
SunFounder PiPower

www.sunfounder.com

11.10.2023

Inhaltsverzeichnis

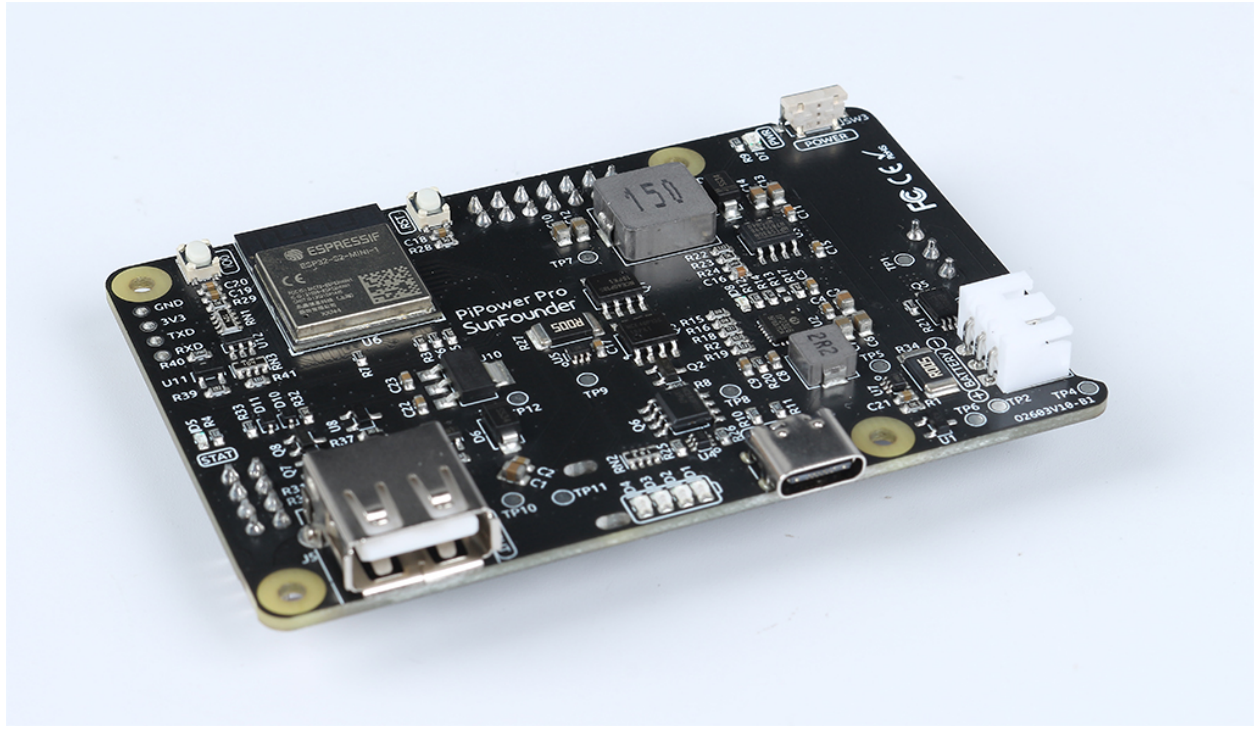
1	Komponentenliste	3
2	PiPower zusammenbauen	5
3	Merkmale	13
3.1	Detaillierte Einführung	14
3.2	Batterieanzeigen	18
4	Erste Schritte	19
4.1	HassOS installieren	19
4.2	PiPower Pro in Home Assistant hinzufügen	29
4.3	Dashboard konfigurieren	35
4.4	Karte mittels des Code-Editors hinzufügen	37
4.5	PiPower Pro Entität	44
4.6	Einrichten des sicheren Herunterfahrens	45
4.7	Coulomb-Zähler (Beta)	54
4.8	Maßgeschneiderte Entwicklung	55
4.9	Mehrere PiPower Pro Einheiten	56
4.10	IO-Erweiterung	57
5	FAQ	59
5.1	PiPower Pro funktioniert nicht?	59
5.2	Mit welchen Einplatinencomputern kann PiPower Pro verwendet werden?	59

Danke, dass Sie sich für unser PiPower entschieden haben.

Bemerkung: Dieses Dokument ist in den folgenden Sprachen verfügbar.

-
-
-

Bitte klicken Sie auf die jeweiligen Links, um das Dokument in Ihrer bevorzugten Sprache aufzurufen.



Was macht eine USV?

Wenn Ihr Raspberry Pi Projekt eine konstante Stromversorgung benötigt, ist es nicht ratsam, sich nur auf das Hauptstromsystem zu verlassen. Je nach Standort können Stromausfälle und -spitzen häufig auftreten und oft stundenlang andauern. Jegliche Stromschwankungen können Ihren Raspberry Pi beschädigen, und ein Stromausfall wird ihn sofort ausschalten. Infolgedessen wird er nicht sicher herunterfahren, was dazu führen kann, dass alle Daten auf der SD-Karte verloren gehen und die Wahrscheinlichkeit erhöht, dass sie zerstört wird.

Deshalb wird die Verwendung einer unterbrechungsfreien Stromversorgung (USV) empfohlen.

Mit einer USV übernimmt bei einem Stromausfall vom Hauptnetz die Batterie oder eine andere Energiequelle und versorgt das Gerät weiterhin ohne es auszuschalten. Eine USV wird oft als Notstromquelle betrachtet. Nachdem die Hauptstromquelle wiederhergestellt ist, wird die USV wieder aufgeladen und ist bereit für den nächsten Notfall.

Über PiPower

Genau aus diesem Grund haben wir PiPower entwickelt. PiPower kann als zweite Energiequelle für den Raspberry Pi verwendet werden. Ein USB-C Netzteil, das an den PiPower angeschlossen ist, versorgt den Raspberry Pi direkt mit Strom und lädt gleichzeitig die Batterie mit geringem Strom auf. PiPower kann einen Raspberry Pi bei einem Stromausfall oder bei Unterbrechung der USB-C Hauptversorgung nahtlos mit Energie versorgen.

PiPower liefert eine 5V/3A Stromversorgung, um verschiedene Nutzungsszenarien des Raspberry Pi zu decken. Es verfügt über 4 Energieindikatoren; jeder Indikator repräsentiert 25% der Energie und ist mit einem Schalter ausgestattet,

um den Raspberry Pi ein-/auszuschalten, ohne das Netzkabel ein- oder auszustecken.

Warnung: Wenn Sie den Akku zum ersten Mal einlegen oder wenn er entfernt und erneut eingesetzt wird, funktioniert der Akku nicht ordnungsgemäß. In diesem Fall müssen Sie das Typ-C-Kabel in den Ladeanschluss stecken, um den Schutzschaltkreis auszuschalten, und der Akku kann dann normal verwendet werden.

Über PiPower Pro

PiPower Pro baut auf PiPower auf und integriert ein ESP32 S2 Modul, das eine Echtzeitüberwachung der Batteriespannung und des aktuellen Zustands des Moduls, Eingangs-/Ausgangsspannung und -strom ermöglicht. Es verfügt zudem über eine intelligente Ladestromanpassung und einen nahtlosen Wechsel zwischen Eingangs- und Batteriestrom, um eine kontinuierliche Stromversorgung zu gewährleisten.

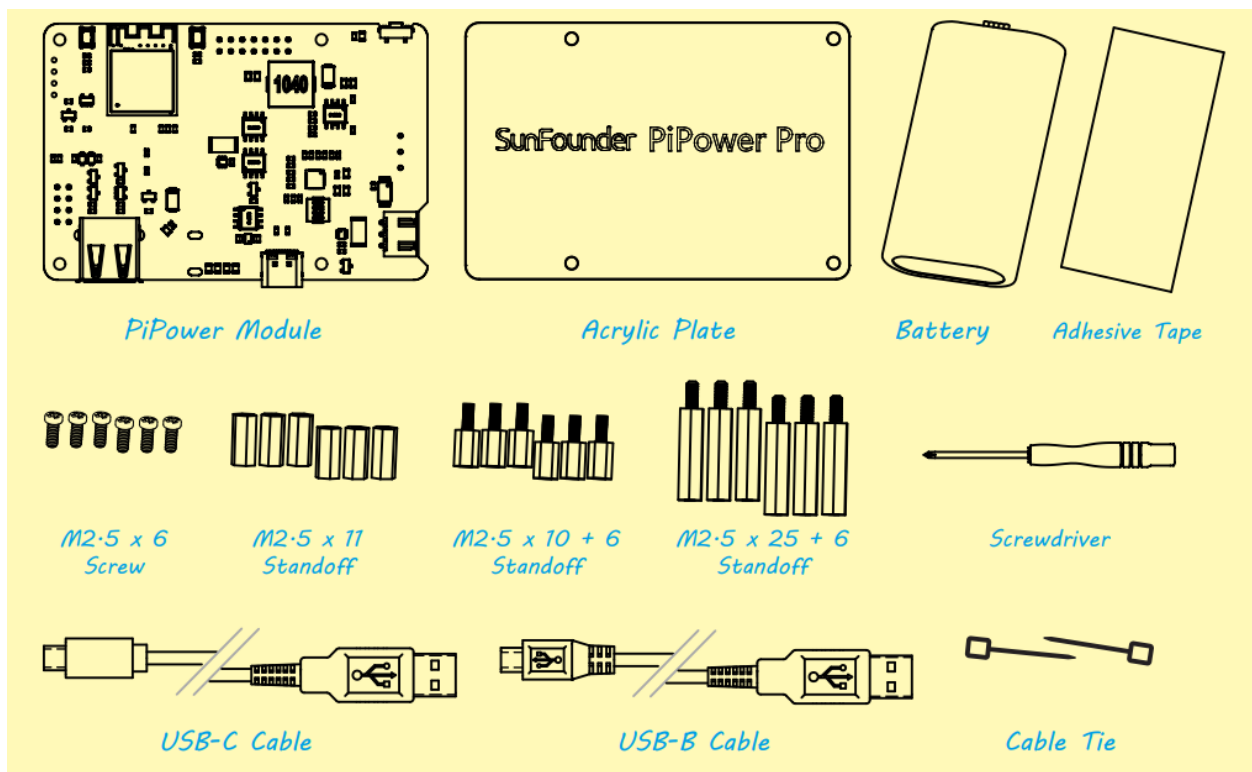
In Verbindung mit Home Assistant können Benutzer problemlos auf alle Parameterdaten zugreifen und diese überprüfen sowie Automatisierungen für intelligente Geräteszenarien konfigurieren.

Darüber hinaus bietet PiPower Pro externe IO-Schnittstellen zur Steuerung des Ein-/Aus-Zustands von Untergeräten. Mit seiner Open-Source-ESPHome-Konfiguration können Benutzer IO-Funktionalitäten anpassen und das System mit weiteren Sensoren erweitern.

Sollten Sie Fragen bei der Nutzung unseres Produkts haben, senden Sie bitte eine E-Mail an service@sunfounder.com. Wir werden so schnell wie möglich antworten.

KAPITEL 1

Komponentenliste

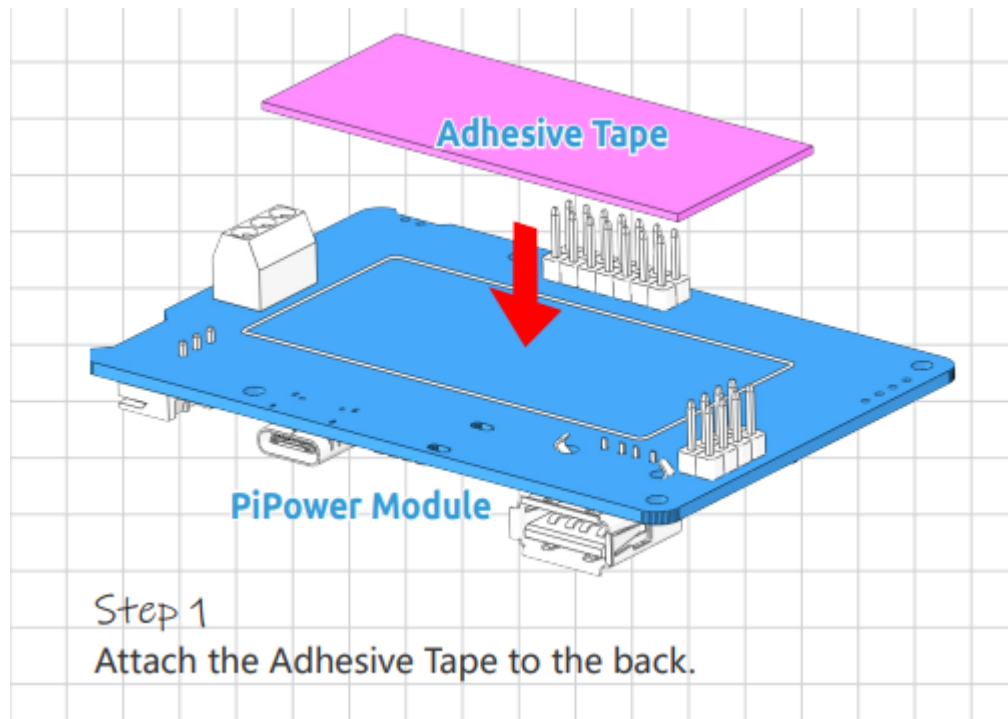


PiPower zusammenbauen

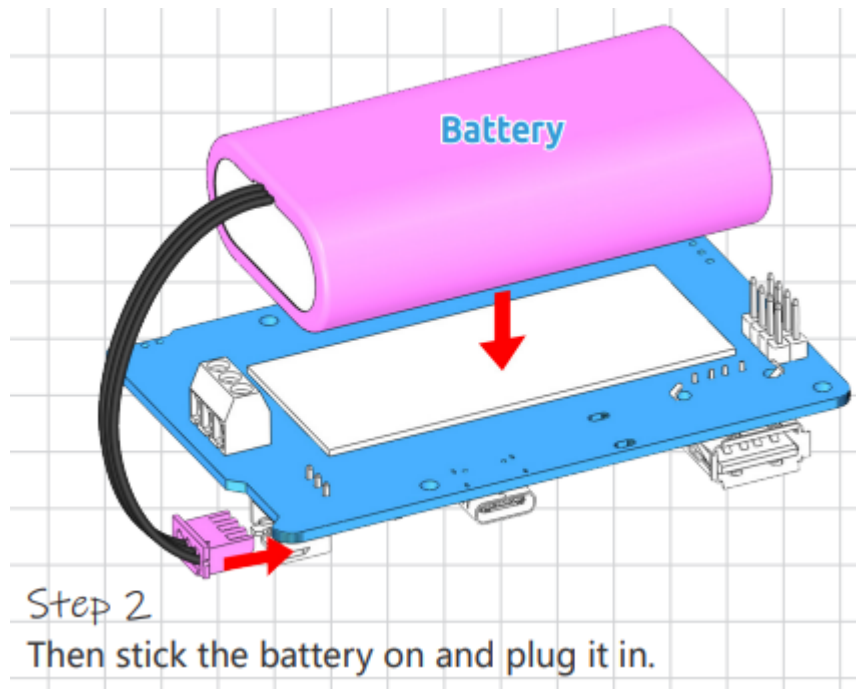
Nachdem Sie sich mit den Komponenten im Paket vertraut gemacht haben, beginnen wir mit dem Zusammenbau von PiPower.

In den folgenden Schritten gibt es viele Details, auf die Sie achten müssen, insbesondere auf die Montageposition des Akkus und der durchsichtigen Acryl-Rückabdeckung.

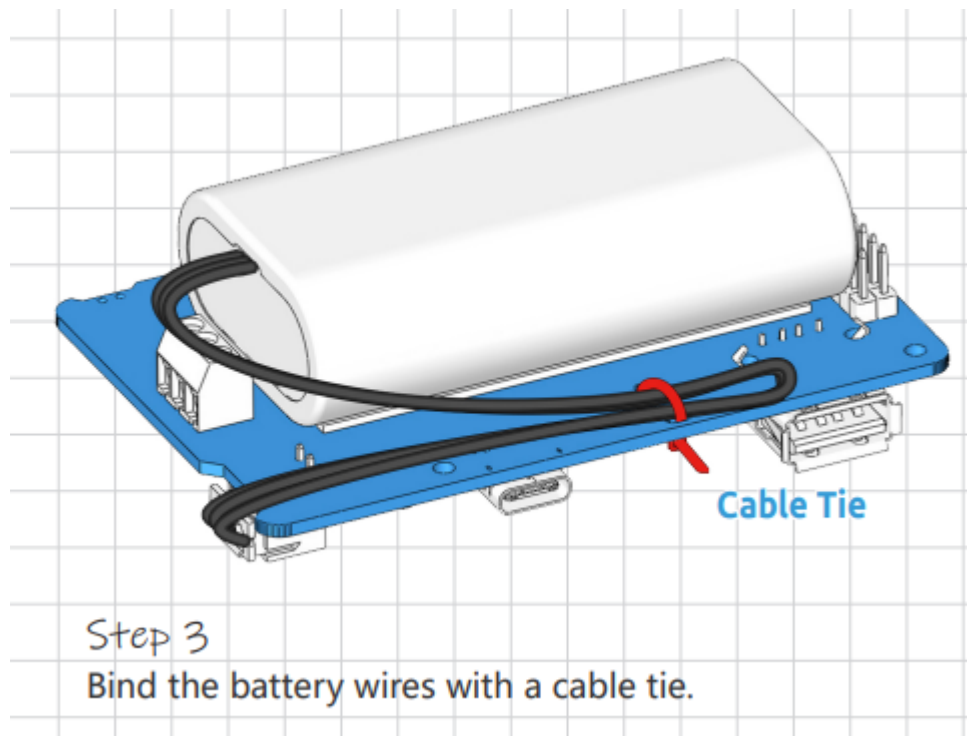
1. Kleben Sie das Klebeband auf die Rückseite.



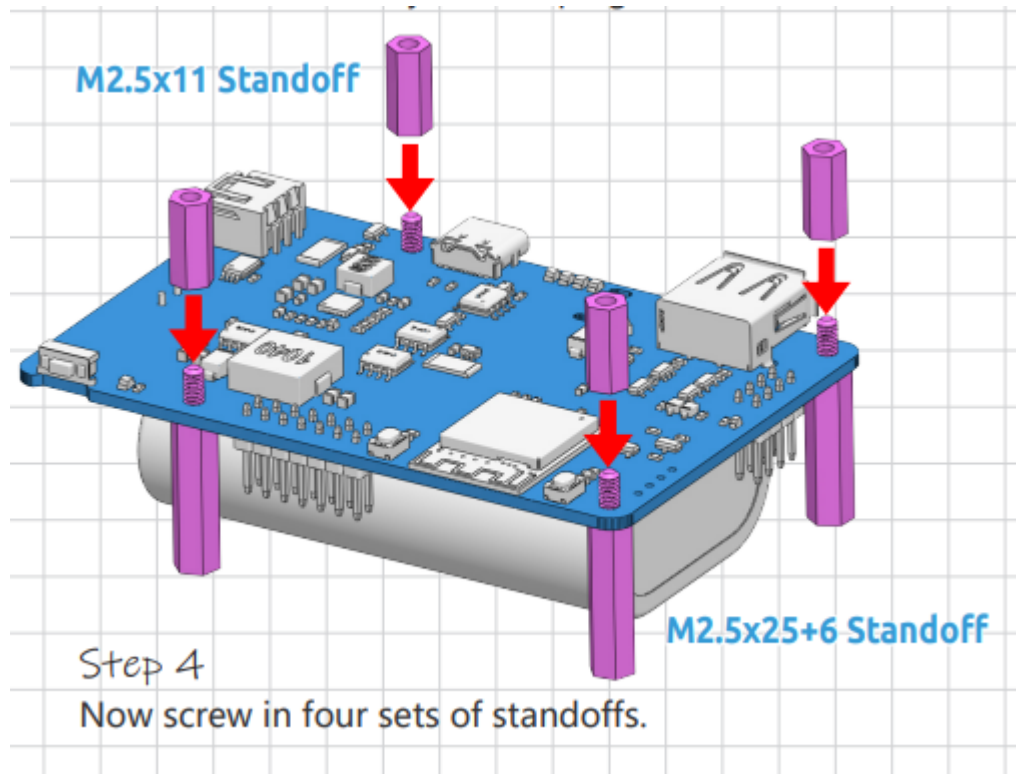
2. Dann kleben Sie den Akku darauf und schließen Sie ihn an.



