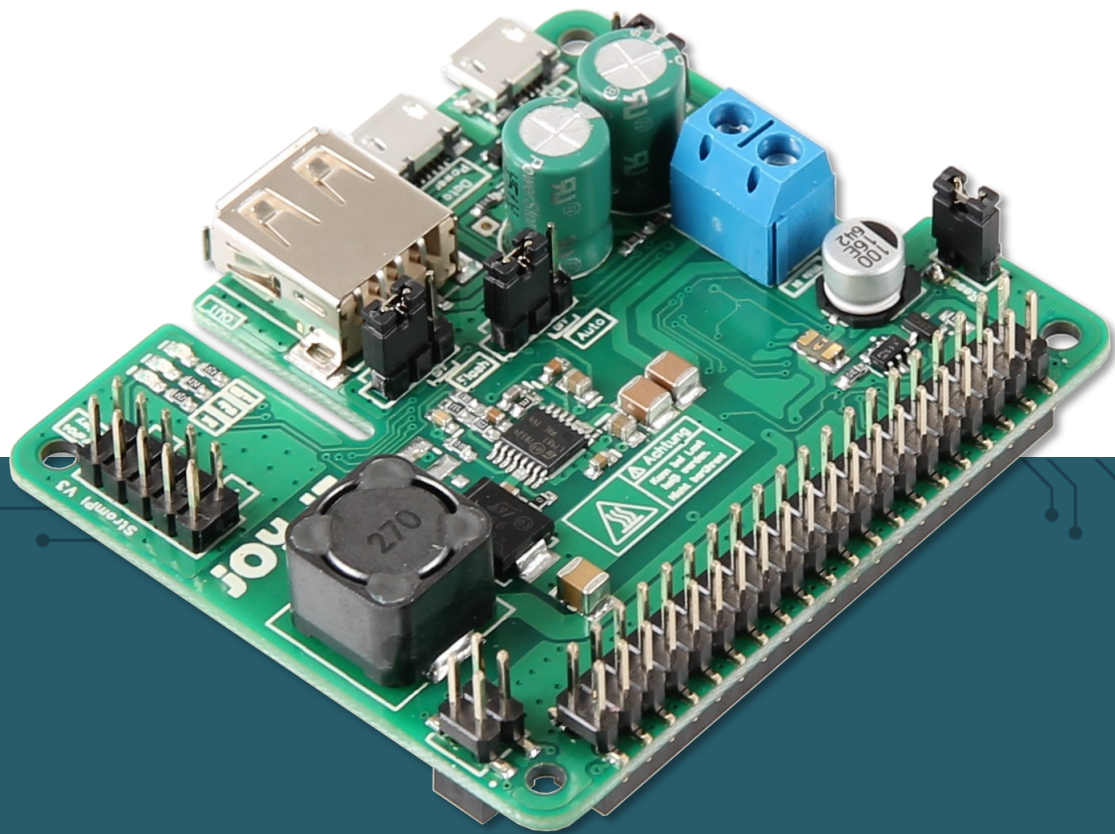




# STROMPI 3

Power-Solution

## 1. INHALTVERZEICHNIS

1. Inhaltsverzeichnis
2. Allgemeine Informationen
3. Übersicht
4. Stromquellen
5. QuickStart
6. Funktionen
  - 6.1 Eingangspriorisierung
  - 6.2 Zeitgesteuerte Anwendung
    - 6.2.1 Real Time Clock
    - 6.2.2 Automatische Zeitkonfiguration
    - 6.2.3 Shutdown Timer
    - 6.2.4 Alarm Modes
    - 6.2.5 PowerOff-Alarm Mode
    - 6.2.6 WakeUp - Alarm Mode
    - 6.2.7 Minute WakeUp-Alarm
    - 6.2.8 Weekend WakeUp-Alarm
    - 6.2.9 Intervall Funktion
7. Zusätzliche Funktionen
  - 7.1 Serialless Mode
  - 7.2 Verwendungsbeispiel anhand der Echtzeituhr
  - 7.3 PowerOn-Button Mode
  - 7.4 Poweroff Mode
  - 7.5 Power Save Mode
8. Konfiguration des StromPi 3
  - 8.1 Aktivierung der seriellen Kommunikation
  - 8.2 Ausführen des GUI Skripts
  - 8.3 VNC für Remotedesktopverbindung
  - 8.4 SSH-Verbindung über PuTTY und Xming
  - 8.5 SSH-Verbindung über MobaXterm
9. StromPi 3 Skripte
  - 9.1 Serial Skripte
  - 9.2 Serialless Skripte
  - 9.3 Raspberry Pi Pin-Layout
  - 9.4 Automatisches Ausführen von Skripten
10. Modifikationen
11. Öffnen der Konsole
12. Konsolen Befehle
13. Serielle Befehle
14. EU-Konformitätserklärung
15. Sonstige Informationen
16. Support

## 2. ALLGEMEINE INFORMATIONEN

Sehr geehrter Kunde,

vielen Dank, dass Sie sich für unser Produkt entschieden haben. Im Folgenden zeigen wir Ihnen, was bei der Inbetriebnahme und der Verwendung zu beachten ist.

Sollten Sie während der Verwendung unerwartet auf Probleme stoßen, so können Sie uns selbstverständlich gerne kontaktieren.

Der StromPi 3 eröffnet dem Raspberry Pi® und vielen weiteren Einplatinencomputer völlig neue Einsatzmöglichkeiten. Zusätzlich zu den bereits verfügbaren Spannungseingängen des StromPi 2 (mUSB, WideRange 6 - 61 V) lässt sich der StromPi 3 um eine optionale, aufsteckbare Batterieeinheit ergänzen, welche Ihren Raspberry Pi mit einer LiFePO4 Batterie um eine wiederaufladbare Notstromquelle erweitert. Unvorhergesehene Stromausfälle gehören damit der Vergangenheit an!

Die einzelnen Eingänge können variabel priorisiert werden, womit sich der StromPi 3 perfekt an Ihr Projekt anpassen lässt. Mit einem Spannungsbereich von 6V-61V und einem Strom von bis zu 3A können auch größere Projekte mit genügend Strom beliefert werden. Zusätzlich dazu besitzt der StromPi 3 nun eine eigene konfigurierbare und autonom agierende Mikrosystemeinheit, welche für ein programmierbares Start-Stopp-Verhalten genutzt werden kann: Lassen Sie Ihr System zu vorgegebenen Zeiten hoch- und runterfahren, um Messungen durchzuführen oder um Geräte ein- oder auszuschalten (auch ideal für Digital Signale oder zur Maschinensteuerung). Die darin enthaltende Echtzeituhr kann auch für einen Zeitabgleich des Raspberry Pis im Betrieb ohne Internetzugang verwendet werden. Zudem wurde der StromPi 3 um eine Steuerung über die serielle Schnittstelle erweitert (Ladezustand des Akkus, Spannungswerte der Ein-/Ausgänge, Steuerung und Konfiguration), was ihn zu einem absoluten Allrounder macht.

Mit dem neuen StromPi 3 sind Sie für jeden Einsatz perfekt ausgerüstet!

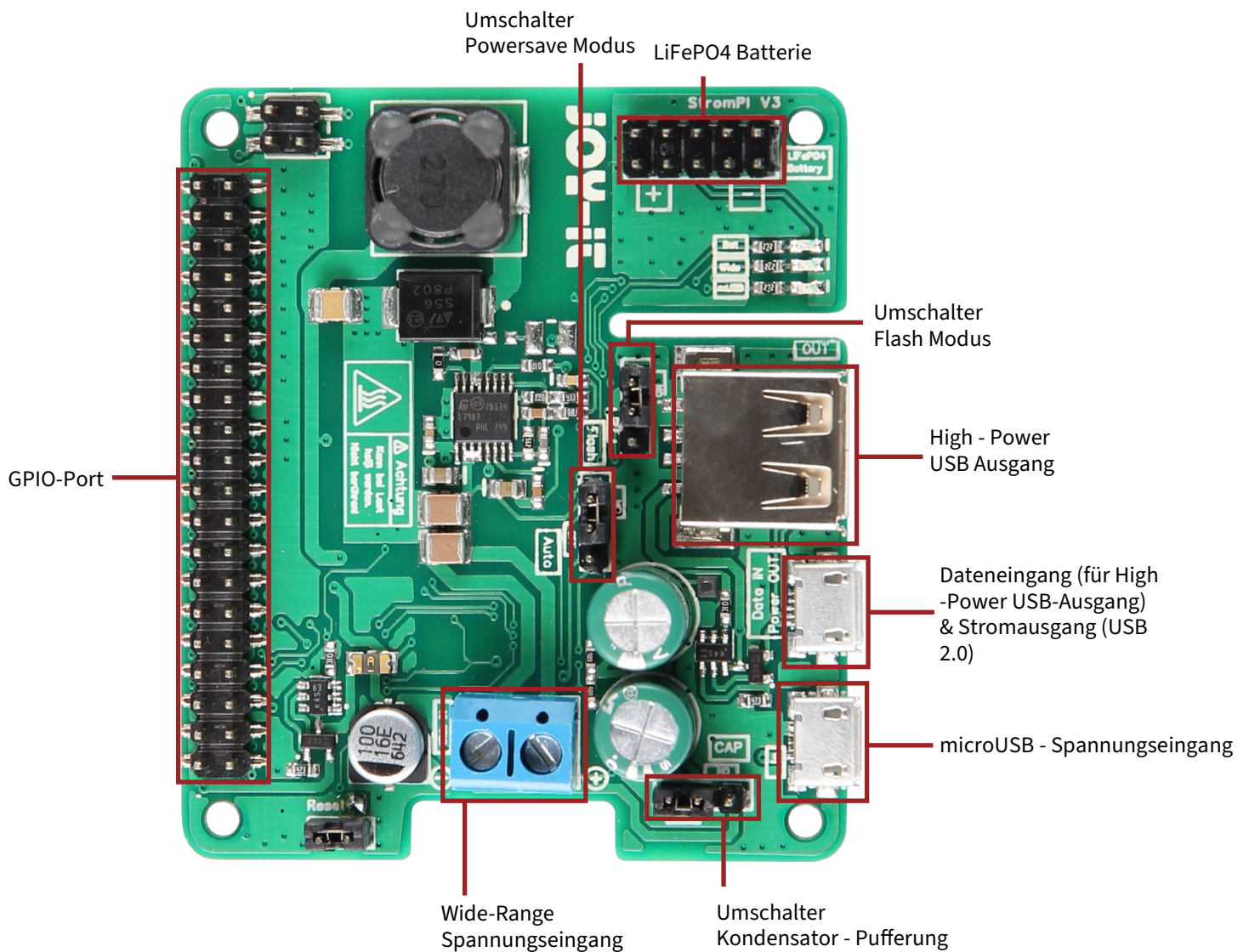


Diese Anleitung ist für Firmware Version **1.72** ausgelegt. Wir empfehlen Ihnen die StromPi Firmware immer aktuell zu halten. Sie können die neuste Firmware [hier](#) herunterladen, eine Anleitung zum Flashen der Firmware befindet sich im Downloadverzeichnis.

### 3. ÜBERSICHT

Es gibt mittlerweile 2 Versionen des StromPi, die erste Version **Rev 1.0** Und die neue Version **Rev 1.1**.

Bei der Rev 1.1 wurden die Modifikationen für den PowerOnButton und den SeriallessMode bereits integriert.

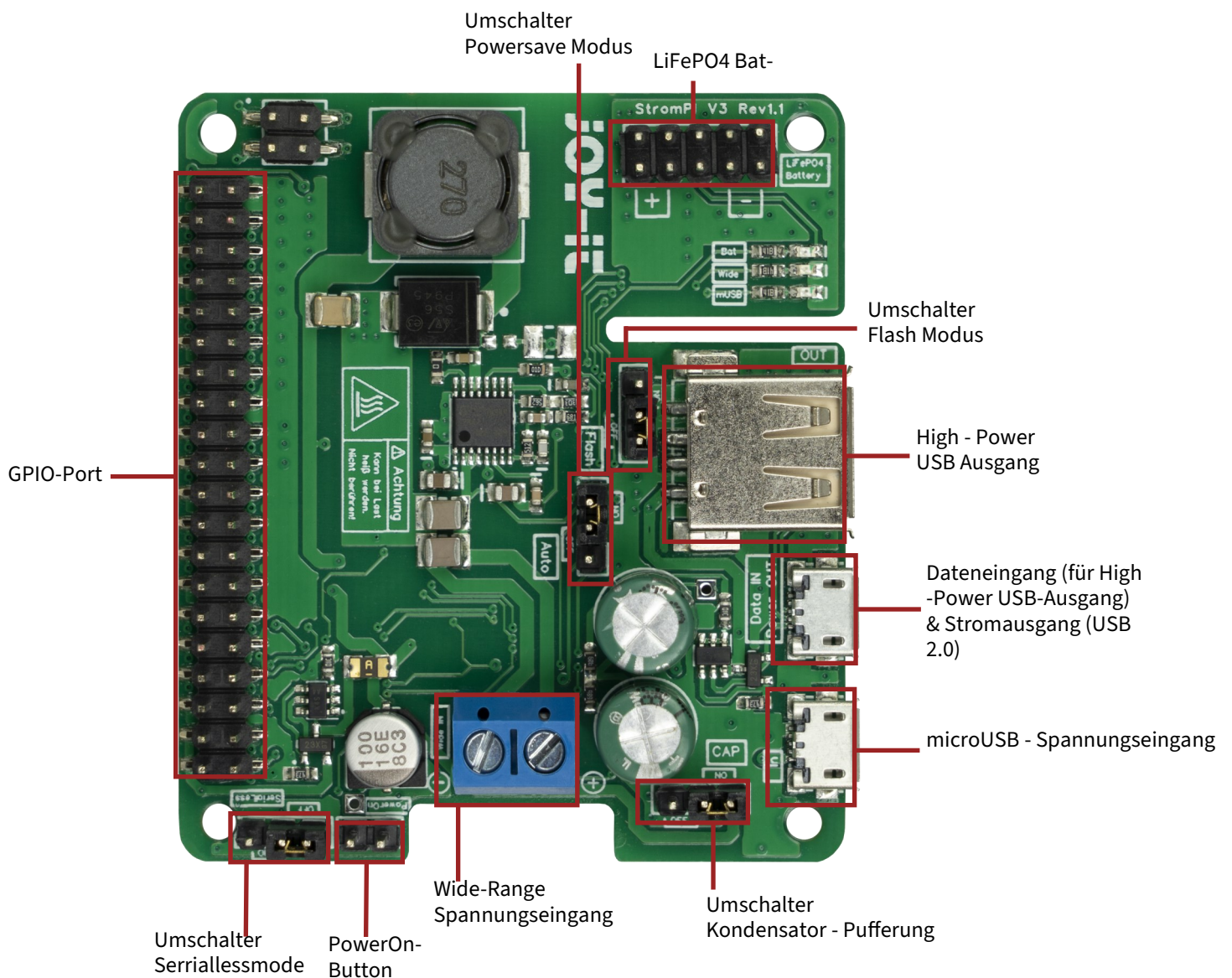


Hinweis zu der Verwendung des High-Power USB Ausgangs, für zum Beispiel die Nutzung einer externen Festplatte:  
Schließen Sie Ihre externe Festplatte an den HighPower USB-Ausgang an und verbinden Sie den micro-USB Dateneingang vom StromPi mit einem USB-Port vom Raspberry Pi für die Datenübertragung.



Bei starker Last kann sich der StromPi 3 erhitzen. Berühren Sie die Platine nicht während des Betriebs, um Verbrennungen zu vermeiden. Ermöglichen Sie zudem auch eine gute Luftzufuhr, um ein Überhitzen des Gerätes auszuschließen.

## Rev 1.1



Der neu hinzugefügte Serialless-Jumper verbindet, wenn auf „ON“ gesteckt, den GPIO 21 (Pin 40) des Raspberry Pi mit einem GPIO des Mikrocontrollers. Die Serialless-Funktion muss trotzdem wie gewohnt in der Konfiguration des StromPi 3 aktiviert werden.

Hinweis zu der Verwendung des High-Power USB Ausgangs, für zum Beispiel die Nutzung einer externen Festplatte:  
Schließen Sie Ihre externe Festplatte an den HighPower USB-Ausgang an und verbinden Sie den micro-USB Dateneingang vom StromPi mit einem USB-Port vom Raspberry Pi für die Datenübertragung.



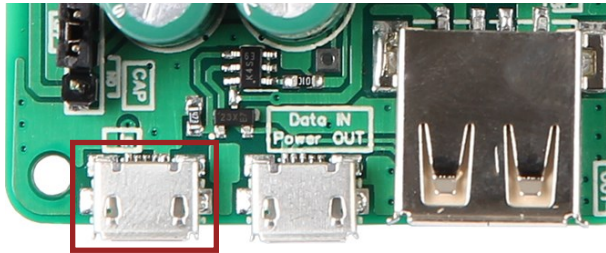
Bei starker Last kann sich der StromPi 3 erhitzen. Berühren Sie die Platine nicht während des Betriebs, um Verbrennungen zu vermeiden. Ermöglichen Sie zudem auch eine gute Luftzufuhr, um ein Überhitzen des Gerätes auszuschließen.

## 4. STROMQUELLEN

Der StromPi 3 kann über drei verschiedene Stromquellen versorgt werden. Im Falle eines Stromausfalls schaltet der StromPi 3 automatisch stufenlos auf die als sekundär konfigurierte Stromquelle und ggf. tertiäre Stromquelle.

### microUSB

Der StromPi 3 kann selbstverständlich, wie gewöhnlich, über einen microUSB - Anschluss mit Strom versorgt werden.

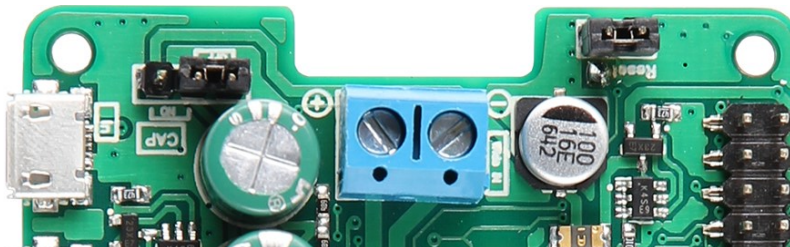


### Wide - Range

Im „Wide“ - Modus kann der StromPi 3 über eine Spannungsquelle im Bereich von 6 - 61 V betrieben werden.

Schließen Sie dazu an den markierten Anschluss einfach eine beliebige Spannungsquelle, wie ein Netzteil oder eine Fahrzeugbordnetzversorgung, an.

