P1 port are active.

At the first commissioning, all consumption (day-month-annual consumption) would be zero.

If this is not the case, the following procedure can be applied:

CLEAR LOGS: Go to the configuration and name the channel to be deleted 'delete'.



Press 'SAVE' + login (adminLOWI) All logs of this channel will now be completely erased. (approx. 4-5 seconds erasure time).

Afterwards, the original name will automatically be filled in again and the consumption of this channel will be set to ZERO. (NOT THE METER READINGS !)

When multiple channels need to be cleared, it is best to do this procedure one by one, to avoid too long a procedure time. This could lead to a 'time-out' for the web server and possibly even to 'crashing'. If this is the case (web page is no longer visible after 're-load'), one must disconnect the LOWI and feed it back. (hard reset)

So much for the P1 port. However, the LOWI can log more than just the P1 port!

IMPORTANT: Do not use comma's, special characters in the input fields because this might corrupt the database and the you have to go to factory settings to restore.

2. HTTP CONNECTIONS

2.1. Connect LEON-THEO with LOWI or ReMI via HTTP:

The status of the P1 port can also be shown visually with an RGB LED via the 'LeON' or THEO' module.

In the redirect page

The LeON or THEO gets the same IP address and port number of the **LOWI** module in the WiFi configuration of this module.

NOTE Mac address for latere configuration !!!

THEO Mac f4cfa25a113f

telenet-DCA6F(B) 🔒 100%

TelenetWiFiFree 56%

telenet-DCA6F 🔒 52%

DIRECT-4HM2020_Series 🔒 48%

SSID:

Password:

HTTP - MQTT:

IP addr LoWi: Port: CH 1-16:

SUBNET:

GATEWAY:

In case of THEO module (temperature measurement), the selector '**HTTP-MQTT**' must be set to '**HTTP**', and a channel number **CH(1-16)** must be entered. **E.g. 15**

(see later) Press '**save**'.

THEO f4cfa25a113f Configuration Saved

Connect LoWi to SSID: telenet-DCA6F
IPaddress: 192.168.0.88:8090

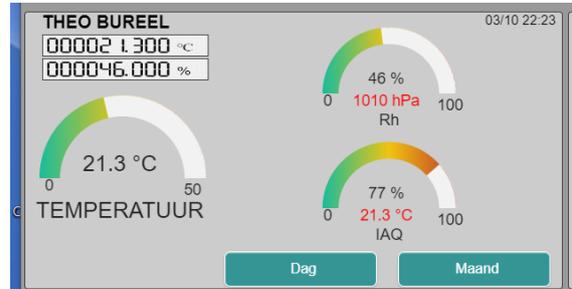
In the LoWi3 setup page

With the linked LOWI, with the same IP address as entered in the WiFi configuration of the THEO, a channel must now be created with the selection '**THEO**'. Under '**Para1**' the chosen channel number is entered (**e.g. 15**), AND the '**Units**' must be on '**IAQ**'. Don't forget to give a name to this channel ! (blank name = not active !)

If there are several THEOs, then **each THEO MUST get a unique channel nr (1-16) !!**

CH10	THEO	0	15	0	IAQ	<input type="checkbox"/>	THEO CH15
CH11	Disable	0	0	0	kWh	<input type="checkbox"/>	

One can now log the temperature, humidity and (indicative) the air quality and air pressure.



3. MQTT CONNECTIONS:

A major change compared to the LOWIv1 is the addition of 'MQTT'.

Now you can exchange all data between multiple LOWIs, THEOs, ReMI's, MEMo3's, PLUG's via an 'intermediate station'.

This intermediate station is called the '**BROKER**'. The user must first create an **account** with a broker of his choice, or a broker of 2-WIRE (with a small activation price).

What are the advantages over HTTP:

Each module, which has **the same account**, can communicate with each other, anywhere in the world.

(of course, all modules must have access to the internet)

There should be no port forwarding in your router because MQTT works in real-time and bi-directional in data traffic. (With HTTP, the LOWI can not reach the THEO or LeON directly, and have to wait for the THEO or LeON data to be sent to the LOWI. This also only works on the local Wi-Fi network and we have noticed that some Wi-Fi routers block this http communication.)

Data storage can also be done in the broker, as a backup of your data.

The biggest advantage of MQTT is that you can exchange data between modules in all possible places.

3.1. WiFi Portal THEO - PLUG - LeON:

In the WiFi portal now select '**MQTT**'

- Enter the address of your **Broker**.

- Enter the obtained '**Token**'. (= login)
- Possibly also **Password**.

Note. Both HTTP **and** MQTT protocols can also be applied simultaneously in THEO !

Then select '**BOTH**' and enter both the IP address and port number of the LOWI and the details for the Broker.

One could now use HTTP when all modules are locally on the same WiFi network, but at the same time send a 'backup' of the data to the Broker.

(this setting may not be interesting in most cases and '**MQTT**' would be preferable)

Remember that INTERNET must always be present on your Wi-Fi network!

Don't forget '**SAVE**'.

THEO f4cfa25a113f Configuration Saved

Connect LoWi to SSID: telenet-DCA6F
IPaddress: 192.168.0.88:8090

3.2. MQTT ENABLE in LOWI en ReMI:

The LOWI3 and ReMI must also receive the same url address from the Broker and Token (= login) as the THEO, LeON or PLUG.

Now check '**MQTT ENABLE**' in the LOWI configuration.