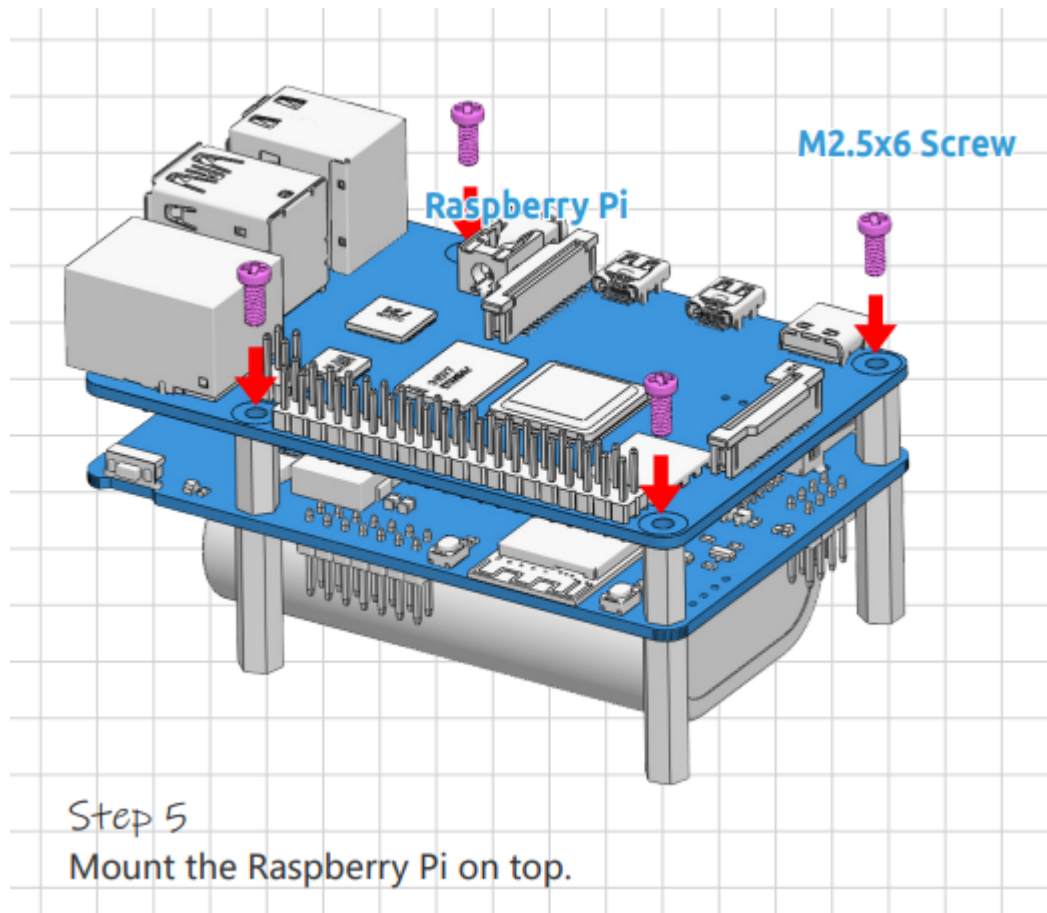
3. Bündeln Sie die Akkukabel mit einem Kabelbinder.



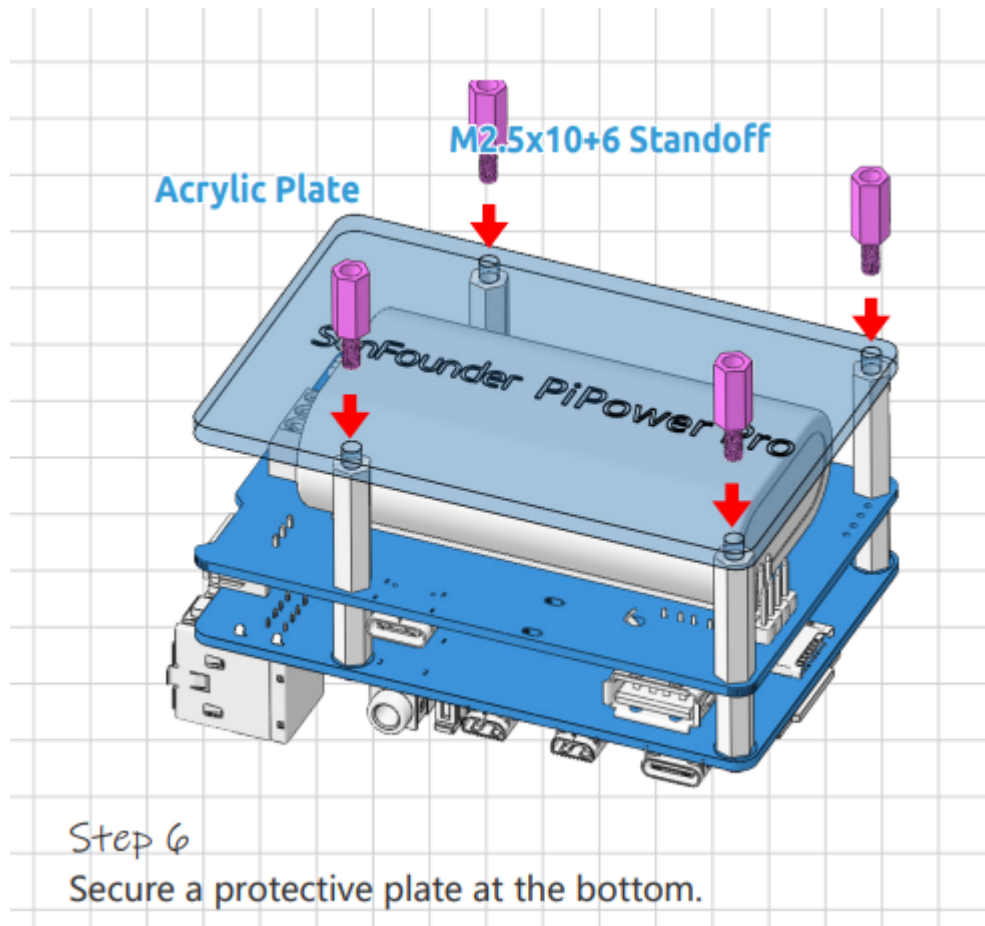
4. Nun schrauben Sie vier Abstandshalter-Sets ein.



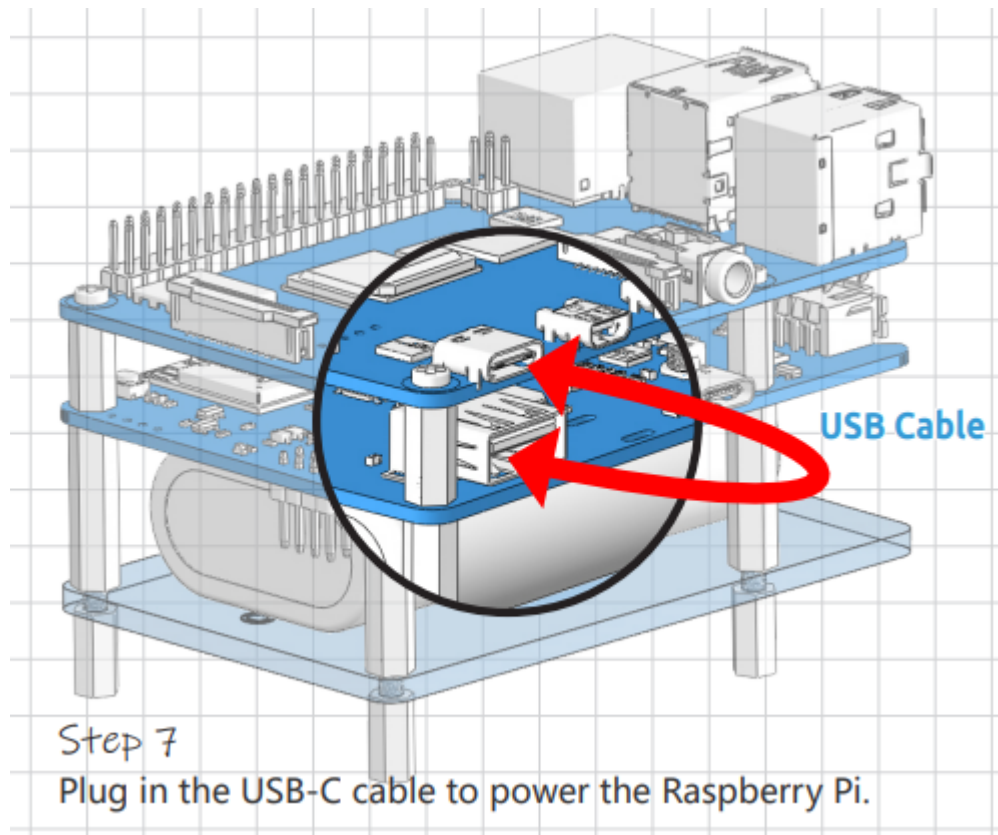
5. Montieren Sie den Raspberry Pi oben drauf.



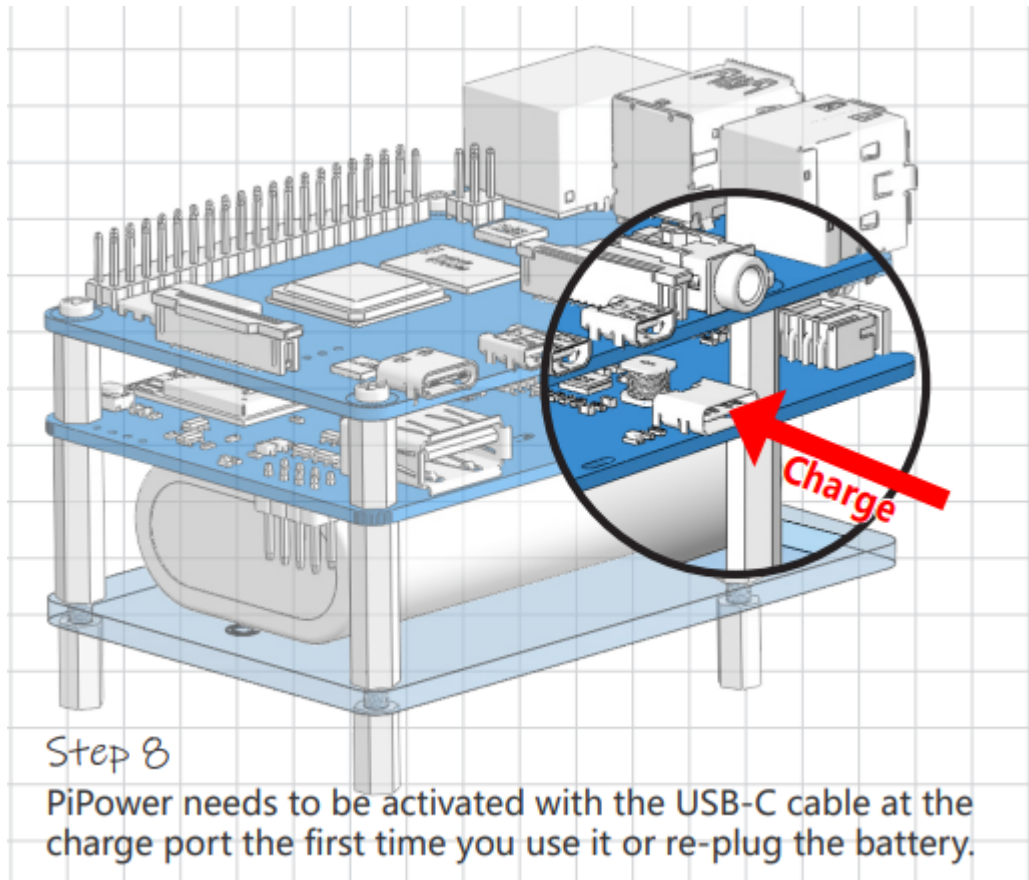
6. Befestigen Sie eine Schutzplatte an der Unterseite.



7. Schließen Sie das USB-C-Kabel an, um den Raspberry Pi mit Strom zu versorgen.

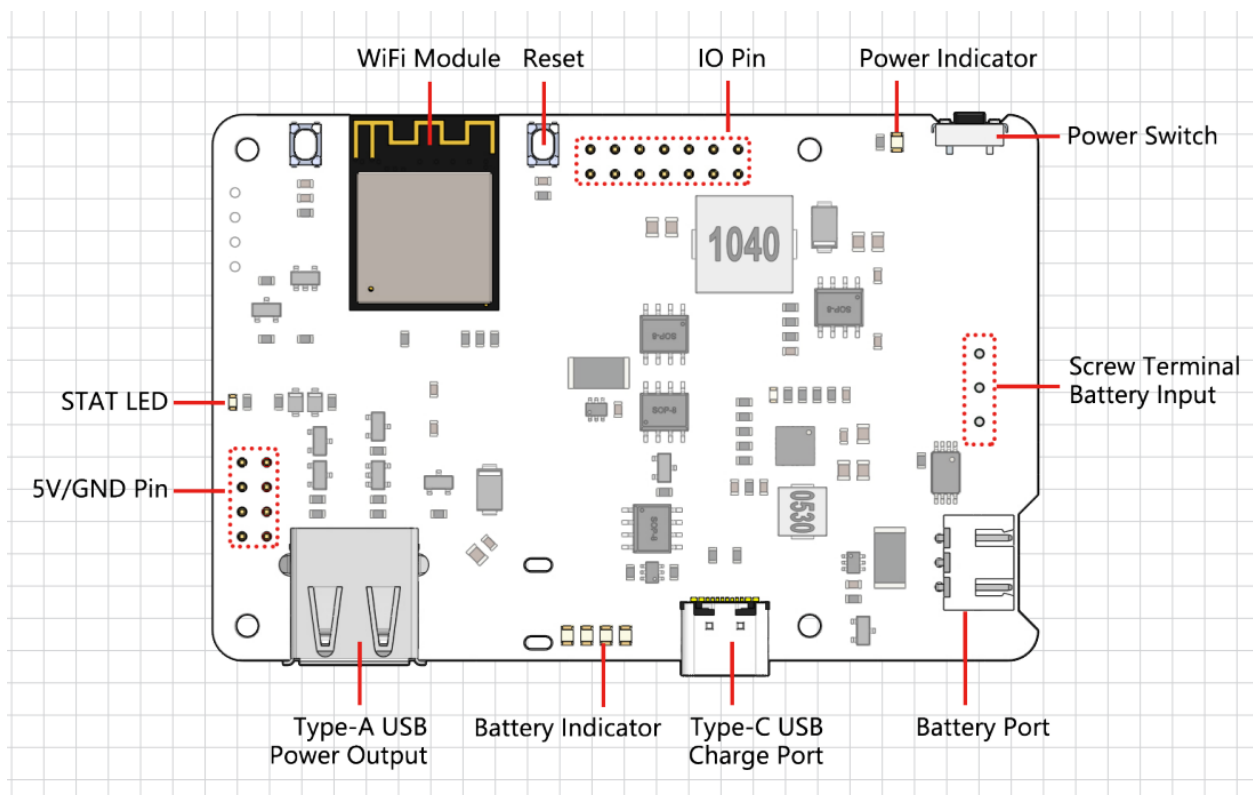


8. PiPower muss beim ersten Gebrauch oder beim erneuten Anschluss des Akkus mit dem USB-C-Kabel am Ladeanschluss aktiviert werden.



Warnung: Wenn Sie den Akku zum ersten Mal einlegen oder wenn er entfernt und erneut eingesetzt wird, funktioniert der Akku nicht ordnungsgemäß. In diesem Fall müssen Sie das Typ-C-Kabel in den Ladeanschluss stecken, um den Schutzschaltkreis auszuschalten, und der Akku kann dann normal verwendet werden.

Merkmale



- **Durchladefunktion**
- **Abschaltstrom < 0.5mA**
- **Eingang:**
 - USB Typ-C, 5V/3A
 - Batterie-Eingang

- **Ausgang:**
 - USB Typ-A, 5V/3A
 - 2x4P P2.54 Pin-Header
- **Ladeleistung: 5V/2A**
- **Integrierter Akku:**
 - Typ: 3,7V Lithium-Ionen-Akkus x 2
 - Kapazität: 2000mAh
 - Anschluss: PH2.0, 3P
- **Überentladungsschutzspannung: 6,0V**
- **Überladeschutzspannung: 8,4V**
- **Abmessungen: 90mm x 60mm x 24,9mm**
- **Bordindikatoren:**
 - 1 x Ladeanzeige (CHG)
 - 1 x Netzindikator (PWR)
 - 4 Batterieindikatoren (D4 ~ D7)
- **Bordnetzschalter**
- **On-board MCU ESP32 S2**

3.1 Detaillierte Einführung

STAT LED

Die STAT LED ist die Statusanzeige für den ESP32 S2.

- Aus: Der ESP32 S2 ist ausgeschaltet.
- Langsames Blinken: Der ESP32 S2 ist eingeschaltet, aber das WLAN ist nicht verbunden.
- Dauerhaft eingeschaltet: Der ESP32 S2 ist eingeschaltet und das WLAN ist verbunden.

Bemerkung: Der sogenannte „ESP32 S2 ausgeschaltet“ Zustand bezieht sich auf die Situation, wenn die USB Type C-Stromversorgung angeschlossen ist. In diesem Zustand ist der ESP32 S2 technisch „ausgeschaltet“, aber nicht vollständig heruntergefahren. Die Strom-LED benötigt immer noch den ESP32 S2, um ihre Beleuchtung zu steuern, und einige Funktionen können weiterhin betriebsbereit sein. Wenn Sie jedoch die USB Type C-Stromversorgung trennen, wird der ESP32 S2 vollständig heruntergefahren.

Umschaltung des Strompfads

Das PiPower Pro integriert eine Strompfad-Funktion, die automatisch die Strompfade umschaltet, um maximalen Ausgangsschutz zu gewährleisten.

1. Wenn eine externe Stromversorgung angeschlossen ist, wird die 5V-Ausgabe direkt über den externen USB Typ C geliefert und kann über den Schalter ausgeschaltet werden. Die externe Stromquelle lädt den Akku mit so viel Strom wie möglich (siehe Lade-Stromstärke), während sichergestellt wird, dass die Eingangsspannung größer als 4,6V ist.

2. Im Moment der Stromtrennung schaltet das System automatisch nahtlos auf Akkustrom um und stellt sicher, dass das System bei Stromausfällen ordnungsgemäß funktionieren kann.
3. Wenn die externe Stromversorgung unter 4,6V liegt, schaltet das System automatisch auf die Akku-Backup-Stromversorgung um, um zu verhindern, dass externe Geräte den Strom verlieren.

Tab. 1: Logik der Ausgangsleistung

Schalter	Externe Stromversorgung	Ausgangsstatus
An	Angeschlossen	Externe Stromversorgung
An	Nicht angeschlossen oder Spannung unter 4,6V	Akku-Stromversorgung
Aus	Angeschlossen	Aus
Aus	Nicht angeschlossen oder Spannung unter 4,6V	Aus

Ladeleistung

Im eingeschalteten Zustand wird der Ladestrom automatisch anhand der Eingangsspannung angepasst.

Tab. 2: Logik des Ladestroms

Schalter	Ladestrom
An	Anhand der Eingangsspannung angepasst
Aus	2A

1. Wenn der Schalter auf „Aus“ steht, liefert das PiPower Pro keinen externen Strom, und der Ladestrom erreicht ein Maximum von 2A, was ein schnelles Laden ermöglicht. Die Ladezeit von 0% auf 100% beträgt ungefähr 2 Stunden und 10 Minuten.
2. Im „An“-Zustand, da das PiPower Pro externen Strom liefern muss, muss der externe USB auch Strom für den Akku liefern. Um sicherzustellen, dass die Spannung der USB-Stromversorgung stabil bleibt, wird der Ladestrom anhand der Eingangsspannung angepasst, wodurch die Spannung über 4,6V bleibt.

Tiefentladeschutz

Wenn die Spannung einer einzelnen Batterie unter 3V fällt, wird der Batterieschutz aktiviert und die Batterie nicht weiter entladen.

Wenn die Batterie ausgesteckt wird, aktiviert der an Bord befindliche Tiefentladeschutzschaltkreis aufgrund seiner Mechanik den Schutz, da die Spannung als zu niedrig betrachtet wird. Wenn Sie die Batterie wieder an das PiPower anschließen, funktioniert die Batterie nicht ordnungsgemäß. In diesem Fall müssen Sie das Type C-Kabel in den Ladeanschluss stecken, um den Schutzschaltkreis auszuschalten, und die Batterie kann dann normal verwendet werden.

Überladeschutz

Das Laden endet, wenn die Gesamtspannung der Batterie 8,4V erreicht.

Ladeausgleich

Wenn die Spannungen der beiden Batterien nicht gleich sind, wird der Ladestrom der beiden Batterien automatisch angepasst, um die beiden Batterien auszugleichen.

Wenn eine einzelne Batterie 4,2V übersteigt, leitet der Spannungsteiler-Widerstandskanal und der Ladestrom der Batterie wird verringert oder sogar entladen.