**Achtung!** Die Spannungsquelle muss im eingeschalteten Zustand im Bereich von 6 - 61 V liegen. Bitte achten Sie auf die Polarität, um Schäden am Raspberry Pi und am StromPi zu vermeiden.



### Batterie - Einheit

Der StromPi 3 kann auch mit einer optionalen 1000 mAh oder 2000mAh LiFePO4 Batterie um eine wieder aufladbare Notstromquelle erweitert werden.

Dazu wird die separat erhältliche Batterieeinheit einfach auf den StromPi 3 aufgesteckt und kann dann als sekundäre oder tertiäre Spannungsquelle priorisiert werden. Dadurch, dass die GPIO - Leiste durchgeführt ist, müssen Sie bei der Verwendung der Batterieeinheit keinerlei Einschränkungen in Kauf nehmen.

Wenn die orangene LED an der Batterie-Einheit leuchtet, bedeutet das, dass die Batterie aufgeladen wird.

Sollte die LED blinken, ist es ein Hinweis darauf, dass die Batterie-Einheit defekt ist.

Um alle Funktionen des StromPis im vollem Umfang nutzen zu können, müssen Sie unsere Python-Skripte benutzen. Diese werden im Kapitel 8 „StromPi 3 Skripte“ aufgelistet und erklärt.



## 5. QUICKSTART

In diesem Kapitel finden Sie die notwendigen Schritte, um den StromPi 3 verwenden zu können.

### Aktivierung der seriellen Kommunikation

#### **Näheres finden Sie in Kapitel 8.1**

Installieren Sie mit den folgenden Befehlen benötigte Abhängigkeiten und öffnen Sie anschließend die Datei config.txt.

```
sudo apt-get update
```

```
sudo apt-get upgrade
```

```
sudo apt remove python3-rpi.gpio
```

```
sudo apt-get install python3-serial minicom python3-pip  
python3-pil python3-pil.imageTk python3-rpi-lgpio  
stm32flash
```

```
sudo nano /boot/firmware/config.txt
```

Nun fügen Sie die folgenden Zeilen an das Ende der Datei an:

```
dtoverlay=miniuart-bt  
enable_uart=1
```

Öffnen Sie die Raspberry Konfiguration und deaktivieren Sie in **5 Interfacing Options** → **P6 Serial** die serielle Login-Shell und aktivieren Sie die serielle Hardware.

### Öffnen der seriellen Konsole des StromPi 3

#### **Näheres finden Sie in Kapitel 11**

Installieren Sie mit den folgenden Befehlen Minicom und stellen Sie eine Verbindung mit dem StromPi 3 her.

```
sudo apt-get install minicom
```

```
sudo minicom -D /dev/ttyAMA0 -b 38400
```

Die Konsole des StromPi 3 öffnen Sie mit:  
(die getippten Zeichen werden nicht angezeigt)

```
startstrompiconsole
```

... oder der Abkürzung:

```
sspc
```

Der Befehl **show-status** öffnet die aktuelle Konfigurationseinstellung und **help** gibt Ihnen eine Übersicht über die Befehle.

Der Startbildschirm der StromPi 3 Konsole sieht wie folgt aus:

```
-----  
Welcome to the StromPi 3 Console  
-----  
Type help to view a list of available commands.  
  
[When you press ENTER the previous command would be executed again]  
  
> |
```

## 6. FUNKTIONEN DES STROMPI 3

Strompi V3 Serial Config Tool

**JOY-IT®**

Options Without Timesettings:  
Choose StromPi V3 Powerpath Mode:  
① mUSB -> Wide ▾

Choose Powerfail Warning Mode:  
② Disabled ▾

Choose Battery Shutdown Mode:  
③ Disabled ▾

Choose Serialless Mode:  
④ Disabled ▾

Choose Power Save Mode:  
⑤ Power Save Mode: Disabled ▾

Poweroff Mode  
Choose Poweroff Mode:  
⑥ Poweroff Mode: Enabled ▾

Alarm Modes  
Choose Alarm Mode:  
⑦ Alarm-Mode: Time-Alarm ▾

Choose WakeUp-Alarm Mode:  
⑧ WakeUp-Alarm: Disabled ▾

Choose Power-Off Alarm Mode:  
⑨ PowerOff-Alarm: Disabled ▾

Interval Timer  
Choose Interval-Alarm Mode:  
⑩ Interval-Alarm: Disabled ▾

Real Time Clock Settings  
Settings for the Clock:  
⑪ ☒ No Change ☐ Manual Setting ☐ Sync with RaspberryPi

Set the Hours: 01 Hours  
Set the Minutes: 13 Minutes  
Set the Days: 01 Day  
Set the Year: 18 Year  
Set the Month: 05  
Set The Weekday: Tuesday ▾

Shutdown Timer  
Choose Shutdown Mode:  
⑫ Enabled ▾

Choose Shutdown timer: 10 Seconds

PowerOnButton  
Choose PowerOn-Button Mode:  
⑬ Enabled ▾

Choose Button Shutdown time: 30 Seconds

Choose Alarm Weekday: Monday ▾

Choose Alarm Day: 01 Day  
Alarm Month: 11 ▾

Setting WakeUp-Alarm minute: 00 Minutes  
Setting WakeUp-Alarm Hour: 00 Hours

Choose Power-Off Minutes: 00 Minutes  
Choose Power-Off Hours: 00 Hours

Interval On-Time in minutes: 00 minutes  
Interval Off-Time in minutes: 00 minutes

Please read the Readme file

UPDATE!

- ① Auswählen des StromPi V3 Powerpath Mode
- ② Aktivieren oder deaktivieren der Powerfail Warning
- ③ Einstellen des Battery Shutdown Mode
- ④ Aktivieren oder deaktivieren des Serialless Mode
- ⑤ Aktivieren oder deaktivieren des Power Save Mode
- ⑥ Aktivieren oder deaktivieren des Poweroff Mode
- ⑦ Einstellen des Alarm Mode
- ⑧ Einstellen des WakeUp-Alarm Modes
- ⑨ Einstellen des PowerOff-Alarm Modes
- ⑩ Einstellen des Interval - Alarm Modes
- ⑪ Einstellungen für die Echtzeituhr
- ⑫ Aktivieren oder deaktivieren des Shutdown-Timers und einstellen der Zeit (in Sekunden)
- ⑬ Aktivieren oder deaktivieren des PowerOn-Button Modes und einstellen der Zeit (in Sekunden)



Für das vornehmen aller Einstellungen die in den folgenden Kapiteln beschrieben werden, ist eine serielle Verbindung notwendig. In Kapitel 8 erfahren Sie, wie Sie die serielle Schnittstelle für den Raspberry Pi konfigurieren.

**StromPi V3 Powerpath Mode:**

Der StromPi V3 Powerpath Mode legt die Priorisierung der einzelnen Spannungsquellen fest.

**Powerfail Warning Mode:**

Der Powerfail Warning Mode warnt bei einem Stromausfall der priorisierten Spannungsquelle.

**Battery Shutdown Mode:**

Der Battery Shutdown Mode legt fest, bei welchem Ladestand der Batterie das Raspberry Pi herunter gefahren werden soll.

**Serialless Mode:**

Der Serialless Mode erlaubt die anderweitige Nutzung der seriellen Ports, indem über einen GPIO - Pin kommuniziert wird. [Rev 1.0: Modifizierung benötigt! Siehe Kapitel 10 Modifikationen]

**Power Save Mode:**

Der Power Save Modus ermöglicht es den Stromverbrauch des StromPi 3 drastisch zu reduzieren, wenn Wide als sekundäre Spannungsquelle eingestellt ist. Allerdings ist dies nur im Betrieb mit niedriger Stromstärke empfohlen (< 2,5 A / z.B. : Raspberry Pi ohne Peripherie).

**Real Time Clock Settings:**

In den RTC Settings kann man die Uhrzeit und das Datum der Echtzeituhr manuell einstellen, oder sie automatisch mit der Zeit des Raspberry Pis vergleichen und synchronisieren.

**Shutdown Timer:**

Wird ein Stromausfall an der primär eingestellten Spannungsquelle detektiert, so wird bei aktiviertem Shutdown ein Signal über die serielle Schnittstelle vom StromPi 3 an den Raspberry Pi gesendet (für das Herunterfahren mittels Skript) und die angegebene Zeit des Timers abgewartet, bis dann der StromPi 3 die Stromversorgung des Raspberry Pis unterbricht.

**PowerOnButton:**

Mit dem PowerOn-Button Mode kann man den ausgeschalteten Raspberry Pi wieder hochfahren, indem man den Kontakt zwischen den Reset Pins schließt. [Rev 1.0: Modifizierung benötigt! Siehe Kapitel 10 Modifizierung]

**Interval Timer:**

Mit dem Interval Timer können Sie Ihr System nach einer bestimmten Zeit herunter- und wieder hochfahren.

**Alarm Mode:**

Der Alarm Mode bezieht sich auf den WakeUp- und den Power-Off Alarm. Er gibt an ob der Alarm täglich, wöchentlich oder nur zu einem bestimmten Datum ausgeführt wird.

**WakeUp-Alarm Mode:**

Der WakeUp-Alarm fährt den Raspberry Pi zu dem angegebenen Zeitpunkt hoch.

**PowerOff -Alarm Mode:**

Der PowerOff-Alarm Mode fährt den Raspberry Pi zu dem angegebenen Zeitpunkt herunter.

**Poweroff Mode:**

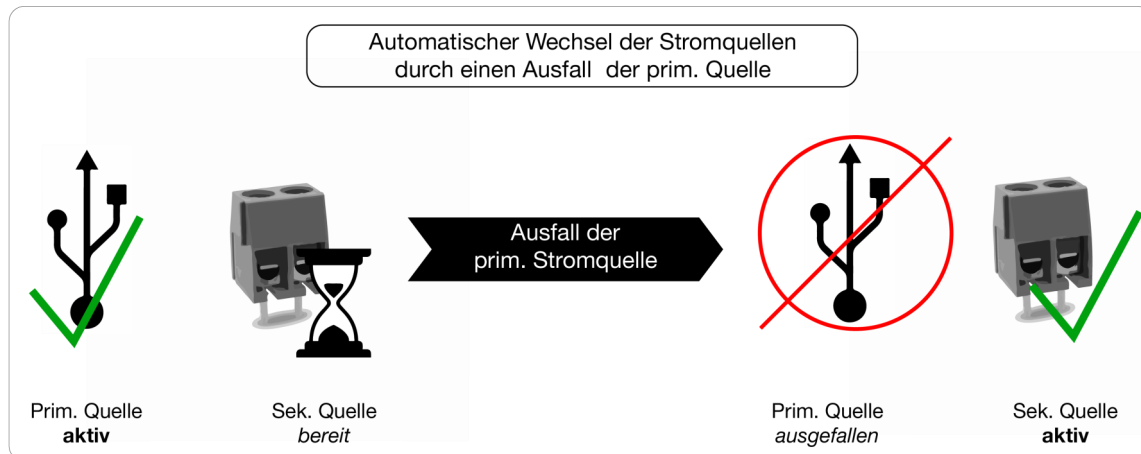
Der Poweroff Mode erfordert das Einschalten des Raspberry Pi mit dem PowerOn-Button nach einem Shutdown durch den Shutdown Timer. Zusätzlich verhindert der Poweroff Mode das Detektieren eines Zurückkehrens der primären Spannungsversorgung.

## 6.1 Eingangspriorisierung

Die Spannungsquelle aus der der StromPi 3 seine Energie bezieht, kann je nach Wunsch und Einsatzzweck eingestellt werden.

Zusätzlich lässt sich eine zweite oder sogar eine dritte Spannungsquelle wählen, die automatisch die Versorgung übernimmt, falls die erste Quelle ausfallen sollte.

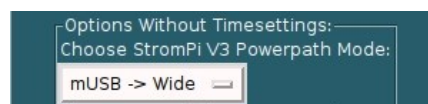
Die Werkseinstellung ist so konfiguriert, dass der microUSB-Eingang als Hauptstromquelle dient. Sollte diese ausfallen, so wird automatisch der Wide-Range-Eingang als alternative Quelle hinzugeschaltet.



Es stehen insgesamt sechs verschiedene Modi zur Verfügung. Die primäre Spannungsquelle dient dabei als Hauptstromquelle. Die sekundäre Spannungsquelle wird automatisch zugeschaltet, falls die primäre Quelle ausfallen sollte.

| MODUS                      | PRIMÄR       | SEKUNDÄR     | TERTIÄR  |
|----------------------------|--------------|--------------|----------|
| Mode 1                     | microUSB     | Wide - Range | -        |
| Mode 2                     | Wide - Range | microUSB     | -        |
| Mode 3                     | microUSB     | Batterie     | -        |
| Mode 4                     | Wide - Range | Batterie     | -        |
| Mode 5<br>(ab Version 1.6) | microUSB     | Wide - Range | Batterie |
| Mode 6<br>(ab Version 1.6) | Wide - Range | microUSB     | Batterie |

Den entsprechenden Modus können Sie über das Configskript unter „Choose StromPi V3 Powerpath Mode“ einstellen.



## 6.2 Zeitgesteuerte Anwendungen

### 6.2.1 Real Time Clock



Der StromPi 3 verfügt zusätzlich über eine Echtzeituhr. Sollte der Strom einmal ausfallen, so sind auch zeitgesteuerte Anwendungen nicht gefährdet. Auch zeitgesteuertes Hoch- und Herunterfahren des Raspberry Pis sind kein Problem! Bitte beachten Sie, dass für einen fortlaufenden Betrieb der RTC zumindest eine der Stromquellen (mUSB, Wide oder Batterie) am StromPi3 angeschlossen bleiben muss — werden alle Stromquellen getrennt, wird die aktuelle Zeit nicht gespeichert.

Die Echtzeituhr (RTC) des StromPi 3 kann mit dem Configskript eingestellt werden:

#### Uhrzeit und Datum manuell einstellen

Um die Uhrzeit und das Datum manuell einzustellen, klicken Sie auf **Manual Setting**. Nun können Sie die Uhrzeit und das Datum in die entsprechenden Felder rechts daneben eintragen. Mit dem Klicken auf **Update** übernehmen Sie die Einstellungen.

#### Uhrzeit und Datum mit dem Raspberry Pi synchronisieren

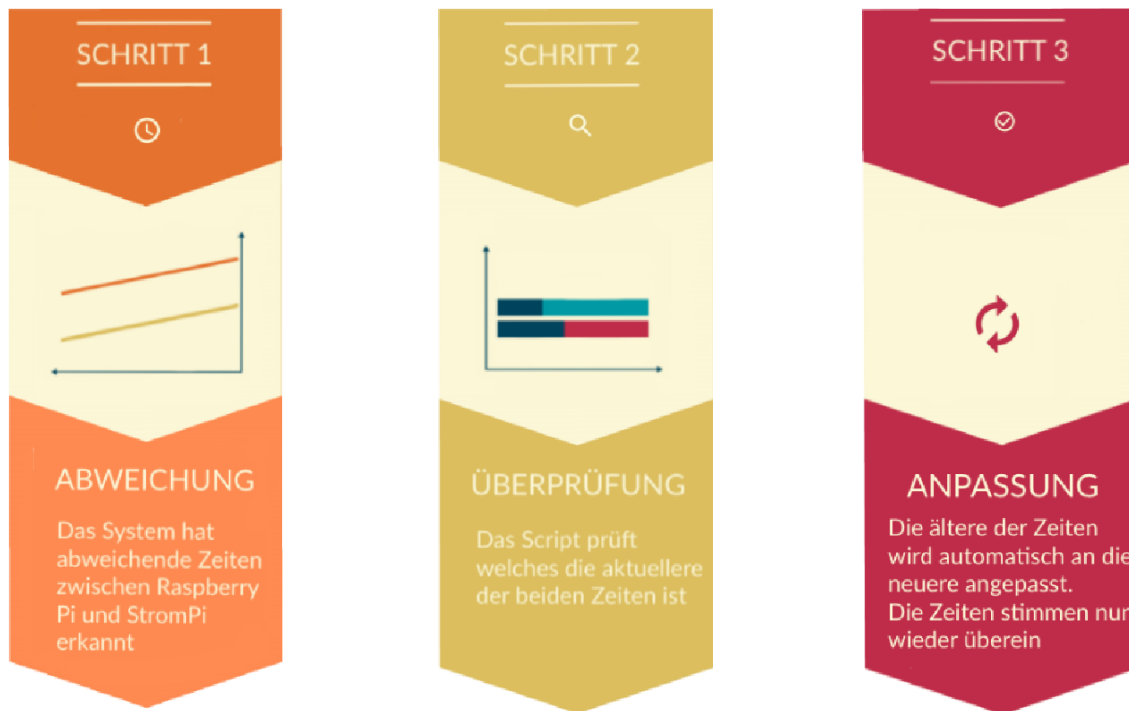
Um die Uhrzeit und das Datum mit dem Raspberry Pi zu synchronisieren, klicken Sie auf **Sync with RaspberryPi**. Mit dem Klicken auf **Update** wird die Uhrzeit der RTC des StromPis und die des Raspberry Pis verglichen. Die ältere Uhrzeit wird mit der aktuelleren überschrieben. Die voreingestellte Zeit vom StromPi ist der 01. Mai 2018 und der Raspberry Pi aktualisiert seine Uhrzeit, wenn er mit dem Internet verbunden ist.

### 6.2.2 Automatische Zeitkonfiguration

Zusätzlich stellen wir Ihnen ein Skript zur Verfügung welches nach dem Start Ihres Raspberry Pis die Echtzeituhr automatisch synchronisiert. Das Skript zur Aktualisierung der Zeit können Sie [hier](#) in unserem Downloadbereich unter „Scripte“ herunterladen.

Laden Sie das Archiv herunter, entpacken Sie das Skript und kopieren Sie dieses auf Ihren Raspberry Pi.

In dem Kapitel 9.4 Automatisches Ausführen von Skripten können Sie nachlesen, wie Sie Skripte zum Autostart hinzufügen, damit sie beim Start des Raspberry Pis automatisch ausgeführt werden.



### 6.2.3 Shutdown Timer

Der StromPi 3 verfügt über einen Shutdown Timer der automatisch, nachdem ein Ausfall der primären Stromversorgung erkannt wurde, die Stromversorgung Ihres Raspberry Pis unterbricht.

Die Zeitspanne, wann der Shutdown nach dem Ausfall der Stromversorgung erfolgt, können Sie individuell, ganz nach Ihren Bedürfnissen, anpassen.

Haben Sie den Shutdown Timer beispielsweise auf 600 gestellt (600 Sekunden → 10 Minuten), so verhält sich das System wie folgt:



Wenn der StromPi3 nun einen Stromausfall erkennt, startet der Timer. Nach 10 Minuten trennt der StromPi3 dann die Stromversorgung zum Raspberry Pi. Zusätzlich stellen wir ein Skript bereit welches entweder über die serielle Konsole oder über einen GPIO Pin einen Stromausfall erkennen kann und dann mit dem Sicheren herunterfahren des Raspberry Pi beginnt.

