

Home Assistant

 TASMOTA



 SONOFF



Inhaltsverzeichnis

1	Flashing Tasmota Software.....	2
1.1	Hardware Preparation.....	2
1.1.1	Serial Connection.....	2
1.1.2	Programming Mode.....	3
1.1.3	Common Mistakes.....	4
1.2	Flashing.....	4
1.3	Initial Configuration.....	5
1.4	After Configuration.....	6
1.5	Alternative Setup of IP-Configuration.....	6
1.6	Ermitteln der IP-Adresse des Tasmota -Geräts.....	7
2	Konfiguration: Sonoff R2 Basic.....	8
3	Konfiguration: Stromzähler D1 Mini.....	12
4	Konfiguration: D1 Mini und PN532 RFID-Reader.....	16
5	Integration der Tasmota Sensoren und Aktoren.....	20
5.1	Über Tasmota Plugin.....	20
5.1.1	Manuelles Erkennen der Tasmota-Geräte (Discovery).....	20
5.1.2	Automatisierung um Status der Geräte zu lesen.....	21
5.2	Tasmota-Geräte über MQTT einbinden.....	22
5.2.1	MQTT Konfiguration für Sonoff Basic R2.....	22
5.2.2	MQTT Konfiguration für Stromzähler Steckdose.....	23
5.2.3	MQTT Konfiguration für D1-Mini+RFID-Reader.....	25
6	Veraltete MQTT Yaml Syntax.....	28

1 Flashing Tasmota Software

1.1 Hardware Preparation

We need to connect to the serial programming interface of the ESP chip. This is done by connecting our serial-to-USB converter TX and RX pins to the ESP RX and TX pins and powering the chip with the 3.3V and GND pins.

In most cases those pins are available on the PCB in the form of pin holes or solder pads but pin headers or jumper wires need to be soldered or otherwise applied. In some cases you will need to solder wires directly on the chip's pins which requires some experience and good soldering equipment.

DO NOT CONNECT DEVICES TO MAINS AC POWER WHILE THE COVER IS OPEN AND CIRCUIT BOARD IS EXPOSED!!!

NEVER TRY TO FLASH WHILE YOUR DEVICE IS CONNECTED TO MAINS POWER!!!

YOU CAN BE ELECTROCUTED IF YOU DO NOT KNOW WHAT YOU ARE DOING!

If you are not careful, your own health will be in danger. Shorting your serial interface with mains AC power will fry your device and serial adapter and will also harm or destroy your computer. It is important to always have all mains power cables disconnected from the device while being connected via serial or even while the case of the device is opened.



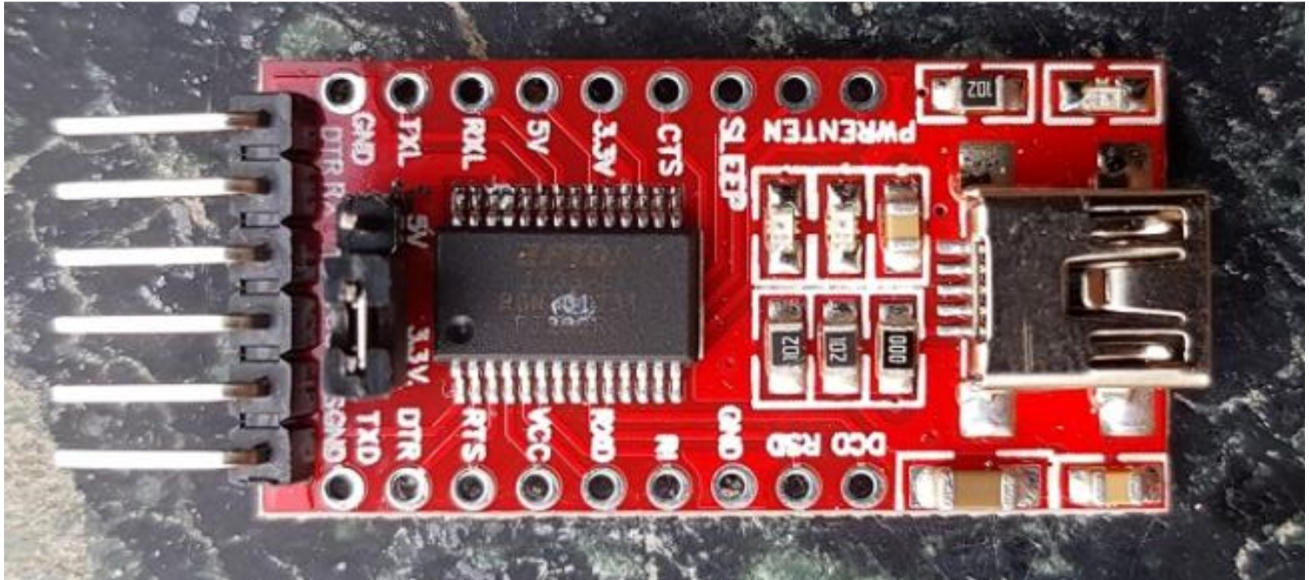
1.1.1 Serial Connection

Each device has its pins labelled differently. If the labelling isn't visible on the PCB please refer to the devices flashing guide or search the Internet for correct pin locations. Device specific instructions and restrictions are documented in the [Tasmota Supported Devices Repository](#). Pinouts for commonly used Wi-Fi modules are [found here](#)

When you have identified pins on your device, connect wires according to the table:

Serial adapter

Power jumper must be set to 3V3 V !