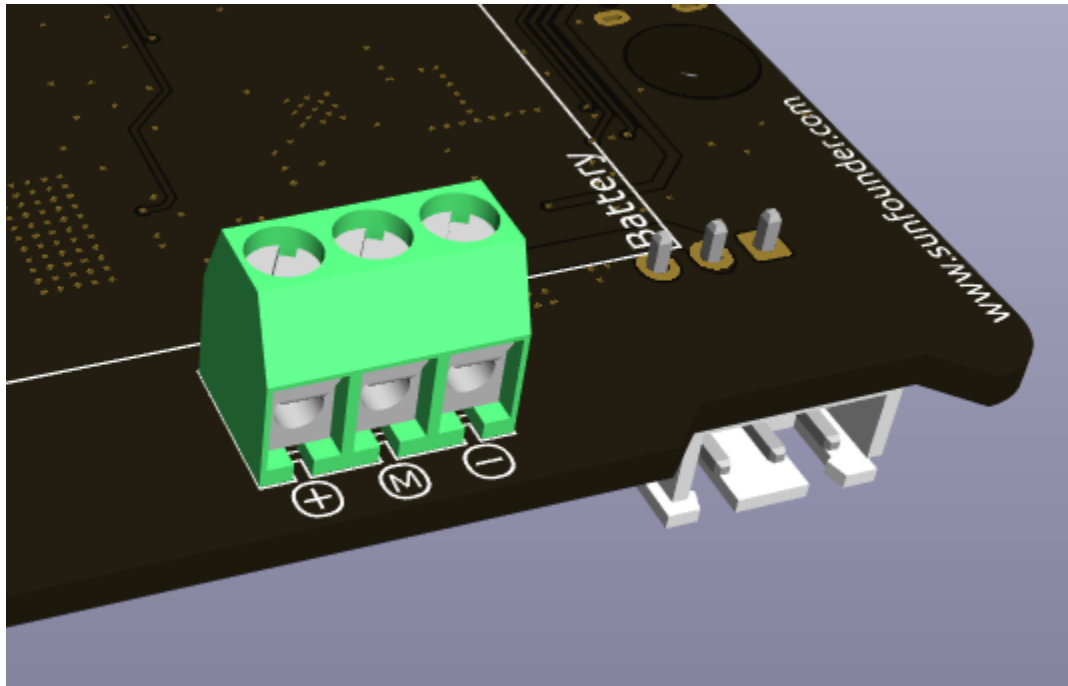
Batterie

Das Produkt wird mit zwei in Reihe geschalteten 3,7V 18650 Lithium-Ionen-Batterien geliefert, die über einen XH2.54 3P-Anschluss verfügen und eine Nennkapazität von 2000mAh haben.

- Zusammensetzung: Li-ion (Lithium-Ion)

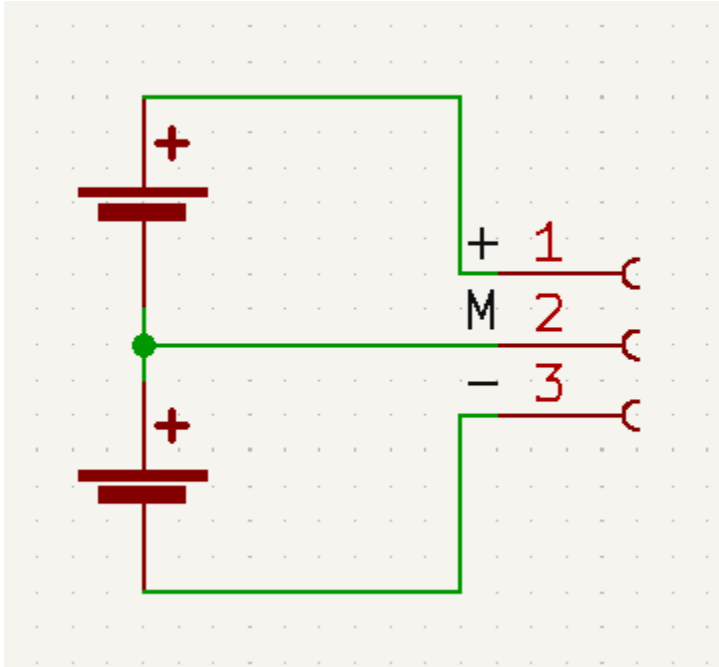
- Kapazität: 2000mAh, 14,8Wh
- Gewicht: 90,8g
- Zellen: 2
- Anschluss: XH2.54 3P
- Überladeschutzspannung: 4,2V pro Zelle
- Tiefentladeschutz: 3V

Externe Batterie



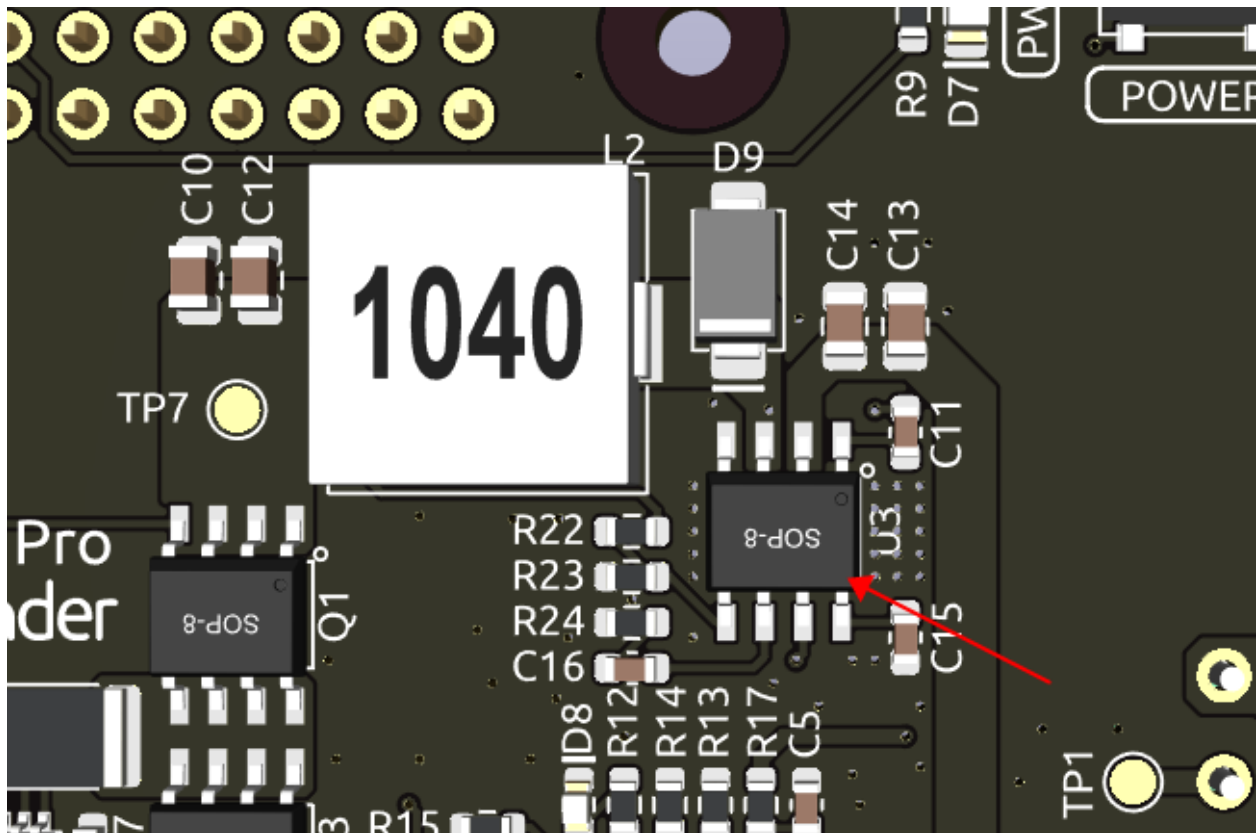
Sie können Ihre eigene Batterie über den Schraubanschluss anschließen. Das Gerät unterstützt nur zwei 3,7V Lithium-Ionen- oder Lithium-Polymer-Batterien. Es ist vorzuziehen, dass die Batterien über eine Schutzplatine verfügen und eine Ausgabe von mehr als 15W sicherstellen.

Warnung: Schließen Sie nicht gleichzeitig die externe Batterie und die im Lieferumfang enthaltene Batterie an!



Temperatur

Wenn die Ausgangsleistung den maximalen Nennwert von 5V/3A erreicht, steigt die Temperatur des DC-DC-Abwärtswandlers U1 auf etwa 70-80 Grad Celsius. Seien Sie also vorsichtig und berühren Sie ihn nicht, um Verbrennungen zu vermeiden und für Belüftung zu sorgen. Wenn die Temperatur die DC-DC-Schutztemperatur von 75 Grad Celsius erreicht, schaltet sich der DC-DC ab, um Überhitzungsschäden zu vermeiden.



D8 LED

Die D8 LED ist eine Ladezustandsanzeige, die vom IP2326-Ladechip bereitgestellt wird. Ursprünglich wurde diese Leuchte entwickelt, um sowohl den Ladezustand als auch eventuelle Anomalien der Batterie anzuzeigen. Sie kann jedoch nur erkennen, ob im Ladeausgang ein Stromfluss vorhanden ist. Dieser Ausgangsstrom kann über einen DC-DC-Wandler umgeleitet werden, um 5V auszugeben. Einfach ausgedrückt: Wenn nicht genügend Eingangsleistung vorhanden ist, ergänzt die Batterie die Stromversorgung, und währenddessen bleibt die LED ständig eingeschaltet, was irreführend sein kann. Die LED wurde jedoch beibehalten, da sie anzeigen kann, ob die Batterie normal funktioniert (die LED blinkt, wenn die Batterie nicht eingesetzt ist).

3.2 Batterieanzeigen

Das Verhältnis zwischen den Batterieanzeigen und der Spannung ist wie folgt:

- 4 LEDs alle eingeschaltet: Spannung > 7,7V
- 3 LEDs eingeschaltet: Spannung > 7,2V
- 2 LEDs eingeschaltet: Spannung > 6,7V
- 1 LED eingeschaltet: Spannung > 6,4V
- 4 LEDs alle ausgeschaltet: Spannung < 6V, zu diesem Zeitpunkt müssen die Batterien geladen werden.

PiPower Pro kann in Home Assistant integriert werden. Um dies zu tun, benötigen Sie einen Raspberry Pi mit HassOS installiert. Bitte folgen Sie dem untenstehenden Link für die Einrichtung.

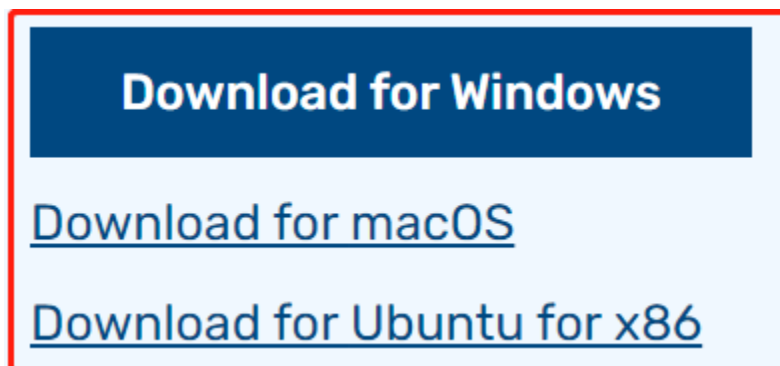
4.1 HassOS installieren

Dieser Abschnitt wird Sie durch die Installation des Home Assistant Betriebssystems auf Ihrem Raspberry Pi führen. Bitte beachten Sie, dass durch diesen Prozess alle vorhandenen Inhalte auf Ihrem Raspberry Pi System gelöscht werden. Es ist wichtig, Ihre Daten vorher zu sichern.

Schritt 1

Raspberry Pi hat ein grafisches SD-Karten-Schreibprogramm entwickelt, das unter Mac OS, Ubuntu 18.04 und Windows läuft und für die meisten Benutzer die einfachste Option ist, da es das Image automatisch herunterlädt und auf die SD-Karte installiert.

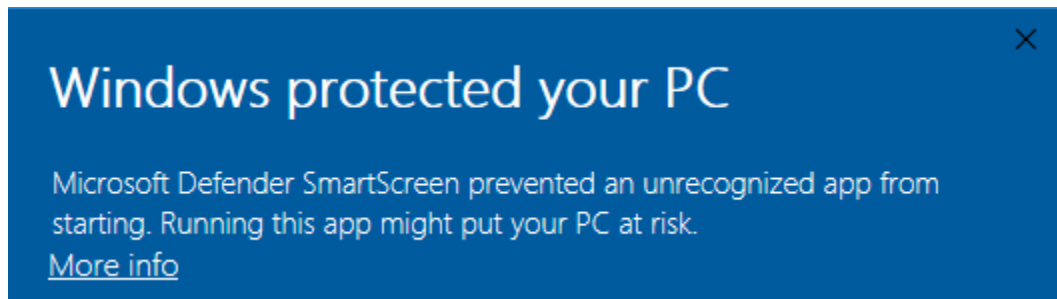
Besuchen Sie die Download-Seite: <https://www.raspberrypi.org/software/>. Klicken Sie auf den Link für den **Raspberry Pi Imager**, der zu Ihrem Betriebssystem passt. Nachdem der Download abgeschlossen ist, klicken Sie darauf, um den Installer zu starten.



Schritt 2

Wenn Sie den Installer starten, kann Ihr Betriebssystem versuchen, die Ausführung zu blockieren. Zum Beispiel erhalten Sie unter Windows folgende Meldung:

Erscheint diese Meldung, klicken Sie auf **Weitere Informationen** und dann auf **Trotzdem ausführen** und folgen Sie den Installationsanweisungen des Raspberry Pi Imagers.

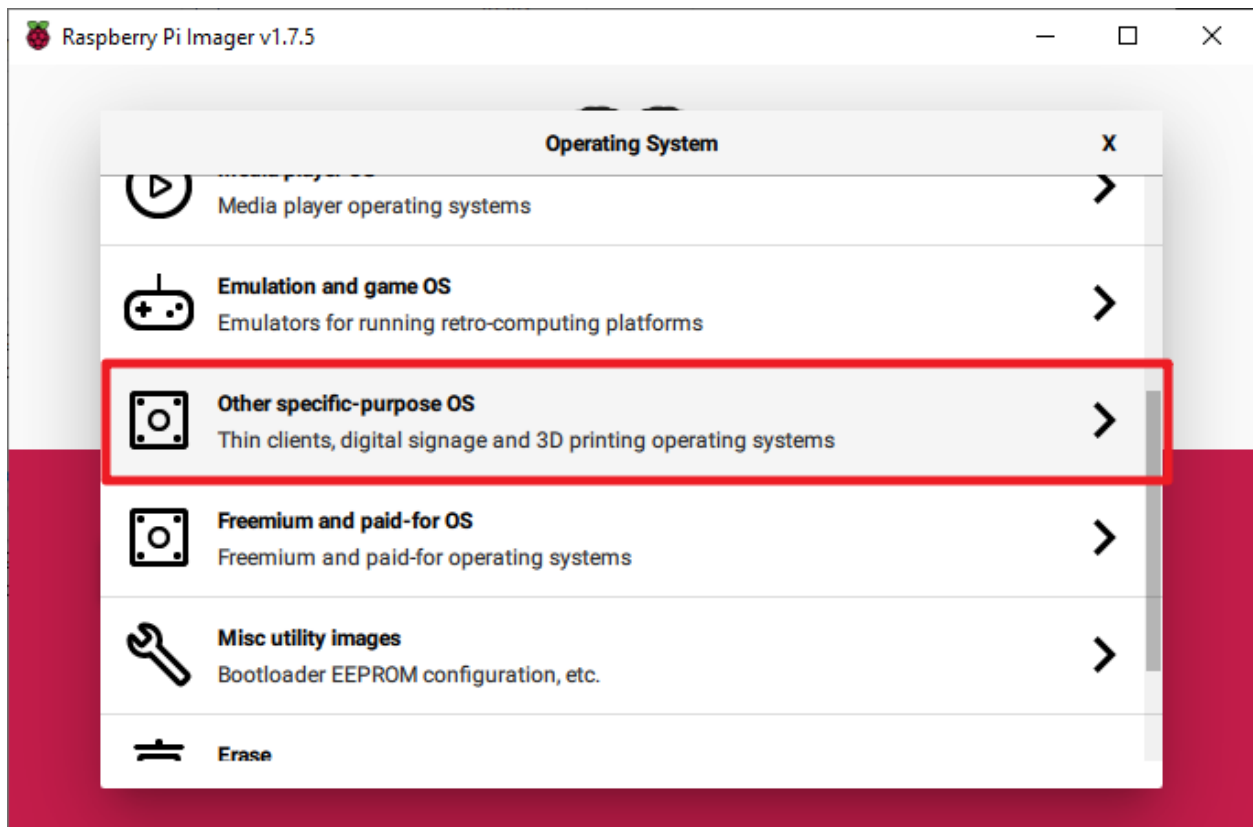


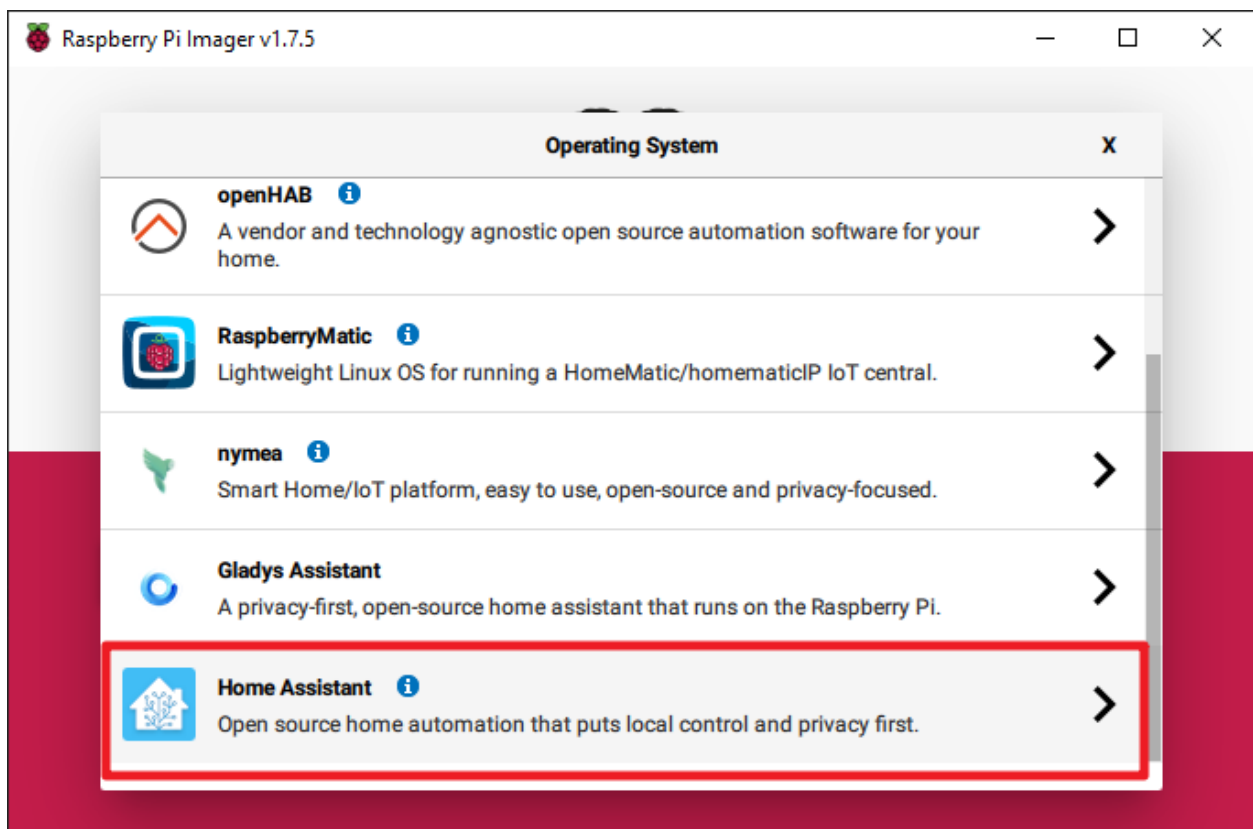
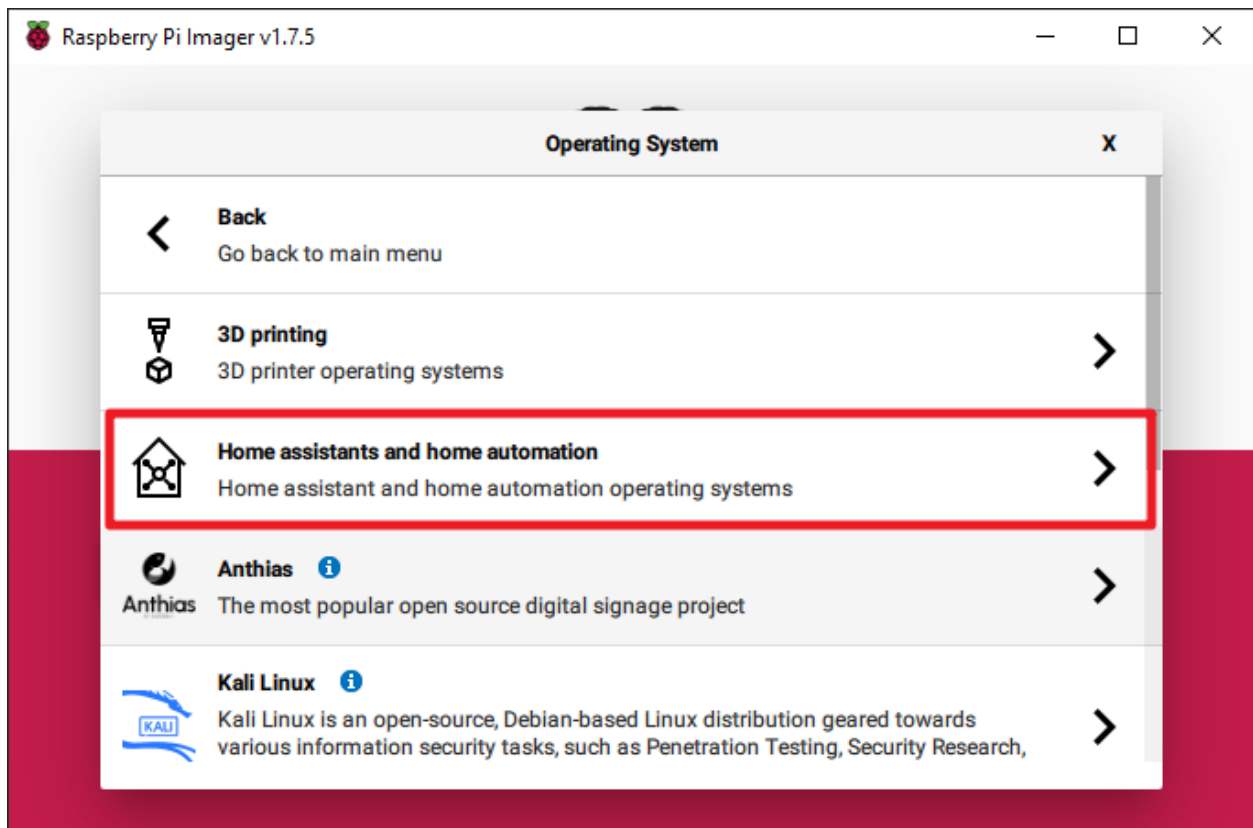
Schritt 3

