# Allgemein

Bateriebetriebene Elektrofahrzeuge beziehen ihre Energie aus Akkus, welche aufgeladen müssen. Hiefür gibt es verscheidene Lösungen. Im privaten Bereich sind Wallboxen eine gute Möglichkeit, dass Fahrzeug in angemessener Zeit aufzuladen. Eine Wallbox ist eine spezielle, auf das Fahrzeug abgestimmte, Steckdose, welche bis zu 11kW oder 22 kW liefern. Auf dem Markt sind viele Wallbaoxmodelle verfügbar. Premium-Modelle besitzen eine eingebaute mandantenfähige Zugriffskontrolle und Abrechung der Stromkosten.

Die günstigen Wallbox-Modelle, wie die Heidelberg Home Eco, können das Fahrzeug auschließlich laden. Eine Zugriffskontrolle oder Abrechnungsfunktionen stehen nicht zu Verfügung.

Das Projekt Make the Wallbaox better (MtWb) erweitert die Fähigkeiten der Wallbox und folgende Funktionen:

* Mandantenfähig
* Entsperrung der Wallbox nur mit Zugangskontrolle
* Zählen der Verbräuche
* Abrechnung je nach Mandanten
* Anzeige der aktuellen Leistungsaufnahme/ Ampere
* Als Webseite (nur im WLAN sichtbar)

# Schaltplan





# GPIO-Belegung

* GPIO 3 – Auslesen S0-Schnittstelle
* GPIO 13 - Relaissteuerung Schütz

# Software

Die Software steht als Open-Source zur freien Verfügung. Die Softwarequellen können über <https://github.com/ReneSchwarzer/MtWb> bezogen werden.

# Einrichtung Raspberry PI

|  |  |  |
| --- | --- | --- |
| **1.** | Raspbian light auf SD-Karte kopieren. | <https://downloads.raspberrypi.org/imager/imager.exe> |
| **2.** | Anmelden als: pi Passwort rasperry |  |
| **3.** | Raspberry pi konfigurieren   * Password ändern: pi Password hallo * WLAN einrichten: SSID=xxxx, KEY=xxxxx * HOST ändern: mtwb | pi@raspberrypi:~ $ sudo raspi-config |
| **4.** | SSH aktivieren: Leere Datei mit den Namen ssh in der BOOT-Partition erstellen. | pi@mtwb:cd /boot  pi@mtwb:sudo touch ssh |
| **5.** | ssh auf Windows starten | ssh pi@mtwb |
| **6.** | MC installieren (optional) | pi@mtwb:~ $ sudo apt-get install mc -y |
| **7.** | Alias erstellen in /etc/profile | alias ll='ls -l' |
| **8.** | Net Core installieren (siehe https://dotnet.microsoft.com/download/linux-package-manager/debian10/runtime-current) | pi@mtwb:~ $ wget https://download.visualstudio.microsoft.com/download/pr/ccbcbf70-9911-40b1-a8cf-e018a13e720e/03c0621c6510f9c6f4cca6951f2cc1a4/dotnet-sdk-3.1.201-linux-arm.tar.gz  pi@mtwb:~ $ sudo mkdir /usr/share/dotnet-sdk  pi@mtwb:~ $ sudo tar zxf dotnet-sdk-3.1.201-linux-arm.tar.gz -C /usr/share/dotnet-sdk/ |
| **9.** | Git installieren | pi@mtwb:~ $ sudo apt-get install git |
| **10.** | Repository klonen | pi@mtwb:~ $ git clone <https://github.com/ReneSchwarzer/MtWb.git>  pi@mtwb:~ $ git clone <https://github.com/ReneSchwarzer/WebExpress.git> |
| **11.** | Rechte erteilen | pi@mtwb:~ $ chmod +x MtWb/src/core/\*.sh |

# Deploy

|  |  |  |
| --- | --- | --- |
| **1.** | Verzeichnis wechseln | pi@mtwb:~ $ cd MtWb/src/core/ |
| **2.** | Repository aktuallisieren | pi@mtwb:~/WebExpress $ git pull  (optional)pi@mtwb:~/MtWb/src/core $ git reset --hard HEAD  pi@mtwb:~/MtWb/src/core $ git pull |
| **3.** | Skript deploy.sh ausführen | (optional)pi@mtwb:~/MtWb/src/core $ sudo chmod +x \*.sh  pi@mtwb:~/MtWb/src/core $ sudo ./deploy.sh |
| **4.** | Reboot | pi@mtwb:~ $ sudo reboot |