You can now enter the address, token and password of your broker.



As already mentioned, with MQTT you can receive data from the THEO, PLUG, ReMI, MEMo3 or LOWI3 at any location.

LoWi sends P1 port data to the MQTT broker every minute and you can read it via your own application. However, if you want to see an update of this data faster, you can check 'P1 FAST PUBLISH'. Now all data from the P1 port will be sent to the broker every 10 seconds (= PUBLISH)

Next to it is a check mark with 'BU CONFIG'. Once your configuration is ready, you can send this configuration to the broker as a backup. Then check 'BU CONFIG' and press 'SAVE'.

Afterwards, you can go back to the last saved configuration in case of a fatal 'crash' or when the new configuration was not good.

NOTE: The logs will NOT be modified when you call up the previous configuration.

So if there are changes in the type of the channels, then possibly wrong data can be shown (e.g. CH01 was temperature IAQ, and has since been changed to kWh... this gives problems !

This function can only work if you were already connected to the broker.

When you enter the broker's details for the first time, the connection will be made about 10 seconds AFTER you have pressed 'SAVE'. This can be recognized by the symbol next to 2-WIRE LOWI:

No connection



Connection in order

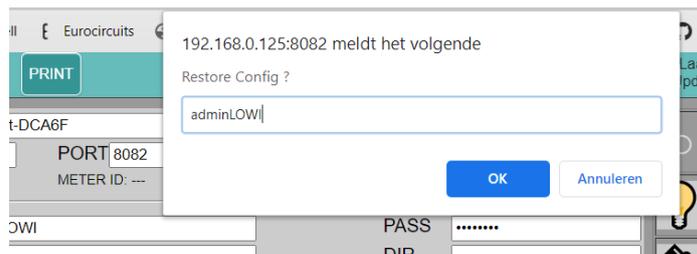


In BRIGHT and in DARK mode.



Retrieve last saved configuration:

Go to 'configuration' and press the MQTT symbol, fill in 'adminLOWI' and press OK.



After about 5 seconds you can reload the page or press 'EXIT'. Your previous configuration is now active.

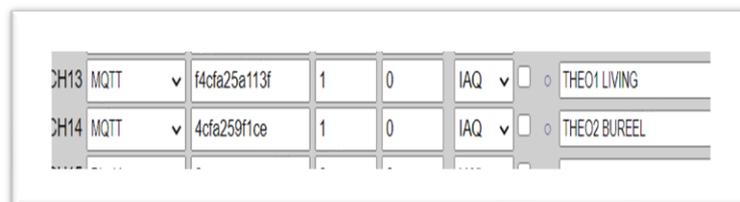
Make sure that you do not activate incorrect or incorrect configuration. After this process, you can't go back to your previous version!

3.3. Configuratie LOWI3 to ReMI:

Once your broker is connected, you know that these settings are in order. Now you can create the desired channels with MQTT.

THEO: read in via MQTT

In the WiFi configuration, THEo must have the same broker and token as this LOWI.



Selection is now 'MQTT', and under 'Device Id' comes the Mac address of the THEO.

The channel number is ALWAYS 1, and is entered under 'Para1' ('Para2' = 0)

The 'Units' MUST be 'IAQ'. Don't forget to enter a name !

This can be repeated if several THEOs, regardless of the location.

LOWI (of ReMI) : connect via MQTT

Indeed, **one can now connect multiple LOWI/ReMI via MQTT!**

Suppose LOWI1 (Mac cc50e3e3b5a6) has a P1 port connected. Then LOWI2 can also obtain this data so that 2 P1 ports can be shown.

Under '**Channel Type**' you select '**MQTT**'. Under '**Device Type**' you enter the **Mac address** of LOWI1.

Under '**Para1**' comes the desired channel number (1-16) from LOWI1. (e.g. P1 IMPORT H+L = CH7)

Channel Type	Device Id	Para1	Para2	Units	Name	
CH01	MQTT	cc50e3e3b5a6	7	0	kWh	LOWI1 IMPORT H+L
CH02	MQTT	cc50e3e3b5a6	10	0	kWh	LOWI1 P1 H+L

Just make sure you don't make loops with multiple LOWIs.

Suppose LOWI2-CH5 retrieves data from LOWI1-CH1. Then LOWI1 may not retrieve the same data from LOWI2-CH5.

However, LOWI1 can retrieve other data from LOWI2 (e.g. THEOs), as long as they were not published by himself.

WiFi SMARTPLUG: read in via MQTT

In the WiFi configuration, it must have the same broker and token as this LOWI.

Selection is now '**MQTT**', and under '**Device Id**' comes the **Mac address** of the PLUG.

The channel number is **ALWAYS 1**, and is entered under '**Para1**' ('**Para2**' = 0)

The '**Units**' **MUST be 'kWh'**. Don't forget to enter a name !

This can be repeated if there are multiple PLUG's, regardless of the location.

CH01	P1 port	Import H+L	0	0	kWh	IMPORT nr18
CH02	P1 port	Gas	0	0	kWh	GAS nr18
CH03	MQTT	f008d1d98b20	1	0	kWh	wifiplug nr12
CH04	MQTT	f4cfa259ee0e	1	0	IAQ	Temp nr18
CH05	Disable	0	0	0	kWh	

MEMo3: read in via MQTT

And indeed, one can **also connect ReMI, MEMo3 with LoWi3, via MQTT!**

Suppose MEMo3 (Mac 98f4ab275424) has an EMM.630 and on log channel 4 is the yield of the solar panels 'Solar Nr 12'.