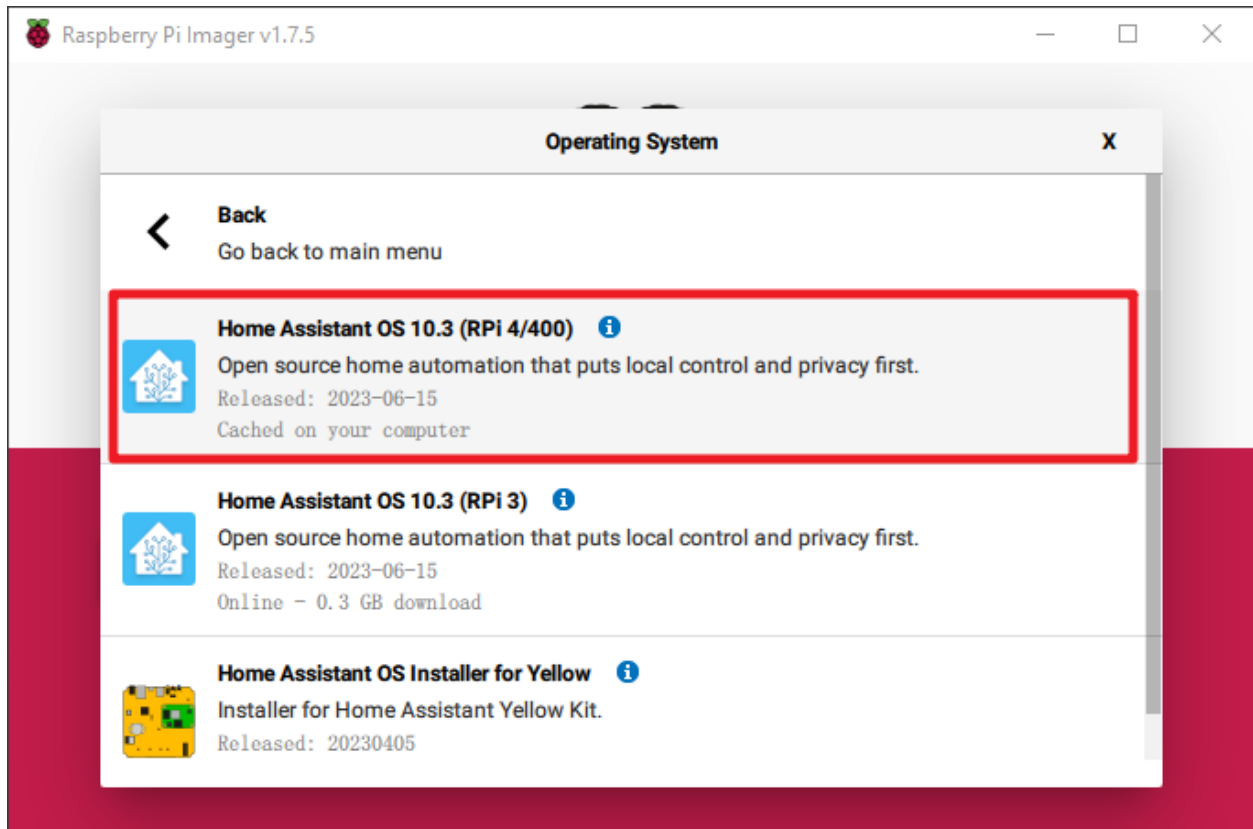
Stecken Sie Ihre SD-Karte in den SD-Karten-Slot Ihres Computers oder Laptops.

Schritt 4

Im Raspberry Pi Imager wählen Sie das Betriebssystem, das Sie installieren möchten, und die SD-Karte, auf die es installiert werden soll.

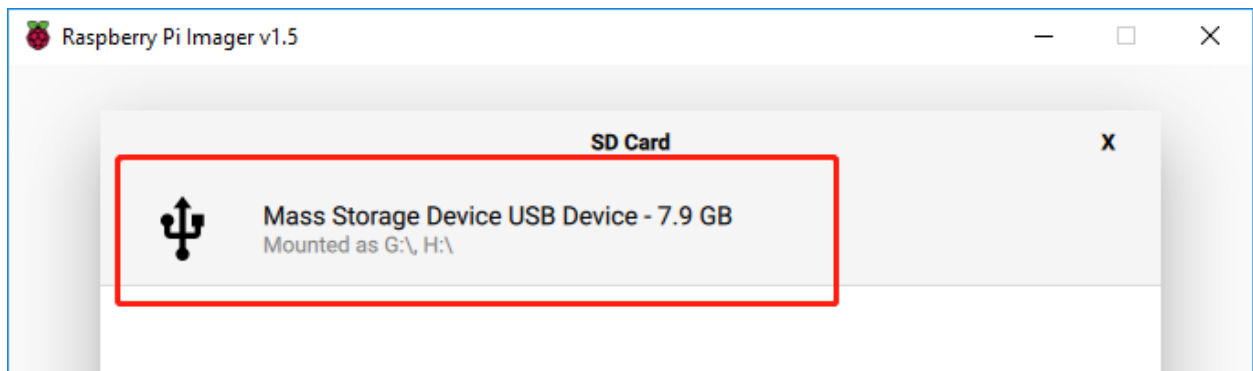






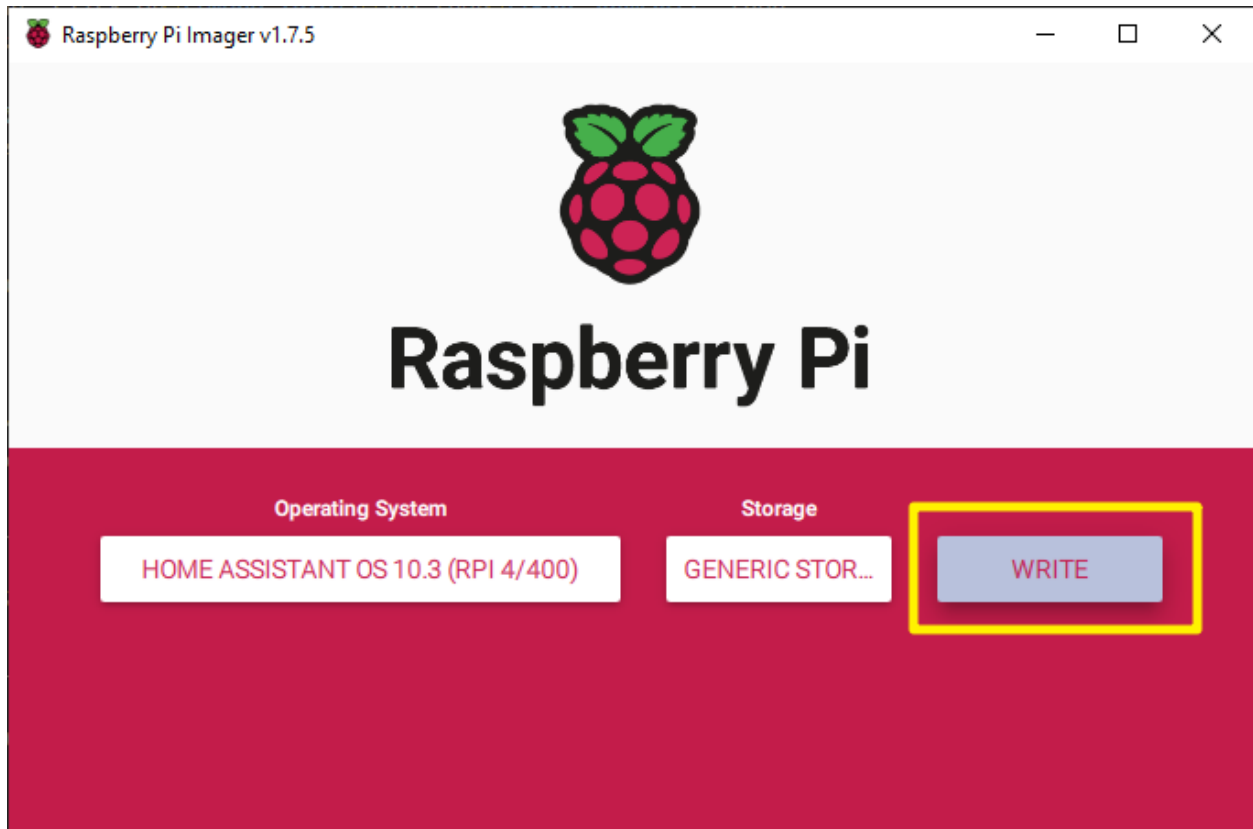
Schritt 5

Wählen Sie die SD-Karte, die Sie verwenden.



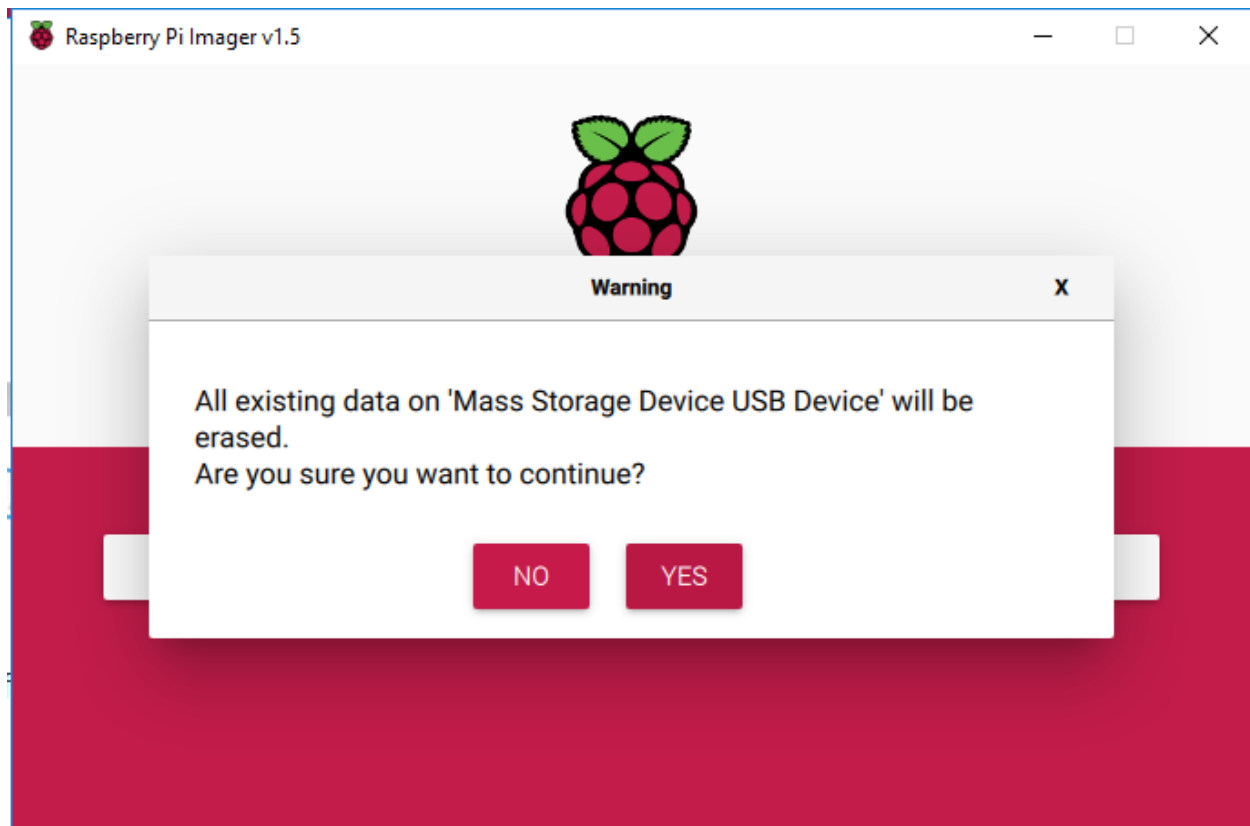
Schritt 6

Klicken Sie auf den **WRITE**-Knopf.



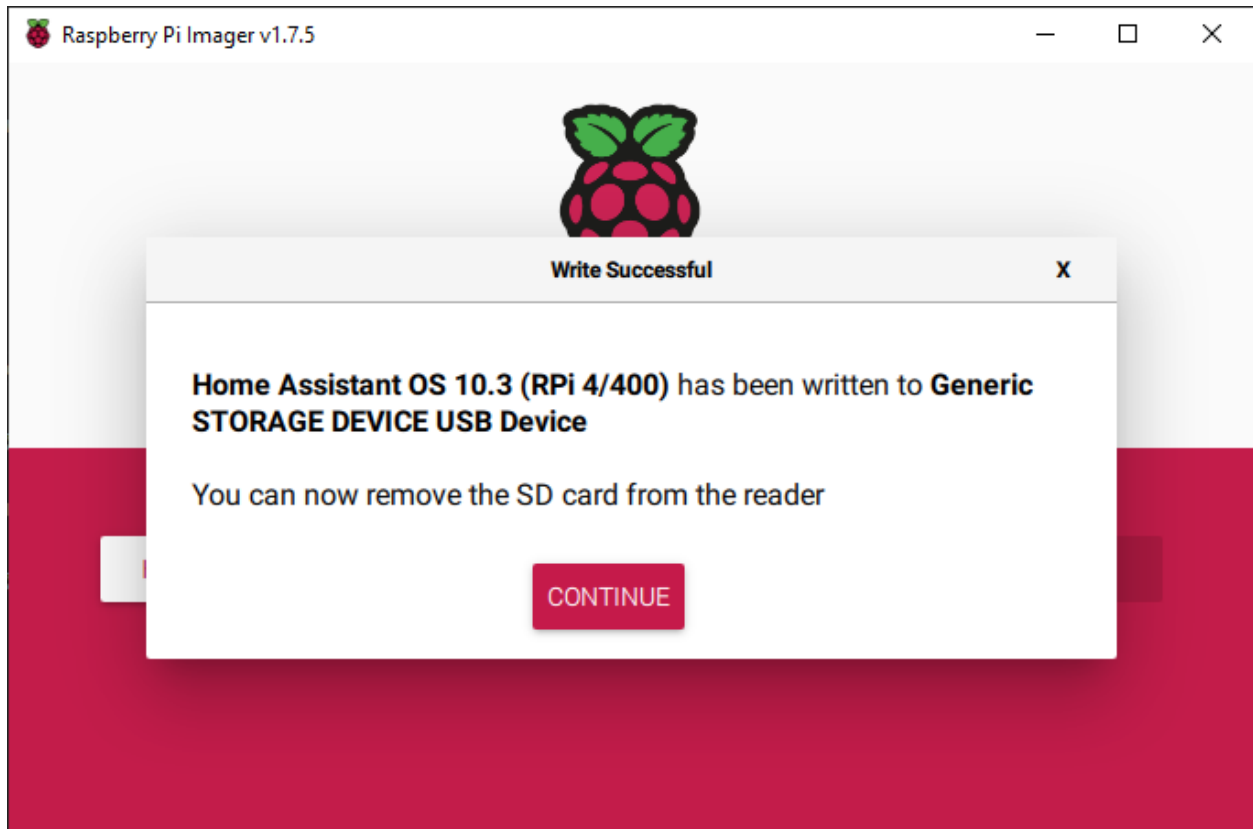
Schritt 7

Falls auf Ihrer SD-Karte aktuell Dateien gespeichert sind, sollten Sie diese eventuell zuerst sichern, um sie nicht dauerhaft zu verlieren. Wenn keine Dateien gesichert werden müssen, klicken Sie auf **Yes**.



Schritt 8

Nach einer Weile wird das folgende Fenster erscheinen, um das erfolgreiche Schreiben zu bestätigen.

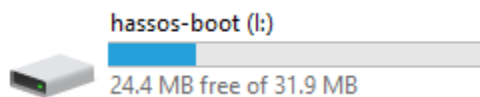


Schritt 9

Als Nächstes konfigurieren wir das WLAN für Pironman.

Bemerkung: Wenn Sie eine kabelgebundene Verbindung für den Netzwerkzugriff verwenden möchten, können Sie diesen Schritt überspringen.

Öffnen Sie den Datei-Explorer und greifen Sie auf die als Hassio-boot benannte SD-Karte zu.



Erstellen Sie im Hauptverzeichnis einen neuen Ordner namens CONFIG.



Innerhalb des CONFIG-Ordners erstellen Sie einen Ordner namens network.

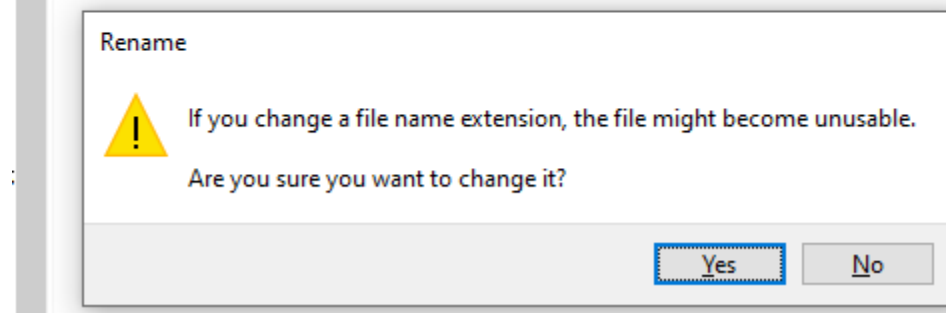
This PC > hassos-boot (I:) > CONFIG >

Name	Date modified	Type
modules	2023/6/28 10:10	File folder
network	2023/6/28 11:34	File folder

Innerhalb des network-Ordners erstellen Sie eine neue Textdatei namens my-network (ohne Dateiendung).