Der Shutdown Timer lässt sich mit dem Configskript unter dem Punkt **Shutdown Timer** einstellen.

Shutdown Timer

Choose Shutdown Mode:

Choose Shutdown timer:

Disabled

10 Seconds

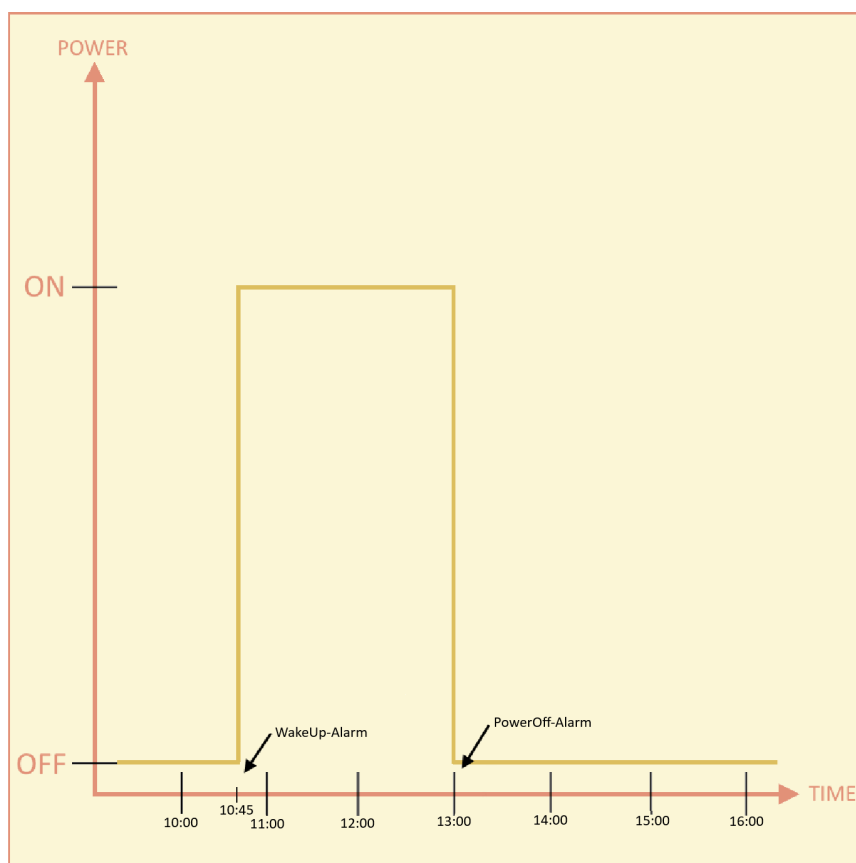
Zum Aktivieren des Shutdown Timers wählen Sie **Enable** bei **Choose Shutdown Mode** und geben Sie die gewünschte Ausschaltzeit in Sekunden in das nebenliegende Feld ein.  
Zum Schluss mit **Update** bestätigen.

#### 6.2.4 Alarm Modes

Der StromPi3 verfügt über die Möglichkeit den Raspberry zeitgesteuert hoch und herunterzufahren.

Mit der Kombination aus WakeUp- und PowerOff-Alarm, lässt sich ein individuelles Start-Stopp-Verhalten für Ihren Raspberry Pi konfigurieren (z.B. jeden Tag um 10:45 Uhr hochfahren und um 13:00 Uhr herunterfahren).

So ist Ihr Raspberry Pi ausschließlich in dem Zeitfenster aktiv, indem die anfallenden Aufgaben bearbeitet werden müssen.



### 6.2.5 PowerOff - Alarm Mode

Der PowerOff-Alarm ermöglicht es Ihnen den Raspberry Pi jeden Tag zu einer voreingestellten Uhrzeit herunterfahren zu lassen. Allerdings kann der Raspberry Pi nur einmal am Tag mit diesem Modus heruntergefahren werden.



Wenn Sie den StromPi3 zeitgesteuert herunterfahren möchten starten Sie das Configskript.

| Alarm Modes                                              |                                            |                                        |                                   |
|----------------------------------------------------------|--------------------------------------------|----------------------------------------|-----------------------------------|
| Choose Alarm Mode:<br>1 = Daily time alarm               | Choose Alarm Weekday:<br>nothing           | Choose Alarm Day:<br>01 Day            | Alarm Month:<br>11                |
| Choose WakeUp-Alarm Mode:<br>WakeUp-Alarm: Disabled      | Setting WakeUp-Alarm minute:<br>00 Minutes | Setting WakeUp-Alarm Hour:<br>00 Hours |                                   |
| Choose Power-Off Alarm Mode:<br>PowerOff-Alarm: Disabled | Choose Power-Off Minutes:<br>41 Minutes    | Choose Power-Off Hours:<br>00 Hours    | Minute Wakeup Time:<br>01 Minutes |

Schalten Sie **Choose PowerOff-Alarm Mode** auf **Enable** um den PowerOff-Alarm, zu aktivieren. Die Uhrzeit um die der Raspberry Pi herunterfahren soll tragen Sie in die zwei Felder rechts daneben ein. Bei **Choose Power-Off Minutes** die Minuten und bei **Choose Power-Off Hours** die Stunden.

### 6.2.6 WakeUp - Alarm Mode

Der WakeUp-Alarm ermöglicht es Ihnen den Raspberry zeitgesteuert hochfahren zu lassen. Zusätzlich haben Sie die Möglichkeit dieses hochfahren an ein spezielles Datum oder an einen Wochentag zu binden.



Wenn Sie den StromPi 3 zeitgesteuert hochfahren möchten starten Sie das Configskript.

| Alarm Modes                                              |                                            |                                        |                                   |
|----------------------------------------------------------|--------------------------------------------|----------------------------------------|-----------------------------------|
| Choose Alarm Mode:<br>1 = Daily time alarm               | Choose Alarm Weekday:<br>nothing           | Choose Alarm Day:<br>01 Day            | Alarm Month:<br>11                |
| Choose WakeUp-Alarm Mode:<br>WakeUp-Alarm: Disabled      | Setting WakeUp-Alarm minute:<br>00 Minutes | Setting WakeUp-Alarm Hour:<br>00 Hours |                                   |
| Choose Power-Off Alarm Mode:<br>PowerOff-Alarm: Disabled | Choose Power-Off Minutes:<br>41 Minutes    | Choose Power-Off Hours:<br>00 Hours    | Minute Wakeup Time:<br>01 Minutes |

Schalten Sie **Choose WakeUp-Alarm Mode** auf **Enable** um den WakeUp-Alarm zu aktivieren. Die Uhrzeit um die der Raspberry Pi hochfahren soll tragen Sie in die zwei Felder rechts daneben ein. Bei **Choose WakeUp-Alarm Minutes** die Minuten und bei **Choose WakeUp-Alarm Hours** die Stunden. Der Raspberry Pi kann nur einmal am Tag über diesen Modus gestartet werden.

Je nachdem was Sie bei **Choose Alarm Mode** eingestellt haben, fährt Ihr Raspberry Pi nun täglich, wöchentlich oder nur zu einem bestimmten Datum zu der eingestellten Zeit hoch. Mit Alarm Mode 1 fährt Ihr Raspberry jeden Tag um die eingestellte Uhrzeit hoch. Mit Alarm Mode 2 fährt Ihr Raspberry an einem speziellem Datum zur angegebenen Zeit hoch. Mit Alarm Mode 3 fährt ihr Raspberry an einem speziellem Wochentag hoch. Mit Alarm Mode 4 aktiviert man den Minute WakeUp Alarm. Der Alarm-Mode bezieht sich ausschließlich auf den WakeUp-Alarm, auf den PowerOff-Alarm hat er keinen Einfluss.



**Achtung!** Ihr Raspberry Pi kann über den WakeUp-Alarm nur gestartet werden, wenn dieser zuvor automatisch über den PowerOff-Alarm heruntergefahren wurde oder er manuell mit dem seriellen Befehl *poweroff* heruntergefahren wurde.

### 6.2.7 Minute WakeUp-Alarm Mode

Der Minute WakeUp-Alarm ermöglicht es Ihnen einen automatischen Timer, zum Starten des Raspberry Pis, zu starten nachdem der Raspberry Pi durch den PowerOff-Befehl oder den PowerOff-Alarm heruntergefahren wurde. Dieser Timer wird in Minuten angegeben. Um diesen Modus zu Aktivieren muss im Configskript den Alarm Modus auf **Minute WakeUp-Alarm** gestellt werden.

Anschließend muss noch die Zeit eingestellt werden (Angabe in Minuten):

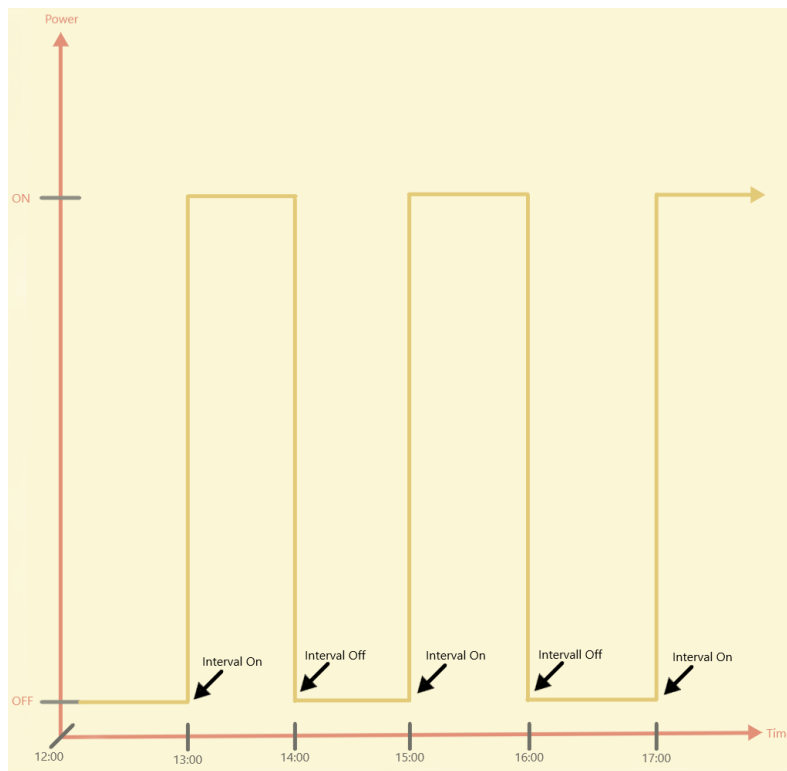
### 6.2.8 Weekend WakeUp-Alarm

Der Weekend WakeUp-Alarm ermöglicht es Ihnen den WakeUp-Alarm für das Wochenende ein oder aus zu schalten.

Wenn Sie diesen Modus auf **Disabled** setzen wird der StromPi 3 am Wochenende (Samstag und Sonntag) nicht durch den WakeUp-Alarm gestartet.

Sollten Sie den Raspberry Pi weiterhin am Wochenende durch den WakeUp-Alarm starten wollen muss der Weekend WakeUp-Alarm auf **Enabled** stehen.

## 6.2.9 Intervall Funktion



Die Intervall Funktion ermöglicht es, Zeitintervalle zu definieren in denen der Raspberry Pi mit Hilfe des StromPi 3 ausgeschaltet und wieder eingeschaltet wird.

Die Funktion können Sie im Configskript unter dem Punkt **Interval Timer** aktivieren und konfigurieren.

| Interval Timer              |                              |                               |
|-----------------------------|------------------------------|-------------------------------|
| Choose Interval-Alarm Mode: | Interval On-Time in minutes: | Interval Off-Time in minutes: |
| Interval-Alarm: Disabled    | 00 minutes                   | 00 minutes                    |

Schalten Sie **Choose Interval-Alarm Mode** auf **Enable** um die Intervall-Funktion zu aktivieren.

Unter **Interval On-Time in minutes** können Sie die Zeit, die der Raspberry Pi eingeschaltet bleiben soll in Minuten angeben.

Unter **Interval Off-Time in minutes** können Sie die Zeit die der Raspberry Pi ausgeschaltet bleiben soll in Minuten angeben.

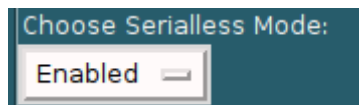
## 7. ZUSÄTZLICHE FUNKTIONEN

### 7.1 Serialless Mode

Der Serialless Modus ermöglicht es Ihnen jegliche serielle Kommunikation des StromPi zu unterdrücken, so können Sie z.B. ein anderes HAT, welches die serielle Schnittstelle benutzt, zeitgleich mit dem StromPi nutzen. Allerdings verhindert dies auch das Konfigurieren des StromPi 3. Aus diesem Grund ist es sinnvoll den StromPi vor dem aktivieren des Serialless Modus zu konfigurieren.

Zusätzlich wird durch diesen Modus die Shutdown-Warnung nicht, wie bisher, über die serielle Schnittstelle, sondern über einen freien GPIO-Pin übertragen.

Um den Serialless Modus zu aktivieren, müssen Sie den Serialless Mode im Configskript unter **Choose Serialless Mode** auf **Enable** stellen.



Außerdem müssen Sie das Skript `Start_Serialless.py` ausführen.

```
pi@raspberrypi:~/StromPi3_Scriptfolder_12062019/Serialless $ sudo python3 Start_Serialless.py
Enabled Serialless
```

Wenn Sie dies erledigt haben, ist der Serialless Modus aktiviert und die serielle Kommunikation wird nun vollständig unterdrückt.

Um den Serialless Modus wieder zu beenden, müssen Sie das Skript `Stop_Serialless.py` ausführen. Dieses Skript setzt den GPIO Pin auf Low, wenn der GPIO Pin für mehr als 5 Sekunden auf Low steht wird der Serialless Modus wieder deaktiviert.

```
pi@raspberrypi:~/StromPi3_Scriptfolder_12062019/Serialless $ sudo python3 Stop_Serialless.py
Setting GPIO to LOW to Disable Serialless Mode.
This will take approx. 10 seconds.
Serialless Mode is Disabled!
```



Für diesen Modus ist, bei der StromPi Rev 1.0, die Serialless - Modifikation notwendig, diese wird im Kapitel 10 Modifikationen genau beschrieben.

Bei der StromPi Rev 1.1 muss nur der Seriallessmode Jumper auf On gestellt werden.



Die hier verwendeten Skripte benötigen Python3.

### 7.2 Verwendungsbeispiel anhand der Echtzeituhr

Um Ihnen nun eine der vielen Anwendungsmöglichkeiten zu zeigen, haben wir ein Skript erstellt (`RTCSerialless.py`), welches nach dem Starten des Skriptes den Serialless Modus deaktiviert anschließend die Echtzeituhr synchronisiert und danach den Serialless Modus wieder aktiviert.

Hier einmal die Konsolenausgabe bei erfolgreichem Ausführen des Skriptes:

```
pi@raspberrypi:~/StromPi3_Scriptfolder_12062019/Serialless $ sudo python3 RTCSerialless.py
Setting GPIO to LOW to Disable Serialless Mode.
This will take approx. 10 seconds.
Serialless Mode is Disabled!
Setting Time!
Enabled Serialless
```

### 7.3 PowerOn-Button Mode

Die PowerOn-Button Mode ermöglicht es, dass der StromPi nach dem Herunterfahren mit dem *poweroff* -Befehl oder dem PowerOff-Alarm ohne trennen der Spannungsversorgungen oder abnehmen des BatteryHATs, wieder neugestartet werden kann.

Für diese Funktion ist, bei der StromPi Rev 1.0, nur eine kleine Modifikation notwendig. Es muss eine Kabelbrücke zwischen zwei Löt pads eingelötet werden. Dann ist es möglich den StromPi 3 durch das Überbrücken des Reset-Pins neu zu starten.

Falls der StromPi 3 schon für den Serialless Modus modifiziert ist oder im Serialless Modus mit der PowerOn-Button Modifikation eingesetzt werden soll, muss eine weitere kleine Modifikation vorgenommen werden.

Im Kapitel 10 Modifikationen, wird genau erklärt was dafür notwendig ist.

Bei der StromPi Rev. 1.1 muss keine Modifikation vorgenommen werden. Sie können die Funktion über die Konfiguration aktivieren und mit den PowerOn Pins den StromPi neustarten.

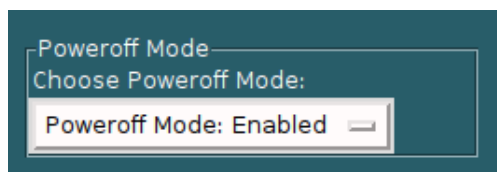


**WICHTIG: Die eingestellte Zeit im Konfigurationsskript ist die Initialisierung des PowerOn-Buttons. Wir empfehlen eine Zeit von mindestens 30 Sekunden.**

### 7.4 Poweroff Mode

Der Poweroff Modus sorgt dafür, dass nach einem Stromausfall Ihr Raspberry Pi solange ausgeschaltet bleibt bis Sie den PowerOn-Button drücken (siehe Kapitel 7.3).

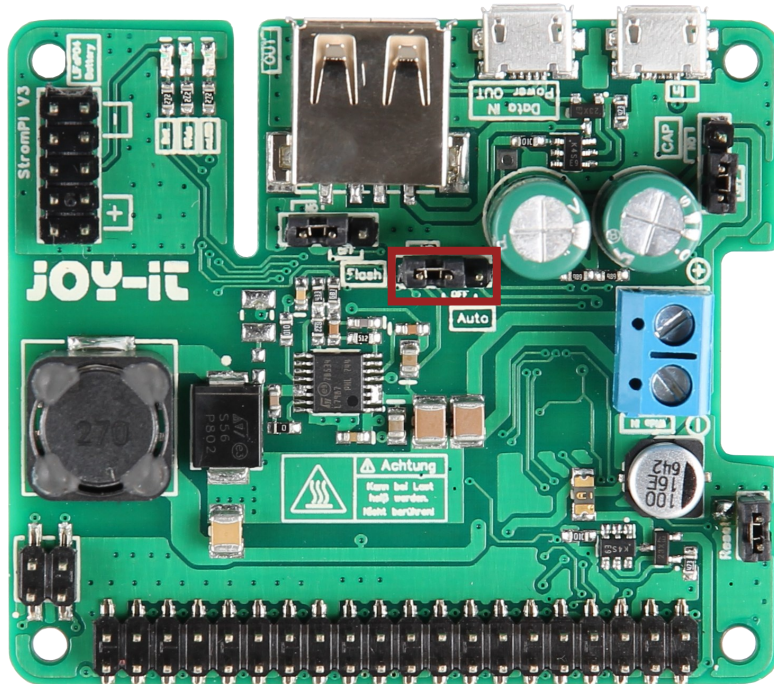
Sie verlieren durch diesen Modus die Option einen Powerback zu erkennen, der Raspberry fährt auf jeden Fall runter und der StromPi wird die Stromversorgung zum Raspberry nach der eingestellten Zeit unterbrechen. Um den Poweroff Modus einzuschalten, müssen Sie im Configskript **Choose Poweroff Mode** auf **Enable** stellen.



