# 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 Visuelles Laden (ViLa) erweitert die Fähigkeiten der Wallbox um 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





# S0-Schnittstelle

Die S0-Schnittstelle (S-Null) dient der Übertragung von Verbrauchsmesswerten (z.B. Strom, Temperatur, Gas) und wird in der EN 62053-31/ DIN 43 864 genormt. Die Übertragung der Daten erfolgt mit Hilfe von gewichteten Impulsen. Wobei die Gewichtung vom verbauten Zählertyp abhängig ist und daher für ViLa dynamisch änderbar sein muss. Weiterhin können die Zähler entweder

* die Impulswertigkeit (z.B. 0,01m3/Impuls) oder
* die Impulsausgangsfrequenz *f* (z.B. 100 Impulse/m3)

angeben. Ein Impuls ist zwischen 30ms und 120ms lang. Dem Impuls folgt ein LOW-Signal, welches mindestens 30ms andauert. ViLa liest den Impulsausgang zyklisch innerhalb von 30ms, um einen Zustandswechsel zu registrieren.

Die Zähler werden in die zwei Klassen A und B eingeteilt.

* Klasse A: 25V/8mA für lange Übertragungswege
* Klasse B: 14V/2mA für kurze Übertragungswege

Die Anschlussspannung liegt zwischen 27V (A)/ 15V (B) und 5V. Der Strom darf 27 mA (A)/ 15 mA (B) nicht übersteigen. Der Raspberry Pi liefert am GPIO 3 (alternativ GPIO 2) eine Spannung von 3,3V und ist mit einem Pull-Up-Widerstand ausgestattet. Der Betrieb außerhalb der S0-Spezifikation stellt für die verbauten Komponenten (siehe Abschnitt Einkaufsliste) kein Problem dar.

# GPIO-Belegung

* GPIO 3 – Auslesen S0-Schnittstelle
* GPIO 13 – Relaissteuerung zum Schalten des Schützes

# Software

Die Software steht als Open-Source zur freien Verfügung. Die Softwarequellen können über <https://github.com/ReneSchwarzer/ViLa> 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: vila * Zeitzone ändern * SSH aktivieren | pi@raspberrypi:~ $ sudo raspi-config |
| **4.** | ssh auf Windows starten | ssh pi@vila |
| **5.** | MC installieren (optional) | pi@vila:~ $ sudo apt-get install mc -y |
| **6.** | Alias erstellen in /etc/profile | alias ll='ls -l' |
| **7.** | Net Core installieren (siehe https://dotnet.microsoft.com/download/linux-package-manager/debian10/runtime-current) | pi@vila:~ $ wget https://download.visualstudio.microsoft.com/download/pr/ccbcbf70-9911-40b1-a8cf-e018a13e720e/03c0621c6510f9c6f4cca6951f2cc1a4/dotnet-sdk-3.1.201-linux-arm.tar.gz  pi@vila:~ $ sudo mkdir /usr/share/dotnet-sdk  pi@vila:~ $ sudo tar zxf dotnet-sdk-3.1.201-linux-arm.tar.gz -C /usr/share/dotnet-sdk/ |
| **8.** | Git installieren | pi@vila:~ $ sudo apt-get install git |
| **9.** | Repository klonen | pi@vila:~ $ git clone [https://github.com/ReneSchwarzer/ViLa.git](https://github.com/ReneSchwarzer/MtWb.git)  pi@vila:~ $ git clone <https://github.com/ReneSchwarzer/WebExpress.git>  pi@vila:~ $ git clone https://github.com/ReneSchwarzer/WebExpressAgent.git |
| **10.** | Rechte erteilen | pi@vila:~ $ chmod +x ViLa/src/core/\*.sh |

# Deploy

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

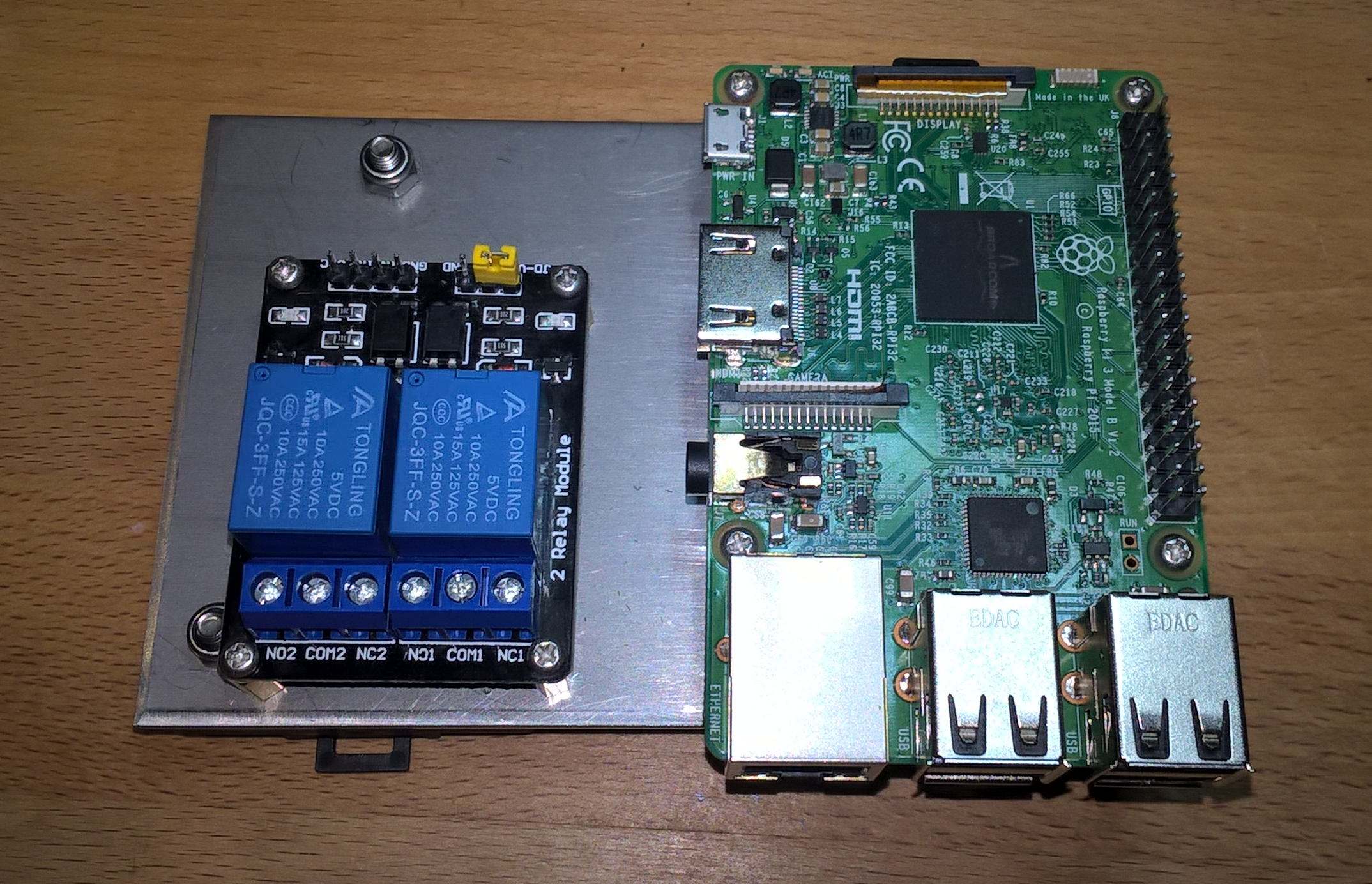
# Update

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