This PC > hassos-boot (I:) > CONFIG > network

Name	Date modified	Type
my-network	2023/6/28 11:34	Text Docu



In der Datei my-network schreiben Sie den folgenden Text und ersetzen dabei MY_SSID und MY_WLAN_SECRET_KEY durch den SSID und das Passwort Ihres Netzwerks:

```
[connection]
id=my-network
uuid=72111c67-4a5d-4d5c-925e-f8ee26efb3c3
type=802-11-wireless

[802-11-wireless]
mode=infrastructure
ssid=MY_SSID
# Entfernen Sie das Kommentarzeichen unten, wenn Ihr SSID nicht gesendet wird
#hidden=true

[802-11-wireless-security]
auth-alg=open
key-mgmt=wpa-psk
psk=MY_WLAN_SECRET_KEY

[ipv4]
method=auto

[ipv6]
addr-gen-mode=stable-privacy
method=auto
```

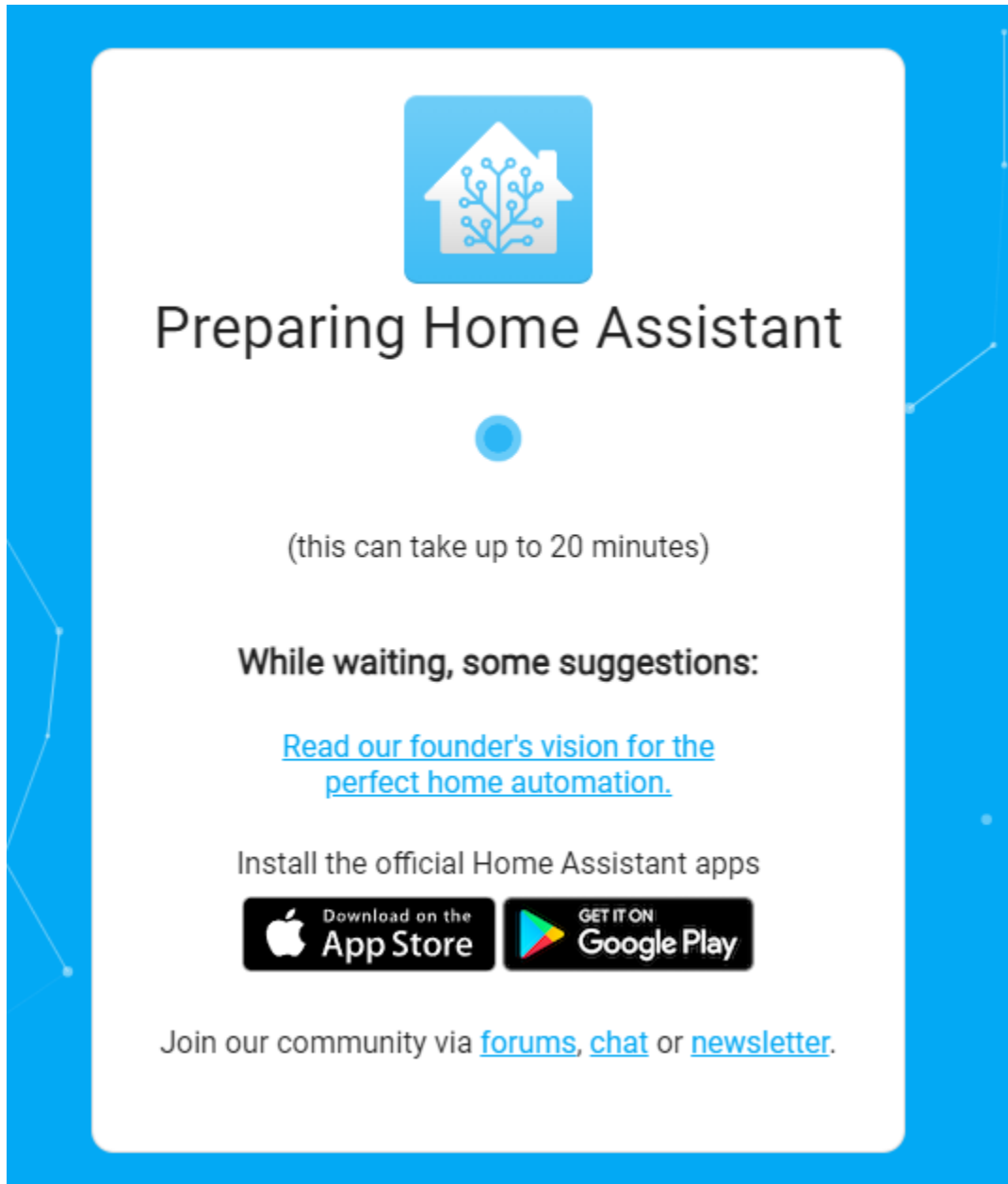
Speichern und schließen Sie die Datei.

Schritt 10

Entfernen Sie die microSD-Karte aus Ihrem Computer und stecken Sie sie in den Raspberry Pi. Schließen Sie dann den Strom an (und bei Bedarf das Ethernet-Kabel).

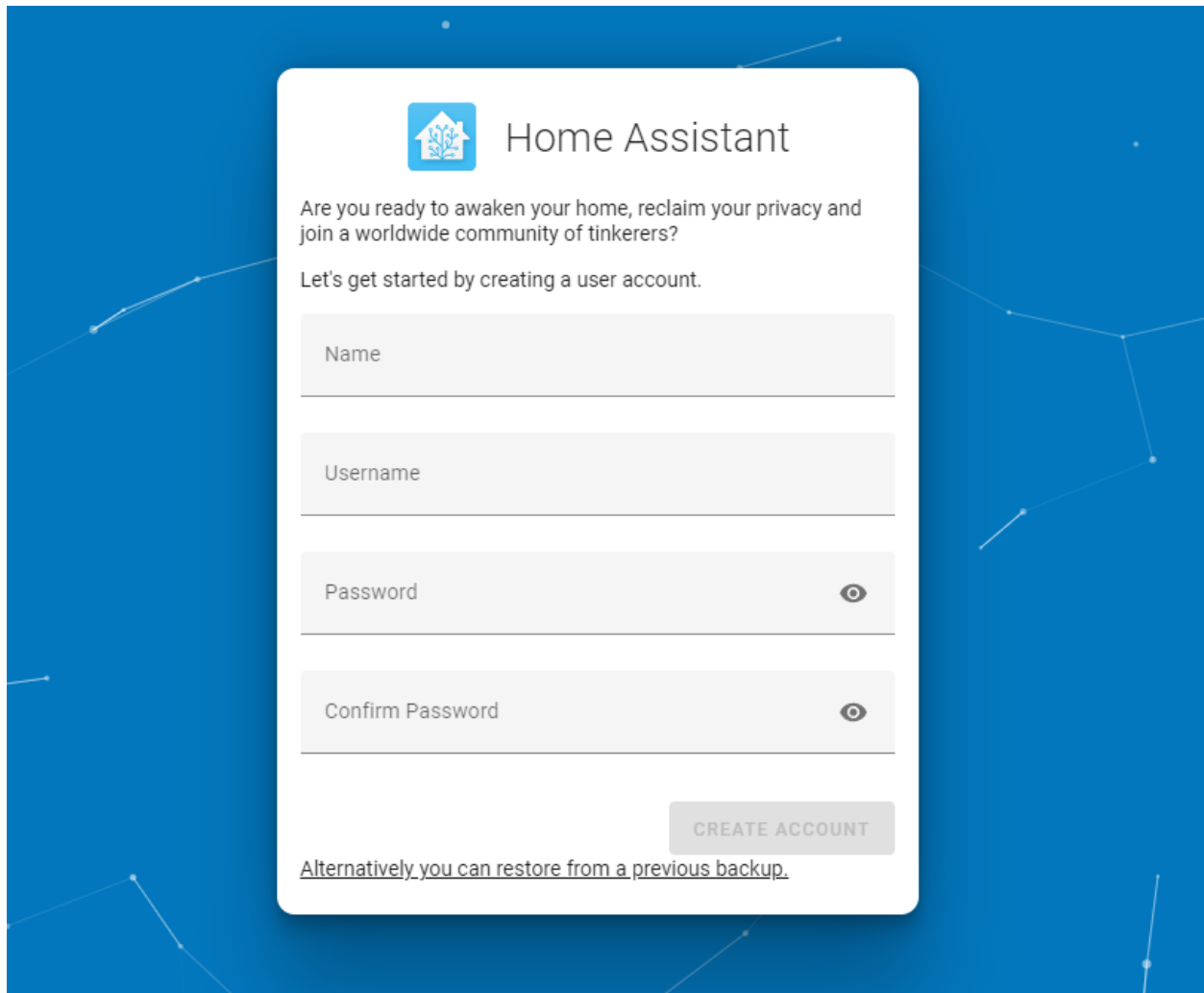
Gehen Sie zurück zu Ihrem Computer und rufen Sie `homeassistant.local:8123` auf. Wenn dies nicht funktioniert, können Sie die IP-Adresse über Ihren Router herausfinden.

Beim ersten Start von Home Assistant kann es eine Weile dauern, da die Ersteinrichtung durchgeführt wird.



Schritt 11

Als Nächstes werden Sie aufgefordert, das erste Konto zu erstellen.

The image shows the Home Assistant account creation interface. It features a blue background with a white central card. At the top of the card is the Home Assistant logo (a house with a plant) and the text 'Home Assistant'. Below this, a message asks if the user is ready to awaken their home and join a community of tinkerers, followed by a prompt to create a user account. There are four input fields: 'Name', 'Username', 'Password', and 'Confirm Password'. The 'Password' and 'Confirm Password' fields have eye icons to toggle visibility. A 'CREATE ACCOUNT' button is located at the bottom right of the card. Below the button, there is a link that says 'Alternatively you can restore from a previous backup.'

Home Assistant

Are you ready to awaken your home, reclaim your privacy and join a worldwide community of tinkerers?

Let's get started by creating a user account.

Name

Username

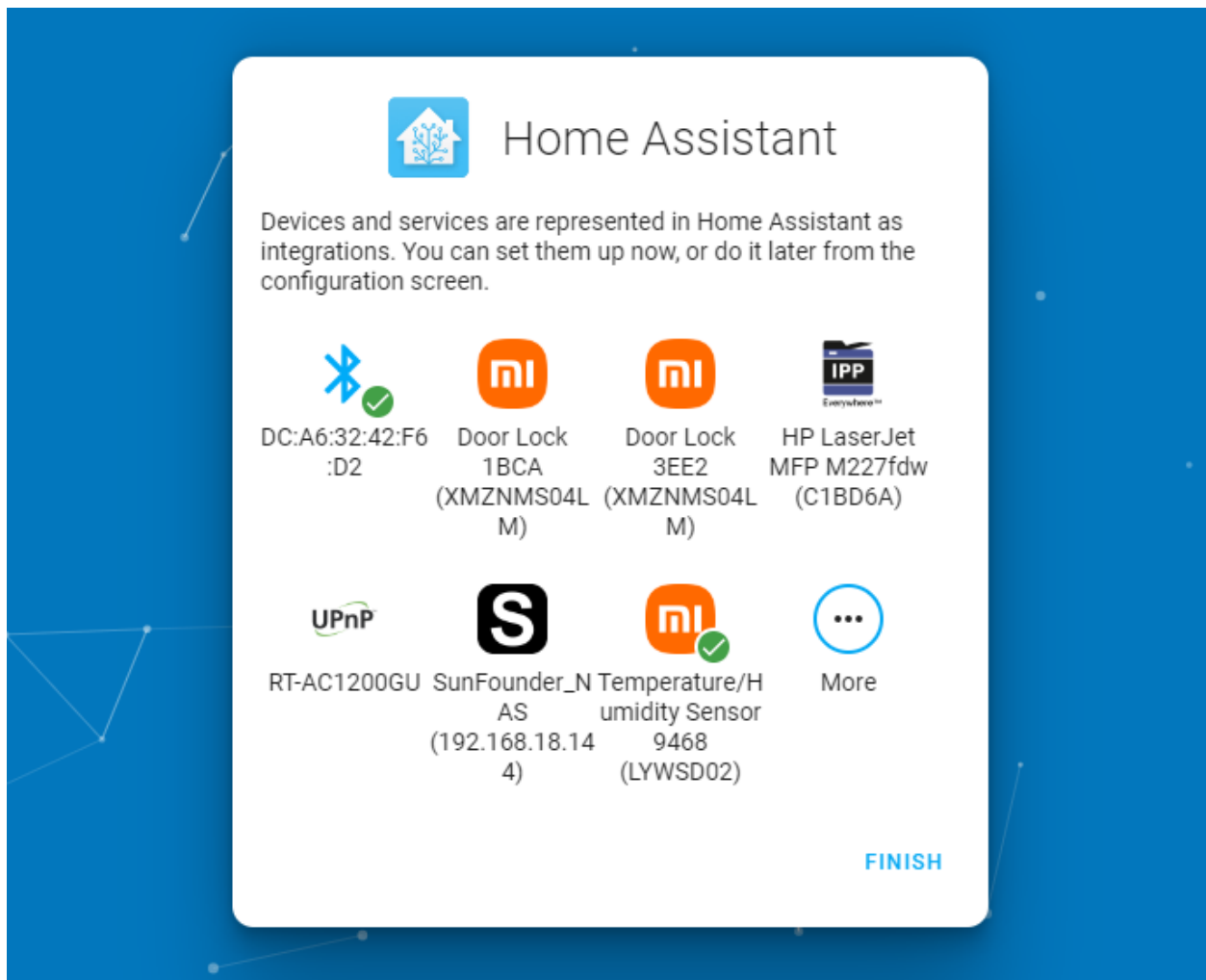
Password

Confirm Password

CREATE ACCOUNT

[Alternatively you can restore from a previous backup.](#)

Das System wird Sie auffordern, einige erkannte Geräte zu installieren. Für den Moment können Sie dies jedoch überspringen und auf FERTIGSTELLEN klicken.



Jetzt haben Sie Home Assistant eingerichtet.

Bemerkung: Wenn Sie bereits Home Assistant haben, bitte ignorieren.

Stellen Sie Ihre PiPower-Pro Karte in Home Assistant ein:

4.2 PiPower Pro in Home Assistant hinzufügen

Schritt 1

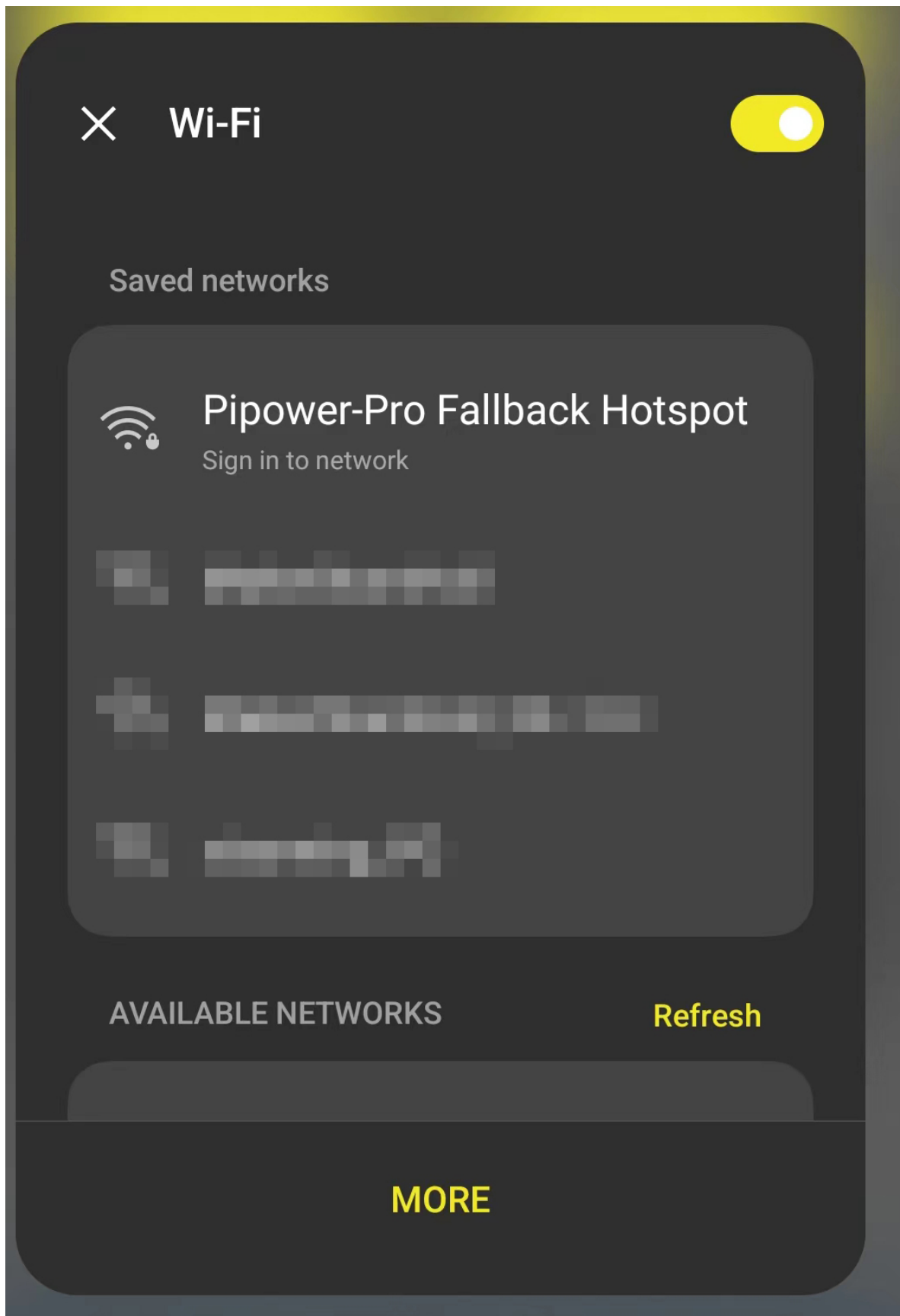
Setzen Sie die Batterie ein.

Schritt 2

Verbinden Sie das USB-C-Ladegerät, bis alle vier Batterieanzeigen leuchten (dies bedeutet, dass die Batterie vollständig geladen ist). Drücken Sie den Ein-/Ausschalter, um das Gerät einzuschalten.

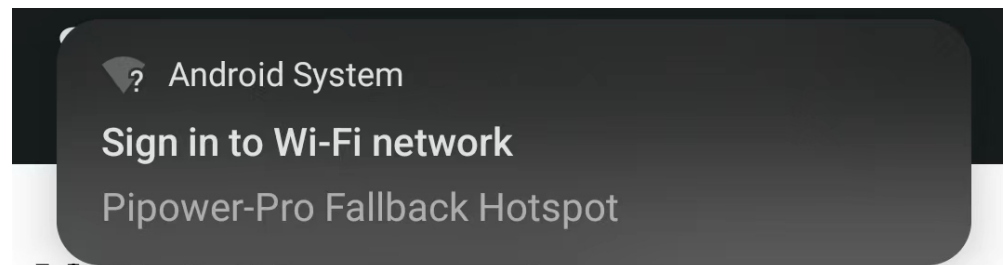
Schritt 3

Konfigurieren Sie das Netzwerk für PiPower Pro. Suchen Sie mit Ihrem Handy (oder einem anderen Gerät) nach Wi-Fi und verbinden Sie sich mit dem PiPower Pro Fallback Hotspot. Das Passwort lautet 12345678.



Schritt 4

Nachdem Sie verbunden sind, wird auf Ihrem Handy eine Konfigurationsseite angezeigt. Vervollständigen Sie hier die Wi-Fi-Konfiguration für PiPower.



WiFi Networks: pipower-pro



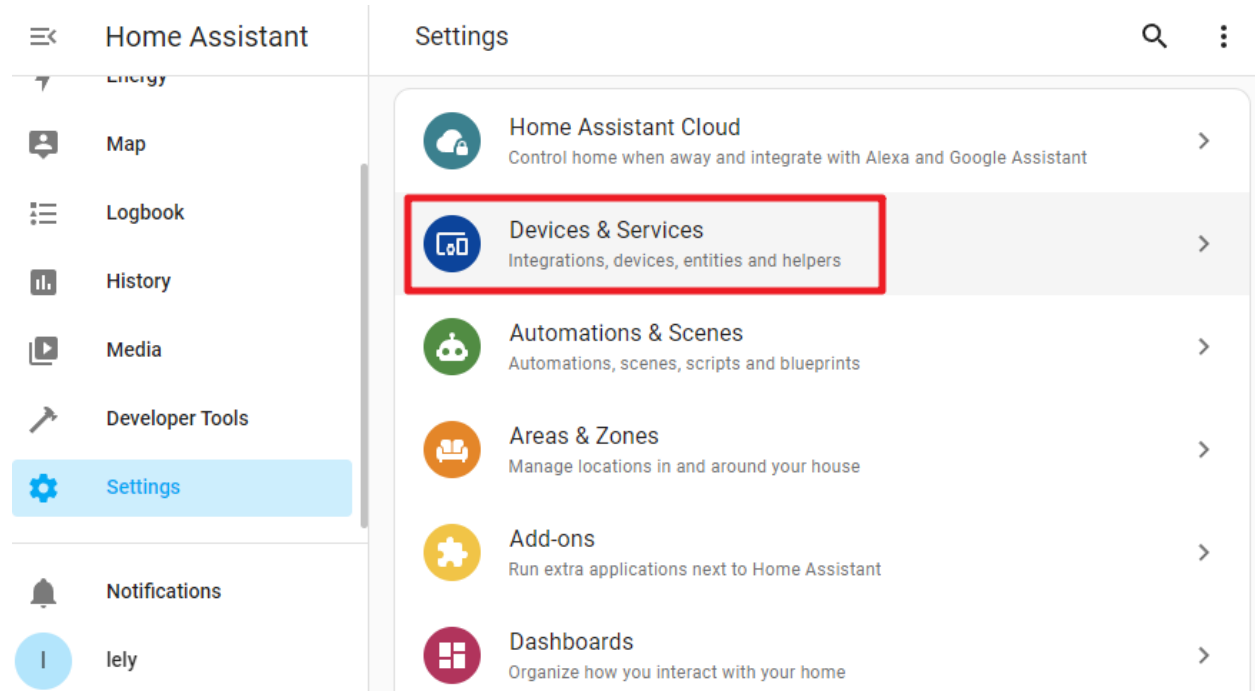
WiFi Settings

SSID
Password

Wenn die Konfigurationsseite nicht automatisch angezeigt wird, können Sie einen Browser öffnen und `pipower-pro.local` besuchen.

Schritt 5

Öffnen Sie Ihre Home Assistant Seite, wählen Sie Konfiguration in der linken Seitenleiste und dann Geräte und Dienste.



Schritt 6


Klicken Sie unten rechts auf + ADD INTEGRATION.




Schritt 7

Wählen Sie ESPHome aus.

Select brand ×

 Search for a brand name
ESPHome ×

 ESPHome >

Schritt 8

Geben Sie `pipower-pro.local` ein und bestätigen Sie.

ESPHome ? ×

Please enter connection settings of your [ESPHome](#) node.

