

GHome Steckdose mit Tasmota flashen

Schritt-für-Schritt-Anleitung

Flashen der freien Firmware Tasmota auf smarte Steckdosen von GHome EP2-A.

“

Achtung!

Das Verändern der Firmware eines Geräts kann immer dazu führen, dass das Gerät unbrauchbar wird. Im schlimmsten Fall kann es danach gar nicht mehr benutzt werden! Die Modifikation geschieht auf eigene Gefahr, wir übernehmen keinerlei Haftung für Schäden!

Was ist Tasmota?

Tasmota ist eine freie Firmware, die auf Geräte installiert (geflasht) werden kann, die auf dem Chip ESP8266 basieren. Günstige smarte Steckdosen beispielsweise hängen häufig von einer nicht ganz so komfortablen China-App ab und können ggf. auch nur über das Internet gesteuert werden.

Vorteile von Tasmota:

- Keine Abhängigkeit vom Internet.
- Kann vollständig im lokalen Netzwerk betrieben und gesteuert werden.
- Bietet eine einfache Weboberfläche, die im Browser aufgerufen werden kann.
- Lässt sich einfach in Smart Home-Systeme wie Home Assistant einbinden.

Dieser Beitrag zeigt den Installationsprozess von Tasmota am Beispiel der bei Amazon erhältlichen smarten Steckdosen [GHome EP2-A](#) (Kein Affiliate-Link).

Voraussetzungen

Folgendes sollte vorhanden sein, um die Steckdosen erfolgreich mit Tasmota flashen zu können:

- Ein WLAN-fähiges Gerät. (hier ein Raspberry Pi 4)
- Ein Smartphone (hier ein iPhone) oder beliebiges anderes zusätzliches WLAN-fähiges Gerät.
- Die Steckdosen, die geflasht werden sollen.

Wenn zusätzlich auch die Strommessfunktion genutzt werden soll, muss diese kalibriert werden, dazu benötigt:

- Ein Strommessgerät, von dem Daten wie Watt, Volt und Ampere abgelesen werden können.
- Einen Verbraucher, der möglichst konstante Strommengen zieht, z.B. eine 60W Lampe.

Wie man den Raspberry Pi installiert und sich damit über SSH verbindet, könnt ihr in unzähligen Anleitung via Google-Suche nachlesen.

Benötigte Software

Zunächst muss ein wenig Software auf dem Raspberry Pi (oder einem beliebigen anderen Debian-basierten System) installiert werden. Dazu verbinden wir uns per SSH auf das Gerät. Da das Flash-Tool viele Abhängigkeiten hat, ist die Durchführung mit Docker am einfachsten. Sollte Docker noch nicht installiert sein, nutzt den nachfolgenden Schnelldurchlauf:

```
sudo apt update
sudo apt install ca-certificates curl
sudo install -m 0755 -d /etc/apt/keyrings
sudo curl -fsSL https://download.docker.com/linux/debian/gpg -o /etc/apt/keyrings/docker.asc
sudo chmod a+r /etc/apt/keyrings/docker.asc
echo \
"deb [arch=$(dpkg --print-architecture) signed-by=/etc/apt/keyrings/docker.asc]
https://download.docker.com/linux/debian \
$(. /etc/os-release && echo "$VERSION_CODENAME") stable" | \
sudo tee /etc/apt/sources.list.d/docker.list > /dev/null
sudo apt update
sudo apt install docker-ce docker-ce-cli containerd.io docker-buildx-plugin docker-compose-plugin
```

Zusätzlich muss auf jeden Fall `git` auf dem System sein:

```
sudo apt install git
```

Jetzt kann das eigentlich Tool **tuya-convert** von Github geladen werden. Dies kann einfach im Home-Verzeichnis gemacht werden:

```
git clone https://github.com/ct-Open-Source/tuya-convert
```