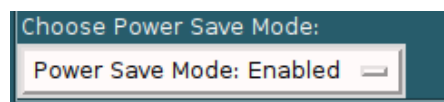
## 7.5 Power Save Mode

Der Powersave Modus ermöglicht es Ihnen den Stromverbrauch des StromPi 3 drastisch zu reduzieren, indem der Spannungswandler bei Nichtnutzung des Wide Anschlusses abgeschaltet wird und nur im Falle eines Spannungsausfalls der primären Stromversorgung eingeschaltet wird. Dies ist nur im Betrieb mit niedriger Stromstärke möglich (unter 2,5 A).

Sollte die Stromstärke zu hoch sein, könnte es zu Komplikationen beim Umschalten der Powerpaths kommen.



Zusätzlich müssen Sie im Configskript **Choose Power Save Mode** auf **Enable** stellen.



## 8. KONFIGURATION DES STROMPI 3

Wir zeigen Ihnen hier wie Sie die serielle Kommunikation für den StromPi einrichten können.

Wir empfehlen Ihnen die neuste Version des Raspberry Pi OS Images zu verwenden.

## 8.1 Aktivierung der seriellen Kommunikation

Um die Funktionen des StromPis nutzen und Einstellungen vornehmen zu können, müssen ein paar Einstellungen an Ihrem Raspberry Pi vorgenommen werden.

Zunächst müssen Sie die Python3-Serial Bibliothek installieren, geben Sie dazu folgenden Befehl in die Konsole des Raspberry Pi ein:

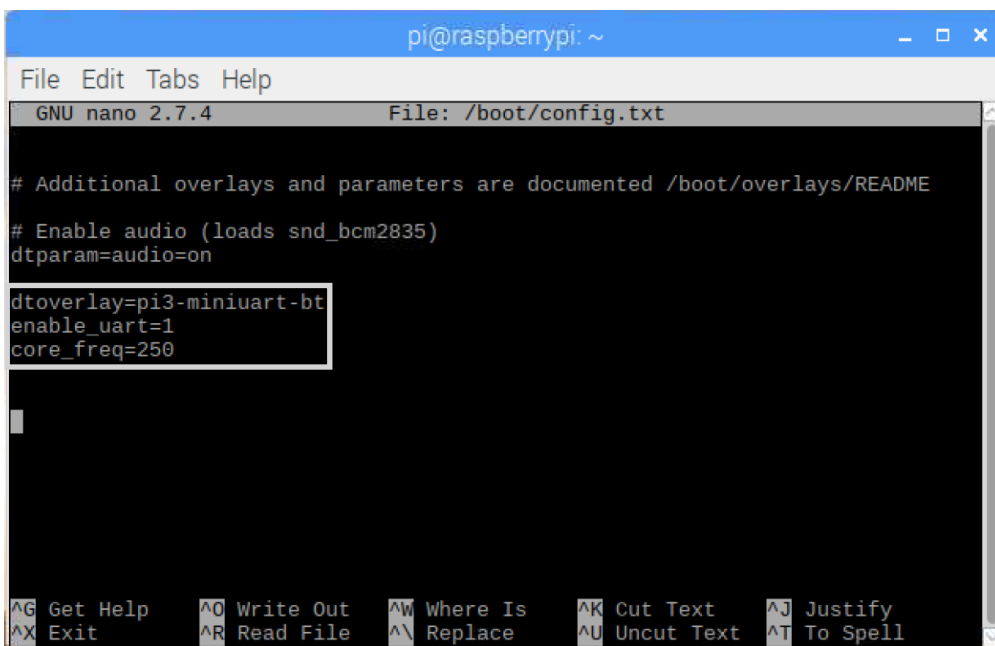
```
sudo apt-get install python3-serial
```

Öffnen Sie danach auf Ihrem Raspberry Pi die Konsole und geben Sie die folgenden Befehle ein:

```
sudo nano /boot/firmware/config.txt
```

Fügen Sie die folgenden Zeilen an das Ende der Datei an:

```
dtoverlay=miniuart-bt  
enable_uart=1
```



Für den Raspberry Pi Zero W benötigen Sie nur die Zeile: `enable_uart=1`.

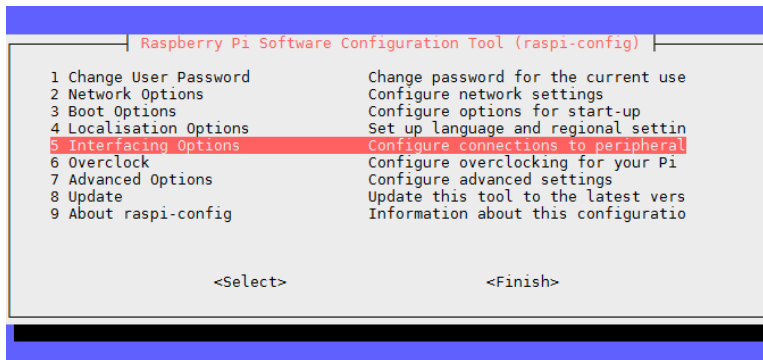
Speichern Sie die Datei mit der Tastenkombination **Strg+O** und verlassen Sie die Datei mit der Kombination **Strg+X**.

Zusätzlich dazu ist es notwendig, die serielle Schnittstelle des Raspberry Pis zu aktivieren und die serielle Konsole zu deaktivieren. Dies können Sie entweder über die Raspberry Pi *Configuration* oder über die *Raspi-Config* vornehmen.

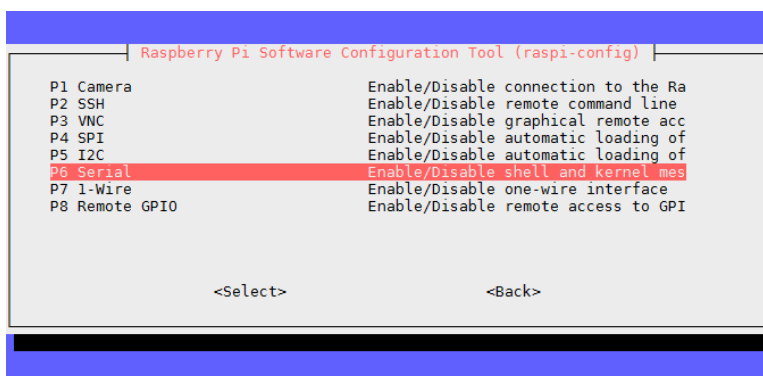
Um dies über die *Raspi-Config* vorzunehmen, geben Sie den folgenden Befehl in ein Terminalfenster an Ihrem Raspberry Pi ein:

```
sudo raspi-config
```

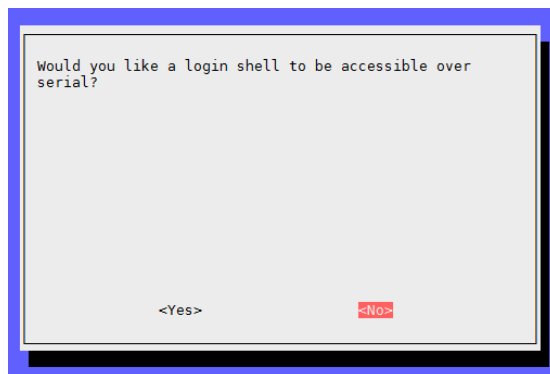
Navigieren Sie hier, mit den Pfeiltasten Ihrer Tastatur, in den Menüpunkt **5 Interfacing Options**.



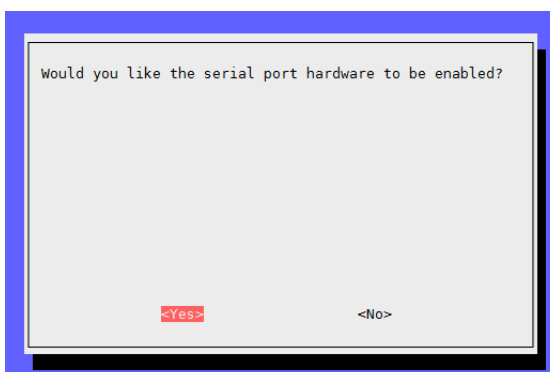
Wählen Sie den Punkt **P6 Serial** aus.



Die Frage ***Would you like a login shell to be accessible over serial?*** beantworten Sie mit **No**.



Die Frage ***Would you like the serial port hardware to be enabled?*** beantworten Sie mit **Yes**.



Nach einer erneuten Bestätigung können Sie das Menü mit **Finish** verlassen.

Die Frage nach einem Neustart des Gerätes beantworten Sie mit **Yes**. Die serielle Schnittstelle ist nun aktiviert und die serielle Konsole deaktiviert.



**Achtung!** Das StromPi Configskript funktioniert nicht mit der CLI Version der StromPi Firmware, ausschließlich mit der Main Version der Firmware. Außerdem muss die serielle Kommunikation aktiviert sein [siehe Kapitel 8.1 Aktivierung der seriellen Kommunikation].

Um das Konfigurationsskript auf Ihrem Raspberry Pi auszuführen, müssen Sie noch eine Bibliothek installieren, dies können Sie über folgenden Befehl erreichen.

```
sudo apt-get install python3-pil python3-pil.imagetk
```

Nun geben wir Ihnen 3 Möglichkeiten mit denen Sie die grafische Oberfläche aufrufen können.

### 8.3 Nutze VNC für Remotedesktopverbindung

Für die Nutzung von VNC müssen Sie mit dem Befehl

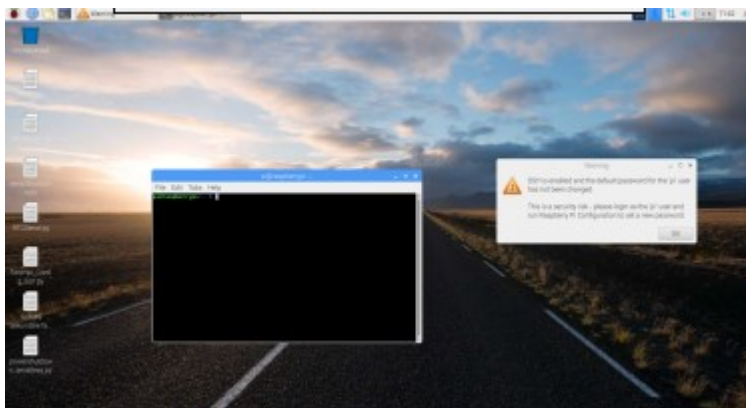
```
sudo raspi-config
```

unter **Interfacing options** → **VNC** einschalten. Anschließend versuchen Sie mit folgendem Befehl Ihre IP-Adresse herauszufinden.

```
ifconfig
```

Laden Sie nun auf dem Gerät von dem Sie den Raspberry Pi steuern wollen *RealVNC* herunter und verbinden Sie sich durch das Eingeben der IP-Adresse.

Nun sollte folgendes Fenster erscheinen:



Hier können Sie dann über die Konsole wie gewohnt mit Ihrem Linux-System weiterarbeiten.

## 8.4 SSH-Verbindung über PuTTY und Xming

Führen Sie folgenden Befehl aus:

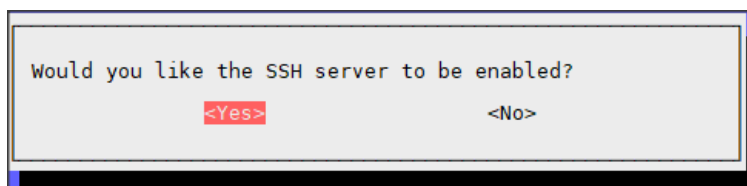
```
sudo raspi-config
```

Gehen Sie auf **Interfacing Options**

```
1 Change User Password Change password for the current user
2 Network Options       Configure network settings
3 Boot Options          Configure options for start-up
4 Localisation Options  Set up language and regional settings to match your location
5 Interfacing Options   Configure connections to peripherals
6 Overclock             Configure overclocking for your Pi
7 Advanced Options      Configure advanced settings
8 Update                Update this tool to the latest version
9 About raspi-config    Information about this configuration tool
```

und aktivieren Sie anschließend **SSH**.

```
P1 Camera Enable/Disable connection to the Raspberry Pi Camera
P2 SSH     Enable/Disable remote command line access to your Pi using SSH
P3 VNC     Enable/Disable graphical remote access to your Pi using RealVNC
P4 SPI     Enable/Disable automatic loading of SPI kernel module
P5 I2C     Enable/Disable automatic loading of I2C kernel module
P6 Serial  Enable/Disable shell and kernel messages on the serial connection
P7 1-Wire  Enable/Disable one-wire interface
P8 Remote GPIO Enable/Disable remote access to GPIO pins
```

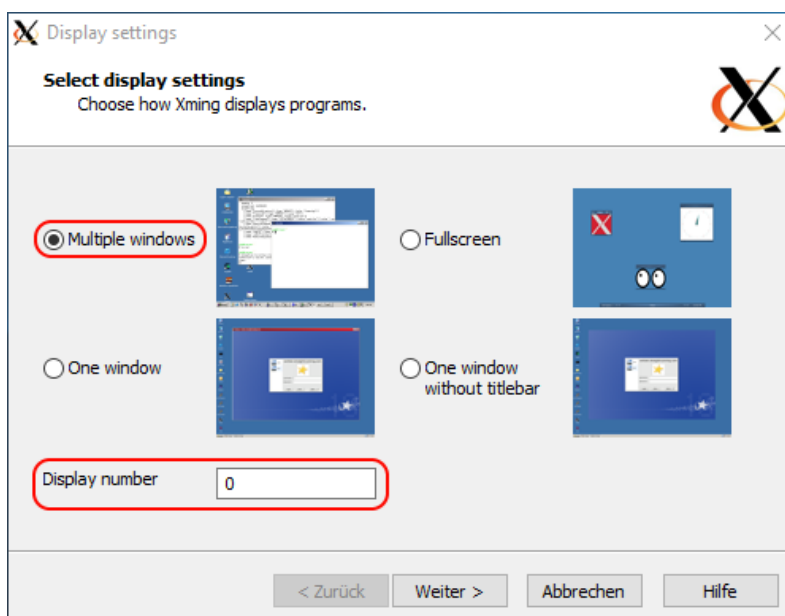


Nun müssen Sie [PuTTY](#) und [Xming](#) herunterladen.

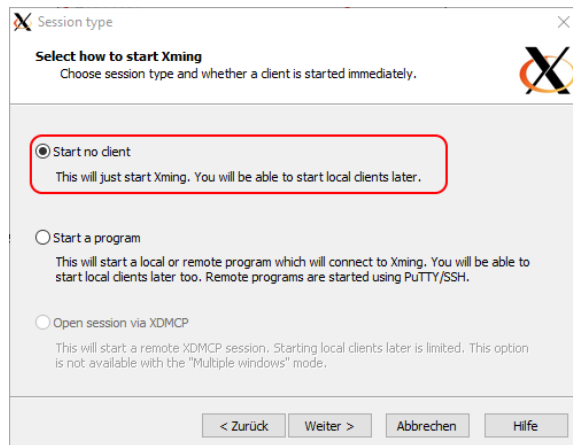
### Xming

Starten Sie nun die *XLaunch.exe* Datei im Installationsverzeichnis von Xming.

Wählen Sie **Multiple Windows** aus und **Display number** als **0**.



Stellen Sie sicher, dass **Start no client** ausgewählt ist:



The dialog box is titled "Session type" and contains the instruction "Select how to start Xming". It offers three radio button options: "Start no client" (selected and highlighted with a red rectangle), "Start a program", and "Open session via XDMCP". Each option has a brief description of its function. At the bottom, there are four buttons: "< Zurück", "Weiter >", "Abbrechen", and "Hilfe".

Session type

Select how to start Xming  
Choose session type and whether a client is started immediately.

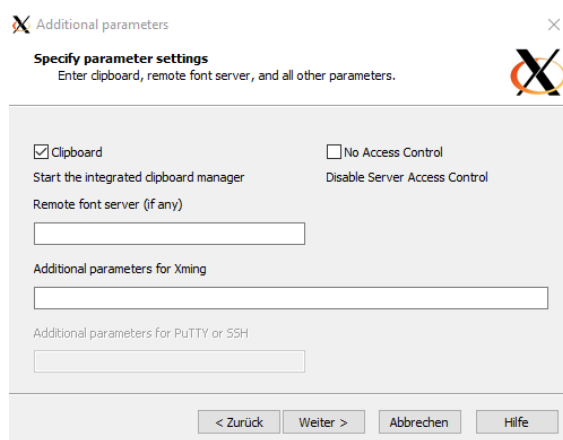
☒ Start no client  
This will just start Xming. You will be able to start local clients later.

☐ Start a program  
This will start a local or remote program which will connect to Xming. You will be able to start local clients later too. Remote programs are started using PuTTY/SSH.

☐ Open session via XDMCP  
This will start a remote XDMCP session. Starting local clients later is limited. This option is not available with the "Multiple windows" mode.

< Zurück Weiter > Abbrechen Hilfe

Drücken Sie hier auf **Weiter** ohne Änderungen vorzunehmen:



The dialog box is titled "Additional parameters" and contains the instruction "Specify parameter settings". It includes checkboxes for "Clipboard" (checked) and "No Access Control". Below these are text input fields for "Remote font server (if any)", "Additional parameters for Xming", and "Additional parameters for PuTTY or SSH". At the bottom, there are four buttons: "< Zurück", "Weiter >", "Abbrechen", and "Hilfe".

Additional parameters

Specify parameter settings  
Enter clipboard, remote font server, and all other parameters.