Host*
pipower-pro.local

Port
6053

SUBMIT

Schritt 9

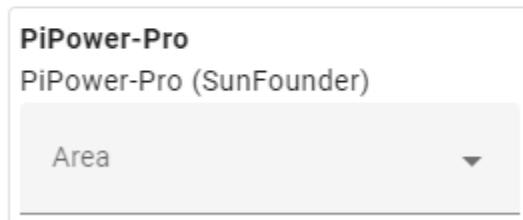
Wählen Sie einen Bereich dafür aus und schließen Sie die Einrichtung ab.

Success!



Created configuration for PiPower-Pro.

We found the following devices:



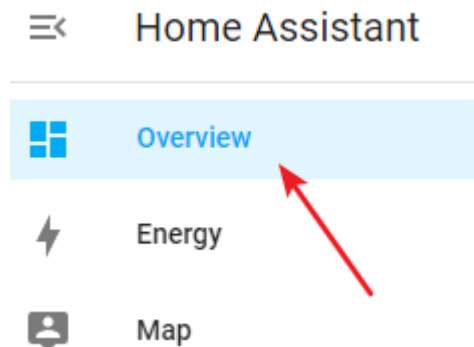
FINISH

Schritt 10

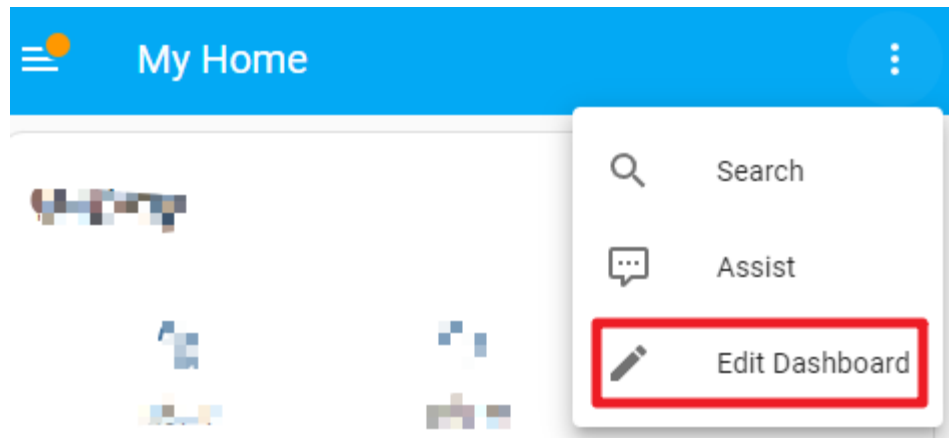
Sie haben PiPower Pro erfolgreich hinzugefügt. Sie können die gewünschten PiPower Pro Konfigurationen im Dashboard hinzufügen.

4.3 Dashboard konfigurieren

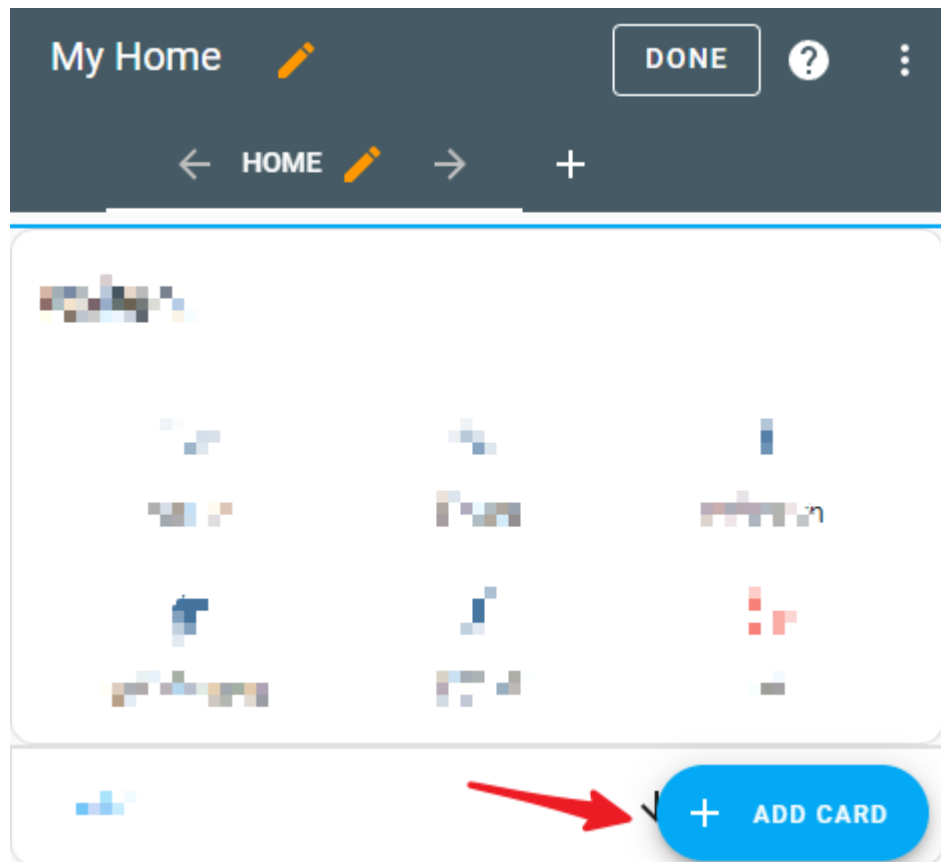
1. Auf der Home Assistant Seite klicken Sie links in der Seitenleiste auf **Overview**, dies führt Sie zur Steuerungsseite.



2. Klicken Sie auf das Menü-Symbol und wählen Sie dann **Edit Dashboard**.



3. Klicken Sie unten rechts, um eine Karte hinzuzufügen, wählen Sie die gewünschte Karte aus den Optionen, konfigurieren Sie sie nach Bedarf und speichern Sie dann.



4.4 Karte mittels des Code-Editors hinzufügen

1. Nachdem Sie eine Karte hinzugefügt haben, können Sie die yaml-Datei der Karte manuell bearbeiten. Klicken Sie dazu auf **SHOW CODE EDITOR** auf der Kartenbearbeitungsseite.

× Alarm Panel Card Configuration ?

Entity*
No matching entities found

Name
Theme (optional)

Available States

☒ Arm home
☒ Arm away
☐ Arm night
☐ Arm vacation
☐ Custom bypass

SHOW CODE EDITOR
CANCEL
SAVE

Invalid configuration

```
type: alarm-panel
states:
  - arm_home
  - arm_away
entity: ''
```

2. Ändern Sie dann direkt die yaml-Datei. Wir bieten einige nützliche Konfigurationen für PiPower Pro an. Bitte kopieren Sie den folgenden yaml-Code direkt in das vorgesehene Feld.

× Alarm Panel Card Configuration ?

```
1 type: alarm-panel
2 states:
3   - arm_home
4   - arm_away
5 entity: ''
6
```

SHOW VISUAL EDITOR
CANCEL
SAVE

Invalid configuration

```
type: alarm-panel
states:
  - arm_home
  - arm_away
entity: ''
```

× Vertical Stack Card Configuration



```

1 type: vertical-stack
2 cards:
3   - type: entities
4     entities:
5       - entity: switch.pipower_pro_output_switch
6       - entity: sensor.pipower_pro_output_source
7       - entity: binary_sensor.pipower_pro_external_power
8       - entity: sensor.pipower_pro_battery_voltage
9       - entity: sensor.pipower_pro_output_voltage
10    title: PiPower Pro
11    show_header_toggle: true
12    state_color: true
13  - square: true
14    type: grid
15    cards:
16      - type: gauge
17        entity: sensor.pipower_pro_battery_current
18        min: -2
19        max: 2
20        severity:
21          green: 0
22          yellow: 2
23          red: 2
24        needle: true
25        name: Battery Current
26      - type: gauge
27        entity: sensor.pipower_pro_output_current
28        min: 0
29        max: 3
30        severity:
31          green: 0
32          yellow: 2
33          red: 2.5
34        needle: true
35        name: Output Current

```

PiPower Pro

PiPower-Pro Output Switch

PiPower-Pro Output Source Battery

PiPower-Pro External Power Plugged in

PiPower-Pro Battery Voltage 5.87 V

PiPower-Pro Output Voltage 4.11 V

☒

☒

0.00 A

Battery Current

0.08 A

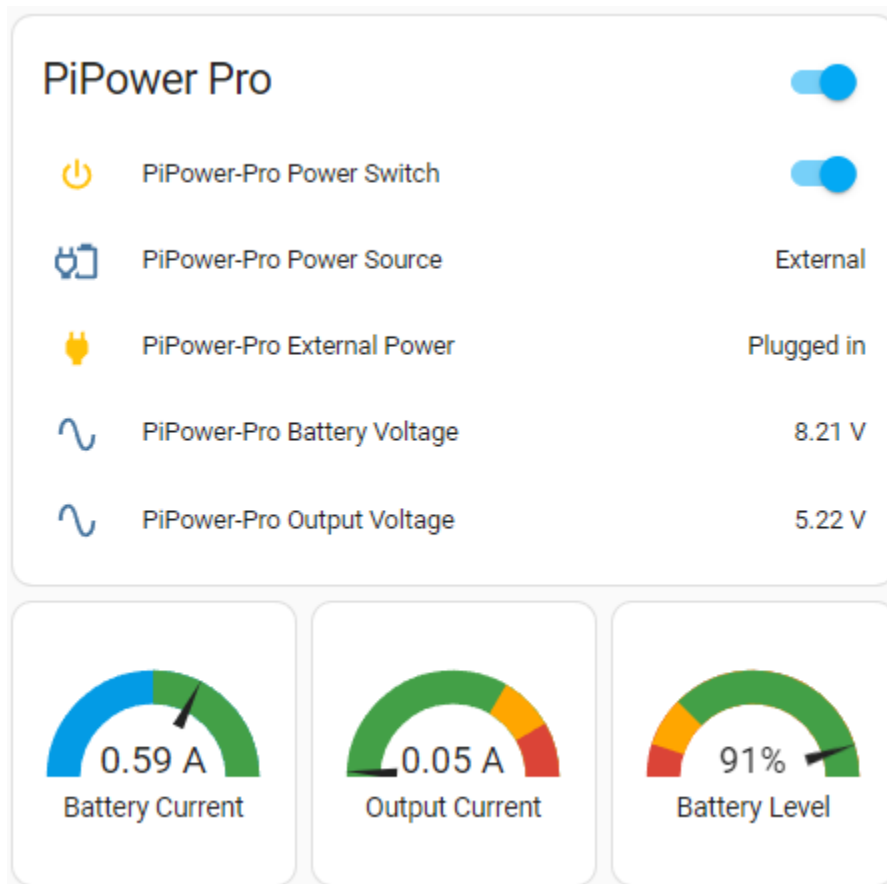
Output Current

0%

Battery Level

SHOW VISUAL EDITOR
CANCEL SAVE

Schnellübersicht



```

type: vertical-stack
cards:
  - type: entities
    entities:
      - entity: switch.pipower_pro_output_switch
      - entity: sensor.pipower_pro_output_source
      - entity: binary_sensor.pipower_pro_external_power
      - entity: sensor.pipower_pro_battery_voltage
      - entity: sensor.pipower_pro_output_voltage
    title: PiPower Pro
    show_header_toggle: true
    state_color: true
  - square: true
    type: grid
    cards:
      - type: gauge
        entity: sensor.pipower_pro_battery_current
        min: -2
        max: 2
        severity:
          green: 0
          yellow: 2
          red: 2
        needle: true

```

(Fortsetzung auf der nächsten Seite)

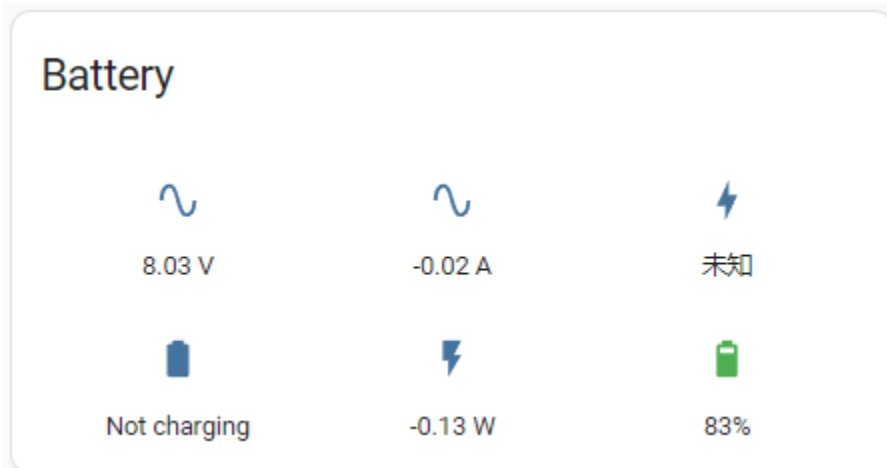
(Fortsetzung der vorherigen Seite)

```

    name: Batteriestrom
  - type: gauge
    entity: sensor.pipower_pro_output_current
    min: 0
    max: 3
    severity:
      green: 0
      yellow: 2
      red: 2.5
    needle: true
    name: Ausgangsstrom
  - type: gauge
    entity: sensor.pipower_pro_battery_level
    name: Batteriestand
    min: 0
    max: 100
    severity:
      green: 25
      yellow: 10
      red: 0
    needle: true
columns: 3

```

Batterieinformationen



```

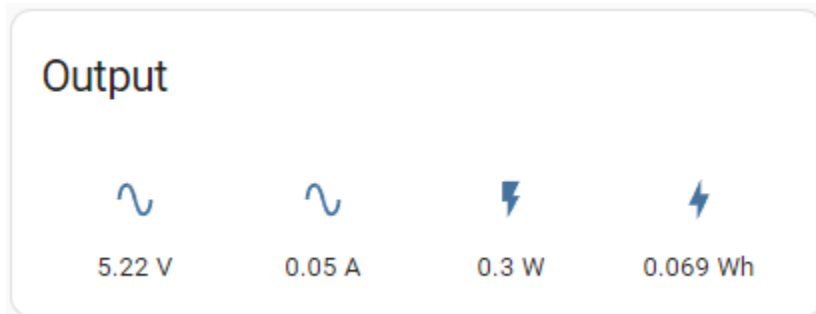
show_name: false
show_icon: true
show_state: true
type: glance
entities:
  - entity: sensor.pipower_pro_battery_voltage
  - entity: sensor.pipower_pro_battery_current
  - entity: sensor.pipower_pro_battery_capacity
  - entity: binary_sensor.pipower_pro_is_charging
  - entity: sensor.pipower_pro_battery_power
  - entity: sensor.pipower_pro_battery_level
title: Batterie

```

(Fortsetzung auf der nächsten Seite)

(Fortsetzung der vorherigen Seite)

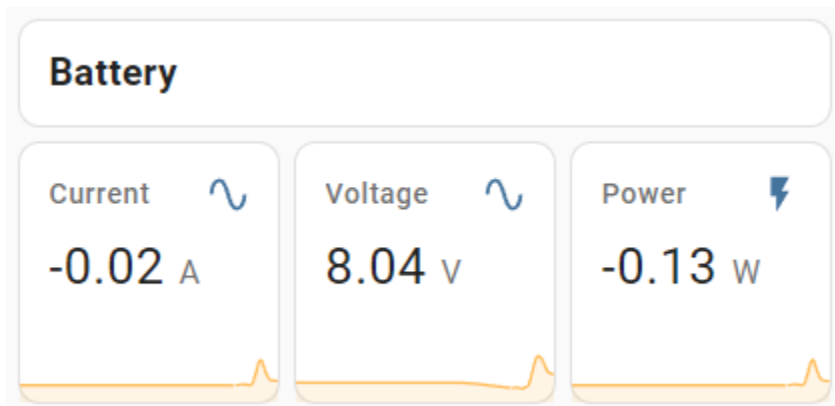
columns: 3

Ausgabeinformationen

```

show_name: false
show_icon: true
show_state: true
type: glance
entities:
  - entity: sensor.pipower_pro_output_voltage
  - entity: sensor.pipower_pro_output_current
  - entity: sensor.pipower_pro_output_power
  - entity: sensor.pipower_pro_output_energy
title: Ausgabe

```

Batterie-Diagramm

```

type: vertical-stack
cards:
  - type: markdown
    content: '## Batterie'
  - square: true
    columns: 3
    type: grid
    cards:
      - hours_to_show: 12
        graph: line
        type: sensor

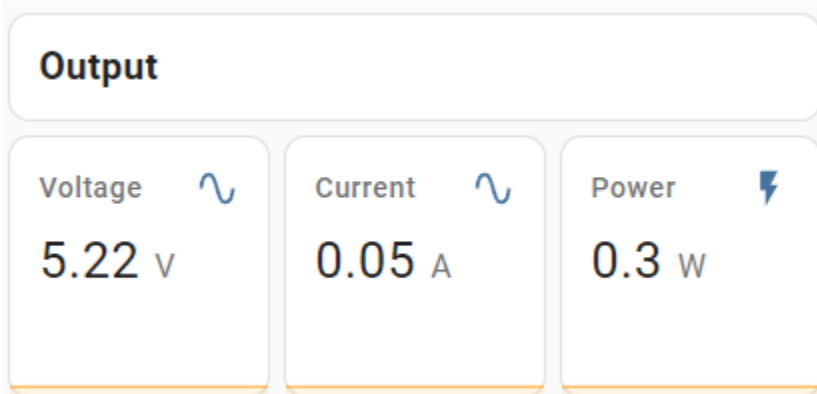
```

(Fortsetzung auf der nächsten Seite)

(Fortsetzung der vorherigen Seite)

```
entity: sensor.pipower_pro_battery_current
detail: 2
name: Strom
- hours_to_show: 12
graph: line
type: sensor
entity: sensor.pipower_pro_battery_voltage
detail: 2
name: Spannung
- hours_to_show: 12
graph: line
type: sensor
entity: sensor.pipower_pro_battery_power
detail: 2
name: Leistung
```

Ausgabe-Diagramm



```
type: vertical-stack
cards:
- type: markdown
  content: '## Ausgabe'
- square: true
  columns: 3
  type: grid
  cards:
    - hours_to_show: 12
      graph: line
      type: sensor
      entity: sensor.pipower_pro_output_voltage
      detail: 2
      name: Spannung
    - hours_to_show: 12
      graph: line
      type: sensor
      entity: sensor.pipower_pro_output_current
      detail: 2
      name: Strom
    - hours_to_show: 12
```

(Fortsetzung auf der nächsten Seite)

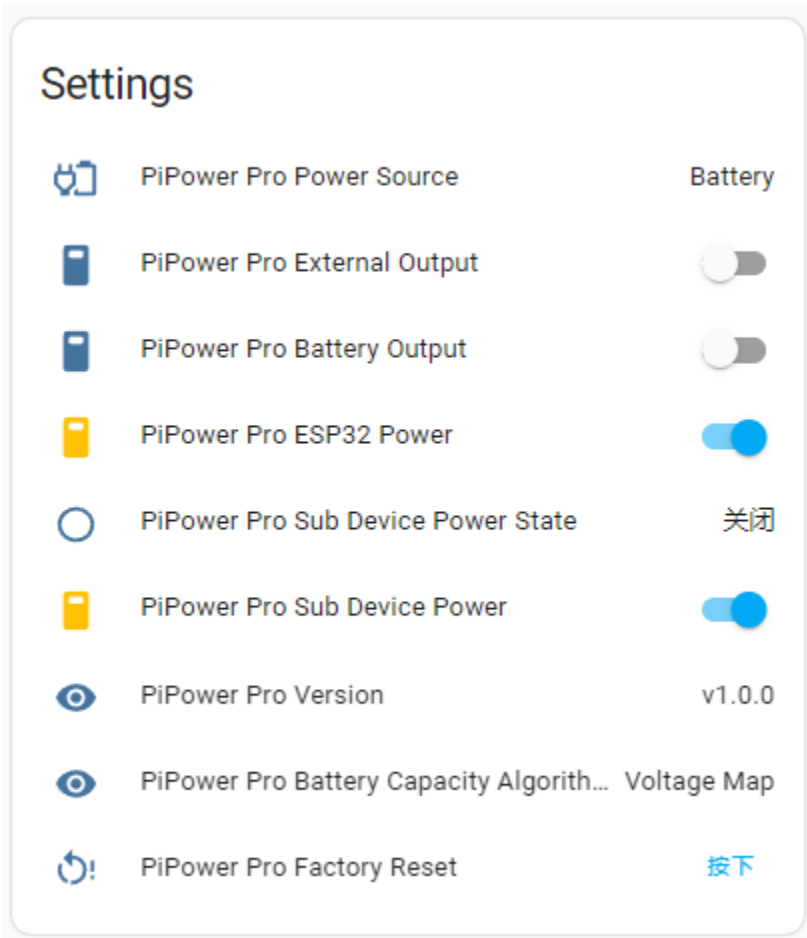
(Fortsetzung der vorherigen Seite)

```

graph: line
type: sensor
entity: sensor.pipower_pro_output_power
detail: 2
name: Leistung

```

Einstellungen



```

type: entities
entities:
- entity: sensor.pipower_pro_input_voltage
- entity: sensor.pipower_pro_output_source
- entity: switch.pipower_pro_external_output
- entity: switch.pipower_pro_battery_output
- entity: switch.pipower_pro_esp32_power
- entity: binary_sensor.pipower_pro_sub_device_power_state
- entity: switch.pipower_pro_sub_device_power
- entity: sensor.pipower_pro_version
- entity: sensor.pipower_pro_battery_capacity_algorithm
- entity: button.pipower_pro_factory_reset
title: Einstellungen
show_header_toggle: false

```

(Fortsetzung auf der nächsten Seite)

```
state_color: true
```

4.5 PiPower Pro Entität

Wenn Sie mit Home Assistant vertraut sind und die Karte selbst anpassen möchten, finden Sie hier eine Liste der PiPower Pro Entitäten, die Sie verwenden können.