Under '**Channel Type**' you select '**MQTT**'. Under '**Device Type**' you enter the **Mac address** of MEMO3.

Under '**Para1**' comes the desired log channel number (1-64) of MEMO3. (e.g. Solar F1 = CH4)

Channel Type	Device Id	Para1	Para2	Units	Name	
CH01	P1 port	Import-Export	0	0	kWh	IMPORT - EXPORT
CH02	P1 port	Import H+L	0	0	kWh	IMPORT H + L
CH03	P1 port	Export H+L	0	0	kWh	EXPORT H + L
CH04	MQTT	98f4ab275424	4	0	kWh	SOLAR Nr12
CH08	MQTT	f4cfa259f1ce	1	0	IAQ	TEmp Nr12 keuken
CH06	MQTT	f4cfa259ee0e	1	0	IAQ	Temp Nr 18

Conversely, **MEMo3** can also 'subscribe' on MQTT meters (see manual MEMo3), condition is the same broker and account.

Create a new channel and select the type '**MQTT**' as a module. The '**MODE**' is in function of the paired device:

- P1 port: P1
- WiFi plug: BI stable
- THEo: AirQ
- Modbus: Energy

The screenshot shows the MQTT configuration interface with three channels:

- CH 0:** MQTT, BiStabiel, VOLT, CURRENT, POWER. Device ID: DAG MAAND. Address: 0, SubAddress: 0. MQTT-SN: ac67b211cfd0, CHnr: 1. Name: WIFI Plug E-Header.
- CH 8:** MQTT, AirQ, ROOM TP, HUMID, VOC. Device ID: DAG MAAND. Address: 0, SubAddress: 0. MQTT-SN: f4cfa259ee0e, CHnr: 1. Name: THEo Nr 18.
- CH 7:** MQTT, P1 port, VOLT, CURRENT, POWER. Device ID: DAG MAAND. Address: 0, SubAddress: 0. MQTT-SN: f4cfa259f19e, CHnr: 2. Name: LoWi 12 imp h+l.

Modbus address and Sub address are zero here

For each MQTT channel, each mac address at '**MQTT SN**', and enter the log channel number of the linked meter.

4. ATTACHMENTS:

4.1 WiFi CONFIG PORTAL:

Bring the LOWI-LEON-THEO-ReMI into "access point" mode by placing it under tension.

The blue LED will light up.

Within 3 seconds, press and hold the **RESET button** until the LED stops blinking and is constantly lit.

On your laptop, go to the WiFi settings:

There would now be an access point with the name: **2WIRE-LOWI, 2WIRE-LEON, 2WIRE-THEO** or **2WIRE-REMI**.

Click this name. When it is the first time, you will have to enter a password:

This is **adminLOWI, adminLEON, adminTHEO** or **adminREMI** according to the type of device.

After a few seconds, a screen would appear in your web browser, with the configuration of this device.

Depending on your operating system (windows, mac, android...) it may sometimes not be possible that the control panel is automatically displayed.

When it is shown that the device is connected, but still shows no configuration, you have to type **8.8.8.8** in the toolbar in your web browser.

Now the control panel will appear.

Select WiFi configuration and the screen below will appear (example THEO)

THEO MAC T4CTAZ5A113T

telenet-DCA6F(B) 100%
 TelenetWiFiFree 56%
 telenet-DCA6F 52%
 DIRECT-4HM2020 Series 48%

SSID: telenet-DCA6F
 Password: ABCDEF0123456789ABCDEF0123

HTTP - MQTT **BOTH**

IP addr LoWi: 192 168 0 88 Port: 8090 CH 1-16 15
 SUBNET: 255 255 255 0
 GATEWAY: 192 168 0 1

MQTT Broker: broker.be
 MQTT Token: testtesttest.....
 MQTT Pass:

save

After the configuration is complete, and 'save' has been made, you need to return the WiFi setting to your normal WiFi network. (the LED on the device must be switched off with a correct setup and reboot)

5. MQTT TOPICS for OEM applications:

For developers, it can be interesting to go deeper into the use of MQTT.

What is needed are the '**TOPICS**' and the '**payload**'. This allows you to create applications via self-developed apps.

At the moment we have the following products that support MQTT:

- **LOWIv3-REMIv1**: all active channels are published to BROKER every minute. (publish)
Topics: MacAdr/PUB/CH0... CH16
- **MEMO3**: All active channels are published to BROKER every minute. (publish)
Topics: MacAdr/PUB/CH0... CH63
- **LEON**: publishes all the data every minute. (publish) Topics: MacAdr/PUB/CH1
- **THEO**: publishes all available data every minute. (publish) THEO can receive status RGB led (subscribe)
Topics:
 - MacAdr/PUB/CH1
 - MacAdr/GET/RGB/CH1
 - MacAdr/SET/RGB/CH1
- **PLUG**: publishes all available data every minute. (publish) Plug can receive status REL
Topics:
 - MacAdr/PUB/CH1
 - MacAdr/GET/REL/CH1
 - MacAdr/SET/REL/CH1

5.1. DATA PUBLISH:

All active channels are 'published' **every minute (PUBLISH)** in **JSON** format and with the following topic: **MAC/PUB/CH1 ... CH63**.

