Anleitung PiLogger WebMonitor

Inhaltsverzeichnis

1 Einleitung....4

2 Vorbereitung Raspberry Pi....5
2.1 Auswahl Betriebssystem....5
2.2 Voraussetzungen für die SD-Kartenerstellung....5
2.3 Download Raspberry Pi Imager....6
2.4 Raspberry Pi Imager verwenden....7
2.5 Erstes Booten, SSH Verbindung....12
2.6 System einrichten....14
2.6.1 Raspi-Config updaten....15
2.6.2 System Options....16
2.6.2.1 S6 Network at Boot....17
2.6.2.2 S8 Power LED....18
2.6.3 Display Options....19
2.6.4 Interface Options....19
2.6.4.1 I5 I2C....20
2.6.5 Performance Options....21
2.6.6 Localisation Options....21
2.6.6.1 L1 Locale....22
2.6.6.2 L3 Keyboard....24
2.6.7 Advanced Options....25
2.6.8 Raspi-Config beenden....25
2.7 Allgemeines Update....26
2.7.1 Firmware Update....26
2.7.2 Distributions-Update....27
2.8 Montieren des PiLogger One....28

3 PiLogger WebMonitor installieren....29
3.1 Router als NTP-Server einrichten....29
3.2 Installer starten....30

4 Bedienungsanleitung PiLogger WebMonitor....35
4.1 Aufruf der Web-Seite mit dem Browser....35
4.2 Die Startseite – *Live Werte*....36
4.3 Die Seite *Diagramme*....37
4.3.1 Anzuzeigende Messwertreihen auswählen....37
4.3.2 Einzelmesswert anzeigen....38
4.3.3 Verschieben und Zoomen....39
4.3.3.1 Maus-Bedienung....39
4.3.3.2 Touch-Bedienung....42
4.3.4 Gleitenden Mittelwert verwenden....42
4.4 Die Seite *Einstellungen 1*....44
4.5 Die Seite *Einstellungen 2*....47
4.6 Die Seite *Kalibration*....51
4.7 Die Seite *Download*....53

5 Konzept PiLogger WebMonitor....55

Copyright © 2019,2022 , G. Weiß-Engel

PiLogger ® and the PiLogger Logo are registered trademarks of G. Weiß-Engel.
Raspberry Pi ® is a trademark of the Raspberry Pi Foundation.
Linux ® is a registered trademark of Linus Torvalds.
Windows ® is a registered trademark of Microsoft Corporation.
All other trademarks are the property of their respective owners.

1 Einleitung

Die freie und kostenlose Software 'PiLogger WebMonitor' ist eine sogenannte 'WebApplication', das heißt, sie ist mit einem Web-Browser über eine Netzwerkverbindung ausführbar. Sie dient dazu die Messwerte des PiLogger One mit dem Raspberry Pi zu verarbeiten, die Messwerte aufzuzeichnen und über das Netzwerk zu visualisieren, sowie die vollständige Verwaltung des Loggers über das Netzwerk zu ermöglichen.
Dafür wird auf dem Raspberry Pi ein kleiner Web-Server installiert, der wie ein Server im Internet auch, die entsprechenden Dateien bereitstellt und die Interaktion ermöglicht.
Diese Mini-Webseite kann sowohl auf demselben Raspberry Pi lokal mit einem Browser aufgerufen werden – wenn ein normales Desktop-System mit Monitor und Tastatur installiert ist – oder auf einem beliebigen Gerät, das mit dem Raspberry über Netzwerk verbunden ist. Das gilt auch für Smartphones und Tablets – es wird keine 'App' benötigt, nur ein Standard-Webbrowser.
Bei dem aufrufenden Browser muss lediglich Javascript aktiviert sein. Es werden keinerlei Verbindungen ins öffentliche Internet benötigt. Selbst die Synchronisation der Zeit auf dem Raspberry Pi kann mit dem eigenem Heimnetzrouter erfolgen. Der Internet-Zugang für den Raspberry kann also nach der vollständigen Installation im Router gesperrt werden.
Damit hat man also einen Netzwerk-verbundenen Logger ohne Cloud und Fremdserver.
Eine entfernte Verbindung über das Internet ist trotzdem per VPN möglich (dann allerdings mit der Vermittlung durch den VPN-Anbieter).
Diese Anleitung gibt eine Schritt-für-Schritt-Anleitung für die Einrichtung eines Raspberry Pi ohne grafische Oberfläche – also ohne Monitor und Eingabegeräte – der rein nur über das lokale Netzwerk angebunden ist. Damit kann der Raspberry Pi im Dauerbetrieb mit minimalem Verbrauch arbeiten. Das Modell Zero W eignet sich dafür besonders gut, da es mit seinem Single-Core-Prozessor und dem integrierten WLAN-Modul mit ca. 600 mW auskommt.