Grundinformationen

- `binary_sensor.pipower_pro_battery_low` - Batteriestatus niedrig (bool)
- `binary_sensor.pipower_pro_is_charging` - Ladezustand (V)

Schalter

- `switch.pipower_pro_battery_output` - Batterieausgangsschalter (bool)
- `switch.pipower_pro_esp32_power` - ESP32 Energieversorgungsschalter (bool)
- `switch.pipower_pro_external_output` - Externer Ausgangsschalter (bool)

Ausgabe

- `sensor.pipower_pro_output_voltage` - Ausgangsspannung (V)
- `sensor.pipower_pro_output_current` - Ausgangsstrom (A)
- `sensor.pipower_pro_output_power` - Ausgangsleistung (W)
- `sensor.pipower_pro_output_energy` - Verbrauchte Ausgangsenergie (Wh), wird zur Berechnung der gesamten Ausgangsenergie verwendet, kann über Dienste zurückgesetzt werden, siehe alle Dienste für Details

Batterie

- `sensor.pipower_pro_battery_voltage` - Batteriespannung (V)
- `sensor.pipower_pro_battery_current` - Batteriestrom, positiv beim Laden, negativ beim Entladen (A)
- `sensor.pipower_pro_battery_power` - Batterieausgangsleistung (W)
- `sensor.pipower_pro_battery_capacity` - Batteriekapazität (mAh)
- `sensor.pipower_pro_battery_level` - Batterieladestand (%)

Eingang

- `sensor.pipower_pro_input_voltage` - Externe Eingangsspannung (V)

Steuerung von Untergeräten

- `switch.pipower_pro_sub_device_power` - Stromsteuersignal des Untergeräts (bool)
- `binary_sensor.pipower_pro_sub_device_power_state` - Betriebszustand des Untergeräts (bool)

Sonstiges

- `sensor.pipower_pro_battery_capacity_algorithm` - Batteriekapazitätsalgorithmus (String)
- `sensor.pipower_pro_power_source` - Aktuelle Energiequelle: Batterie/Extern (String)
- `sensor.pipower_pro_battery_factory_capacity` - Nominale Kapazität der Batterie ab Werk (mAh)

- `binary_sensor.pipower_pro_external_power` - Externer Eingangszustand (bool)
- `button.pipower_pro_factory_reset` - Taster für Werksreset (bool)
- `update.pipower_pro_firmware` - Firmware aktualisieren
- `switch.pipower_pro_power_switch` - Ausgangsschalter (bool)
- `sensor.pipower_pro_version` - PiPower Pro Version (String)

Alle Dienste

- `set_battery_factory_capacity` - Ändern Sie die nominale Kapazität der Batterie ab Werk (capacity: int, mAh), Standardwert 2000
- `enable_coulomb_count_beta` - Coulomb-Zählalgorithmus aktivieren (enable: bool), Standardwert false
- `reset_capacity` - Aktuelle Kapazität auf nominale Werkkapazität zurücksetzen
- `reset_output_energy` - Ausgangsenergie auf 0 zurücksetzen
- `set_edv2` - Festlegen der Spannung für das Ende der Entladungskalibrierung 2, Standard 6.8. Siehe Coulomb-Zählung für Details
- `set_edv1` - Festlegen der Spannung für das Ende der Entladungskalibrierung 1, Standard 6.5. Siehe Coulomb-Zählung für Details
- `set_edv0` - Festlegen der Spannung für das Ende der Entladungskalibrierung 0, Standard 6.2. Siehe Coulomb-Zählung für Details
- `set_rcv` - Festlegen der Spannung zum Zurücksetzen des Kalibrierungsstatus, Standard 8.0. Siehe Coulomb-Zählung für Details
- `simulate_low_power` - Simulieren eines niedrigen Energiezustands zum Testen von Szenarien mit niedrigem Energieauslöser

Erweiterte Nutzung:

4.6 Einrichten des sicheren Herunterfahrens

PiPower Pro verfügt über zwei vorab konfigurierte Pins, um den Energiezustand (im Folgenden als Untergeräte bezeichnet) von angeschlossenen Geräten zu überwachen. Dies ermöglicht das Fern-Einschalten, das Fern-Ausschalten sowie ein automatisches sicheres Herunterfahren, wenn der Akku schwach ist.

Bemerkung: Wenn der mit HassOS betriebene Host als Untergerät von PiPower Pro eingerichtet ist, verliert er ebenfalls seine Funktion, wenn der Host heruntergefahren wird, und ein Fernstart ist nicht möglich.

- Pin 42 und Sensor-Entität `binary_sensor.pipower_pro_sub_device_power_state` lesen den aktuellen Zustand des Geräts.
- Pin 41 und Entität `switch.pipower_pro_sub_device_power` steuern die Energieversorgung des Untergeräts.

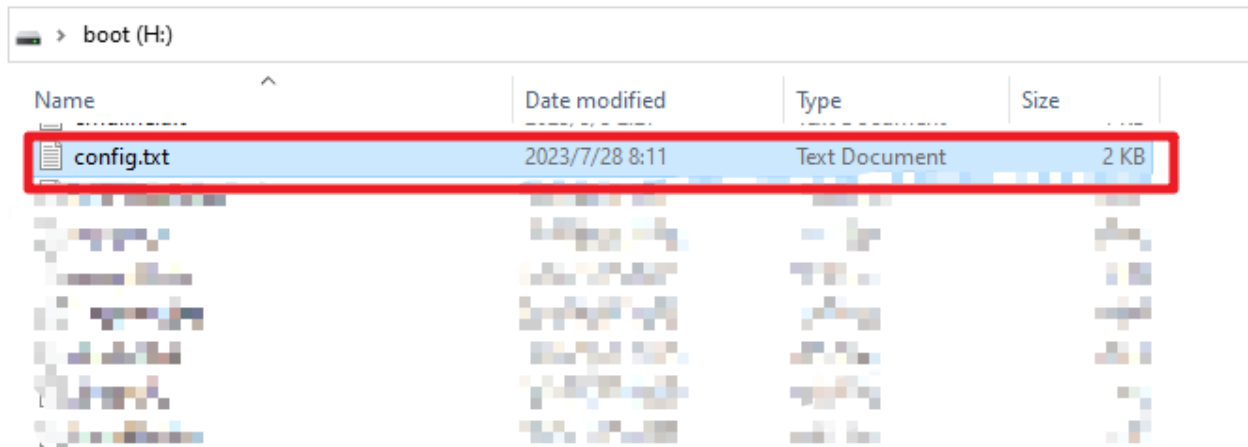
Zum Beispiel verwenden wir PiPower Pro als eine USV-Stromquelle für einen Raspberry Pi, überwachen dessen Status und schalten ihn automatisch sicher ab, wenn die externe Stromversorgung unterbrochen wird und der Akkuladestand niedrig ist.

Schritt 1

Konfigurieren Sie den Raspberry Pi.

Legen Sie die beiden Raspberry Pi-Pins für den **Power Status Signal Pin** und den **Shutdown Signal Pin** fest. Dies kann über devicetree gemacht werden.

Stecken Sie die SD-Karte mit dem Raspberry Pi-System in Ihren Computer. Im Stammverzeichnis der Boot-Partition finden Sie `config.txt`.



Öffnen Sie diese und fügen Sie die folgenden zwei Zeilen am Ende unter `[all]` hinzu.

```
dtoverlay=gpio-poweroff,gpiopin=17
dtoverlay=gpio-shutdown,gpio_pin=18
```

- `gpio-poweroff` ist der Ein-/Ausschaltzustand des Raspberry Pi. Nach erfolgreicher Konfiguration setzt der Raspberry Pi diesen Pin beim Einschalten auf hoch und zieht ihn beim Ausschalten auf niedrig.
- `gpio-shutdown` steuert das Signal zum Herunterfahren des Raspberry Pi. Nach erfolgreicher Konfiguration bewirkt das Ziehen dieses Pins auf niedrig, dass der Raspberry Pi heruntergefahren wird.

Schritt 2

- Verbinden Sie den Pin 42 von PiPower Pro mit dem `gpio-poweroff`-Pin des Raspberry Pi, hier mit Pin 17.
- Verbinden Sie den Pin 41 von PiPower Pro mit dem `gpio-shutdown`-Pin des Raspberry Pi, hier mit Pin 18.

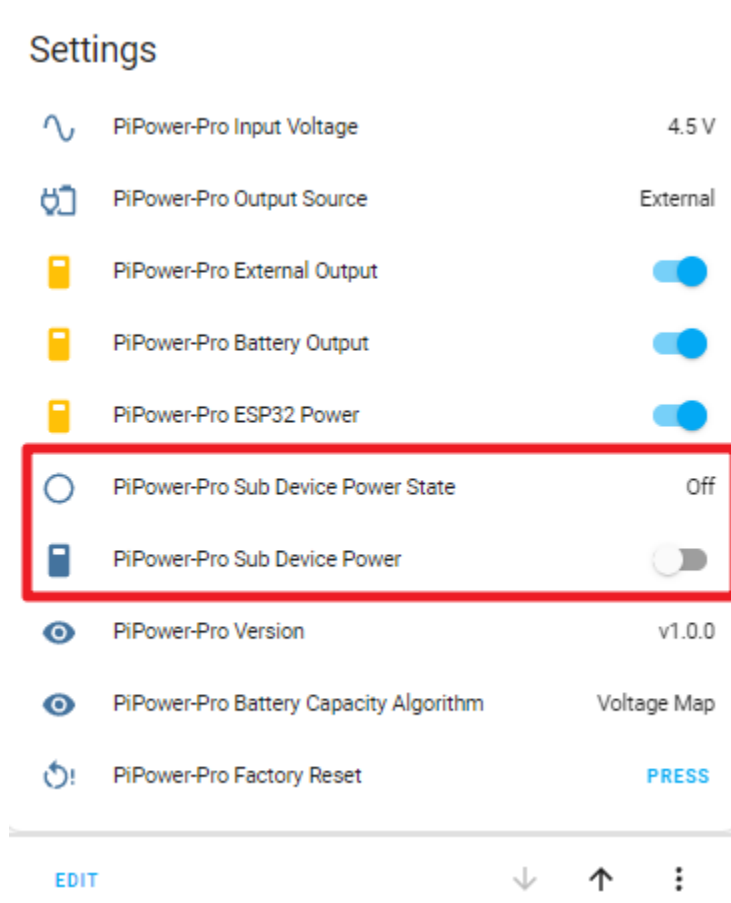
Schritt 3

Testen Sie nun, ob die beiden oben genannten Signale ordnungsgemäß funktionieren.

Fügen Sie dem Dashboard zwei Entitäten hinzu:

- `binary_sensor.pipower_pro_sub_device_power_state`
- `switch.pipower_pro_sub_device_power`

Wenn Sie die **Settings**-Karte hinzufügen (siehe [Karte mittels des Code-Editors hinzufügen](#) für Anweisungen zum Hinzufügen von Karten), werden diese beiden Entitäten enthalten sein und als **PiPower-Pro Sub Device Power State** und **PiPower-Pro Sub Device Power** bezeichnet.



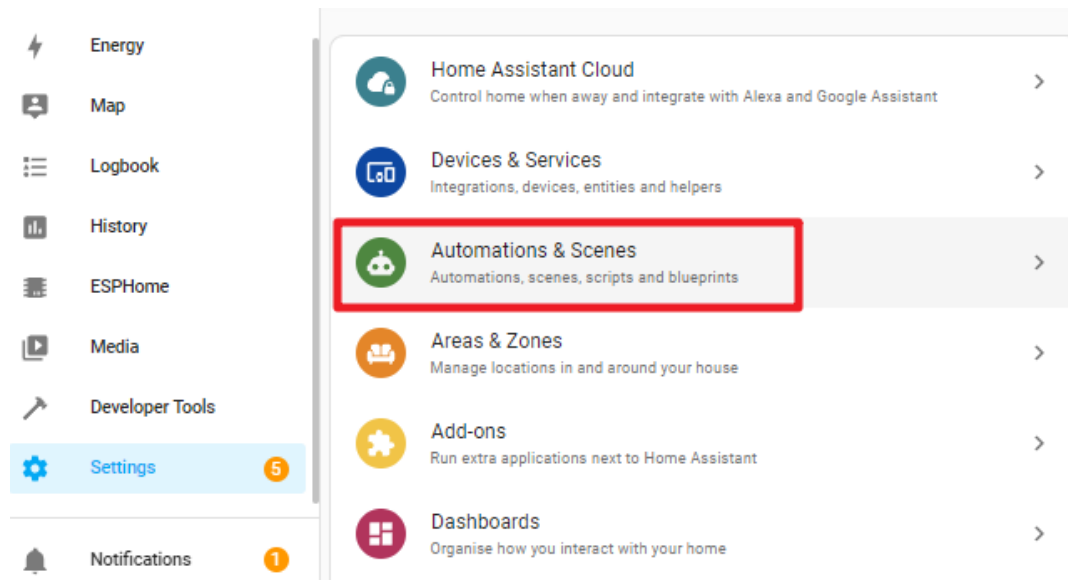
Mit Ersterem können Sie überprüfen, ob der Raspberry Pi funktioniert, und mit Letzterem können Sie den Raspberry Pi ausschalten.

Bemerkung: PiPower-Pro Untergeräte-Strom kann den Raspberry Pi nur ausschalten. Um ihn einzuschalten, müssen Sie ihm weiterhin Strom zuführen (d.h., den Hauptschalter auf der **PiPower Pro**-Karte einschalten).

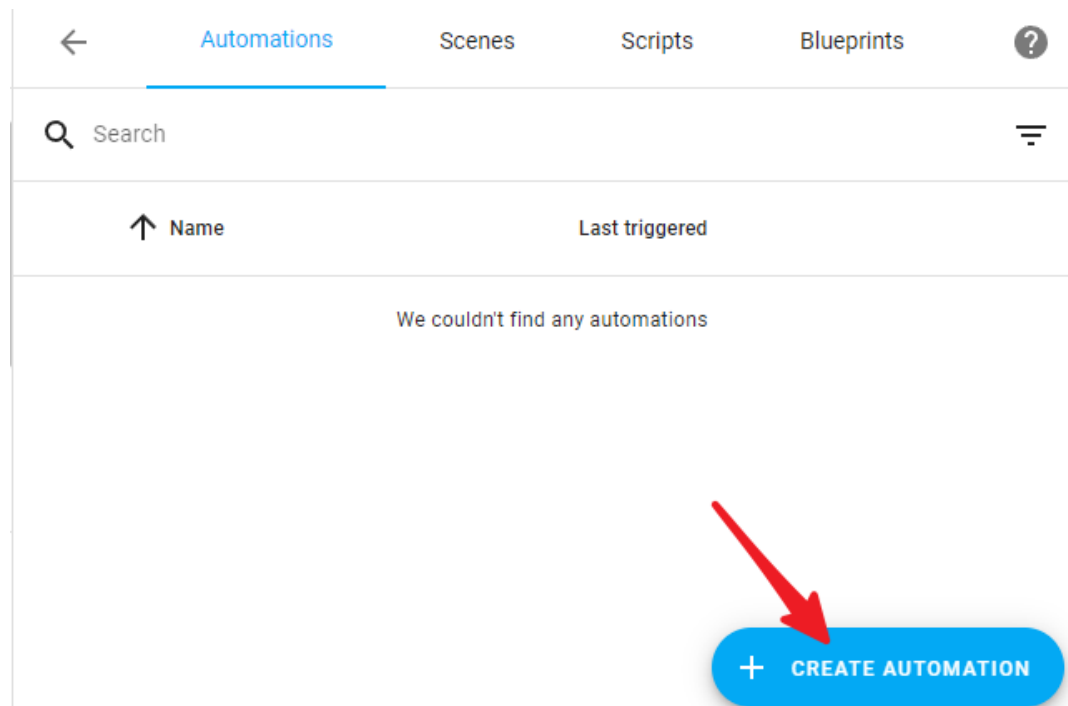
Schritt 4

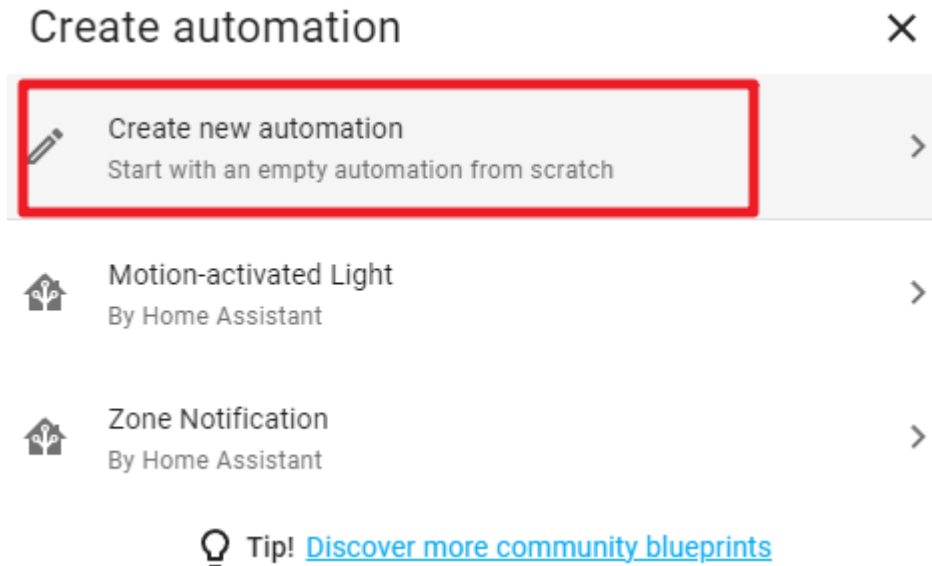
Konfigurieren Sie nun Automatisierungen, um PiPower Pro einen sicheren Herunterfahren des Raspberry Pi zu ermöglichen:

1. Öffnen Sie die Home Assistant Konfigurationsseite, klicken Sie in der linken Seitenleiste auf „Einstellungen“ und wählen Sie „Automatisierungen.“

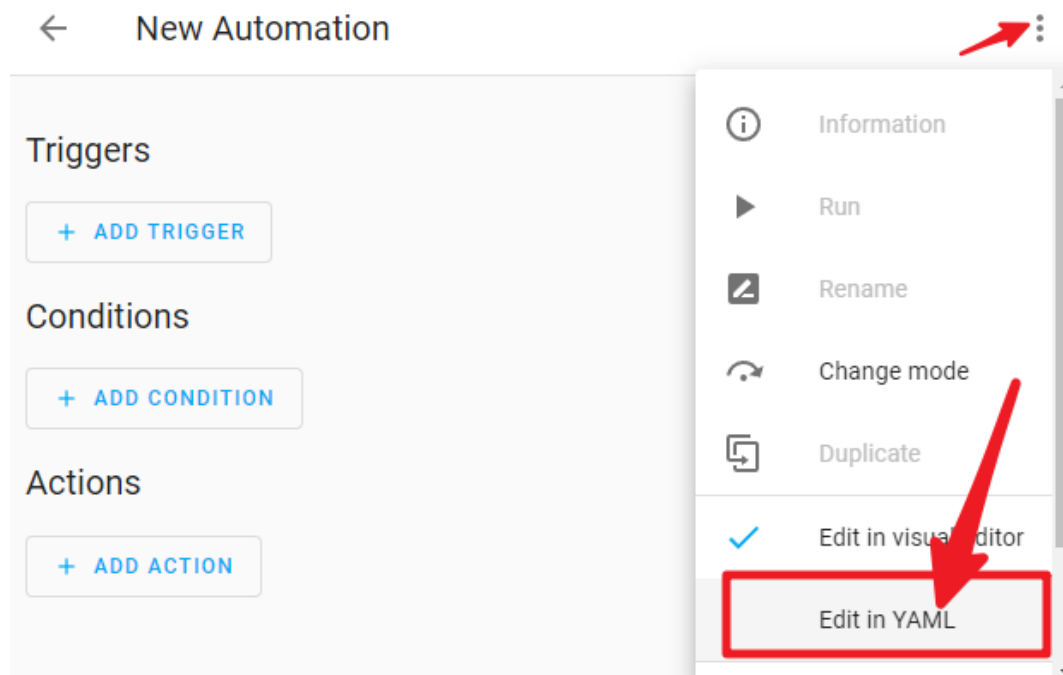


2. Erstellen Sie eine neue Automatisierung.





3. Klicken Sie auf „In YAML bearbeiten.“



4. Ersetzen Sie den vorhandenen Code durch den folgenden Code.

```
alias: Sicherer Shutdown RPi
description: Raspberry Pi ausschalten, wenn keine externe Stromversorgung
angeschlossen ist und der Akku schwach ist
trigger:
  - platform: state
    entity_id:
      - binary_sensor.pipower_pro_external_power
    from: "on"
```