E.g. Mac LOWI = cc50e3e3b5a6, channel 12 then the topic is: **cc50e3e3b5a6/PUB/CH12**

De data (payload) is JSON:

- "ident" = MACadres (always 12 characters)
- "device_CH" = channel number 1-16 (up to 63 for MEMO)
- "Name" = name
- "Type" = type
- "Units" = units
- **"T" = rate (1 = High, 0 = Low) only present if Type = P1**

- "U" = voltage (V)
- "I" = current (mA)
- "P" = power (W of liter bij Gas - Water)
- "HC" = hourly consumption (Wh or liter for Gas - Water)
- "MC" = maandverbruik (Wh of liter bij Gas - Water)
- "CH" = meter reading (High Rate if data from P1 port)
- "CL" = meter reading (Low Rate if data from P1 port, otherwise idem CH)

Type, name and units in connection with connected channel:

Abbreviations used and corresponding units:

- ident: Mac Address module (LEON-THEO-LOWI-REMI-PLUG-MEMO3)
- device_CH: channel number in configuration 1-16 (up to 63 at MEMO3)
- name: channel name
- Type:
 - P1 (P1 data)
 - MB (modbus of MQTT data)
 - AIR (temp-humidity ...)
- Units:
 - kWh electricity
 - m3G gas
 - m3W water
 - Miscellaneous (mixed data P1 CH0)
 - IAQ (mixed data THEO)

5.2 Data by type:

The screenshots are from the toolbox on the flespi website.

5.2.1 Data van P1 fast publish:

If '**P1 FAST PUBLISH**' is checked for LoWi3 and ReMI, the P1 port will be published **every 10 seconds**:

Topic: (LoWi of ReMI) **MAC/PUB/CH0**

De data (payload) is JSON:

- "ident" = LOWI of ReMI MACadres (always 12 Characters)
- "device_CH" = 0
- "Name" = P1_ALL
- "Type" = P1
- "Units" = Varies
- "U" = voltage (V)

- "I" = current (mA)
- "PI" = power import (W)
- "PE" = power export (W)
- "T" = rate (1 = High, 0 = Low)
- "CIH" = counter import H (Wh)
- "CIL" = counter import L (Wh)
- "CEH" = counter export H (Wh)
- "CEL" = counter export L (Wh)
- "CG" = counter gas (liter)
- "CW" = counter water (liter)

```
94b97e6a3260/PUB/CHO
qos: 0, dup: -, retain: -, 26/08/2022 16:15:01

{
  ident : "94b97e6a3260"
  device_CH : "0"
  Name : "P1_ALL"
  Type : "P1"
  Units : "Varia"
  U : "237"
  I : "1700"
  PI : "68"
  PE : "0"
  T : "1"
  CIH : "4910622"
  CIL : "6553936"
  CEH : "3870209"
  CEL : "1375789"
  CG : "25"
  CW : "0"
}
```

5.2.2 Data from Log channels CH1... CH16 or CHO... CH 63(MEMo3)

Receive from Modbus of MQTT

- "U" = voltage (V)
- "I" = current (mA)
- "P" = power (W/liter Gas/liter Water)
- "HC" = hour consumption (Wh/liter Gas/liter Water)
- "DC" = day consumption (Wh/liter Gas/liter Water)
- "MC" = monthly consumption (Wh/liter Gas/liter Water)
- "CH" = counter H (Wh/liter Gas/liter Water)
- "CL" = counter L : always 0

```
94b97e6a3260/PUB/CH8
qos: 0, dup: -, retain: +, 26/08/2022 16:10:12

{
  ident : "94b97e6a3260"
  device_CH : "8"
  Name : "P1_EH"
  Type : "P1"
  Units : "kWh"
  T : "1"
  U : "241"
  I : "2140"
  P : "2015"
  HC : "197"
  DC : "197"
  MC : "197"
  CH : "3870100"
  CL : "0"
}
```

Only for P1 and channel = Import H+L and Export H+L:

- "CIHL" = import H + L (Wh)
- "CEHL" = export H + L (Wh)

AIR:

- "Tp" = temperature (°C decimal vb 24.8)
- "Rh" = humidity (%)
- "Pa" = air pressure (hPa approximate)
- "IAQ" = air quality (%) approximate)
- "RGB" = RGB color hex format #RRGGBB (vb. #00FF00)

Note: The RGB LED tracks the status of the P1 port when connected. The following values are valid:

- #990000rood
- #009900groen
- #000099blauw
- #FF0000rood flash
- #00FF00groen flash
- #0000FFblauw flash
- #000000uit

You can manipulate the colour of the THEO via an app:

Topic= MacAdr/SET/RGB/CH1

Dates: as mentioned above, back to status P1 = #FFFFFF

5.2.4 Data of the Plug

Topic: MacAdr/PUB/CH1

Sent every minute or sent in case of status change

- ident : "ac67b211cfd0" Mac address
- device_CH : "1" Channel number
- Name : "PLUG"
- Type : "PLUG"
- Units : "kWh"
- REL : "0" 0=OFF; 1=ON
- U : "237" Voltage (V)
- I : "0" Current (mA)
- P : "0" Power in (W)
- CH : "4047591652" counter H (Wh)
- CL : "0" counter L : always 0

- RSSI : "59" signal strength WiFi in dBm

```
ac67b211cfd0/PUB/CH1
qos: 0, dup: -, retain: -, 23/08/2022 16:41:54
{
  ident: "ac67b211cfd0"
  device_CH: "1"
  Name: "PLUG"
  Type: "PLUG"
  Units: "kWh"
  REL: "0"
  U: "239"
  I: "0"
  P: "0"
  CH: "4047591652"
  CL: "0"
  RSSI: "64"
}
```