Jumer must be set to 3V3 !

Serial adapter	ESP device
3V3	3V3 or VCC
TX	RX
RX	TX
GND	GND

Note that TX from your adapter goes to RX on the ESP device and RX from adapter goes to TX on the device!

1.1.2 Programming Mode

ESP needs to be put into programming mode or flash mode before the firmware can be uploaded. This is done by connecting GPIO0 pin to GND while the chip is booting.

On many devices the installed control button is connected to GPIO0 and GND, making entering Programming Mode easy. On others you will need to bridge the pins on the PCB or directly on the chip with a jumper wire. GPIO0 locations for popular modules can be found in [Pinouts](#)!

Device specific instructions are documented in [Tasmota Supported Devices Repository](#)..

To put the ESP into Programming Mode:



1. Disconnect serial programmer and power
2. Bridge GPIO0 and GND (by pressing the on-board button or connection with a wire)
3. Connect the serial programmer to your computer
4. After a few seconds disconnect GPIO0 from GND (release button or remove the wire connection). On devices that do not provide the GPIO0 connected button, it may be easier to leave the wired bridge in place throughout the entire flashing process (erase & upload). Doing so will not create any problems. After the firmware is uploaded successfully, remove the bridge. This allows the device to boot normally.

If everything went well, you are now in Programming Mode and ready to continue with [flashing](#). If the flashing process is unable to start, disconnect the device and retry the steps.

1.1.3 Common Mistakes

- Wire connections and solder joints - Double check all connections and also check for solder overflow.
- Use a USB data cable - Some USB cables are for charging only and do not connect the data lines needed to load the firmware onto the device.
- Insufficient power delivered over the serial programmer. This leads to flashing failures or corrupted flash altogether. Supply more power with a separate **3.3V** power supply or get an adapter with a better power supply. Be sure all DC voltages use the same GND line.
- Recheck your serial programmer so to ensure that it **supplies 3.3V** voltage and NOT 5V. 5V will damage the ESP chip!
- Releasing GPIO0 button/wire before booting is finished - It is safe to leave GPIO0 connected to GND during the entire programming process (erase & upload). Just be sure to remove the GPIO0 to GND bridge before booting the device for regular operation.
- Make sure that the RX pin is connected to the TX pin between the serial adapter and your ESP device, and vice versa.
- Erase the flash first and power cycle before uploading the Tasmota firmware binary. Not erasing can leave behind remnants of the previous flash contents which can interfere with the new firmware operation.

1.2 Flashing

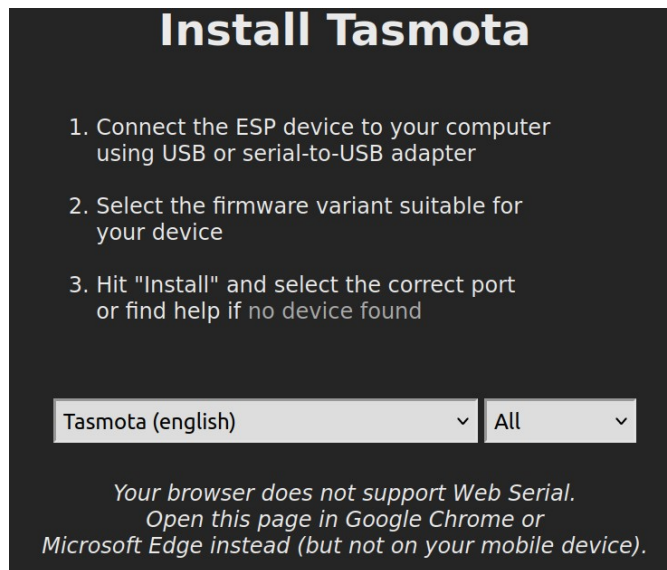
If you have followed [Hardware preparation](#), your device should be in Programming Mode and ready for a Tasmota firmware binary to be installed.

You may want to back up the device manufacturer's firmware on the one in a million chance you don't like Tasmota.

Flash Tasmota directly from your web browser.

Web Installer (no support for firefox! Use Chrome)