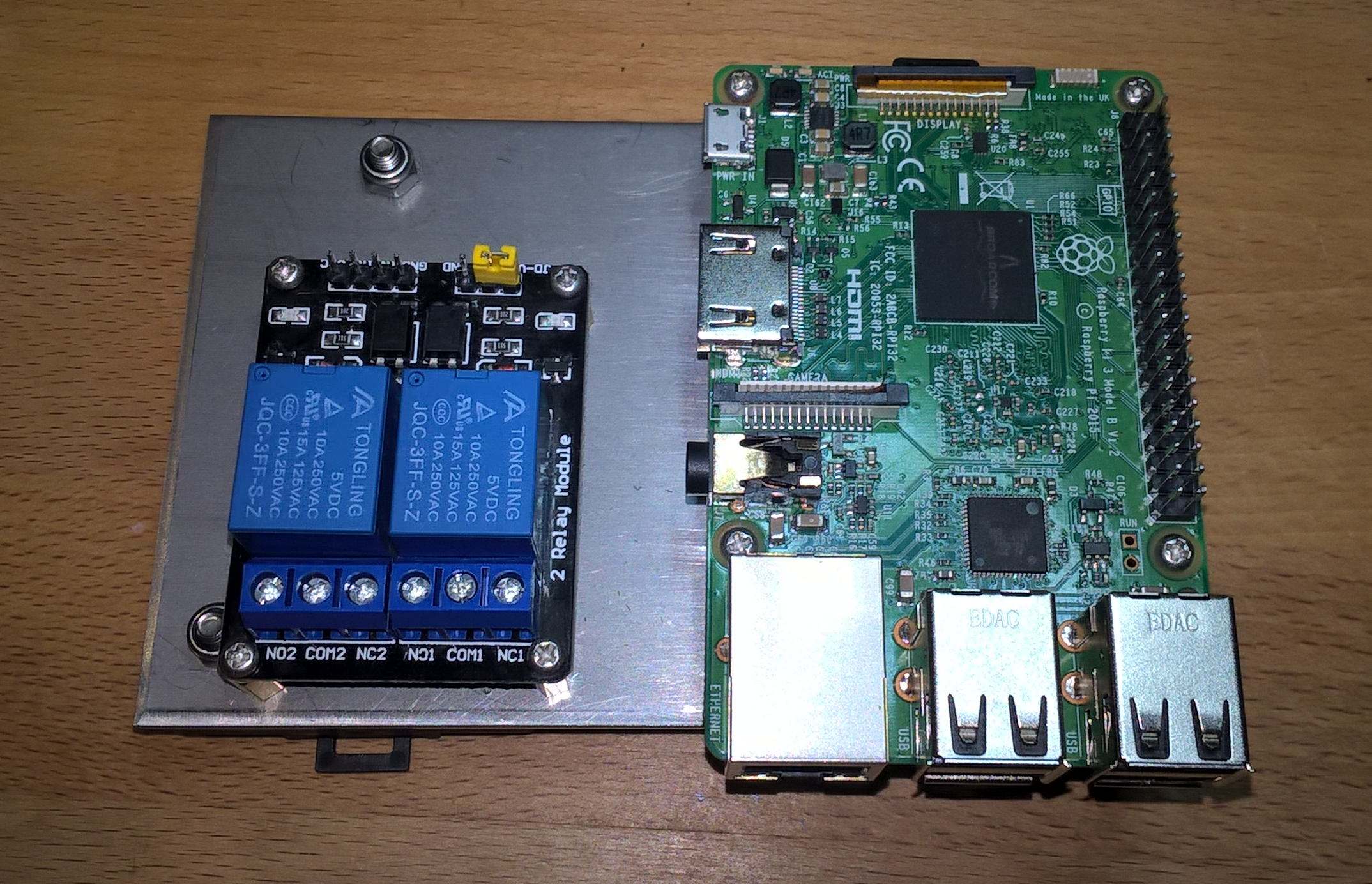
# Update

|  |  |  |
| --- | --- | --- |
| **1.** | Raspbian light aktualisieren | pi@mtwb:~ $ sudo raspi-config |
| **2.** | Apt-get updaten | pi@mtwb:~ $ sudo apt-get update  pi@mtwb:~ $ sudo apt-get upgrade |
| **3.** | Git aktualisieren | pi@mtwb:~ $ sudo apt-get install git -y |

# Einkaufsliste

|  |  |  |
| --- | --- | --- |
| **Item** | **Link** | **Preis** |
| Distanzmuttern 2,5mm | <https://www.amazon.de/gp/product/B01MY07FSU/ref=oh_aui_detailpage_o01_s00?ie=UTF8&psc=1> | 15€ |
| Raspberry Pi | https://www.amazon.de/gp/product/B01CD5VC92/ref=ppx\_yo\_dt\_b\_asin\_title\_o00\_s00?ie=UTF8&psc=1 | 40€ |
| Relaisboard | <https://www.amazon.de/gp/product/B019Z8LWFE/ref=oh_aui_detailpage_o06_s00?ie=UTF8&psc=1> | 10€ |
| Grundplatte V2A 125x80x3mm |  | € |

# Aufbau



# Bedienung