Topic: MacAdr/SET/REL/CH1

Change status to 'ON' or 'OFF'

```
c44f3378ced5/SET/REL/CH1
qos: 0, dup: -, retain: +, 26/08/2022 16:44:35
ON
{"userProperties":{"timestamp":"1661525075.418682","cid":"1584409"}}

c44f3378ced5/SET/REL/CH1
qos: 0, dup: -, retain: -, 26/08/2022 16:46:59
OFF
{"userProperties":{"timestamp":"1661525219.286801","cid":"1584409"}}
```

Application example: One can create a fictitious plug on one of the 16 channels of a ReMI. Then create a logic that switches this fictitious plug. An external application can then 'subscribe' to this topic and use this MQTT contact as an entry based on logic in the ReMI module.

```
aabbccddeeff/SET/REL/CH1
qos: 0, dup: -, retain: -, 26/08/2022 17:20:28
ON
{"userProperties":{"timestamp":"1661527228.918995","cid":"1584409"}}

aabbccddeeff/SET/REL/CH1
qos: 0, dup: -, retain: -, 26/08/2022 17:20:33
OFF
{"userProperties":{"timestamp":"1661527233.152799","cid":"1584409"}}
```

Topic: MacAdr/GET/REL/CH1

Receive status 'ON' or 'OFF'

5.2.4 Examples of data received (JSON)

1. LEON: {"ident":"MacAdr","device_CH":"1","name":"NAAM", "Type":"RGB", "Units":"HEX", "T":"rgb val"}

E.g. {"ident":"f4cfa25a9b38","device_CH":"1","name":"living", "Type":"RGB", "Units":"HEX", "T":"#000099"} (T = Tariff, blue, not flash)

{"ident":" f4cfa25a9b40", "device_CH":"1","name":"keuken", "Type":"RGB", "Units":"HEX", "T":"#FF0000"} (rood, flash)

2. THEO: {"ident":"MacAdr", "device_CH":"1","name":"NAAM", "Type":"AIR", "Units":"IAQ", "Tp":"temp val", "Rh":"vocht val", "Pa":"druk val", "IAQ":"IAQ val", "RGB":"RGB val"}

Vb. {"ident":" f4cfa25a9b38", "device_CH":"1","name":"bureel", "Type":"AIR", "Units":"IAQ", "Tp":"23.6", "Rh":"45", "Pa":"1012", "IAQ":"75", "RGB":"#009900"}

(RGB = green, not flash)

Controlling LED on THEO:

topic: BY MAC/SET/RGB/CH1

Data: RGB status as above (24 bit RGB) #000000 = off

3. LOWI3-REMI: all P1 channel:

- Units: according to P1 channels:
- CHO: all P1 meter readings

{"ident":"MacAdr","device_CH":"1","name":"NAAM", "Type":"P1", "Units":"P1 type", "U":"volt val", "I":"curr val", "P":"power val", "HC":"uurverbruik", "DC":"dagverbruik", "MC":"maandverbruik", "CH":"tellerH", "CL":"tellerL", "T":"tarief"}

Vb. {"ident":" f4cfa25a9b74", "device_CH":"1","name":"import H", "Type":"P1", "Units":"IHL", "U":"230", "I":"450", "P":"120", "HC":"145", "DC":"2480", "MC":"12987", "CH":"12345", "CL":"6789", "T":"1"}

CH1: IHL = import H+L, CH2: EHL = export H+L, CH3: G = gas, CH4: W = water, CH5: IH = import H, CH6: IL = import L, CH7: EH = export H, CH8: EL = export L

{"ident":"MacAdr","device_CH":"1", "name":"import", "Type":"P1", "Units":"P1 type", "U":"volt val", "I":"curr val", "P":"power val", "HC":"uurverbruik", "DC":"dagverbruik", "MC":"maandverbruik", "CH":"tellerH", "CL":"tellerL", "T":"tarief"}

Vb. {"ident":" f4cfa25a9b38", "device_CH":"1", "name":"import H", "Type":"P1", "Units":"IHL", "U":"230", "I":"450", "P":"120", "HC":"145", "DC":"2480", "MC":"12987", "CH":"12345", "CL":"6789", "T":"1"}

5.3 SUBSCRIBE:

If you want to receive data with an app, you have to do a 'subscribe':

TOPIC: MacAdr/PUB/CH1... CH16 where MacAdr can be a LOWI, REMI, MEMO3 or THEO.

At THEO, the channel is always CH1.

Data can be filtered with JSON 'deserialize'

Syntax on most apps: \$.xx where xx are the keywords mentioned above (U -I -P etc)

(e.g. at THEO: \$. Tp = temperature, \$. CH = counter H)

5.4 GET: Request day logs:

With an external application, data can be retrieved for daily or monthly graphs and configuration.

ONLY POSSIBLE AT LOWI3 or REMI !

DALogs G

PUBLISH TOPIC: MacAdr/**GET**/Day/CH1... 16, and payload is the number of days back to max 9

E.g. request day log CH4 yesterday :

MacAdr/**GET**/Day/CH4, payload = 1 (1 day back)

SUBSCRIBE TOPIC: MacAdr/**POST**/Day/CH1... 16

The answer (payload) is csv – format, separation '|'

After each receipt of payload, it is advisable to do an '**UNSUBSCRIBE**' so that with each reload this data is not received again (and unsolicited)!