☒ Clipboard  
Start the integrated clipboard manager

☐ No Access Control  
Disable Server Access Control

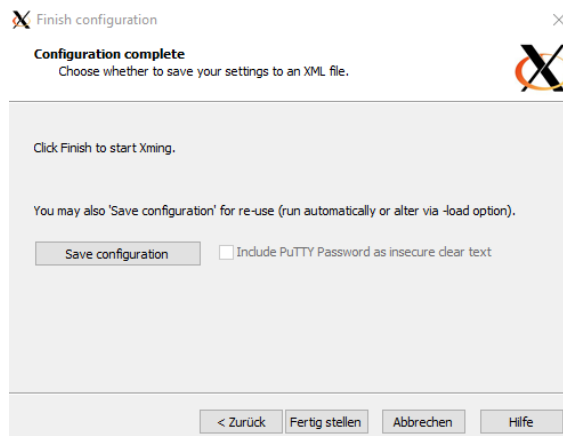
Remote font server (if any)

Additional parameters for Xming

Additional parameters for PuTTY or SSH

< Zurück Weiter > Abbrechen Hilfe

Drücken Sie nun auf **Fertig stellen**:



The dialog box is titled "Finish configuration" and contains the instruction "Configuration complete". It includes a "Save configuration" button and a checkbox for "Include PuTTY Password as insecure clear text". At the bottom, there are four buttons: "< Zurück", "Fertig stellen", "Abbrechen", and "Hilfe".

Finish configuration

Configuration complete  
Choose whether to save your settings to an XML file.

Click Finish to start Xming.

You may also 'Save configuration' for re-use (run automatically or alter via -load option).

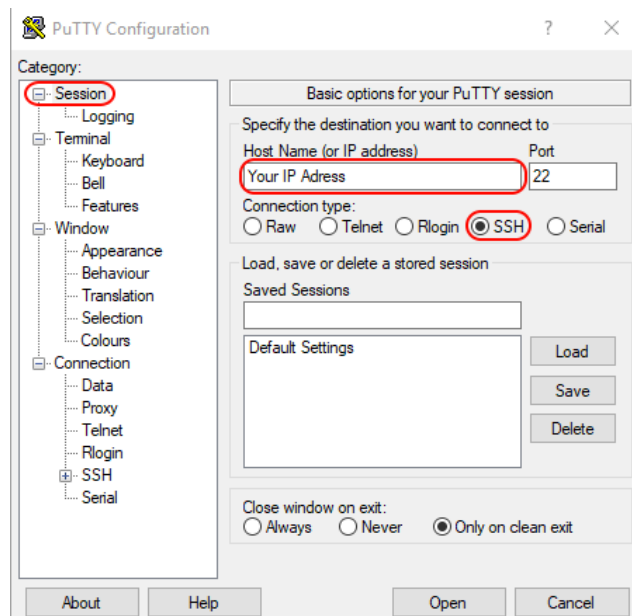
Save configuration ☐ Include PuTTY Password as insecure clear text

< Zurück Fertig stellen Abbrechen Hilfe

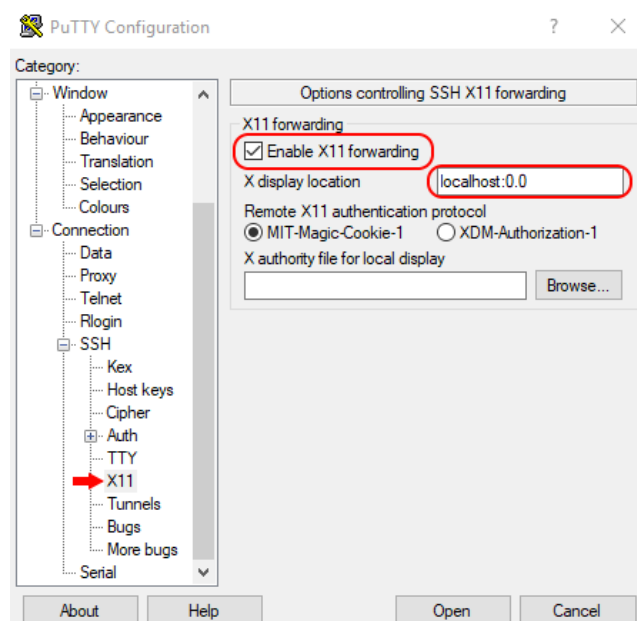
## PuTTY

Nun müssen Sie noch PuTTY einstellen.

Stellen Sie sicher, dass Sie unter **Session** → **SSH** ausgewählt haben und



Versichern Sie sich nun das unter **X11** → **Enable X11 forwarding** eingeschaltet ist und geben Sie unter **Xdisplay location** → **localhost:0.0** ein.



Drücken Sie nun **Open** und Sie sollten folgendes sehen:

```
Using username "pi".
pi@192.168.1.114's password:
```

Geben Sie nun Ihre Login Daten ein.

Sie haben die SSH-Verbindung erfolgreich hergestellt.

Geben Sie nun folgenden Befehl ein:

```
sudo -E python3 strompi_config_gui.py
```

Nun öffnet sich die grafische Oberfläche, welche wie folgt aussieht:

StromPi V3 Serial Config Tool

JOY-IT®

**Options Without Timesettings:**  
Choose StromPi V3 Powerpath Mode: mUSB -> Wide  
Choose Powerfail Warning Mode: Enabled  
Choose Battery Shutdown Mode: Disabled  
Choose Serialless Mode: Disabled  
Choose Power Save Mode: Power Save Mode: Disabled

**Real Time Clock Settings**  
Settings for the Clock:  
☒ No Change  
☐ Manual Setting  
☐ Sync with RaspberryPi  
Set the Hours: 00 Hours  
Set the Minutes: 32 Minutes  
Set the Days: 01 Day  
Set the Month: 05  
Set the Year: 18 Year  
Set The Weekday: Tuesday

**Shutdown Timer**  
Choose Shutdown Mode: Disabled  
Choose Shutdown timer: 10 Seconds

**PowerOnButton**  
Choose PowerOn-Button Mode: Disabled  
Choose Button Shutdown time: 30 Seconds

**Alarm Modes**  
Choose Alarm Mode: Alarm-Mode: Time-Alarm  
Choose WakeUp-Alarm Mode: WakeUp-Alarm: Disabled  
Choose Power-Off Alarm Mode: PowerOff-Alarm: Disabled  
Choose Alarm Weekday: Monday  
Setting WakeUp-Alarm minute: 00 Minutes  
Choose Power-Off Minutes: 00 Minutes  
Choose Alarm Day: 01 Day  
Setting WakeUp-Alarm Hour: 00 Hours  
Choose Power-Off Hours: 00 Hours  
Alarm Month: 01

**Interval Timer**  
Choose Interval-Alarm Mode: Interval-Alarm: Disabled  
Interval On-Time in minutes: 00 minutes  
Interval Off-Time in minutes: 00 minutes

Please read the Readme file

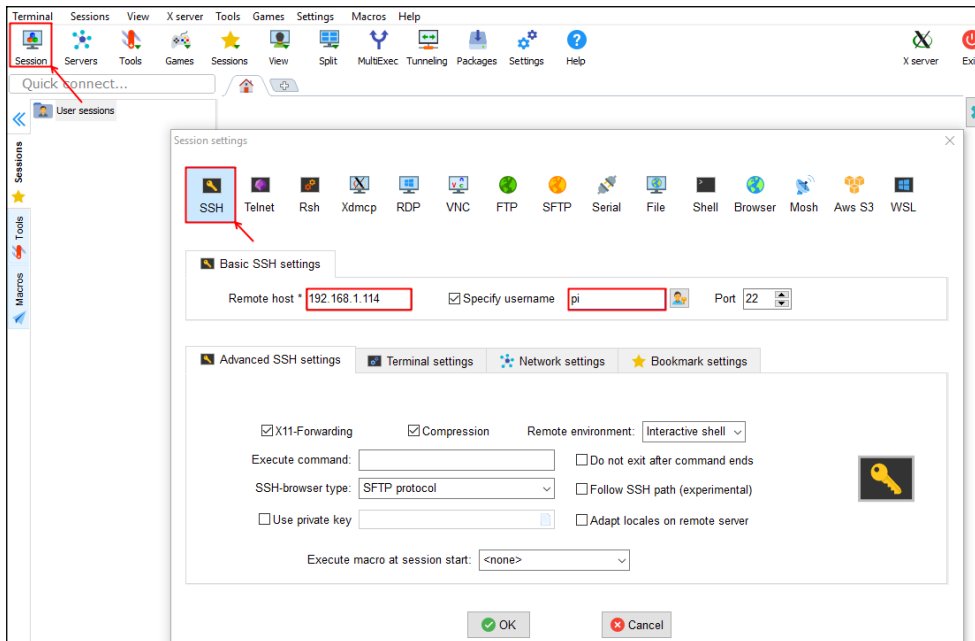
UPDATE!

Die angezeigten Werte werden beim Start des Skriptes, mit den von Ihnen im StromPi eingestellten Werten synchronisiert.

Sie können nun beliebig Werte ändern und anschließend auf **Update** klicken, um die Einstellungen Ihres StromPi zu ändern.

## 8.5 SSH-Verbindung über MobaXTerm

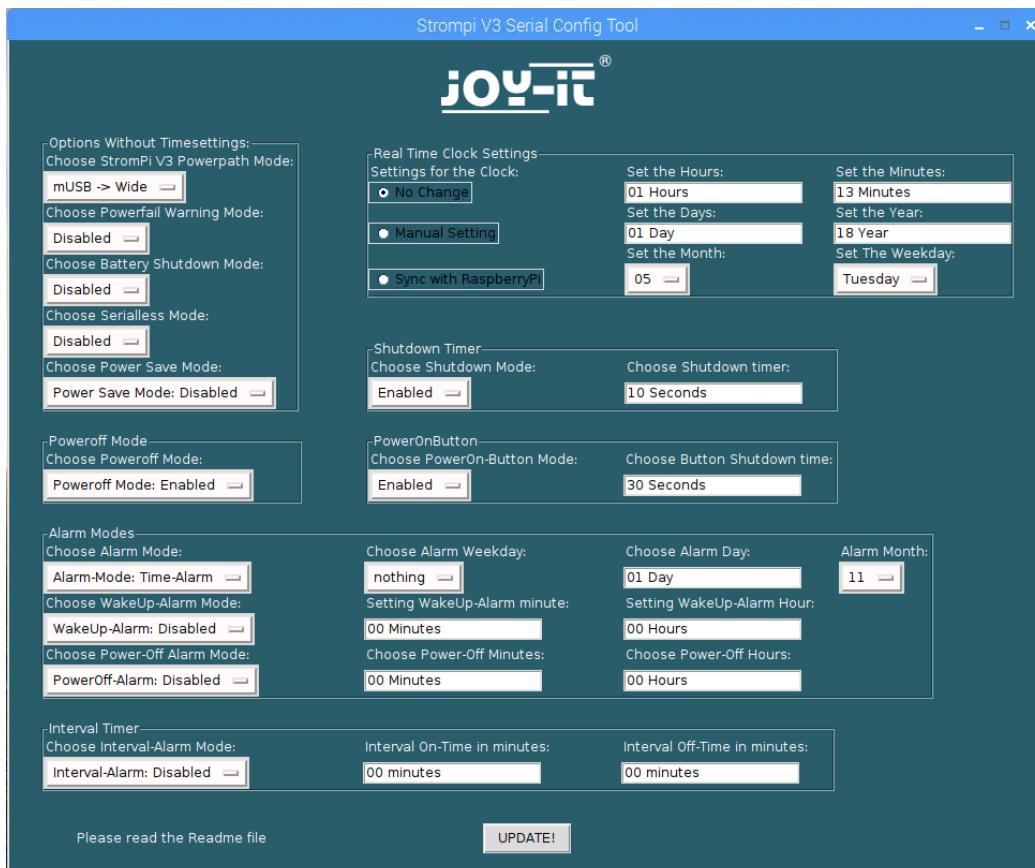
Stellen Sie sicher, dass SSH aktiviert ist, falls dies noch nicht geschehen ist, schauen Sie im letzten Kapitel nach wie Sie SSH aktivieren können. Laden Sie sich als nächstes MobaXTerm herunter.



Drücken Sie nun auf Session und wählen Sie dann SSH aus.  
Geben Sie nun bei **Remote host** die IP-Adresse Ihres Raspberry Pis ein und bei **Specify username** **pi**. Drücken Sie dann auf **OK** und geben Sie dann das Passwort Ihres Raspberry Pis ein sobald Sie dazu aufgefordert werden.

```
sudo -E python3 strompi_config_gui.py
```

die grafische Oberfläche aufrufen. Diese sollte wie folgt aussehen:



Die angezeigten Werte werden beim Start des Skriptes mit den von Ihnen im StromPi eingestellten Werten synchronisiert.  
Sie können nun beliebig Werte ändern und anschließend auf **Update** klicken, um die Einstellungen Ihres StromPi zu ändern.



Es ist möglich, dass Ihr Skript anders aussieht. Der Poweroff Mode und der Power Save Mode wurden erst für eine spätere Firmware Version hinzugefügt.

## 9. STROMPI 3 SKRIPTE

Alle von uns veröffentlichten StromPi Skripte können Sie [hier](#) herunterladen.

Die Serial Skripte lesen und senden die benötigten Informationen über die serielle Schnittstelle, die Serialless Skripte benutzen stattdessen einen GPIO-Pin.

Neben dem ConfigSkript, welches Sie bereits kennen gelernt haben, gibt es noch die folgenden Skripte:

### 9.1 Serial Skripte

#### **serialShutdown.py**

Dieses Skript fährt den Raspberry Pi herunter, wenn die primäre Spannungsversorgung ausfällt. Das Skript hat einen eigenen Shutdown Timer. Er gibt die Zeit an die gewartet werden soll, ob die primäre Spannungsversorgung zurück ist, bevor der Raspberry Pi heruntergefahren wird. Diese ist voreingestellt auf 10 Sekunden. Sie können die Zeit in Zeile 14 des Skripts anpassen.

```
13 #####  
14 wait_for_shutdowntimer = 10;  
15 #####
```

Der Shutdown Timer im Skript muss kleiner sein als der Shutdown Timer den Sie mit dem Configskript festlegen, da sonst die Stromzufuhr des Raspberry Pis unterbrochen wird bevor er heruntergefahren wurde. Sie müssen dieses Skript im Autostart Ihres Raspberry Pi haben damit es funktioniert.

Eine Anleitung dazu finden Sie im Kapitel 9.4 Automatisches Ausführen von Skripten.

#### **shutdown.py**

Dieses Skript fährt beim Ausführen Ihren StromPi und Raspberry Pi über den Poweroff Befehl herunter.

Der Poweroff Befehl ermöglicht es den StromPi über den WakeUp Alarm zu Starten.

#### **RTCSerial.py**

Dieses Skript vergleicht beim Start die Zeit des Raspberry Pis mit der Zeit der RTC des StromPis und die ältere wird mit der aktuelleren Zeit überschrieben.

Wenn Sie dieses Skript in den Autostart Ihres Raspberry Pis packen, wird die Zeit automatisch bei jedem Start des Raspberry Pi verglichen.

Eine Anleitung dazu finden Sie im Kapitel 9.4 Automatisches Ausführen von Skripten.

## Poweralarm\_serial.py

Dieses Skript sendet, wenn die primäre Spannungsversorgung ausfällt, eine Email an die im Skript angegebene Email-Adresse. Eine weitere Email wird gesendet, wenn die primäre Spannungsversorgung wieder vorhanden ist oder wenn der Raspberry Pi neugestartet wurde.

```
1  #!/usr/bin/env python
2  # -*- coding: utf-8 -*-
3  import RPi.GPIO as GPIO
4  import time
5  import serial
6  import os
7  import smtplib
8  from email.mime.text import MIMEText
9  #####
10 # This is The config for the EMAIL notification
11 #####
12 SERVER = 'SMTP.Server.Com'
13 PORT = 587
14 EMAIL = 'Example@Example.com'
15 PASSWORT = 'Password'
16 EMPFAENGER = ['Empfänger@Example.com' ,]
17 SUBJECT_Powerfail = 'Raspberry Pi Powerfail!' #Powerfail Email Betreff
18 SUBJECT_Powerback = 'Raspberry Pi Powerback!' #Powerback Email Betreff
19 SUBJECT_Restart = 'Raspberry Pi Restart!' #Restart Email Betreff
20 #####
21 # Here you can choose whether you want to receive an email when the Raspberry Pi restarts - 1 to activate - 0 to deactivate
22 Restart_Mail = 1
23 #####
```

In Zeile 22 können Sie die Benachrichtigungsmail beim Neustart des Systems ein (1) oder aus (0) schalten.

*SERVER* ist Ihr SMTP E-Mail Server (z.B. smtp.ihr-e-mail-server.de).

*PORT* ist Ihr Serverport

*EMAIL* ist Ihre E-Mail Adresse

*PASSWORT* ist Ihr E-Mail Passwort

*EMPFAENGER* sind die E-Mail Adressen welche die Emails bekommen

*SUBJECT\_Powerfail* wird in den Betreff für die Spannungsverlust E-Mail eingetragen

*SUBJECT\_Poweback* wird in den Betreff der Powerback E-Mail eingetragen

Sie müssen dieses Skript im Autostart Ihres Raspberry Pi haben damit es funktioniert.

Eine Anleitung dazu finden Sie im Kapitel 9.4 Automatisches Ausführen von Skripten.

In Kapitel 9.3 Raspberry Pi Pin-Layout können Sie sich die Pinbelegung des Raspberry Pi anschauen.

## 9.2 Serialless Skripte

### poweralarm\_serialless.py

Dieses Skript vergleicht die Uhrzeit des Raspberry Pis mit der des StromPis und überschreibt die ältere mit der neueren Zeit. Zusätzlich sendet es wenn die primäre Spannungsversorgung ausfällt eine Email an die im Skript angegebene Email-Adresse. Eine weitere Email wird gesendet wenn die primäre Spannungsversorgung wieder vorhanden ist oder wenn der Raspberry Pi neugestartet wurde.

```

10 # Here you can choose the connected GPIO-Pin
11 GPIO_TPIN = 21
12
13 # Here you can choose whether you want to receive an email when the Raspberry Pi restarts
14 Restart_Mail = 1
15
16 # This is The config for the EMAIL notification
17 #-----
18 SERVER = 'SMTP.Server.Com'
19 PORT = 587
20 EMAIL = 'Example@Example.com'
21 PASSWORD = 'Password'
22 EMPFAENGER = ['Empfänger@Example.com' ,]
23 SUBJECT_Powerfail = 'Raspberry Pi Powerfail!' #Powerfail Email Betreff
24 SUBJECT_Powerback = 'Raspberry Pi Powerback!' #Powerback Email Betreff
25 SUBJECT_Restart = 'Raspberry Pi Restart!' #Restart Email Betreff
26 #-----

```

In Zeile 11 des Skripts können Sie den GPIO-Pin den Sie zur alternativen Kommunikation für den Serialless Modus benutzen ändern. In Zeile 14 können Sie die Benachrichtigungsmail beim Neustart des Systems ein (1) oder aus (0) schalten.

*SERVER* ist Ihr SMTP E-Mail Server (z.B. smtp.ihr-e-mail-server.de)

*PORT* ist Ihr Serverport

*EMAIL* ist Ihre E-Mail Adresse.