Anschließend wechseln wir in das neue Verzeichnis:

```
cd tuya-convert
```

Anpassung an tuya-convert

Jedes Gerät in einem Netzwerk hat eine MAC-Adresse. An den ersten Stellen einer MAC-Adresse (Präfix) kann man ableiten, von welchem Hersteller das Gerät kommt. Das Tool **tuya-convert** hat eine Liste dieser Präfixe hinterlegt, um flashbare Geräte identifizieren zu können. Die Steckdosen von GHome scheinen aber MAC-Adressen zu haben, dessen Präfixe dort noch nicht enthalten sind. Deswegen erkennt das Tool diese nicht und wir müssen die Präfixe ergänzen.

In den Amazon Rezensionen ist die Rede davon, dass der Präfix `34987A` hinzugefügt werden muss. Bei meinen Steckdosen war es der Präfix `3CE90E`. Deswegen fügen wir beide in der Liste hinzu, es kann aber sein dass ihr dort noch einen anderen Präfix eintragen müsst, dies erfahrt ihr dann aber erst später im Prozess.

```
nano scripts/oui/esp.txt
```

Und dort ganz am Ende jeweils in eine eigene Zeile die beiden zusätzlichen Präfixe eintragen. Mit **STRG + O -> Eingabetaste -> STRG + X** (Mac: *command statt STRG*) speichern und den Editor beenden.

Steckdose flashen

Jetzt ist nahezu alles vorbereitet, um den Flash-Vorgang zu starten.

.env

Um das Flash-Tool mit Docker zu starten, wird noch eine .env Datei benötigt. In unserem aktuellen Verzeichnis **tuya-convert** befindet sich dazu eine Beispieldatei mit dem Namen `.env-template`, diese kopieren wir:

```
cp .env-template .env
```

Der wichtigste Eintrag in dieser Datei ist der Name des WLAN-Adapters auf unserem System. Voreingestellt ist dort `wlan0` und das sollte auf dem Raspberry Pi und vielen anderen Systemen auch genau so passen. Alle Adapter anzeigen lassen kann man sich unter anderem mit `ip addr`. Sollte der Name abweichen, müsst ihr den vorab in der neuen Datei `.env` anpassen.

Docker Image bauen und Container starten

Jetzt ist endlich alles bereit für den Start. Das Docker Image wird direkt auf dem System gebaut, dann ein Container erstellt und dieser gestartet. Sobald der Vorgang abgeschlossen ist wird der Container wieder automatisch gelöscht:

Hinweis

Der Build-Vorgang kann beim ersten Mal sehr lange dauern. Auf meinen Raspberry Pi 4 lag die Dauer bei 4-5 Minuten.

```
sudo docker compose build && sudo docker compose run --rm tuya
```

Es kann sein, dass noch einige Warnhinweise bestätigt werden müssen. Sobald die nachfolgende Meldung erscheint, müssen zunächst die dort erwähnten Schritte durchgeführt werden, die nachfolgend auch detaillierter beschrieben werden, bevor es mit "Enter" weitergehen kann.

IMPORTANT

1. Connect any other device (a smartphone or something) to the WIFI vtrust-flash

This step is IMPORTANT otherwise the smartconfig may not work!

2. Put your IoT device in autoconfig/smartconfig/pairing mode (LED will blink fast). This is usually done by pressing and holding the primary button of the device

Make sure nothing else is plugged into your IoT device while attempting to flash.

3. Press ENTER to continue

1. Das Tool hat jetzt ein WLAN mit dem Namen **vtrust-flash** geöffnet. Mit diesem verbinden wir uns jetzt mit dem Smartphone oder einem anderen beliebigen WLAN-fähigen Gerät.