PAYLOAD:
channelNr|year|month|date|units|data1|data2|data3|data4|data1| etc

Total there are 24h*12*4 = 1152 data's (resolution 5'), each data is 32bits

Units: 0 = kWh electr / 1 = m3G gas / 2 = m3W water / 3 = IAQ

Data1 = volt(IAQ) – Data2 = current (Rh) – Data3 = power (t°) Data4 = dagverbruik (luchtdruk)

MAANDLOGS

PUBLISH TOPIC: MacAdr/**GET**/Month/CH1... 16, and payload is the number of months back to max 11

E.g. request monthly log 3 months back :

MacAdr/**GET**/Month/CH4, payload = 3 (3 months back)

SUBSCRIBE TOPIC: MacAdr/**POST**/Month/CH1... 16

The answer (payload) is csv – format, separation '|'

After each receipt of payload, it is advisable to do an '**UNSUBSCRIBE**' so that with each reload this data is not received again (and unsolicited)!

PAYLOAD:
channelNr|year|month|date|units|data1|data2|data1| etc

Total there are 30(31) days*2 = 62 data's (resolution 24h), each data is 32bits

Units: 0 = kWh electr / 1 = m3G gas / 2 = m3W water / 3 = IAQ

With electr: Data1 = daily consumption (Wh), Data2 = daily cost (eurocents)

At IAQ: Data1/65536 = max temp, Data1 % 65536 = min temp

Data2/65536 = max Rh, Data2 % 65536 = min Rh

Examples : (FLESPI)

Example log THEO4 at 23:17

```
{
  "ident": "f4cfa2bf41",
  "name": "THEO2",
  "IAQ": "76",
  "Pa": "1008",
  "RGB": "#000099",
  "Rh": "48",
  "Tp": "18.8",
  "Type": "AIR",
  "Units": "IAQ",
  "channel.id": "32753",
  "device.id": "1082077",
  "device.name": "THEO",
  "9b38",
  "device.type.id": "172",
  "ident": "f4cfa25a9b38",
  "pugin": "THERMO1",
  "THERMO1": "",
  "protocol.id": "39",
  "server.timestamp": "1598733641.607368",
  "timestamp": "1598733597.466124"
}
```

act	timestamp	ident	server.s	Tp	Rh	IAQ	CH	U	CH	Cl	Typ	device.name
Q	31/08/2020 09:14:09	f4cfa2995f1	31/08/2	24.9	39	81	216	240			PI	PI_L
Q	31/08/2020 09:14:09	f4cfa2995f1	31/08/2	20.8	45	78					AIR	THEO MOTT CH14W
Q	31/08/2020 09:14:09	f4cfa2995f1	31/08/2	17.6	50	79					AIR	THEO MOTT CH15L
Q	31/08/2020 09:14:09	f4cfa2995f1	31/08/2	21.7	45	78					AIR	THEO MOTT CH15L
Q	31/08/2020 09:14:09	f4cfa2995f1	31/08/2	24.9	39	81					AIR	THEO IME
Q	31/08/2020 09:14:09	f4cfa2995f1	31/08/2	20.9	53	74					AIR	THEO MOTT CH15W
Q	31/08/2020 09:14:09	f4cfa2995f1	31/08/2				31	240	44773	6	CHL	IMPORT HLL
Q	31/08/2020 09:14:09	f4cfa2995f1	31/08/2				31	240	44773	0	CH	IMPORT H
Q	31/08/2020 09:14:09	f4cfa2995f1	31/08/2				0	240	60440	0	CL	IMPORT L

With a 'REST API' one can use this data for backup (containers) or calculations

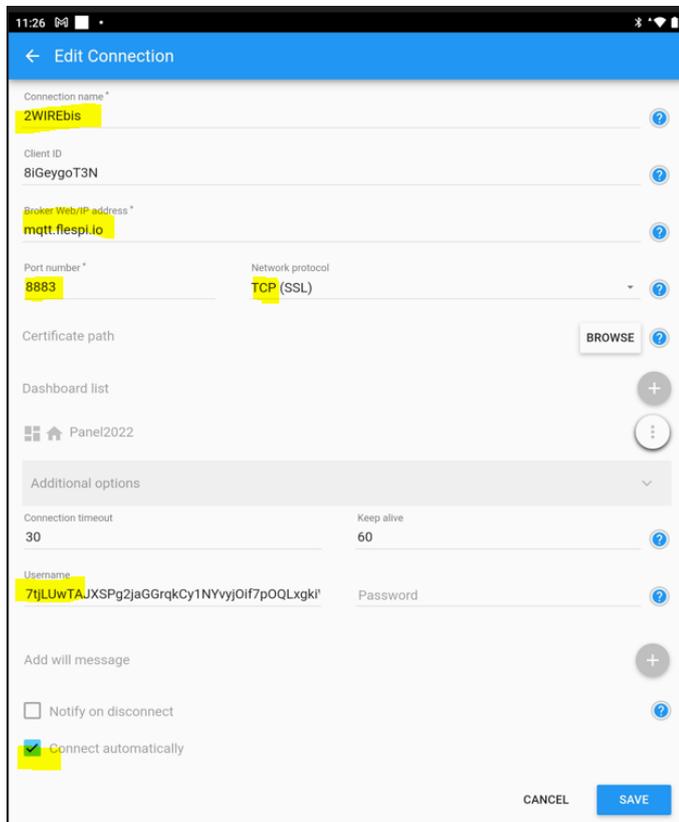
(zie: Flespi Rest Api Documentation)