*PASSWORD* ist Ihr E-Mail Passwort.

*EMPFAENGER* sind die E-Mail Adressen welche die Emails bekommen.

*SUBJECT\_Powerfail* wird in den Betreff für die Spannungsverlust E-Mail eingetragen

*SUBJECT\_Poweback* wird in den Betreff der Powerback E-Mail eingetragen

Sie müssen dieses Skript im Autostart Ihres Raspberry Pi haben damit es funktioniert. Eine Anleitung dazu finden Sie im Kapitel 9.4 Automatisches Ausführen von Skripten.

In Kapitel 9.3 Raspberry Pi Pin-Layout können Sie sich die Pinbelegung des Raspberry Pi anschauen.

### **powershutdown\_serialless.py**

Dieses Skript fährt den Raspberry Pi herunter, wenn die Primäre Spannungsversorgung ausfällt. Das Skript hat einen eigenen Shutdown Timer, er gibt die Zeit an die gewartet werden soll, ob die primäre Spannungsversorgung zurück ist, bevor der Raspberry Pi heruntergefahren wird. Diese ist voreingestellt auf 10 Sekunden. Sie können die Zeit in Zeile 11 des Skripts anpassen, genau wie den GPIO-Pin den Sie zur alternativen Kommunikation für den Serialless Modus benutzen (Zeile 10).

```

9 # Here you can choose the connected GPIO-Pin and the ShutdownTimer
10 GPIO_TPIN = 21
11 ShutdownTimer = 10

```

Der Shutdown Timer im Skript muss kleiner sein als der Shutdown Timer den Sie mit dem Configskript festlegen, da sonst die Stromzufuhr das Raspberry Pi unterbrochen wird bevor er heruntergefahren wurde.

In Kapitel 9.3 Raspberry Pi Pin-Layout können Sie sich die Pinbelegung des Raspberry Pi anschauen.

Sie müssen dieses Skript im Autostart Ihres Raspberry Pi haben damit es funktioniert. Eine Anleitung dazu finden Sie im Kapitel 9.4 Automatisches Ausführen von Skripten.

### RTC\_serialShutdown\_serialless.py

Dieses Skript hat alle Funktionen des *powershutdown\_serialless.py* - Skripts und vergleicht zusätzlich noch beim Start die Uhrzeit des Raspberry Pi und die des StromPis. Die ältere Zeit wird mit der aktuelleren überschrieben.

Den Shutdown Timer und den GPIO-Pin, den Sie zur alternativen Kommunikation für den Serialless Modus benutzen, können in Zeile 11 und 12 geändert werden.

```
10 # Here you can choose the connected GPIO-Pin and the ShutdownTimer
11 GPIO_TPIN = 21
12 ShutdownTimer = 10
```

Der Shutdown Timer im Skript muss kleiner sein als der Shutdown Timer den Sie mit dem Configskript festlegen, da sonst die Stromzufuhr des Raspberry Pi unterbrochen wird bevor er heruntergefahren wurde.

In Kapitel 9.3 Raspberry Pi Pin-Layout können Sie sich die Pinbelegung des Raspberry Pi anschauen.

Sie müssen dieses Skript im Autostart Ihres Raspberry Pi haben damit es funktioniert. Eine Anleitung dazu finden Sie im Kapitel 9.4 Automatisches Ausführen von Skripten.

### RTC\_Email\_serialless.py

Dieses Skript vergleicht die Uhrzeit des Raspberry Pis mit der des StromPis und überschreibt die ältere mit der neueren Zeit. Zusätzlich sendet es, wenn die primäre Spannungsversorgung ausfällt eine Email, an die im Skript angegebene Email-Adresse. Eine weitere Email wird gesendet wenn die primäre Spannungsversorgung wieder vorhanden ist, oder wenn der Raspberry Pi neugestartet wurde.

```
13 # Here you can choose the connected GPIO-Pin
14 GPIO_TPIN = 21
15
16 #Here you can choose whether you want to receive an email when the Raspberry Pi restarts
17 Restart_Mail = 1
18
19 # This is The config for the EMAIL notification
20 #-----
21 SERVER = 'SMTP.Beispiel.DE'
22 PORT = 587
23 EMAIL = 'Beispiel@abc.de'
24 PASSWORT = 'Passwort'
25 EMPFAENGER = ['Empfänger1@abc.de', 'Empfänger2@abc.com']
26 SUBJECT_Powerfail = 'Raspberry Pi Powerfail!'
27 SUBJECT_Powerback = 'Raspberry Pi Powerback!'
28 SUBJECT_Restart = 'Raspberry Pi Restart!'
29 #-----
```

In Zeile 14 des Skripts können Sie den GPIO-Pin den Sie zur alternativen Kommunikation für den Serialless Modus benutzen ändern. In Zeile 17 können Sie die Benachrichtigungsmail beim Neustart des Systems ein (1) oder aus (0) schalten.

*SERVER* ist Ihr SMTP E-Mail Server (z.B. smtp.ihr-e-mail-server.de)

*PORT* ist Ihr Serverport

*EMAIL* ist Ihre E-Mail Adresse

*PASSWORT* ist Ihr E-Mail Passwort

*EMPFAENGER* sind die E-Mail Adressen welche die Emails bekommen

*SUBJECT\_Powerfail* wird in den Betreff für die Spannungsverlust E-Mail eingetragen

*SUBJECT\_Poweback* wird in den Betreff der Powerback E-Mail eingetragen

In Kapitel 9.3 Raspberry Pi Pin-Layout können Sie sich die Pinbelegung des Raspberry Pi anschauen.

Sie müssen dieses Skript im Autostart Ihres Raspberry Pi haben damit es funktioniert. Eine Anleitung dazu finden Sie im Kapitel 9.4 Automatisches Ausführen von Skripten.

### **RTCSerialless.py**

Dieses Skript wurde extra für den Serialless Modus geschrieben. Beim Start des Skripts wird der Serialless Modus beendet, damit die Uhrzeit des Raspberry Pi und die des StromPi verglichen werden können und die ältere wird mit der aktuelleren Zeit überschrieben. Danach wird der Serialless Modus wieder aktiviert. Weil die serielle Kommunikation nun deaktiviert ist können Sie keine weitere Änderungen an der Konfiguration des StromPi vornehmen. In Zeile 9 des Skripts können Sie den GPIO-Pin, den Sie zur alternativen Kommunikation für den Serialless Modus benutzen ändern.

```
9 GPIO_TPIN = 21
```

Um den Serialless Modus zu beenden muss der Kommunikationspin für min. 5 Sekunden auf *Low* gesetzt werden. In Kapitel 9.3 Raspberry Pi Pin-Layout, können Sie sich die Pinbelegung des Raspberry Pi anschauen.

Wenn Sie dieses Skript in den Autostart Ihres Raspberry Pis packen, wird die Zeit automatisch bei jedem Start des Raspberry Pi verglichen. Und der Serialless Modus anschließend aktiviert.

Eine Anleitung dazu finden Sie im Kapitel 9.4 Automatisches Ausführen von Skripten.

### **Start\_Serialless.py**

Dieses Skript startet den Serialless Modus, um sämtliche serielle Kommunikation zwischen dem Raspberry Pi und dem StromPi zu unterdrücken.

In Zeile 9 des Skripts können Sie den GPIO-Pin den Sie zur alternativen Kommunikation für den Serialless Modus benutzen ändern.

In Kapitel 9.3 Raspberry Pi Pin-Layout, können Sie sich die Pinbelegung des Raspberry Pi anschauen.

### **Stop\_Serialless.py**

Dieses Skript stoppt den Serialless Modus, um serielle Kommunikation zwischen dem Raspberry Pi und dem StromPi wieder zu ermöglichen. In Zeile 9 des Skripts können Sie den GPIO-Pin den Sie zur alternativen Kommunikation für den Serialless Modus benutzen ändern.

```
9 GPIO_TPIN = 21
```

In Kapitel 9.3 Raspberry Pi Pin-Layout, können Sie sich die Pinbelegung des Raspberry Pi anschauen.

## 9.3 Raspberry Pi Pin-Layout

Hier können Sie die Pin-Belegung des Raspberry Pis sehen:

Der StromPi verwendet nur GPIO14 und GPIO15 für die serielle Kommunikation, und GPIO21 im Serialless Modus. Alle anderen Datenpins können frei verwendet werden.

|           |                     |                                                                                     |                                                                                     |           |                |
|-----------|---------------------|-------------------------------------------------------------------------------------|-------------------------------------------------------------------------------------|-----------|----------------|
| <b>1</b>  | 3.3 V DC            |    |    | <b>2</b>  | 5 V DC         |
| <b>3</b>  | GPIO 2 (SDA1, I2C)  |    |    | <b>4</b>  | 5 V DC         |
| <b>5</b>  | GPIO 3 (SCL1, I2C)  |    |    | <b>6</b>  | Ground         |
| <b>7</b>  | GPIO 4              |    |    | <b>8</b>  | GPIO 14 (TXD0) |
| <b>9</b>  | Ground              |    |    | <b>10</b> | GPIO 15 (RXD0) |
| <b>11</b> | GPIO 17             |    |    | <b>12</b> | GPIO 18        |
| <b>13</b> | GPIO 27             |   |   | <b>14</b> | Ground         |
| <b>15</b> | GPIO 22             |  |  | <b>16</b> | GPIO 23        |
| <b>17</b> | 3.3 V               |  |  | <b>18</b> | GPIO 24        |
| <b>19</b> | GPIO 10 (SPI, MOSI) |  |  | <b>20</b> | Ground         |
| <b>21</b> | GPIO 9 (SPI, MISO)  |  |  | <b>22</b> | GPIO 25        |
| <b>23</b> | GPIO 11 (SPI, CLK)  |  |  | <b>24</b> | GPIO 8 (SPI)   |
| <b>25</b> | Ground              |  |  | <b>26</b> | GPIO 7 (SPI)   |
| <b>27</b> | ID_SD (I2C, EEPROM) |  |  | <b>28</b> | ID_SC          |
| <b>29</b> | GPIO 5              |  |  | <b>30</b> | Ground         |
| <b>31</b> | GPIO 6              |  |  | <b>32</b> | GPIO 12        |
| <b>33</b> | GPIO 13             |  |  | <b>34</b> | Ground         |
| <b>35</b> | GPIO 19             |  |  | <b>36</b> | GPIO 16        |
| <b>37</b> | GPIO 26             |  |  | <b>38</b> | GPIO 20        |
| <b>39</b> | Ground              |  |  | <b>40</b> | GPIO 21        |

## 9.4 Automatisches Ausführen von Skripten

Skripte lassen sich wie folgt in den Autostart Ihres Raspberry Pis kopieren, um beim Start des Raspberry Pis ausgeführt zu werden:

```
sudo nano /home/pi/.bashrc
```

Hier kann das Skript dem Autostart hinzugefügt werden. Der folgende Befehl muss hierfür vor dem Befehl **exit 0** eingefügt werden.

```
echo Running at boot  
sudo python3 /home/pi/StromPi3_Scriptfolder_2024-11-22/  
Serial/serialShutdown.py &
```

Beispiel mit dem *serialShutdown.py* Skript:

```
GNU nano 7.2 .bashrc  
# See /usr/share/doc/bash-doc/examples in the bash-doc package.  
  
if [ -f ~/.bash_aliases ]; then  
    . ~/.bash_aliases  
fi  
  
# enable programmable completion features (you don't need to enable  
# this, if it's already enabled in /etc/bash.bashrc and /etc/profile  
# sources /etc/bash.bashrc).  
if ! shopt -oq posix; then  
    if [ -f /usr/share/bash-completion/bash_completion ]; then  
        . /usr/share/bash-completion/bash_completion  
    elif [ -f /etc/bash_completion ]; then  
        . /etc/bash_completion  
    fi  
fi  
echo Running at boot  
sudo python3 /home/pi/StromPi3_Scriptfolder_2024-11-22/Serial/serialShutdown.py &
```

Den Skriptnamen und den Pfad müssen Sie für Ihr verwendetes Skript entsprechend anpassen.

Speichern Sie die Datei mit der Tastenkombination **STRG + O**, bestätigen Sie mit **Enter** und verlassen Sie den Editor mit der Kombination **STRG + X**.

Die Änderungen werden nach einem Neustart wirksam.

## 10. MODIFIKATIONEN

In diesem Kapitel beschreiben wir alle Modifikationen, welche für manche Funktionen des **StromPi3 Rev 1.0** benötigt werden.

**Diese Modifikationen sind nicht notwendig, wenn Sie bereits die StromPi3 Rev 1.1 verwenden.**

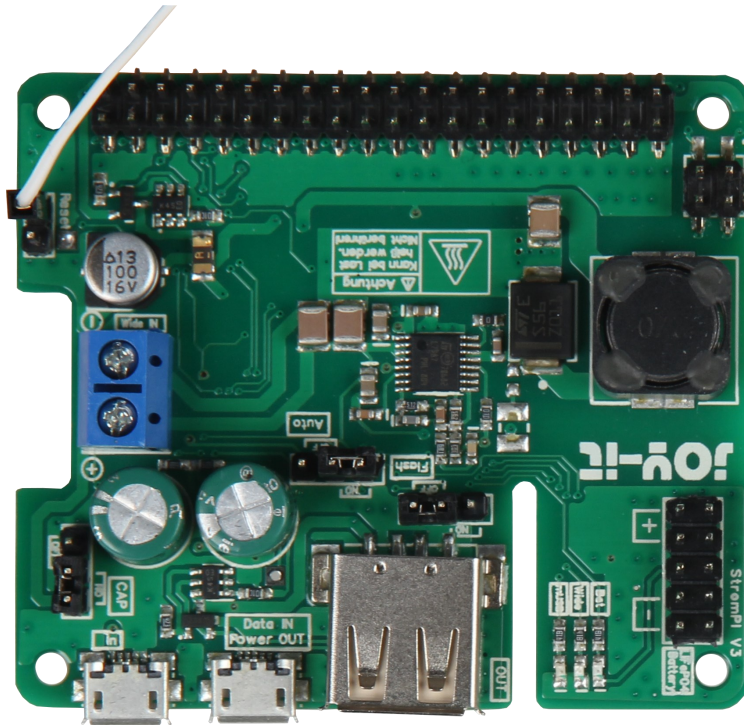
### Serialless Mode

Für die Modifizierung für den Serialless Mode gibt es 3 verschiedene Varianten, welche Ihnen im Folgenden erklärt werden.

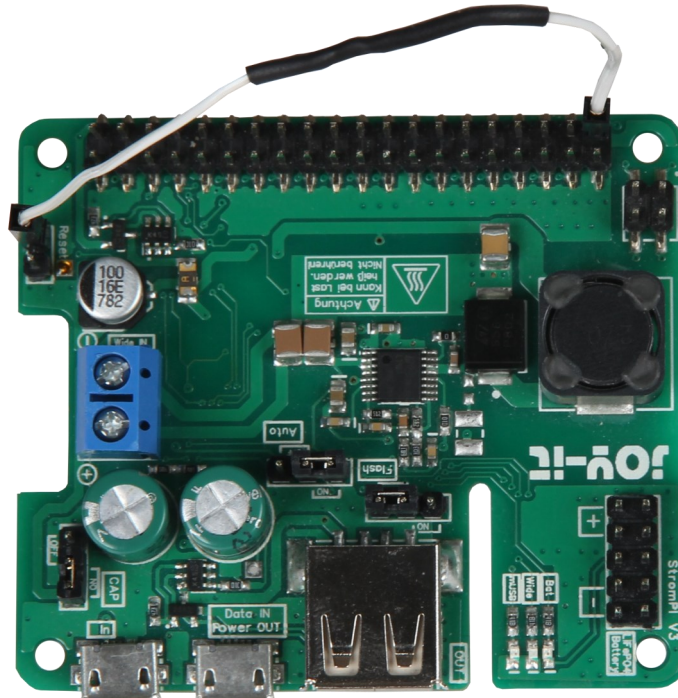
### **Variante 1**

Variante 1 ist die schnellste Lösung, kann jedoch nur verwendet werden, wenn Sie keinen Battery Hat benutzen.

Stecken Sie dazu einfach, wie im folgenden Bild zu sehen ist, ein Kabel auf den Reset-Pin. Dafür müssen Sie zunächst den Jumper entfernen.



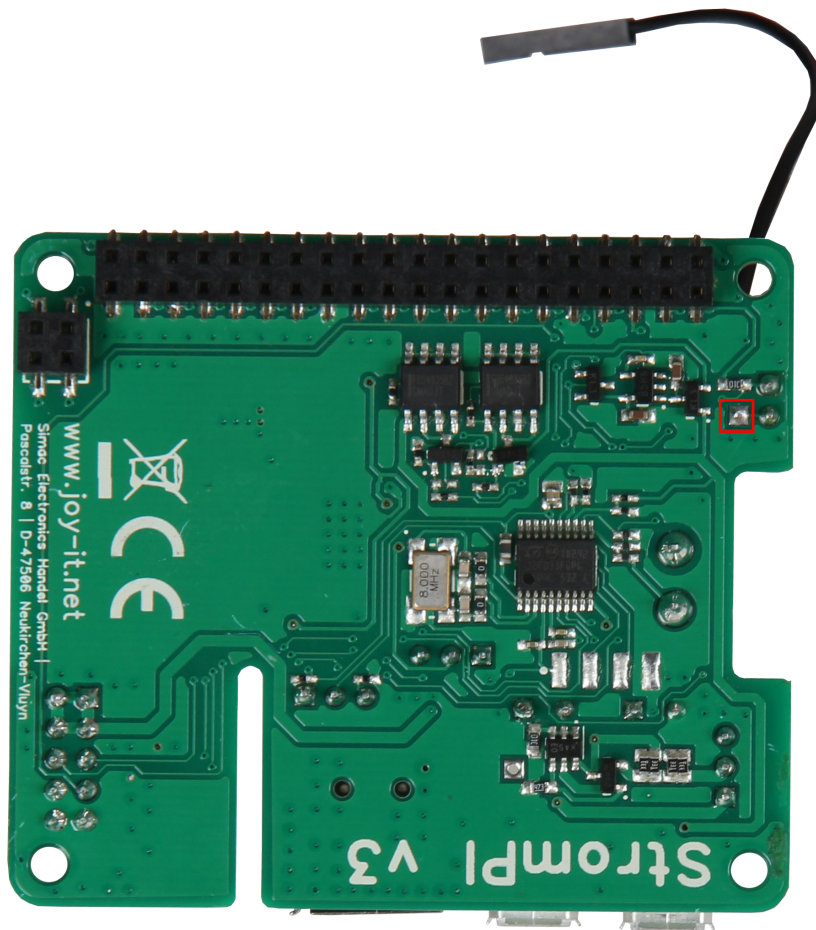
Das andere Ende des Kabels müssen Sie auf einen GPIO-Pin stecken.



