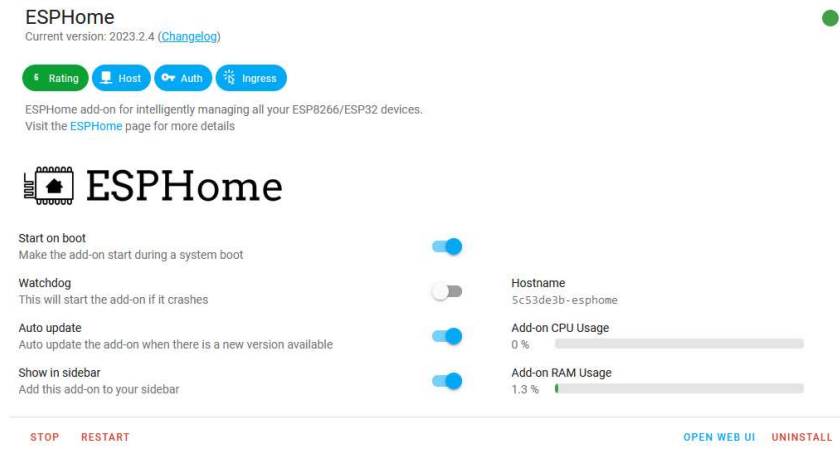


Installation:

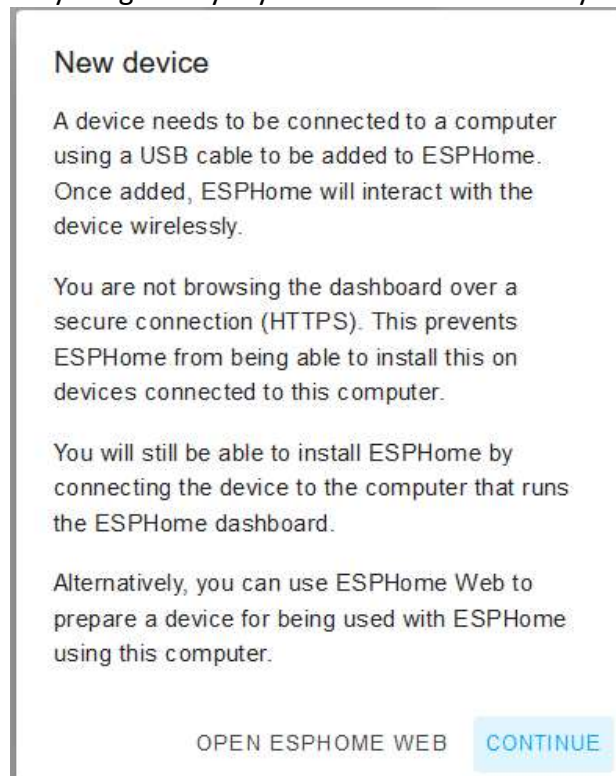
1.

If not done, install <ESPHome> to you HomeAssistant:
HomeAssistant > Settings > Add-ons > ADD-one Store >
select <ESPHome> > install it > activate <Show in Sidebar> and Start it.



2.

Select select <ESPHome> from the sidebar and click on <NEW DEVICE>.
A pop-up window occurs, notice the content about how a device has to be connected.
I prefer to use the device to be connected to the computer running HomeAssistand
and to have do anything locally If you choose the same way hit <CONTINUE>.



3.

In the next window you will be asked for the desired device name,
Simply enter one , and if it is the first time you create an ESPHome device additionally
you will be asked for your WiFi informations, provide the needed values and click <NEXT>

Create configuration

Name*

Enter the credentials of the Wi-Fi network that you want your device to connect to.

This information will be stored in your secrets and used for this and future devices. You can edit the information later by editing your secrets at the top of the page.

Network name *

Password

CANCEL

NEXT

4.

At the next pop-up please select <ESP8266>

Select your device type

Select the type of device that this configuration will be installed on.