6.Example link to the Android APP 'IoT MQTT'

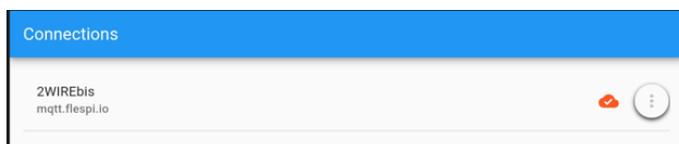
'IoT MQTT' is an Android application where you can download either the free version (with advertising) or the paid pro version to link your MQTT products from there. You can also connect the 2-WIRE web servers LoWi3, ReMI, MEMo3 and the WiFi Plug, THEO stick.

In this chapter we show you how to make the MQTT link with 2-WIRE on the basis of a number of screenshots.

Once the APP has been downloaded on your tablet or smartphone, you can open this app and create a new connection by pressing + A screen will open where you can enter the name, url and token for this new connection:

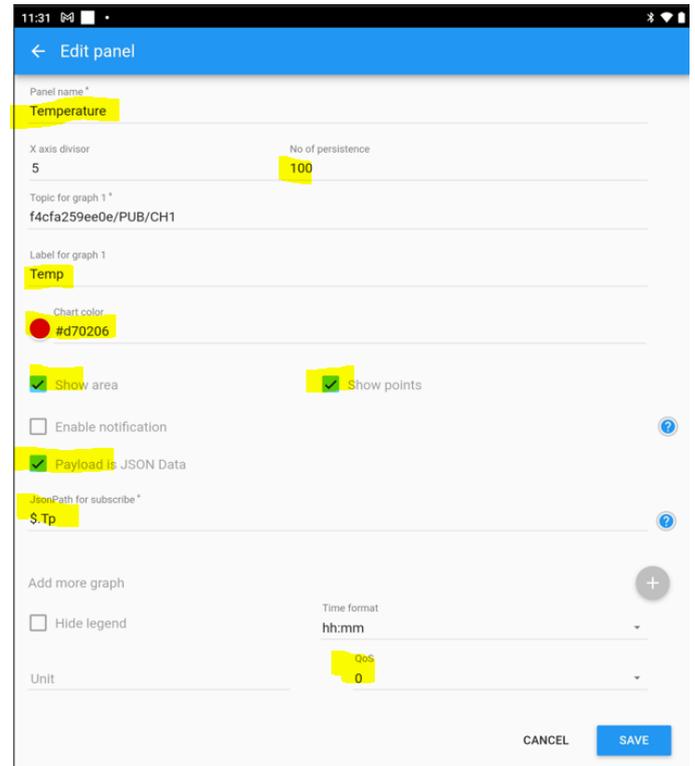


Once the connection is created, you can add a dashboard. '+ ADD A NEW DASHBOARD' and in this dashboard you can add multiple PANELS. We show a few examples :



6.1 Linegraph:

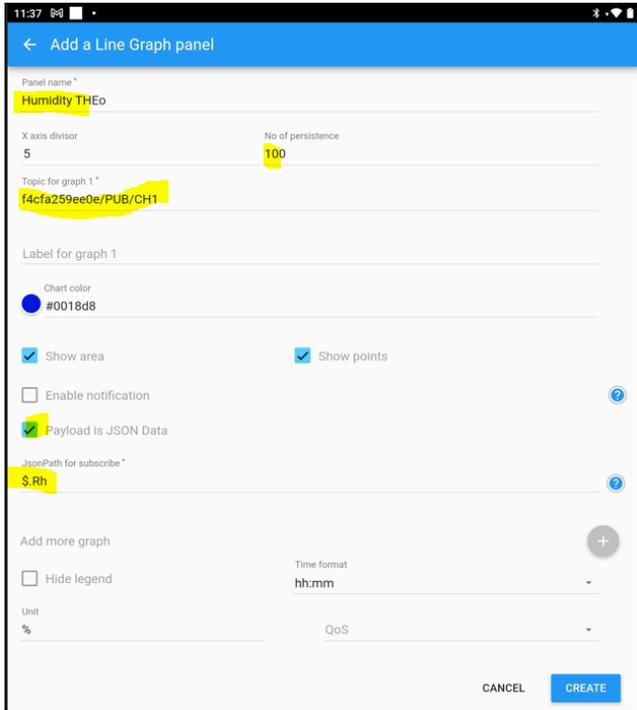
Add a line graph 'Linegraph' where we subscribe to the topic **MAC/PUB/CH1** s of a THEo and the payload 'Tp' = temperature



The result:

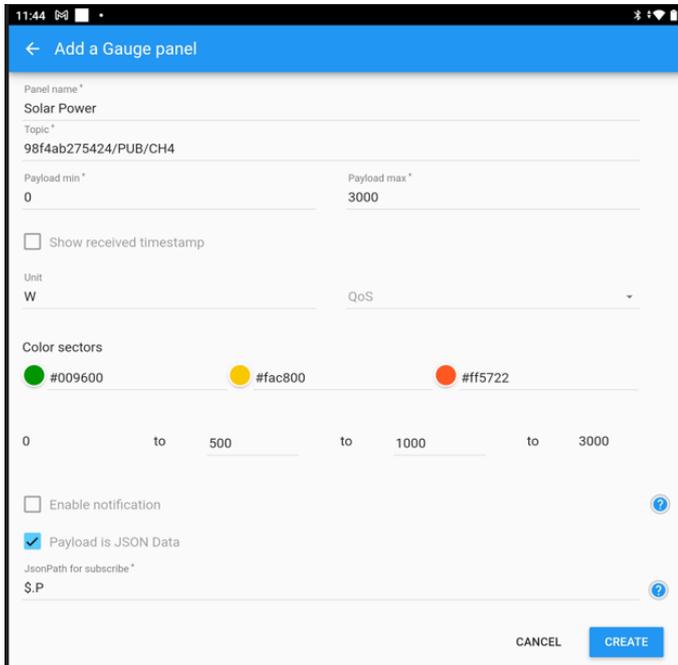


Another example of a line graph with RH%, the same topic of a THEo but here payload 'RH' or the relative humidity



6.2 Speedo meters

In this example a topic **MAC/PUB/CH4** from a MEMo3 web server, Channel4 : here solar with payload "P"= current power

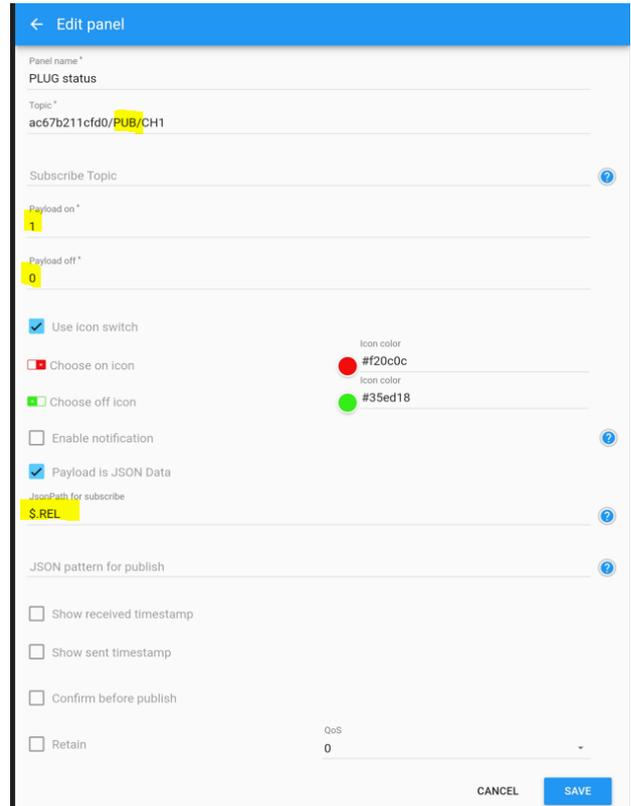


Result:



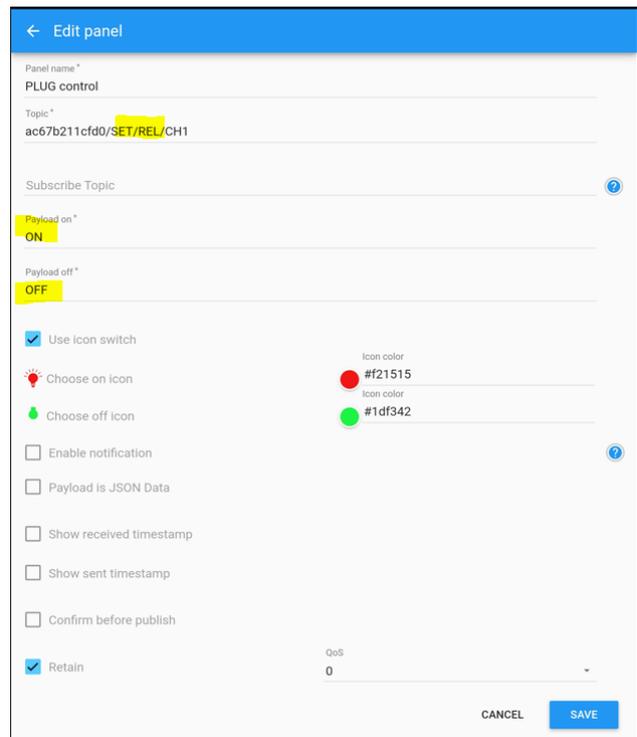
6.3. Status tracking WiFi Plug:

We choose a 'switch' panel here and read the **MAC/PUB/CH1** topic and the payload 'REL' of the WiFi plug to display the 0 or 1 status.



6.4 Operating WiFi plug

We choose a 'switch' panel here and take the **MAC/SET/CH1** topic here to operate the plug ON , OFF



6.5 Text display

We choose a text panel and as a topic we use the wildcard '#' to display all topics **MAC/#** of the Plug live in this case

← Add a Text Log panel

Panel name*
PLUG Matt payloads

Topic*
ac67b211cfd0/#

Additional options >

QoS

Enable notification
Notification is not available for wildcard

Payload is JSON Data

CANCEL CREATE

6.5 Display

If you have checked the option 'work in the background', the topics will continue to come in the background.

You can customize the layout by filling panels on a part of the screen, or by adding text panels, using 'dark mode', change colors,



We show this example on the app' IoT MQTT' because it could help you to develop your own integration or even your own application. And if you are willing to share your application/integration with other 2WIRE users, please contact us.