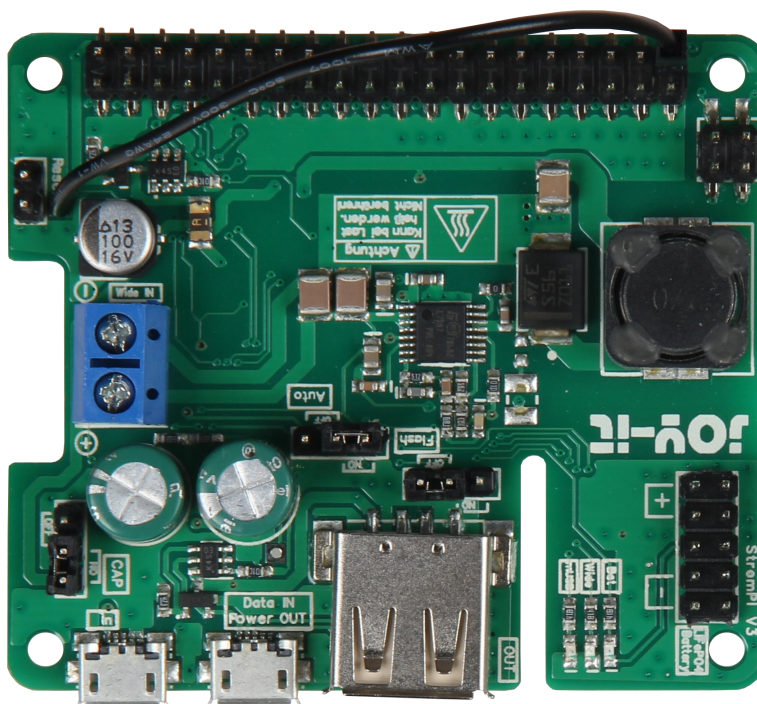
Man kann auf jeden anderen GPIO-Pin verwenden, allerdings muss dafür die Software dementsprechend angepasst werden.

## Variante 2

Löten Sie ein Ende des Kabels, wie im Bild zu sehen an das freie Pad.  
Beachten Sie das der Jumper an dem Reset-Pin dafür gesetzt sein muss.

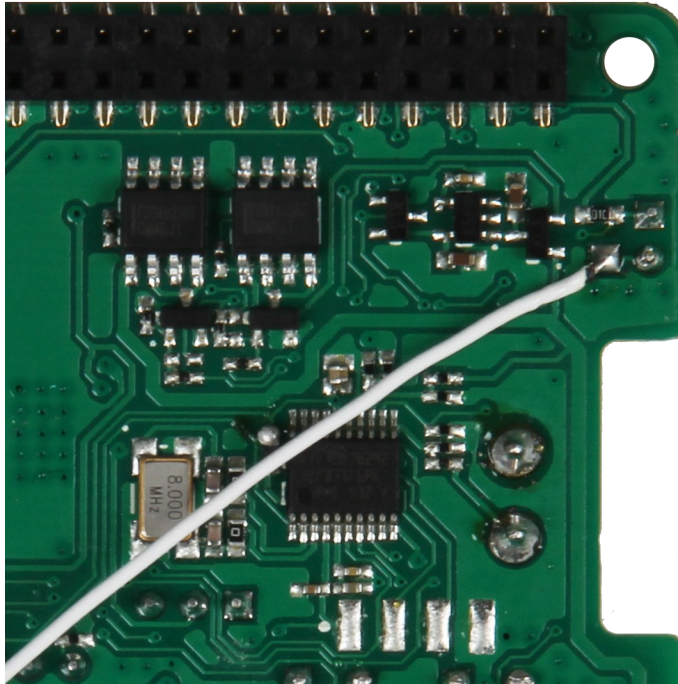


Das andere Ende des Kabels wird wie bei Variante 1 auf einen freien GPIO-Pin gesteckt.

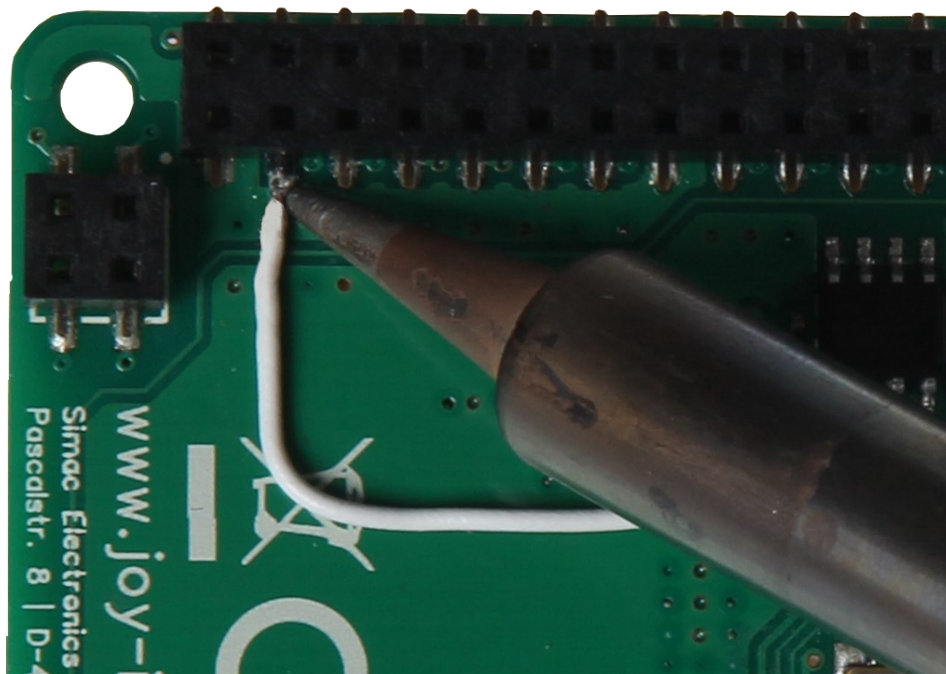


### **Variante 3**

Für Variante 3 löten Sie, wie bei Variante 2 ein Ende des Kabels an das freie Pad des Reset-Pins. Auch hier muss der Jumper gesetzt bleiben.

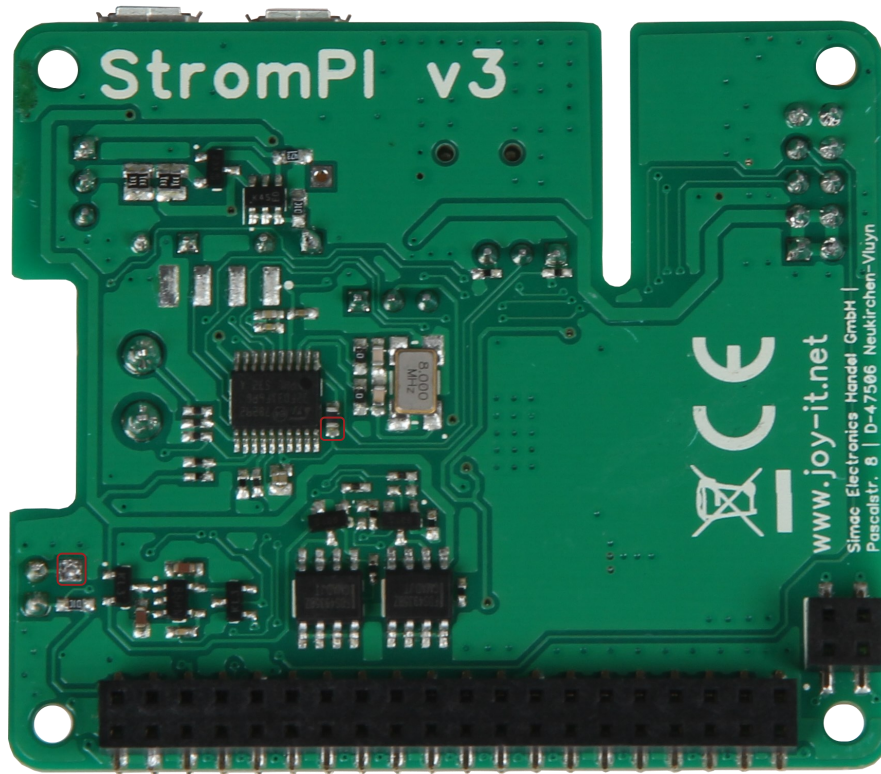


Das andere Ende des Kabels löten Sie, wie im Bild unten zu sehen an einen der GPIO-Pins

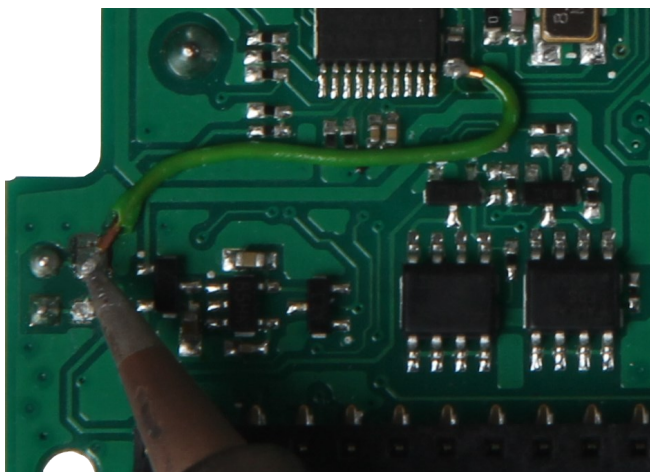
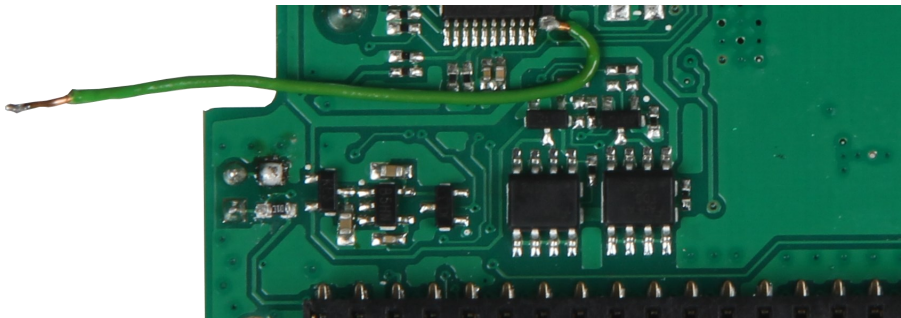


## PowerOn-Button

Die beiden rot markierten Löt pads müssen mit einem Kabel überbrückt werden.

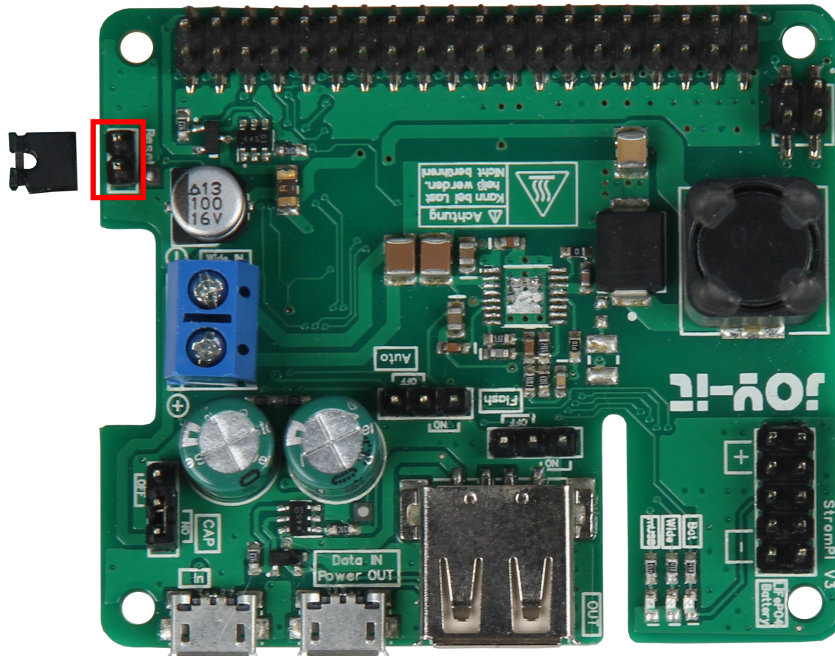


Geben Sie dazu zunächst etwas Lötzinn auf beide Pads. Löten Sie nun ein Ende des Kabels an das obere Pad.



Als letztes wird das andere Kabelende an dem unteren Löt pad angelötet.

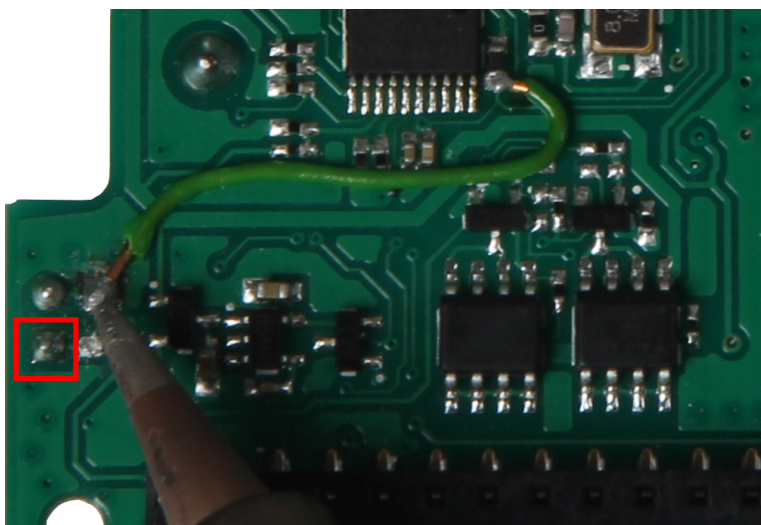
Jetzt kann man das System nach einem Poweroff - Befehl durch das Kurzschließen der rot markierten Reset-Pins neustarten. Im Betrieb dürfen die Pins nicht gebrückt sein. Für mehr Komfort ist es möglich einen Taster (Schließer) an die Pins anzuschließen um diese für den Neustart kurzzuschließen.

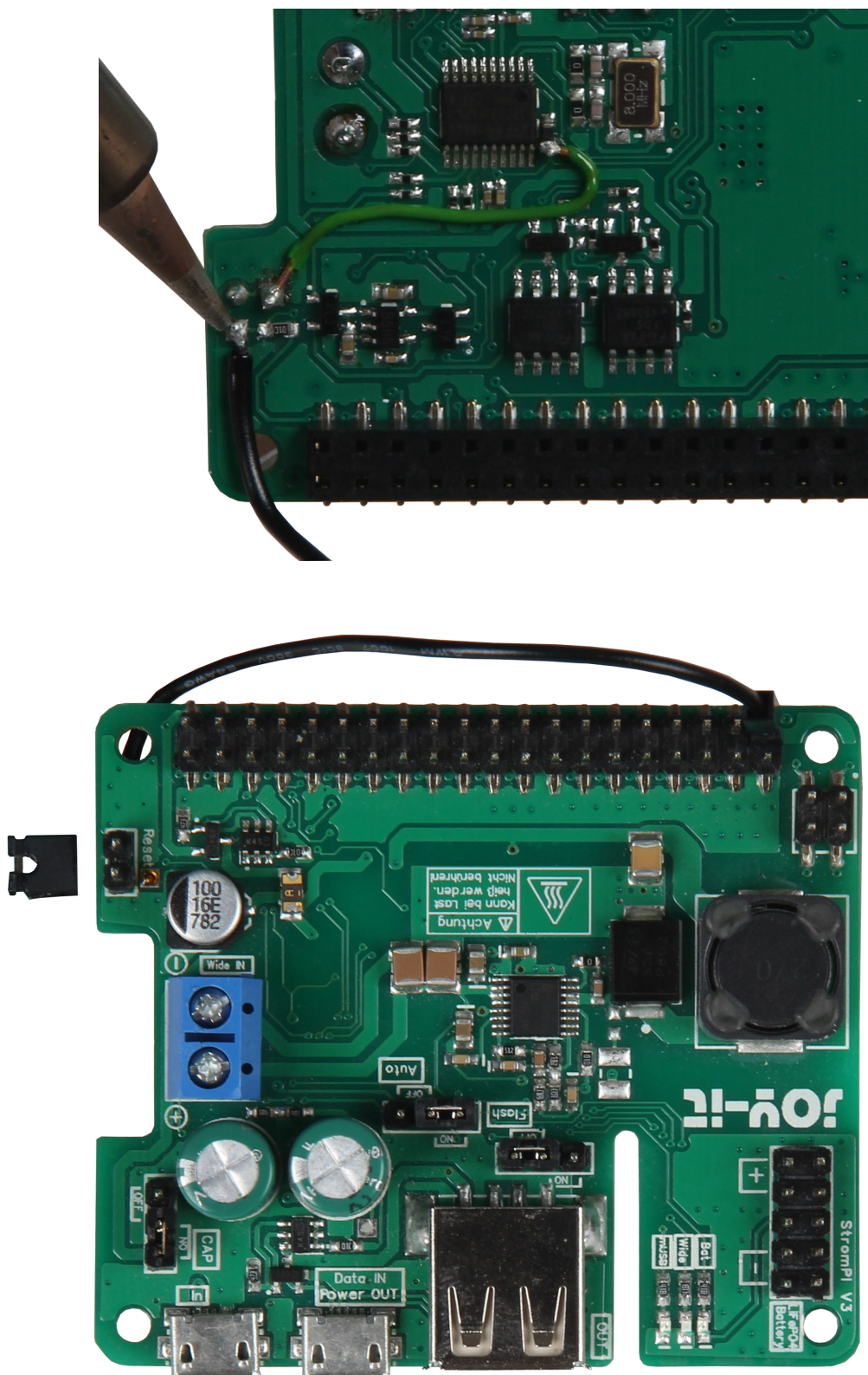


**WICHTIG! Zwischen Poweroff und dem Neustart müssen mindestens 30 Sekunden liegen.**

### Serialless und PowerOn-Button

Da das Pad, dass normalerweise für die Serialless Modifikation genutzt wird durch die Poweroff Modifikation belegt ist, muss das Kabel für den Serialless-Modus jetzt and das rot markierte Pad angelötet werden.





Für mehr Informationen über den Serialless-Modus lesen Sie bitte die **Serialless-Anleitung**.