2. Dann stecken wir die smarte Steckdose ist eine stromführende Steckdose. Sie sollte blau blinken und befindet sich damit im Pairing-Modus. Falls nicht, den Button an der Steckdose lange drücken, bis die smarte Steckdose einmal neu startet.
3. Ist beides erledigt, kann die Meldung auf dem Raspberry Pi mit der Eingabetaste bestätigt werden.
4. Jetzt läuft der vollautomatische Flashing-Prozess, bei dem vorerst nur ein minimales Zwischensystem aufgespielt wird.

Sobald das abgeschlossen ist, erscheint folgende Meldung auf dem Raspberry Pi:

```
You can also provide your own image by placing it in the /files directory
Please ensure the firmware fits the device and includes the bootloader
MAXIMUM SIZE IS 512KB
```

```
Available options:
```

- 0) return to stock
- 1) flash espurna.bin
- 2) flash tasmota.bin
- q) quit; do nothing

```
Please select 0-2: 2
```

```
Are you sure you want to flash tasmota.bin? This is the point of no return [y/N] y
```

Da wir Tasmota auf dem Gerät haben möchten, wählen wir Option 2 und bestätigen dies anschließend nochmal mit “y”. Jetzt wird Tasmota auf das Gerät übertragen, anschließend startet die Steckdose neu. Die Frage, ob ein weiteres Gerät geflasht werden soll, kann mit “n” beantwortet werden.

Vorgang bricht ab und findet kein Gerät

Wenn es zu keinem Problem gekommen ist, kann dieser Abschnitt ignoriert werden.

Möglicherweise erhaltet ihr folgende Meldung, bevor der Flash-Vorgang starten kann:

```
Timed out while waiting for the device to (re)connect
```

```
=====
```

```
Attempting to diagnose the issue...
```

```
No ESP82xx based devices connected according to your wifi log.
```

```
Here is a list of all the MAC addresses that connected:
```

```
3c:e9:0e:f2:23:b5
```

Damit muss in der Datei `scripts/oui/esp.txt` ggf. ein weiterer MAC-Präfix eingetragen werden. Die Meldung gibt aus, welche beiden Geräte gefunden worden sind, hier:

- 3c:e9:0e:f2:23:b5
- a6:02:6b:4b:b6:38

Eins davon wird die Steckdose sein, das andere das Smartphone. Dort lässt sich leicht nachschauen, welche MAC-Adresse das Gerät hat, bei meinem iPhone war es die `a6:02:6b:4b:b6:38`, damit gehört die andere Adresse zur Steckdose. Der Präfix sind die ersten drei Blöcke ohne Doppelpunkt: `3c:e9:0e:f2:23:b5` = `3CE90E`

Nach der Ergänzung kann der Vorgang dann einfach wiederholt werden, ggf. muss die Steckdose wieder durch langes Drücken des Buttons in den Pairing-Modus versetzt werden.

Konfiguration

Die Steckdose läuft jetzt mit Tasmota. Jetzt folgen noch ein paar Schritte, um sie korrekt zu konfigurieren.

Steckdose mit WLAN verbinden

Um die Steckdose mit dem gewünschten WLAN zu verbinden, sind folgende Schritte notwendig:

1. Die neu geflashte Steckdose öffnet ein WLAN mit einem Namen, der mit "tasmota_XXXX-XXXX" startet.

2. Das Smartphone mit diesem WLAN der Steckdose verbinden.



3. Es öffnet sich nach dem Verbinden automatisch eine Webseite zum Eintragen der Login-Daten für das gewünschte WLAN, mit das sich die Steckdose verbinden soll (kann etwas dauern). Falls nicht, folgende IP im Browser aufrufen: `http://192.168.4.1`
4. **Scan for wifi networks** auswählen -> Es wirkt so als wenn nichts passiert. Einfach etwas länger warten, er sucht nach WLANs und zeigt diese dann an.
5. WLAN auswählen, dann das dazugehörige Passwort bei **AP1** eingeben.
6. Mit Save bestätigen. Die Steckdose startet neu, danach ist die Steckdose im WLAN und per Browser erreichbar.
7. Die IP der Steckdose kann im Router gefunden werden.