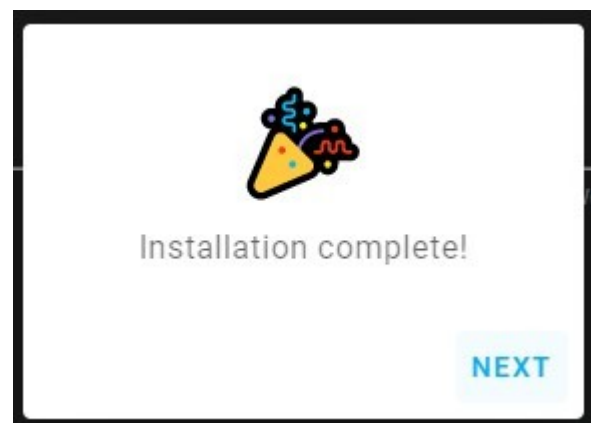
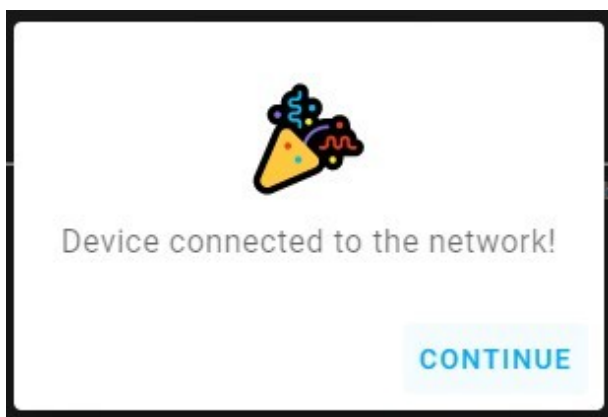
<https://tasmota.github.io/install/>

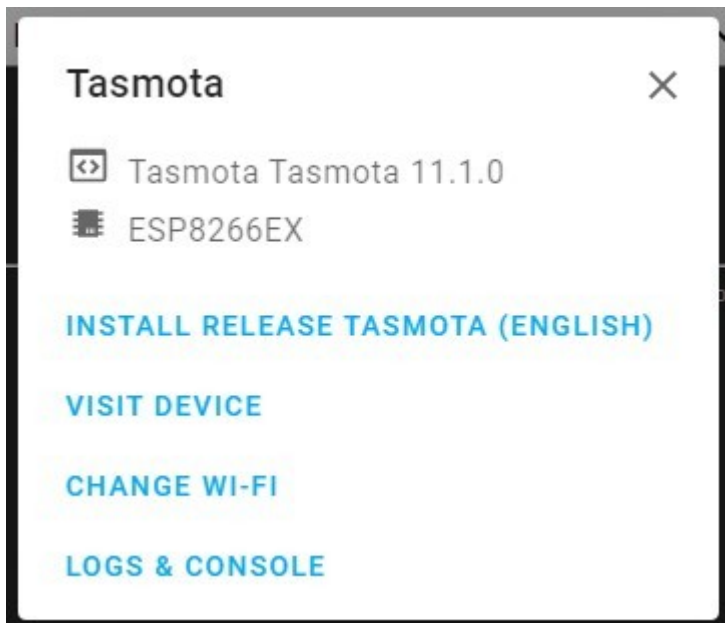


1.3 Initial Configuration

Once the installation is successful, click on NEXT. The installer will scan for Wi-Fi networks and select the strongest one. Use the dropdown to select between networks.

Enter the Wi-Fi password.





Wait until configuration completes and click CONTINUE

From here click on VISIT DEVICE to enter the [WebUI](#) for further configuration.

1.4 After Configuration

Your device running Tasmota is now ready to be [controlled](#)

Check out all the Tasmota [features](#) and ways to [integrate it](#) with other platforms.

1.5 Alternative Setup of IP-Configuration

If the ESP8266/ESP32 device is not correctly configured for the home WLAN, the device will start its own hotspot. Connect to this **TASMOTA_XXXXXX** WLAN network. Then open a web-browser and enter ip **192.168.4.1** address.

1.6 Ermitteln der IP-Adresse des Tasmota -Geräts

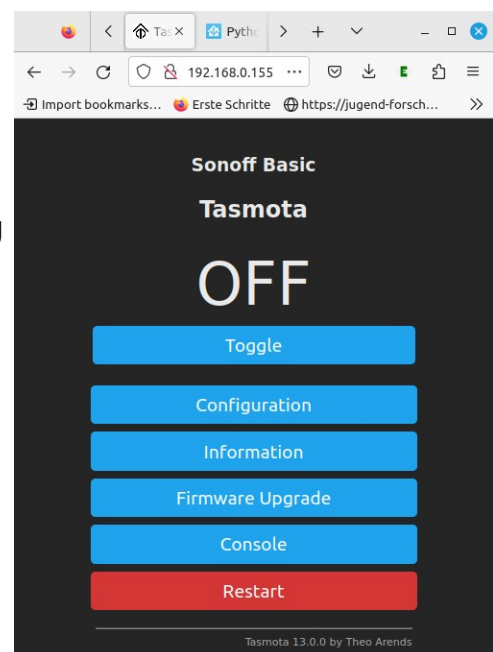
Über den zentrale Router des LANs kann die Liste aller Geräte im Heimnetz ausgelesen werden. Die Fritzbox stellt über den Menüpunkt „Heimnetz → Netzwerk“ können die IP-Adressen von allen Geräten ausgelesen werden:

Heimnetz > Netzwerk

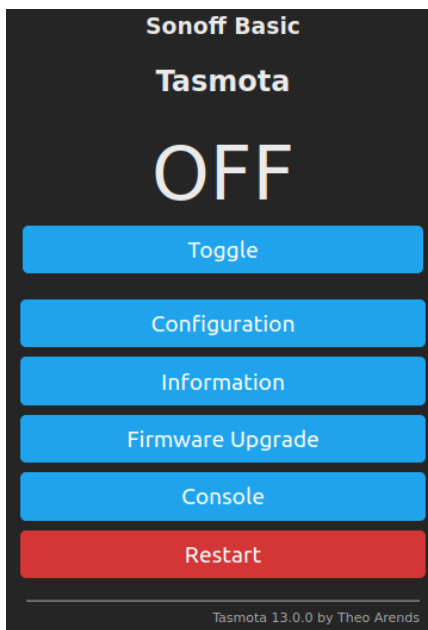
Netzwerkverbindungen		Netzwerkeinstellungen	
Name	Verbindung	IP-Adresse	Eigenschaften
julianFAS20M-S2M	LAN 1 mit 1 Gbit/s	192.168.0.99	
M2007J22G	WLAN	192.168.0.45	2,4 GHz, 96 / 65 Mbit/s
MIWIFI-R3	LAN 1 mit 1 Gbit/s	192.168.0.108	
nextcloud	LAN 1 mit 1 Gbit/s	192.168.0.84	Portfreigabe selbst. Portfreigabe erlaubt
PC-38-1F-8D-07-96-BA	WLAN	192.168.0.102	2,4 GHz, 72 / 58 Mbit/s
relais-1-2609	WLAN	192.168.0.155	2,4 GHz, 72 / 54 Mbit/s
softIQ-EntkalkungsAnlage	WLAN	192.168.0.23	2,4 GHz, 39 / 1 Mbit/s
Strommessung-3652	WLAN	192.168.0.153	2,4 GHz, 72 / 54 Mbit/s

Das Tasmota Gerät kann über den Webbrowser konfiguriert werden. Hierzu wird die Adresse des Tasmota Geräts als URL in die oberste Zeile des Browsers eingetragen. Die Tasmota Software unterstützt nur den HTTP Dienst. Die IP-Adresse kann in die URL-Leiste des Browsers eingetragen werden: **HTTP://192.168.XXX.XXX**. ist die vom Router vergebene IP-Adresse.

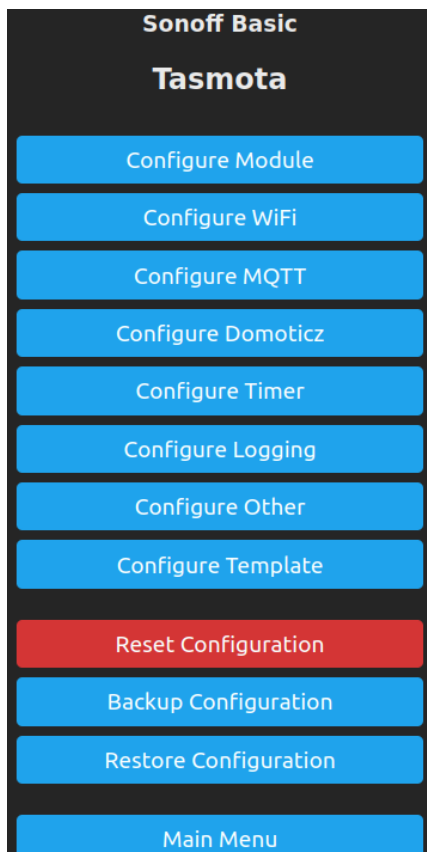
If the IP-Address is e.g.**192.168.0.155** the local configuration of the device can be opened, by entering the IP-address in the URL line of the browser (e.g. **http://192.168.0.155**)



2 Konfiguration: Sonoff R2 Basic



Configuration



Configure Module

Sonoff Basic

Tasmota

Module parameters

Module type (Sonoff Basic)

Sonoff Basic (1) ▾

GPI01

None ▾

GPI02

None ▾

GPI03

None ▾

GPI04

None ▾

GPI014

None ▾

Save

Configuration

Tasmota 13.0.0 by Theo Arends

Configure WIFI

Sonoff Basic

Tasmota

Schottenberg27 

@ebsWLAN 

mountain 

Scan for all WiFi Networks

Wifi parameters

WiFi Network ()

Schottenberg27

WiFi Password 

....

WiFi Network 2 ()

Type your Alternative WiFi Network

WiFi Password 

....

Hostname (%s-%04d)

%s-%04d

Save

Configuration

Configure MQTT

Sonoff Basic

Tasmota

MQTT parameters

Host ()

192.168.0.202

Port (1883)

1883

Client (DVES_5F8A31)

DVES_%06X

User (DVES_USER)

jürgen

Password ■

....

Topic = %topic% (tasmota_5F8A31)

relais_1

Full Topic (%prefix%/%%topic%/)

%prefix%/%%topic%/

Save

Configuration

Configure Other

Sonoff Basic

Tasmota

Other parameters

Template

{"NAME":"Generic","GPIO":[1,1,1,1,1,1,1,

■ Activate

Web Admin Password ■

....