ESP32

>

ESP32-S2

>

ESP32-S3

>

ESP32-C3

>

ESP8266

>

Raspberry Pi Pico W

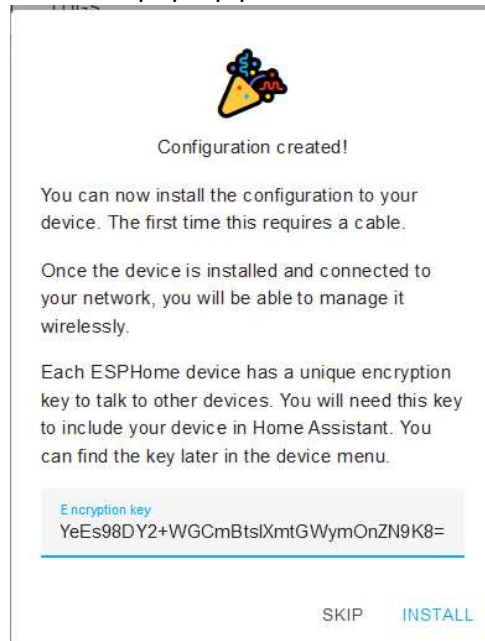
>

☒ Use recommended settings

CANCEL

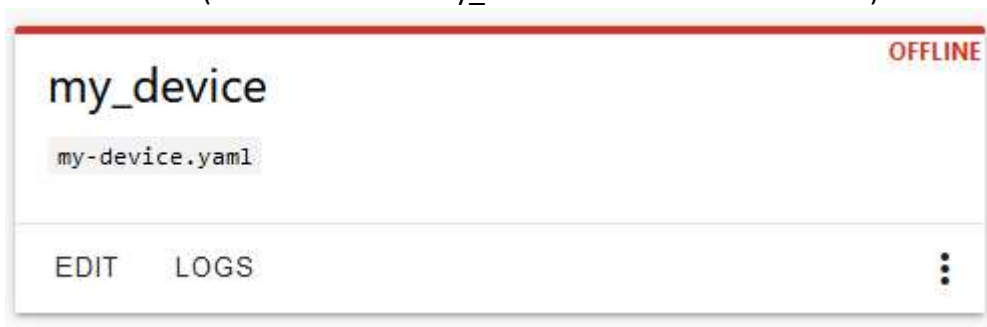
5.

At the next pop-up please select <SKIP> !



6.

Now you should be back to the ESPHome desktop and see a template with your chosen name (I have choosen <my_device> for these instructions)



7.

Now we have to do the most important steps.

Click on <EDIT> on the new template and you should see something like this

```
× my-device.yaml
1 | esp8266:
2 |   name: my-device
3 |   friendly_name: my_device
4 |
5 |   esp8266:
6 |     board: esp01_1m
7 |
8 |   # Enable logging
9 |   logger:
10 |
11 |   # Enable Home Assistant API
12 |   api:
13 |     encryption:
14 |       key: "ddI98kv/s6XYeEs98DY2+WGCmBtsIXmtGWymOnZN9K8="
15 |
16 |   ota:
17 |     password: "3ffda77ba97c39f4fbd3623dbebc6992"
18 |
19 |   wifi:
20 |     ssid: !secret wifi_ssid
21 |     password: !secret wifi_password
22 |
23 |   # Enable fallback hotspot (captive portal) in case wifi connection fails
24 |   ap:
25 |     ssid: "My-Device Fallback Hotspot"
26 |     password: "cxHBewB1f3hm"
27 |
28 |   captive_portal:
29 |
```

8.

Now you have to open the provided example .yaml file

https://github.com/PitHerm/esphome-externalcomponents/blob/main/Docs/s5_gr3p15kww3.yaml

within a new browser session !

now you have to insert/replace parts from the sample .yaml into your .yaml file but take care not to overwrite the parts which I marked in RED of your file shown here:

```
esphome:
  name: my-device
  friendly_name: my_device

esp8266:
  board: esp01_1m

# Enable logging
logger:

# Enable Home Assistant API
api:
  encryption:
    key: "ddI98kv/s6XYeEs98DY2+WGCmBtsIXmtGWymOnZN9K8="

ota:
  password: "3ffda77ba97c39f4fbd3623dbebc6992"

wifi:
  ssid: !secret wifi_ssid
  password: !secret wifi_password

  # Enable fallback hotspot (captive portal) in case wifi
  connection fails
  ap:
    ssid: "My-Device Fallback Hotspot"
    password: "cxHBewBLf3hm"

captive_portal:
```

after all it should look like this (I marked the parts to keep with RED again):

```
esphome:
  name: my-device
  friendly_name: my_device

external_components:
  source:
    type: git
    url: https://github.com/PitHerm/esphome-externalcomponents

esp8266:
  board: esp01_1m

# Enable logging
logger:
  baud_rate: 0

web_server:
  port: 80

# Enable Home Assistant API
api:
  encryption:
    key: "ddI98kv/s6XYeEs98DY2+WGCmBtsIXmtGWymOnZN9K8="

ota:
  password: "3ffda77ba97c39f4fbd3623dbebc6992"

wifi:
  ssid: !secret wifi_ssid
  password: !secret wifi_password

  # Enable fallback hotspot (captive portal) in case wifi
  connection fails
  ap:
    ssid: "My-Device Fallback Hotspot"
```

password: "cxHBewBLf3hm"

captive_portal:

sensor:

- platform: uptime
name: Solis Inverter ESPHome Uptime
update_interval: 60s
- platform: wifi_signal
name: Solis Inverter ESPHome Wifi Signal
update_interval: 60s

uart:

id: uart_bus
tx_pin: GPIO5
rx_pin: GPIO4
rx_buffer_size: 4096
#tx_pin: 1
#rx_pin: 3
baud_rate: 9600

debug:

direction: BOTH
dummy_receiver: false
after:
delimiter: "\n"

sequence:

- lambda: UARTDebug::log_string(direction, bytes);

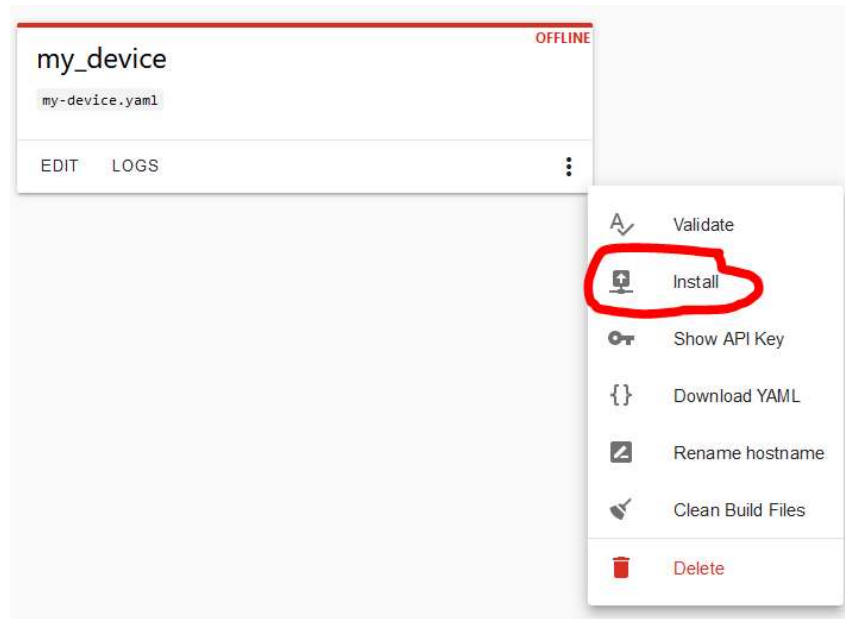
s5_gr3p15kww3:

id: solisinverter
uart_id: uart_bus
update_interval: 60s
voltage_dc_1:
name: "Solis Inverter DC Voltage 1"
voltage_dc_2:
name: "Solis Inverter DC Voltage 2"
voltage_ac_u:

```
    name: "Solis Inverter AC Voltage U"
voltage_ac_v:
    name: "Solis Inverter AC Voltage V"
voltage_ac_w:
    name: "Solis Inverter AC Voltage W"
current_dc_1:
    name: "Solis Inverter DC Current 1"
current_dc_2:
    name: "Solis Inverter DC Current 2"
current_ac_u:
    name: "Solis Inverter AC Current U"
current_ac_v:
    name: "Solis Inverter AC Current V"
current_ac_w:
    name: "Solis Inverter AC Current W"
power_dc_1:
    name: "Solis Inverter DC Power 1"
power_dc_2:
    name: "Solis Inverter DC Power 2"
power_ac_total:
    name: "Solis Inverter AC Power Total"
va_ac_total:
    name: "Solis Inverter AC VA Total"
powerfactor_ac:
    name: "Solis Inverter AC Power Factor"
energy_today:
    name: "Solis Inverter Energy Today"
energy_thismonth:
    name: "Solis Inverter Energy This Month"
energy_total:
    name: "Solis Inverter Energy Total"
temperature_inv:
    name: "Solis Inverter Temperature"
```


9.

Now click on <SAVE> to save the file. and then on <X> to close it.
After that click on the three vertically point on the template of your new device and select
<Install>



10.

Within the next pop-up select how you want to install the software to the D1-mini.
Before you select a way to, read the following lines and connect the D1-mini on your choice
Then select the way to.

Wirelessly : not possible for first installation

Plug into this computer : if you are connected remotely to the computer running HomeAssistant and the D1-mini is connected to this computer ,
works only with Chrome or Edge browser.

Plug into the computer running the ESPHome Dashboard : (my preferred) the D1-mini is plugged into the computer running HomeAssistant.

How do you want to install my-device.yaml on your device?

- Wirelessly >
Requires the device to be online
- Plug into this computer >
For devices connected via USB to this computer
- Plug into the computer running ESPHome Dashboard >
For devices connected via USB to the server
- Manual download >
Install it yourself using ESPHome Web or other tools

CANCEL

11.

A new windows opens and you will see the progress of compiling etc.
It takes some time for compiling etc , please be patient.

If it is ready, close the window and select Settings > Devices
After some seconds your new device should appear (if not reboot HomeAssistant)
Now <Configure> the new device.

12.

Connect the new device to your inverter and after (depending at what time the stick did the last request) maximum 10 Minutes your values will appear.