192.168.4.1
tasmota_F22FCD-4045



Captive WLAN

[Abbrechen](#)

Sonoff Basic Module

Tasmota

[Scan for wifi networks](#)

Wifi parameters

AP1 SSId ()

AP1 Password ☐

AP2 SSId ()

AP2 Password ☐

Hostname (%s-%04d)

CORS Domain

Save

Restore Configuration

Reset Configuration

Firmware updaten

Die geflashte Version von Tasmota ist sehr alt, deswegen ist der erste Schritt jetzt, die Firmware zu aktualisieren. Dazu öffnen wir in einem Browser die Weboberfläche der Steckdose, indem wir dessen IP eingeben. Ggf. kommt eine Warnung weil es sich um eine unverschlüsselte HTTP-Verbindung handelt, dann muss einmal bestätigt werden dass die Seite wirklich aufgerufen werden soll.

Auf der Oberfläche dann **Firmware Upgrade** auswählen und dann auf den oberen Button bei der OTA-URL auf **Start upgrade** drücken. Jetzt gibt es zwei Möglichkeiten:

1. Entweder startet jetzt das vollautomatische Update und die Steckdose startet anschließend neu.
2. Oder es wird eine Datei **tasmota.bin.gz** im Browser heruntergeladen. Dann muss im unteren Feld diese Datei ausgewählt werden und dort **Start upgrade** gedrückt werden, damit dann der Updateprozess startet und die Steckdose neu startet.

Template anpassen

Nach dem Update kommt man wieder auf die Weboberfläche. Damit Tasmota weiß, wie alle Funktionen korrekt angesteuert werden, muss das Template angepasst werden.

Dazu auf **Configuration** -> **Configure other** und im ersten Feld "Template" folgendes eintragen (gilt nur für die GHome-Steckdosen, bei anderen Geräten ggf. abweichend):

```
{ "NAME": "GHome EP2-A", "GPIO": [576,0,320,0,2656,2720,0,0,2624,32,0,224,0,0], "FLAG": 0, "BASE": 45 }
```

Direkt da drunter bei **Activate** muss ein Häkchen gesetzt werden. Zusätzlich können **Device Name** und **Friendly Name 1** angepasst werden. Die Steckdose an der mein PC ist hat zum Beispiel den Namen "tasmota-pc".

Dann wird das ganze mit **Save** bestätigt, das Gerät startet dann wieder neu.

Strommessung kalibrieren (optional)

Wenn auch die Strommessung sinnvolle Werte ergeben soll, muss dies einmal kalibriert werden. Wie schon erwähnt wird dazu ein Spannungsmessgerät und ein konstanter Verbraucher benötigt, wie zum Beispiel eine 60W Lampe.

Vorgehen dabei:

1. Tasmota Steckdose in eine Steckdose stecken, darin dann das Spannungsmessgerät einstecken und darin dann wiederum den Verbraucher (Lampe o.ä.).
2. In der Weboberfläche den Menüpunkt **Console** aufrufen.

In der Konsole folgende Befehle eintippen (mit Eingabetaste jeweils bestätigen):

- `savedata 1`
- `VoltageSet ###.##` (in V)
- `PowerSet ###.##` (in W)
- `CurrentSet ###.##` (in mA)
- `savedata 0`

Anschließend wieder ins Hauptmenü gehen, dort sollten jetzt passende Stromwerte zu sehen sein.

Abschluss

Die smarte Steckdose läuft jetzt mit Tasmota und ist einsatzbereit. Sie kann jetzt problemlos über die Weboberfläche gesteuert werden, eine Einbindung in ein Smart Home-System ist ebenfalls kein Problem mehr.

Mit freundlicher Genehmigung von viertelwissen.de

Revision #5

Created 27 March 2025 00:42:29 by Christian

Updated 5 December 2025 07:40:04 by Christian