☒ HTTP API enable

☒ MQTT enable

Device Name (Tasmota Relais 1)

Tasmota

Friendly Name 1 (Tasmota)

Tasmota Relais 1

Emulation

☒ None

☐ Belkin WeMo single device

☐ Hue Bridge multi device

Save

Settings:

WIFI:

WIFI Network(): SSID des Hausnetzes

WIFI Passowort: Passwort des angegebene Hausnetzes

MQTT

Host(): IP-Adress of MQTT broker (192.168.0.202)

User: User name configured in MQTT broker (jürgen)

Password: Passwort of the specified user (neuneuneuneu)

Topic: relais_1

Template:

```
{"NAME":"Generic","GPIO":[1,1,1,1,1,1,1,1,1,1,1,1,1,1],"FLAG":0,"BASE":18}
```

HTTP API: enabled

MQTT: enabled

Friendly Name1: Tasmota Relais 1

3 Konfiguration: Stromzähler D1 Mini

NOUS A1T
Tasmota

Voltage	306	V
Current	0.000	A
Active Power	0	W
Apparent Power	0	VA
Reactive Power	0	VA _r
Power Factor	0.00	
Energy Today	0.000	kWh
Energy Yesterday	0.000	kWh
Energy Total	0.086	kWh

ON

Toggle

Configuration

Information

Firmware Upgrade

Console

Restart

Configuration

NOUS A1T
Tasmota

Configure Module

Configure WiFi

Configure MQTT

Configure Domoticz

Configure Timer

Configure Logging

Configure Other

Configure Template

Reset Configuration

Backup Configuration

Restore Configuration

Main Menu

Configure Module

NOUS A1T

Tasmota

Module parameters

Module type (Sonoff Basic)
NOUS A1T (0) ▾

Save

Configuration

Tasmota 13.0.0 by Theo Arends

Configure WIFI

NOUS A1T

Tasmota

Schottenberg27 
mountain 

Scan for all WiFi Networks

Wifi parameters

WiFi Network ()
Schottenberg27

WiFi Password 
....

WiFi Network 2 ()
Type your Alternative WiFi Network

WiFi Password 
....

Hostname (%s-%04d)
%s-%04d

Save

Configuration

Configure MQTT

NOUS A1T

Tasmota

MQTT parameters

Host ()

192.168.0.202

Port (1883)

1883

Client (DVES_178E44)

DVES_178E44

User (DVES_USER)

jürgen

Password ☐

....

Topic = %topic% (tasmota_178E44)

Strommessung

Full Topic (%prefix%/%topic%/)

%prefix%/%topic%/

Save

Configuration

Configure Other

NOUS A1T

Tasmota

Other parameters

Template

{"NAME":"NOUS A1T","GPIO":[32,0,0,0,2

☒ Activate

Web Admin Password ☐

....

☒ HTTP API enable

☒ MQTT enable

Device Name (Tasmota Strommesser)

Tasmota

Friendly Name 1 (Tasmota)

Tasmota Strommesser

Emulation

☒ None

☐ Belkin WeMo single device

☐ Hue Bridge multi device

Save

Settings:

WIFI:

WIFI Network(): SSID des Hausnetzes

WIFI Passowort: Passwort des angegebene Hausnetzes

MQTT

Host(): IP-Adress of MQTT broker (192.168.0.202)

User: User name configured in MQTT broker (jürgen)

Password: Passwort of the specified user (neuneuneuneu)

Topic: Strommessung

Template:

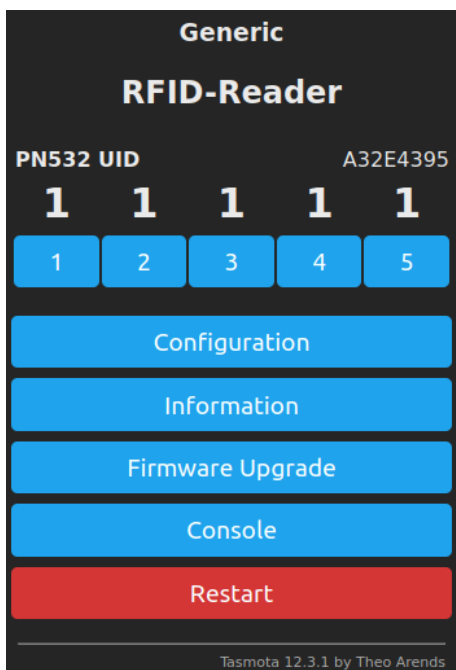
```
{"NAME":"NOUS A1T","GPIO":  
[32,0,0,0,2720,2656,0,0,2624,320,224,0,0,0],"FLAG":0,"BASE":49}
```