(Fortsetzung auf der nächsten Seite)

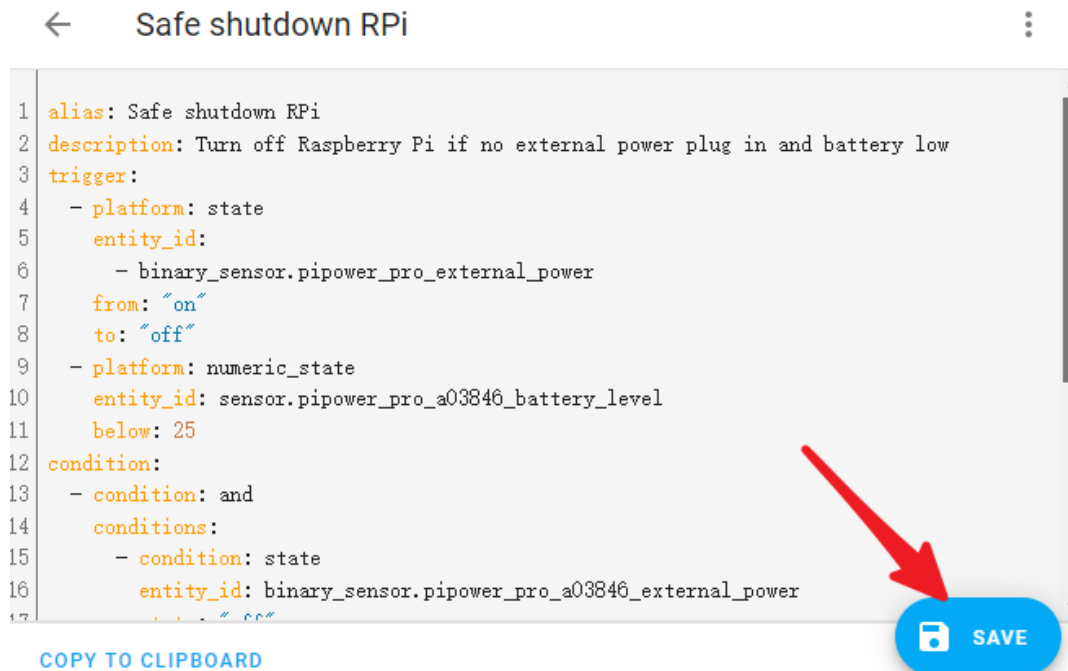
(Fortsetzung der vorherigen Seite)

```

    to: "off"
  - platform: numeric_state
    entity_id: sensor.pipower_pro_a03846_battery_level
    below: 25
condition:
  - condition: and
    conditions:
      - condition: state
        entity_id: binary_sensor.pipower_pro_a03846_external_power
        state: "off"
  - condition: and
    conditions:
      - condition: state
        entity_id: switch.pipower_pro_sub_device_power
        state: "on"
action:
  - type: turn_off
    device_id: a0ee4e356c85c4f69f765ed72baad129
    entity_id: switch.pipower_pro_sub_device_power
    domain: switch
mode: single

```

5. Klicken Sie auf „Speichern.“



6. Klicken Sie auf „Umbenennen.“

Rename



Name*

Safe shutdown RPi

Description

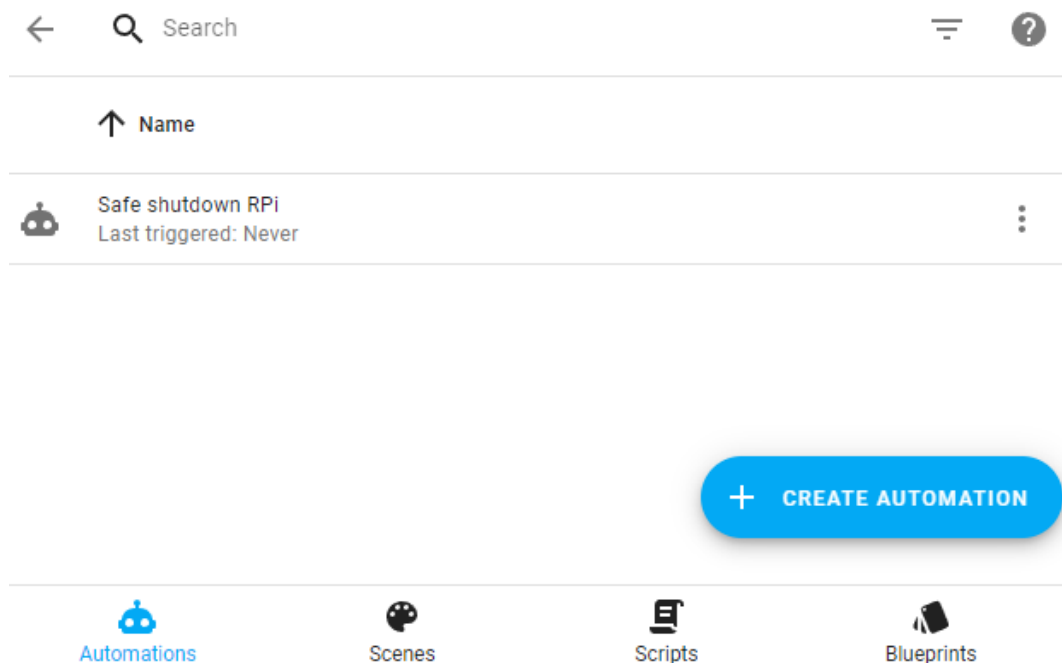
Turn off Raspberry Pi if no external power plug in and battery low

CANCEL

RENAME



7. Gehen Sie einen Schritt zurück. Nun sollten Sie die neu eingerichtete Automatisierung sehen.



Bemerkung: Weitere Automatisierungen müssen erstellt werden. Befolgen Sie dazu die vorherigen Schritte.

Strom sparen

```
alias: Save Power
description: Turn off if raspberry pi power off
```

(Fortsetzung auf der nächsten Seite)

(Fortsetzung der vorherigen Seite)

```

trigger:
  - platform: state
    entity_id:
      - binary_sensor.pipower_pro_sub_device_power_state
    from: "on"
    to: "off"
condition:
  - condition: state
    entity_id: switch.pipower_pro_sub_device_power
    state: "off"
action:
  - delay:
    hours: 0
    minutes: 0
    seconds: 2
    milliseconds: 0
  - type: turn_off
    device_id: a0ee4e356c85c4f69f765ed72baad129
    entity_id: switch.pipower_pro_a03846_power_switch
    domain: switch
  - type: turn_off
    device_id: a0ee4e356c85c4f69f765ed72baad129
    entity_id: switch.pipower_pro_a03846_esp32_power
    domain: switch
mode: single

```

Sync Ausschalten RPi

```

alias: Sync Power Off RPi
description: Power Off Raspberry Pi is Switch Off
trigger:
  - platform: state
    entity_id:
      - switch.pipower_pro_a03846_power_switch
    from: "on"
    to: "off"
condition: []
action:
  - type: turn_off
    device_id: a0ee4e356c85c4f69f765ed72baad129
    entity_id: switch.pipower_pro_sub_device_power
    domain: switch
mode: single

```

Sync Einschalten RPi

```

alias: Sync Power On RPi
description: Power On Raspberry Pi is Switch On
trigger:
  - platform: state
    entity_id:
      - switch.pipower_pro_a03846_power_switch

```

(Fortsetzung auf der nächsten Seite)

(Fortsetzung der vorherigen Seite)

```

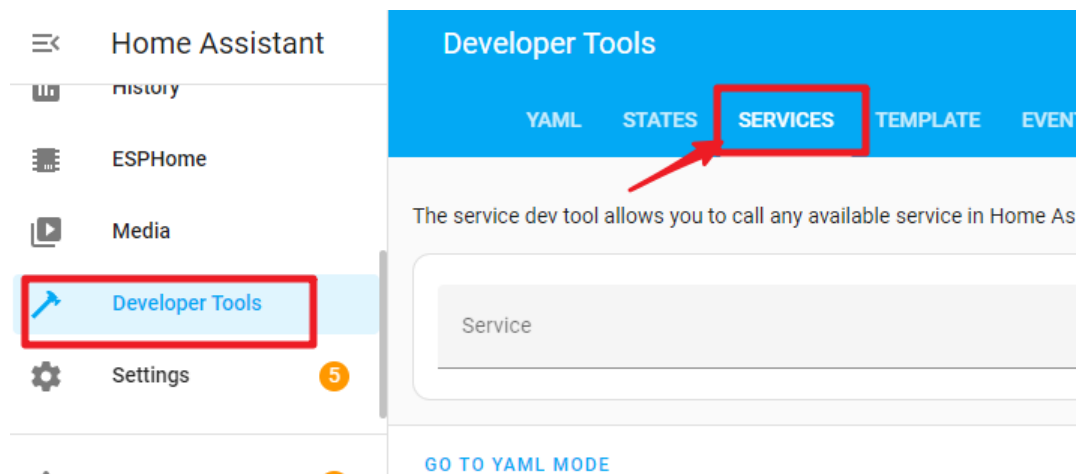
    from: "off"
    to: "on"
condition: []
action:
  - type: turn_on
    device_id: a0ee4e356c85c4f69f765ed72baad129
    entity_id: switch.pipower_pro_sub_device_power
    domain: switch
mode: single

```

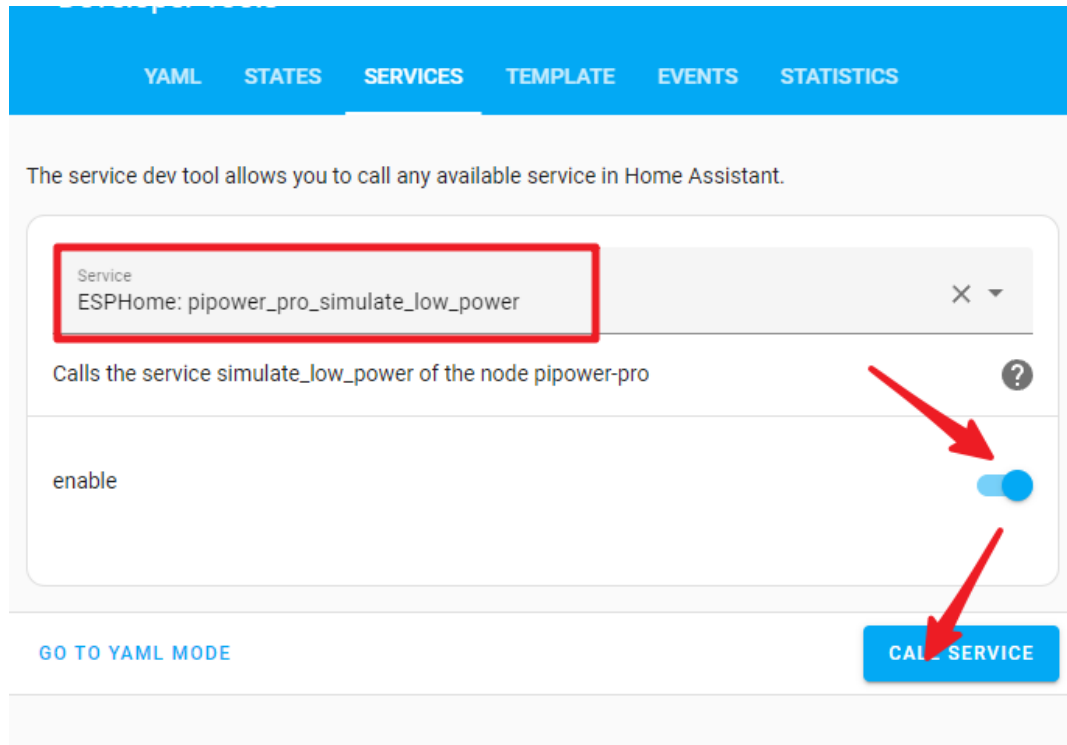
Schritt 5

Um den Test auszulösen, simulieren wir einen Zustand niedriger Leistung:

1. Öffnen Sie die SERVICES-Oberfläche in den Entwicklertools.



2. Finden Sie ESPHome: `pipower_pro_simulate_low_power`, aktivieren Sie es und klicken Sie auf die Schaltfläche „Call Service“.



Sie werden sehen, dass das PiPower-Batterielicht erlischt und der Batteriestand im Überblick auf 10% sinkt.

Der Raspberry Pi wird dann heruntergefahren, und 2 Sekunden nachdem er das Herunterfahren abgeschlossen hat, wird der PiPower Pro ausgeschaltet und die PWR-Leuchte erlischt.

4.7 Coulomb-Zähler (Beta)

Der Coulomb-Zähler-Algorithmus kann die Genauigkeit der Batteriekapazitätsberechnung verbessern, befindet sich jedoch derzeit im Beta-Stadium und kann zu erheblichen Ungenauigkeiten führen. Bitte mit Vorsicht verwenden.

Coulomb-Zähler aktivieren

1. Gehen Sie zur Home Assistant-Seite und klicken Sie auf „Entwicklertools“ in der linken Seitenleiste.
2. Auf der Entwicklertools-Seite wählen Sie den Tab „Dienste“ aus.
3. In der Liste der Dienste wählen Sie ESPHome: pipower_pro_enable_coulomb_count_beta.
4. Schalten Sie den Schalter für enable_coulomb_count_beta ein.
5. Klicken Sie unten auf den Button **Dienst aufrufen**.
6. Die aktuell ausgewählte Batteriekapazitätsalgorithmus können Sie im Entity sensor.pipower_pro_battery_capacity_algorithm prüfen.

Algorithmus

Der Coulomb-Zähler-Algorithmus berechnet die Energie, indem er jede Sekunde die Strom- und Spannungsmessungen der Batterie integriert.

Kapazität += Spannung * Strom

Abgleich

Die durch diese Integration berechnete Kapazität entspricht nur der Lade-/Entladeenergie ab dem aktuellen Zeitpunkt. Um sie mit der tatsächlichen Kapazität der Batterie in Verbindung zu bringen, ist ein Abgleichprozess erforderlich. Die hier verwendete Abgleichmethode ist einfach. Die Standardbatteriekapazität des PiPower Pro entspricht der Nennkapazität der Batterie, also 2000mAh. Die tatsächliche Batteriekapazität wird geringer sein als dieser Wert. Solange die Batterie geladen wird, wird die Kapazität auf maximal 2000mAh eingestellt (kann über den Dienst `set_battery_factory_capacity` geändert werden). Wenn die Batterie vollständig geladen ist, entspricht der Kapazitätswert der tatsächlichen Batteriekapazität von 2000mAh, und der Integrationsberechnungswert entspricht dem tatsächlichen Batteriekapazitätswert.

Automatische Kalibrierung

Integrationen können Fehler ansammeln, und die Batteriekapazität wird im Laufe der Zeit abnehmen, wenn die Batterie verwendet wird, was möglicherweise nicht die nominelle Kapazität von 2000mAh erreicht. Deshalb müssen einige Kalibrierungsmethoden verwendet werden, um die Batteriekapazität zu kalibrieren.

Hier wird die Methode der Kompensierten Endladespannung (CEDV) verwendet. Das Prinzip der CEDV-Kalibrierungsmethode ist, dass die Spannung am Ende der Batterieentladung relativ genau ist und die Spannungs-kurve zu diesem Zeitpunkt auch am steilsten ist. Die Verwendung dieser Spannung als Kalibrierungspunkt ist daher angemessener. Deshalb setzen wir hier 3 EDV-Punkte: `edv2` (7%), `edv1` (3%) und `edv0` (0%).

Nach dem Festlegen dieser 3 Kalibrierspannungen wird PiPower Pro die Batterie kalibrieren, wenn sie zu diesen 3 Punkten entladen wird: $\text{MaxKapazität} = \text{MaxKapazität} - \text{Kapazität} + \text{MaxKapazität} * 7\%$ Um eine unbegrenzte Kalibrierung am selben Punkt aufgrund von Spannungsschwankungen zu vermeiden, wird die Kalibrierung auf einmal vor dem Erreichen des RCV (Zurücksetzen der Kalibrierspannung, Standard 8,0V) begrenzt. Sowohl `edv2`, `edv1`, `edv0` als auch `rcv` können im Dienst Service konfiguriert werden, siehe *PiPower Pro Entität* für Details.

Anzeige

Wenn der Coulomb-Zähler-Algorithmus aktiviert ist, wechselt die Batterieanzeige ebenfalls in den Coulomb-Zähler-Modus. Es besteht jedoch eine geringe Chance, dass die Batterieanzeige falsch ist oder sogar die Batterieanzeige zurückgesetzt wird.

Das Verhältnis zwischen den Batterieanzeigen und der Leistung ist wie folgt:

- 4 LEDs alle an: 75%
- 3 LEDs an: 50%
- 2 LEDs an: 25%
- 1 LED an: 10%
- 4 LEDs alle aus: 0%, Batterien müssen geladen werden.

4.8 Maßgeschneiderte Entwicklung

Sollten Sie feststellen, dass die Grundfunktionen von PiPower Pro nicht ausreichend für Ihre Anforderungen sind, können Sie individuelle Entwicklungen für PiPower Pro durchführen.

Die gesamte Software für PiPower Pro ist Open Source. Hier finden Sie das Grundtutorial und die Vorbereitung für die individuelle Entwicklung.

1. **Aktivieren Sie den Entwicklermodus von Home Assistant.**
 - a. Öffnen Sie die Home Assistant Verwaltungsseite.
 - b. Wählen Sie „Configuration“ in der unteren linken Ecke.
2. **Installieren Sie ESPHome.**
 - a. Öffnen Sie die Home Assistant Verwaltungsseite.

- b. Wählen Sie „Configuration“ in der unteren linken Ecke.
- c. Wählen Sie „Add-ons“.
- d. Klicken Sie auf den „Add“-Button.
- e. Suchen Sie nach „esphome“.
- f. Klicken Sie auf „Installieren“.
- g. Nach der Installation klicken Sie auf „Start“.
- h. Wählen Sie „Add to Sidebar“.

3. Erstellen Sie ein neues Gerät.

- a. Klicken Sie in der Seitenleiste auf „ESPHome“, um die ESPHome Verwaltungsseite aufzurufen.
- b. Wählen Sie „New Device“.
- c. Geben Sie den Gerätenamen ein, z.B. „PiPower Pro“.
- d. Bei der ersten Konfiguration müssen Sie auch das WLAN-Konto und das Passwort eingeben.
- e. Wählen Sie „ESP32 S2“.
- f. Bestätigen Sie und überspringen Sie die Installation.

4. Konfigurieren Sie das neue Gerät.

- a. Wählen Sie das gerade erstellte Gerät aus und klicken Sie auf „Edit“, um die YAML-Bearbeitungsseite zu öffnen.
- b. Fügen Sie am Ende die PiPower Pro Vorlage hinzu:

```
packages:  
  remote_package: github://sunfounder/pipower-pro/pipower-pro-template.  
  ↪yaml@main
```

- c. Klicken Sie in der oberen rechten Ecke auf „Installieren“, um es auf PiPower Pro zu installieren.

4.9 Mehrere PiPower Pro Einheiten

Wenn Sie mehrere PiPower Pro Einheiten in der gleichen Home Assistant Umgebung verwenden möchten, müssen Sie die YAML-Einstellungen ändern. Fügen Sie `name_add_mac_suffix: true` unter „esphome“ hinzu.

```
esphome:  
  name: pipower-pro  
  friendly_name: PiPower-Pro  
  name_add_mac_suffix: true
```

4.10 IO-Erweiterung

J4 wird zur Erweiterung verwendet. Der IO stammt vom ESP32 S2.

Tab. 1: IO-Erweiterung

Funktionen	Pin	Pin	Funktionen
5V	5V	3V3	3V3
ADC,Touch,GPIO8	8	GND	Ground
ADC,Touch,GPIO9	9	10	GPIO10,Touch,ADC
ADC,DAC,GPIO18	18	36	GPIO36
GPIO37	37	38	GPIO38
GPIO39	39	40	GPIO40
GPIO41	41	42	GPIO42

5.1 PiPower Pro funktioniert nicht?

Wenn Sie den Akku zum ersten Mal einlegen oder den Akku ausstecken und erneut einsetzen, wird der Akku nicht ordnungsgemäß funktionieren.

Das liegt daran, dass, wenn der Akku entfernt wird, aufgrund des Mechanismus der integrierten Tiefentladeschutzschaltung, die Spannung als zu niedrig betrachtet wird, wodurch die Schutzschaltung aktiviert wird;

In diesem Fall müssen Sie das **Type C**-Kabel in den Ladeanschluss stecken, um die Schutzschaltung freizugeben, und der Akku kann normal verwendet werden.

5.2 Mit welchen Einplatinencomputern kann PiPower Pro verwendet werden?

Die mit PiPower Pro kompatiblen Einplatinencomputer sind unten aufgeführt.

Bemerkung: Funktionell kompatibel bedeutet, dass es normal von PiPower Pro mit Strom versorgt werden kann.