## 11. ÖFFNEN DER SERIELLEN KONSOLE DES STROMPI 3

Um Einstellungen manuell über die StromPi Konsole vorzunehmen, müssen Sie Minicom herunterladen und installieren, geben Sie dazu

```
sudo apt-get update
```

```
sudo apt-get upgrade
```

```
sudo apt-get install minicom
```

Nachdem Minicom erfolgreich installiert wurde, können Sie mit folgendem Befehl eine serielle Verbindung zum StromPi 3 herstellen:

```
sudo minicom -D /dev/ttyAMA0 -b 38400
```



**Achtung!** Das System wartet nun auf eine weitere Eingabe und zeigt Ihre nächsten Tastaturanschläge unter Umständen nicht an.

Geben Sie den folgenden Befehl ein und bestätigen Sie diesen mit **Enter**, auch wenn Sie Ihre Eingabe möglicherweise nicht sehen:

Die serielle Konsole des StromPi 3 öffnen Sie mit:

```
startstrompiconsole
```

oder der Abkürzung:

```
sspc
```

Die serielle Konsole des StromPi 3 öffnet sich nun. Die aktuelle Konfigurationseinstellung Ihres Gerätes können Sie mit dem Befehl **show-status** einsehen.

```
-----  
Welcome to the StromPi 3 Console  
-----  
Type help to view a list of available commands.  
[When you press ENTER the previous command would be executed again]  
> |
```

Eine allgemeine Übersicht liefert der Befehl **help**.

Mit der Tastenkombination **STRG + A / X / Enter** schließen Sie Minicom.

Mit den Tastenkombinationen **SHIFT + Bild auf / ab** können Sie durch das Terminalfenster scrollen. Das Mausrad funktioniert dazu auch.

Sollte nach dem Eingeben des Befehls:

```
sudo minicom -D /dev/ttyAMA0 -b 38400
```

**Port/dev/tty8** (wie im Bild) und nicht **Port/dev/ttyAMA0** stehen, müssen Sie Minicom manuell konfigurieren.

```
Welcome to minicom 2.7

OPTIONS: I18n
Compiled on Apr 22 2017, 09:14:19.
Port /dev/tty8, 12:04:03

Press CTRL-A Z for help on special keys
```

Um das zu erreichen, geben Sie den folgenden Befehl ein:

```
sudo minicom -s
```

Sie sehen nun das folgende Fenster:

```
+-----[configuration]-----+
| Filenames and paths          |
| File transfer protocols      |
| Serial port setup            |
| Modem and dialing            |
| Screen and keyboard          |
| Save setup as dfl             |
| Save setup as..              |
| Exit                         |
| Exit from Minicom            |
+-----+-----+
|                               |
+-----+-----+
```

Wählen Sie hier **Serial port setup** aus.

```
+-----+
| A - Serial Device      : /dev/ttyAMA0 |
| B - Lockfile Location  : /var/lock    |
| C - Callin Program     :              |
| D - Callout Program    :              |
| E - Bps/Par/Bits       : 38400 8N1    |
| F - Hardware Flow Control : Yes       |
| G - Software Flow Control : No        |
| H - RS485 Enable       : No          |
| I - RS485 Rts On Send  : No          |
| J - RS485 Rts After Send : No         |
| K - RS485 Rx During Tx : No          |
| L - RS485 Terminate Bus : No          |
| M - RS485 Delay Rts Before: 0         |
| N - RS485 Delay Rts After : 0         |
|                               |
| Change which setting? █ |
+-----+
```

Drücken Sie nun **A** und ändern Sie das Serial Device zu **/dev/ttyAMA0**.  
Bestätigen Sie anschließend mit **Enter**.

Geben Sie nun **E** ein und das folgende Fenster erscheint:

```
+-----[Comm Parameters]-----+
| A - Serial De|          |          |          |          |
| B - Lockfile Loc|      Current: 38400 8N1          |
| C - Callin Proj| Speed      Parity      Data      |
| D - Callout Proj| A: <next>    L: None      S: 5      |
| E - Bps/Par/B| B: <prev>    M: Even      T: 6      |
| F - Hardware Flo| C: 9600      N: Odd      U: 7      |
| G - Software Flo| D: 38400      O: Mark      V: 8      |
| H - RS485 En| E: 115200      P: Space          |
| I - RS485 Rts|          |          |          |
| J - RS485 Rts A| Stopbits          |          |
| K - RS485 Rx Du| W: 1              Q: 8-N-1          |
| L - RS485 Termi| X: 2              R: 7-E-1          |
| M - RS485 Delay|          |          |          |
| N - RS485 Delay|          |          |          |
|          |          |          |          |
|          | Choice, or <Enter> to exit? █ |
+-----+
Change which
```

Drücken Sie nun **D** um eine Baudrate von 38400 einzustellen. Bestätigen Sie anschließend zwei mal mit **Enter** um in das Konfigurationsmenü zurückzukehren. Drücken Sie danach auf ***Speichern als dfl*** und dieses Fenster erscheint als Bestätigung.

```
+-----+
| Configuration saved |
+-----+
```

Sie können nun mit **ESC** das Konfigurationsmenü verlassen und anschließend **STRG + A** drücken. Nun sehen Sie im unteren Teil der Konsole folgendes:

```
CTRL-A Z for help | 38400 8N1 | NOR | Minicom 2.8 | VT102 | Offline | ttyAMA0
```

Wenn bei Ihnen alles so aussieht, wie auf dem Bild ist Minicom richtig konfiguriert. Nun können Sie den folgenden Befehl eingeben, um Minicom zu starten.

```
sudo minicom
```

Nun können Sie mit...

```
startstrompiconsole
```

... oder der Abkürzung...

```
sspc
```

... die StromPi 3 Konsole öffnen.

Auf den nächsten Seiten sind alle StromPi-Konsolenbefehle aufgelistet.

## 12. STROMPI 3 KONSOLENBEFEHLE

### Befehle für die CLI und Main Firmwareversionen

StromPi - Konsole starten:

```
startstrompiconsole
```

oder

```
sspc
```

StromPi - Konsole beenden:

```
quit
```

Alle Befehle auflisten:

```
help
```

Einstellen der Uhrzeit der RTC des StromPis:

```
set-clock <hour> <minutes> <seconds>
```

Einstellen des Datums der RTC des StromPis:

```
set-date <day> <month> <year> <weekday>
```

Ausgeben der gemessenen Spannungen:

```
adc-output
```

Einstellen des StromPi-Mode:

(<1>: mUSB → Wide, <2>: Wide → mUSB, <3>: mUSB → Bat, <4>: Wide → Bat, <5>: mUSB → Wide, <6>: Wide → mUSB → Bat)

```
strompi-mode <number>
```

Ausgeben der aktuellen Zeit der RTC des StromPis:

```
time-output
```

Anzeigen der aktuellen globalen Einstellungen:

```
show-status
```

Anzeigen der aktuellen Alarmeinstellungen:

```
show-alarm
```

Ausschalten des Raspberry Pi mit dem StromPi:

```
poweroff
```

## **Befehle nur für die CLI Firmwareversion**

Setzen des Shutdown Timers in Sekunden:

```
set-timer <shutdown-seconds>
```

Aktivieren <1> oder deaktivieren <0> des Raspberry Pis Shutdown:

```
shutdown-enable <option>
```

Aktivieren <1> oder deaktivieren <0> der Warnung bei Verlust der primären Spannungsversorgung:

```
warning-enable <option>
```

Einstellen des WakeUp-Alarms:  
(<1>: Uhrzeit, <2>: Datum, <3>: Wochentag)

```
alarm-mode <mode-number>
```

Aktivieren <1> oder deaktivieren <0> des WakeUp-Alarms:

```
alarm-enable <option>
```

Einstellen der Zeit des WakeUp-Alarms:

```
alarm-set-time <hour> <minutes>
```

Einstellen des Wochentags des WakeUp-Alarms:

```
alarm-set-weekday <weekday>
```

Einstellen des Datums des WakeUp-Alarms:

```
alarm-set-date <date> <month>
```

Einstellen der Zeit des Poweroff-Alarms:

```
poweroff-set-time <hour> <minutes>
```

Aktivieren <1> oder deaktivieren <0> des Poweroff-Alarms:

```
poweroff-enable <option>
```

Aktivieren <1> oder deaktivieren <0> des Serialless - Modes:

```
serialless-mode <option>
```

Einstellen der Batteriespannung bei der, der Raspberry Pi ausgeschaltet wird:

```
batlevel-shutdown <level-number>
```

Aktivieren <1> oder deaktivieren <0> des PowerOff-Interval-Alarms:

```
interval-enable <option>
```

Einstellen der On time des Interval-Alarms:

```
interval-ontime <min>
```

Einstellen der Off time des Interval-Alarms:

```
interval-offtime <min>
```

Aktivieren <1> oder deaktivieren <0> des PowerOn-Buttons:

```
poweronbutton-enable <option>
```

Einstellen der Zeit der PowerOn-Button Initialisierung:

```
poweronbutton-timer <seconds>
```

### 13. SERIELLE BEFEHLE

Die seriellen Befehle ermöglichen es den StromPi zu konfigurieren, ohne die StromPi Config-Skripte zu verwenden.

Beim Nutzen der Befehle sind folgende Dinge zu beachten:

- Jeder Befehl muss mit der Enter-Taste (Carriage Return, \r) abgeschlossen werden.
- Die Kommunikation erfolgt mit folgenden Parametern: Baudrate: 38400, Datenbits: 8, Stoppbits: 1, Parität: keine
- Vor dem Konfigurieren sollte der Befehl 'quit' gesendet werden, um eventuelle Menüzustände zu verlassen.
- Nach dem Setzen mehrerer Werte muss 'set-config 0 0' (nur speichern) oder 'set-config 0 1' (speichern und neustarten) gesendet werden. Das ändern des StromPi-Mode erfordert das speichern mit 'set-config 0 1'.
- Beim Aktivieren eines bestimmten Alarm-Modus (z. B. Time-, Date- oder Weekday-Alarm) müssen alle anderen Alarm-Modi deaktiviert werden.

Nur **ein Alarm-Modus kann gleichzeitig aktiv** sein. Dies gilt **nicht** für den PowerOff-Alarm oder den Interval-Alarm – diese können unabhängig von den anderen Alarmen aktiv sein.

| Befehl        | Werte  | Funktion                                                                                                                                         | Beispiel         |
|---------------|--------|--------------------------------------------------------------------------------------------------------------------------------------------------|------------------|
| quit          | -      | Beendet die StromPi Console                                                                                                                      | quit             |
| set-config 0  | 0 / 1  | Konfiguration speichern (0 = ohne Neustart, 1 = mit Neustart)                                                                                    | set-config 0 1   |
| set-config 1  | 1 - 6  | Setzt den StromPi-Modus (1 = mUSB->Wide, 2 = Wide->mUSB, 3 = mUSB->Battery, 4 = Wide->Battery, 5 = mUSB->Wide->Battery, 6 = Wide->mUSB->Battery) | set-config 1 1   |
| set-config 2  | 0 / 1  | Date-Alarm aktivieren/deaktivieren                                                                                                               | set-config 2 1   |
| set-config 3  | 0 / 1  | Weekday-Alarm aktivieren/deaktivieren                                                                                                            | set-config 3 1   |
| set-config 4  | 0 / 1  | Time-Alarm aktivieren/deaktivieren                                                                                                               | set-config 4 1   |
| set-config 5  | 0 / 1  | PowerOff-Alarm aktivieren/deaktivieren                                                                                                           | set-config 5 1   |
| set-config 6  | 0 - 59 | Minute für Wake-Up Alarm                                                                                                                         | set-config 6 30  |
| set-config 7  | 0 - 23 | Stunde für Wake-Up Alarm                                                                                                                         | set-config 7 12  |
| set-config 8  | 0 - 59 | Minute für PowerOff-Alarm                                                                                                                        | set-config 8 30  |
| set-config 9  | 0 - 23 | Stunde für PowerOff-Alarm                                                                                                                        | set-config 9 12  |
| set-config 10 | 1 - 31 | Alarm-Tag (nur bei Date-Alarm)                                                                                                                   | set-config 10 15 |
| set-config 11 | 1 - 12 | Alarm-Monat (nur bei Date-Alarm)                                                                                                                 | set-config 11 6  |
| set-config 12 | 1 - 7  | Alarm-Wochentag (1=Montag, ..., 7=Sonntag)                                                                                                       | set-config 12 4  |

### 13. SERIELLE BEFEHLE

| Befehl        | Werte     | Funktion                                                             | Beispiel         |
|---------------|-----------|----------------------------------------------------------------------|------------------|
| set-config 13 | 0 / 1     | Wake-Up Alarm aktivieren/deaktivieren                                | set-config 13 1  |
| set-config 14 | 0 / 1     | Raspberry Pi Shutdown aktivieren/deaktivieren                        | set-config 14 1  |
| set-config 15 | 0 - 65535 | Shutdown-Zeit in Sekunden                                            | set-config 15 30 |
| set-config 16 | 0 / 1     | Powerfail-Warnung aktivieren/deaktivieren                            | set-config 16 1  |
| set-config 17 | 0 / 1     | Serial-Less-Mode aktivieren/deaktivieren                             | set-config 17 1  |
| set-config 18 | 0 - 3     | Battery-Level Shutdown (0 = aus, 1 = < 10%, 2 = < 25%,<br>3 = < 50%) | set-config 18 3  |
| set-config 19 | 0 / 1     | Interval-Alarm aktivieren/deaktivieren                               | set-config 19 1  |
| set-config 20 | 0 - 65535 | Interval-On-Time in Minuten                                          | set-config 20 60 |
| set-config 21 | 0 - 65535 | Interval-Off-Time in Minuten                                         | set-config 21 60 |
| set-config 22 | 0 / 1     | PowerOn-Button aktivieren/deaktivieren                               | set-config 22 1  |
| set-config 23 | 0 - 65535 | Initialisierungszeit PowerOn-Button in Sekunden                      | set-config 23 30 |
| set-config 24 | 0 / 1     | Power-Save-Mode aktivieren/deaktivieren                              | set-config 24 1  |
| set-config 25 | 0 / 1     | PowerOff-Mode aktivieren/deaktivieren                                | set-config 25 1  |
| set-config 26 | 0 / 1     | Wakeup-Timer aktivieren/deaktivieren                                 | set-config 26 1  |
| set-config 27 | 1 - 65535 | Wakeup-Timer-Dauer in Sekunden                                       | set-config 27 60 |
| set-config 28 | 0 / 1     | Wakeup-Weekend aktivieren/deaktivieren                               | set-config 28 1  |

In diesem Codebeispiel wird, nach der Einrichtung der seriellen Schnittstelle, die StromPi Konsole beendet, der StromPi-Mode auf Mode 2 gesetzt und der Raspberry Pi shutdown aktiviert. Zum Schluss werden die Einstellungen gespeichert und die serielle Schnittstelle wieder geschlossen.

```
# -*- coding: utf-8 -*-
import serial
import time

# Configure serial interface
serial_port = serial.Serial(
    port='/dev/ttyAMA0',
    baudrate=38400,
    timeout=1,
    bytesize=8,
    stopbits=1,
    parity=serial.PARITY_NONE
)

time.sleep(0.2)

#close StromPi console
serial_port.write(b'quit\r')
time.sleep(0.5)

#set StromPi mode to mode 2 (wide -> musb)
serial_port.write(b'set-config 1 2\r')
time.sleep(0.2)

#enable Raspberry Pi shutdown
serial_port.write(b'set-config 14 1\r')
time.sleep(0.2)

#safe the config
serial_port.write(b'set-config 0 1\r')
time.sleep(0.5)

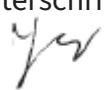
#close the serial port
serial_port.close()
```

## 14. EU-KONFORMITÄTSERKLÄRUNG

Als Hersteller erklären wir, die SIMAC Electronics GmbH, dass unser Produkt RB-StromPi3 bei bestimmungsgemäßer Verwendung die grundlegenden Anforderungen der folgenden Richtlinien erfüllt: EMV-Richtlinie 2014/30/EU (EMV) & 2011/65/EU (Rohs).

Die nachfolgend aufgeführten Normen wurden zur Beurteilung des Geräts angewandt:

EN 55022:2010+ AC:2011  
EN55024:2010

| Datum      | Name des Verantwortlichen | Unterschrift                                                                        | Stellung im Betrieb |
|------------|---------------------------|-------------------------------------------------------------------------------------|---------------------|
| 24.08.2018 | Xu Lu                     |  | Geschäftsführer     |

## 15. SONSTIGE INFORMATIONEN

Unsere Informations- und Rücknahmepflichten nach dem Elektrogesetz (ElektroG)



**Symbol auf Elektro- und Elektronikgeräten:**

Diese durchgestrichene Mülltonne bedeutet, dass Elektro- und Elektronikgeräte **nicht** in den Hausmüll gehören. Sie müssen die Altgeräte an einer Erfassungsstelle abgeben. Vor der Abgabe haben Sie Altbatterien und Altakkumulatoren, die nicht vom Altgerät umschlossen sind, von diesem zu trennen.

### **Rückgabemöglichkeiten:**

Als Endnutzer können Sie beim Kauf eines neuen Gerätes, Ihr Altgerät (das im Wesentlichen die gleiche Funktion wie das bei uns erworbene neue erfüllt) kostenlos zur Entsorgung abgeben. Kleingeräte bei denen keine äußere Abmessungen größer als 25 cm sind können unabhängig vom Kauf eines Neugerätes in Haushaltsüblichen Mengen abgeben werden.

### **Möglichkeit Rückgabe an unserem Firmenstandort während der Öffnungszeiten:**

SIMAC Electronics GmbH, Pascalstr. 8, D-47506 Neukirchen-Vluyn

### **Möglichkeit Rückgabe in Ihrer Nähe:**

Wir senden Ihnen eine Paketmarke zu mit der Sie das Gerät kostenlos an uns zurücksenden können. Hierzu wenden Sie sich bitte per E-Mail an [Service@joy-it.net](mailto:Service@joy-it.net) oder per Telefon an uns.

### **Informationen zur Verpackung:**

Verpacken Sie Ihr Altgerät bitte transportsicher, sollten Sie kein geeignetes Verpackungsmaterial haben oder kein eigenes nutzen möchten kontaktieren Sie uns, wir lassen Ihnen dann eine geeignete Verpackung zukommen.

## 16. SUPPORT

Wir sind auch nach dem Kauf für Sie da. Sollten noch Fragen offen bleiben oder Probleme auftauchen stehen wir Ihnen auch per E-Mail, Telefon und Ticket-Supportsystem zur Seite.

E-Mail: [service@joy-it.net](mailto:service@joy-it.net)

Ticket-System: <https://support.joy-it.net>

Helpdesk: <https://www.pienergy.joy-it.net/de/helpdesk>

Telefon: +49 (0)2845 9360 – 50

Für weitere Informationen besuchen Sie unsere Website:

[www.joy-it.net](http://www.joy-it.net)