HTTP API: enabled

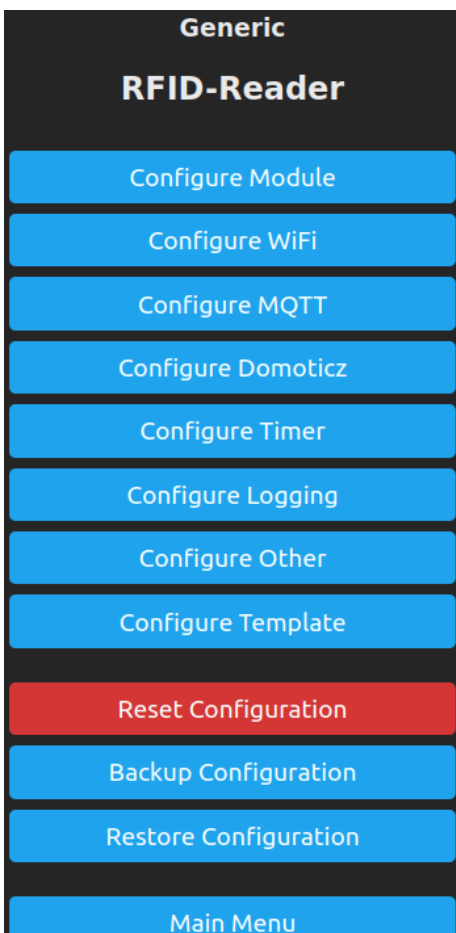
MQTT: enabled

Friendly Name1: Tasmota Strommesser

4 Konfiguration: D1 Mini und PN532 RFID-Reader



Configuration



Configure Module

Generic
RFID-Reader

Module parameters

Module type (Sonoff Basic)
Generic (18) ▾


D3 GPIO0	Button ▾	1 ▾
TX GPIO1	None ▾	
D4 GPIO2	None ▾	
RX GPIO3	Button ▾	2 ▾
D2 GPIO4	PN532 Rx ▾	
D1 GPIO5	PN532 Tx ▾	
D6 GPIO12	Relay ▾	1 ▾
D7 GPIO13	Relay ▾	2 ▾
D5 GPIO14	Relay ▾	3 ▾
D8 GPIO15	Relay ▾	4 ▾
D0 GPIO16	Relay ▾	5 ▾
A0 GPIO17	None ▾	


Save

Configure WIFI

Generic
RFID-Reader

Schottenberg27 

@ebsWLAN 

softliQ:SC_bfad69 

Scan for all WiFi Networks

Wifi parameters

WiFi Network ()
Schottenberg27

WiFi Password ■
....

WiFi Network 2 ()
Type your Alternative WiFi Network

WiFi Password ■
....

Hostname (%s-%04d)
%s-%04d

Save

Configuration

Configure MQTT

Generic
RFID-Reader

MQTT parameters

Host ()
192.168.0.202

Port (1883)
1883

Client (DVES_E4989F)
DVES_%06X

User (DVES_USER)
jürgen

Password ☐
....

Topic = %topic% (tasmota_E4989F)
D1MINI

Full Topic (%prefix%/ %topic%/)
%prefix%/ %topic%/

Save

Configuration

Configure Other

Template

```
{"NAME":"Generic","GPIO":[1,1,1,1,1,1,1,
```

☐ **Activate**

Web Admin Password ☐
....

☒ **HTTP API enable**

☒ **MQTT enable**

Device Name (RFID-Reader)
RFID-Reader

Friendly Name 1 (Tasmota)
RFID-Reader

Friendly Name 2 (Tasmota2)
Tasmota2

Friendly Name 3 (Tasmota3)
Tasmota3

Friendly Name 4 (Tasmota4)
Tasmota4

Friendly Name 5 (Tasmota5)
Tasmota5

Emulation

☒ **None**

☐ **Belkin WeMo** single device

☐ **Hue Bridge** multi device

Settings:

WIFI:

WIFI Network(): SSID des Hausnetzes

WIFI Passwort: Passwort des angegebenen Hausnetzes

MQTT

Host(): IP-Adress of MQTT broker (192.168.0.202)

User: User name configured in MQTT broker (jürgen)

Password: Passwort of the specified user (neuneuneuneu)

Topic: D1MINI

Template:

```
{"NAME":"Generic","GPIO":[1,1,1,1,1,1,1,1,1,1,1,1,1,1],"FLAG":0,"BASE":18}
```

HTTP API: enabled

MQTT: enabled

Friendly Name1: RFID-Reader

5 Integration der Tasmota Sensoren und Aktoren

5.1 Über Tasmota Plugin

Das Tasmota-Plugin muss hierzu als Integration installiert sein.

Siehe auch: <https://tasmota.github.io/docs/Home-Assistant/>

Das Tasmota-Plugin merkt sich alle Tasmota-Geräte, die es während der Laufzeit des Home-Assistant jemals gesehen hat.

Das bedeutet aber, wenn eine neues Tasmota Gerät im Haus installiert wird, muss das Tasmota-Plugin davon Kenntnis erhalten. Dieses Anlernen des Tasmota Plugin kann auf zwei Arten durchgeführt werden.

5.1.1 Manuelles Erkennen der Tasmota-Geräte (Discovery)

<https://community.home-assistant.io/t/tasmota-and-setoption19-ha-discovery/112481>

Über die Option19 steuert Tasmota die Tasmota discovery Funktion. Das heißt: soll ein Tasmota-Gerät von Home-Assistant erkannt werden, dann muss man die Option 19 einschalten und dann wieder ausschalten.

Tasmota-Geräte stellen über das Web-Interface eine Console bereit. Hier muss man den Befehl:

SetOption19 ON

eingeben und mit Enter bestätigen.

Danach muss man diese Option auch wieder abstellen, damit das Tasmota Gerät richtig funktioniert. D.h. 10-20 s nach dem Einschalten, kann die Funktion durch Senden des Befehls

SetOption19 OFF

auch wieder abgeschaltet werden.

Diese Funktion muss für jedes Tasmota-Gerät ausgeführt werden.

5.1.2 Automatisierung um Status der Geräte zu lesen

<https://community.home-assistant.io/t/how-to-set-setoption19-tasmota-devices-marked-unavailable/561597/4>

Alternativ kann eine Automatisierung im Home-Assistant gestartet werden, die den Status der jedes Tasmota-Geräts einliest.

configuration.yaml:

```
- id: '1565194745362'
  alias: Status at start HA
  trigger:
    - event: start
      platform: homeassistant
  condition: []
  action:
    - data:
        payload: ''
        topic: cmd/tasmotas/status
        service: mqtt.publish
```

Dieser Abschnitt muss für jedes Tasmota-Gerät angepasst und dann in die configuration.yaml Datei eingetragen werden.

5.2 Tasmota-Geräte über MQTT einbinden

5.2.1 MQTT Konfiguration für Sonoff Basic R2

Sonoff Basic R2, Smart Switch, WiFi für Tasmotta



```
mqtt:
  switch:
    - name: "CONF: Sonoff Relais 1"
      command_topic: "cmdnd/relais_1/POWER"
      state_topic: "stat/relais_1/POWER"
      qos: 0
      unique_id: 2225d76a-3370-11ee-baeb-325096b39f47
```

5.2.2 MQTT Konfiguration für Stromzähler Steckdose

NOUS Tasmota Steckdose A1T - Wlan smart plug 16A mini WiFi Steckdose



configuration.yaml:

```
mqtt:
  #Intelligente Steckdose
  sensor:
    - name: "CONF: Stromzähler Leistung"
      state_topic: "tele/Strommessung/SENSOR"
      unit_of_measurement: "W"
      value_template: "{{ value_json.ENERGY.Power }}"
      device_class: power
      unique_id: 0969280e-3370-11ee-b74b-325096b39f47

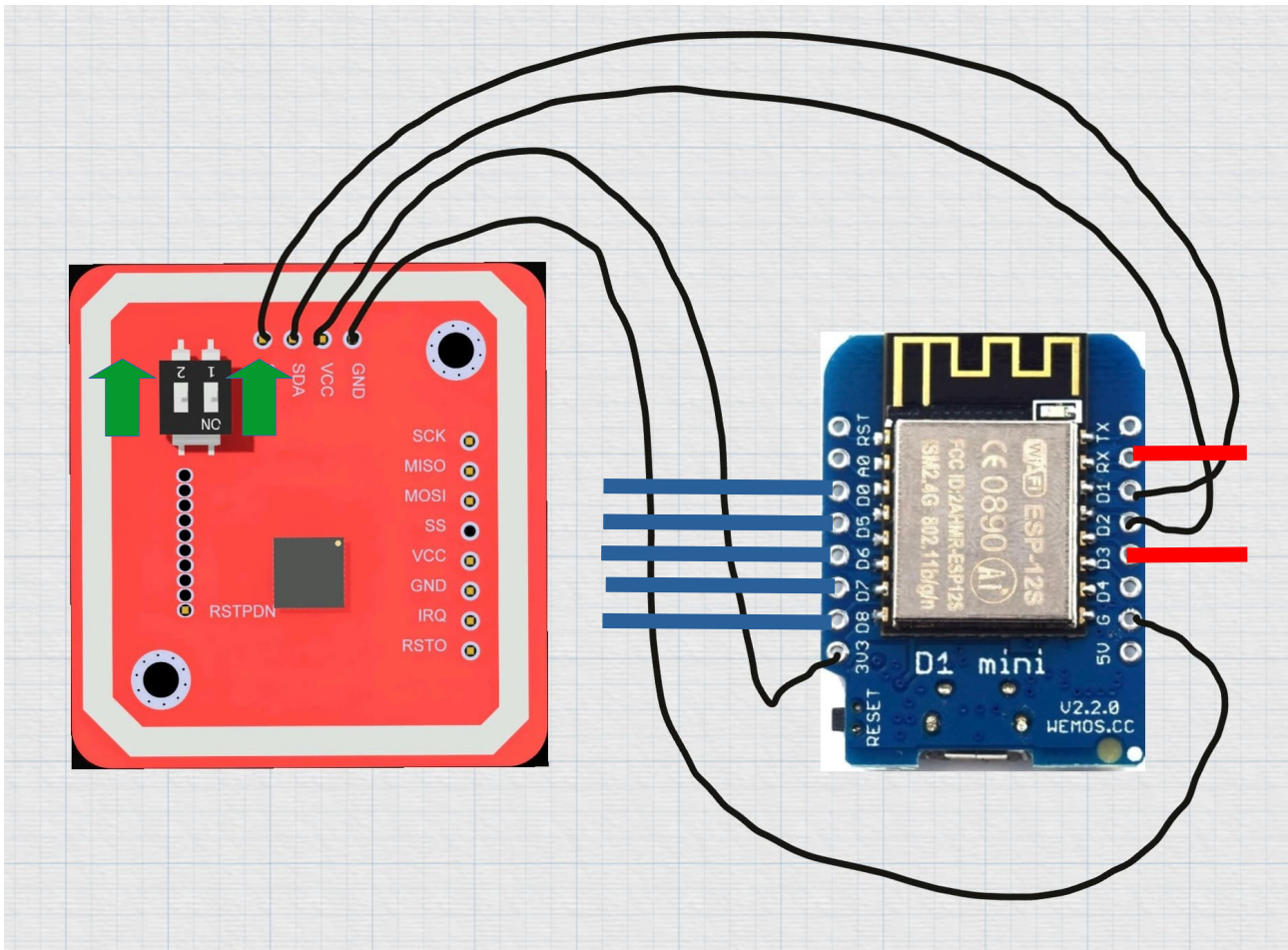
    - name: "CONF: Stromzähler Spannung"
      state_topic: "tele/Strommessung/SENSOR"
      unit_of_measurement: "V"
      value_template: "{{ value_json.ENERGY.Voltage }}"
      device_class: voltage
      unique_id: e0a1a72a-336f-11ee-b5a4-325096b39f47

mqtt:
  switch:
    - name: "CONF: Steckdosenschalter"
      command_topic: "cmd/Strommessung/POWER"
      state_topic: "stat/Strommessung/POWER"
      payload_on: "ON"
      payload_off: "OFF"
      unique_id:
```

3d751d56-3370-11ee-af30-325096b39f47

```
# {  
# "Time": "2023-08-04T19:05:47",  
# "ENERGY": {  
#   "TotalStartTime": "2023-07-30T17:52:55",  
#   "Total": 0.086,  
#   "Yesterday": 0,  
#   "Today": 0,  
#   "Period": 0,  
#   "Power": 0,  
#   "ApparentPower": 0,  
#   "ReactivePower": 0,  
#   "Factor": 0,  
#   "Voltage": 0,  
#   "Current": 0  
# }  
#}
```

5.2.3 MQTT Konfiguration für D1-Mini+RFID-Reader



Rote Linien: Anschlüsse für Knöpfe (Steuerung von R1 und R2)

Blaue Linien: Anschlüsse für Relais oder LEDs R1 - R5

Buttons müssen gegen GND gelegt werden: Die Buttons sind als "Button" definiert: Button active low, internal pull-up resistor

siehe auch: <https://tasmota.github.io/docs/Buttons-schematics/>

Mit den beiden Buttons können die Schaltfunktionen R1 und R2 ausgeführt werden. Möchte man nur die Zustände der Buttons abfragen, dann dürfen an den Ausgängen D6 und D7 keine LEDs oder Relais angeschlossen werden.

configuration.yaml:

```
mqtt:
  #Intelligente Steckdose
  sensor:
    #D1-Mini
    - name: "CONF: RFID_Reader_Scan"
      state_topic: tele/D1MINI/SENSOR
      # {"Time":"2023-08-03T21:14:11","PN532":{"UID":"A32E4395"}}
      qos: 0
      value_template: "{{ value_json.PN532.UID | string }}"
      unique_id: 259c276a-3370-11ee-baeb-325096b39f47

    - name: "CONF: RFID_Reader_Update"
      state_topic: tele/D1MINI/SENSOR
      # {"Time":"2023-08-03T21:14:11","PN532":{"UID":"A32E4395"}}
      qos: 0
      value_template: "{{ value_json.Time }}"
      unique_id: 259c276a-36750-11ee-baeb-32567839f47

  switch:
    #-----
    # D1 MINI
    #-----
    #stat/D1MINI/RESULT = {"POWER1":"OFF"}
    #tele/tasmota/SENSOR = {"Time":"2021-01-01T00:00:00","Switch1":"OFF"}
    - name: "CONF: RFID relais 1"
      command_topic: "cmd/D1MINI/POWER1"
      state_topic: stat/D1MINI/POWER1
      qos: 0
      value_template: "{{ value_json.POWER1 }}"
      unique_id: 2526276a-36750-11ee-baeb-32567839f47

    - name: "CONF: RFID relais 2"
      command_topic: "cmd/D1MINI/POWER2"
      state_topic: stat/D1MINI/POWER2
      qos: 0
      value_template: "{{ value_json.POWER2 }}"
      unique_id: 2527276a-36750-11ee-baeb-32567839f47

    - name: "CONF: RFID relais 3"
      command_topic: "cmd/D1MINI/POWER3"
      state_topic: stat/D1MINI/POWER3
      qos: 0
      value_template: "{{ value_json.POWER3 }}"
      unique_id: 2529276a-36750-11ee-baeb-32567839f47

    - name: "CONF: RFID relais 4"
      command_topic: "cmd/D1MINI/POWER4"
      state_topic: stat/D1MINI/POWER4
      qos: 0
```


value_template: "{{ value_json.POWER4 }}"
unique_id: 2523276a-36750-11ee-baeb-32567839f47

- name: "CONF: RFID relais 5"
command_topic: "cmd/D1MINI/POWER5"
state_topic: stat/D1MINI/POWER5
qos: 0
value_template: "{{ value_json.POWER5 }}"
unique_id: 2529726a-36750-11ee-baeb-32567839f47

6 Veraltete MQTT Yaml Syntax

Alt:

```
mqtt:  
  broker: 127.0.0.1  
sensor:  
  - platform: mqtt
```

Neu:

```
mqtt:  
  sensor:  
    - name: device01
```

Home Assistant unterstützt nur noch das neue Format! Das alte Format findet man häufig noch im Internet in Beispielen.