Viel Spaß und Erfolg mit dem PiLogger WebMonitor !

2 Vorbereitung Raspberry Pi

2.1 Auswahl Betriebssystem

Der PiLogger WebMonitor baut auf der Raspberry-Pi-Seite vollständig auf Software-Komponenten auf Python-Basis. Damit ist prinzipiell der Einsatz auf jedem Betriebssystem (OS) mit Python-Unterstützung möglich. Lediglich die spezielle Hardware-Unterstützung für die Raspberry Pi GPIOs unter Python wird noch benötigt.
Das Ziel der weiteren Anleitung ist die einfache und direkte Beschreibung einer sogenannten 'headless'-Installation – also der Betrieb des Raspberry Pi ohne angeschlossenes Display und Eingabegerät. Damit läuft der Raspberry im Normalbetrieb ganz allein mit minimalem Hardware-Einsatz und somit geringstmöglichem Stromverbrauch als web-gesteuerter Logger.
Deshalb ist hier die Empfehlung : Raspberry Pi OS in der Lite-Version.
Dies ist das offizielle, von der Raspberry Pi Foundation bereitgestellte Betriebssystem, welches auf Debian Linux basiert und breit unterstützt wird.

2.2 Voraussetzungen für die SD-Kartenerstellung

Um eine SD-Karte für einen Raspberry Pi vorzubereiten ist natürlich ein Computer mit einem SD-Karten-Laufwerk notwendig – egal, ob intern oder extern.

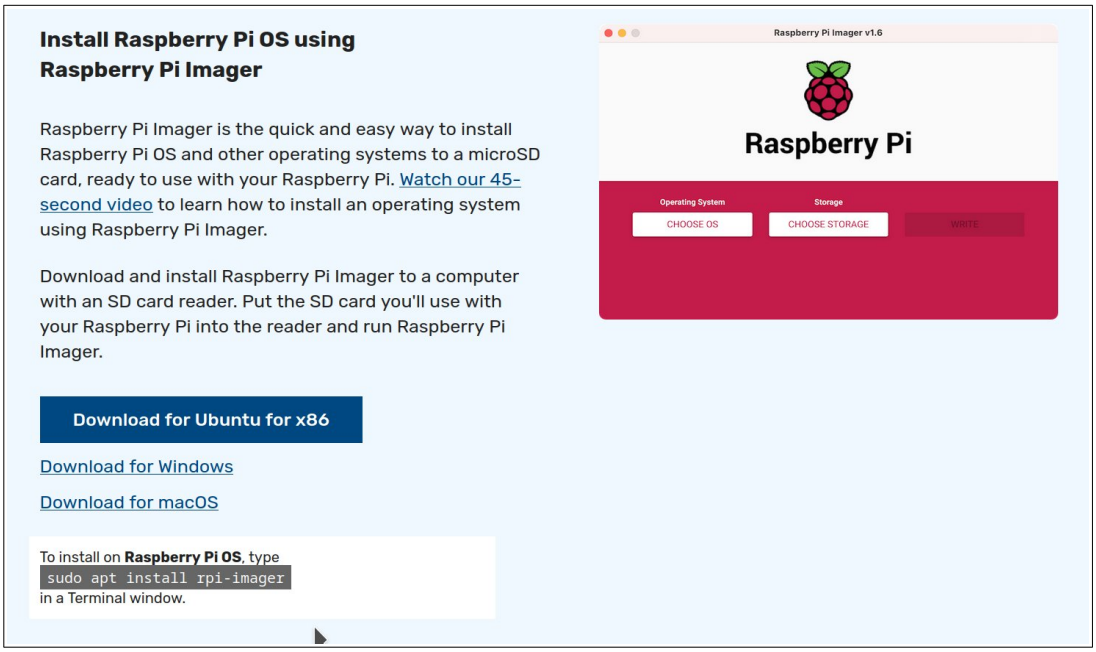
Außerdem muss eine Image-Datei des Betriebssystems per Internet heruntergeladen werden. Dafür muss der Rechner natürlich eine Internetverbindung haben.
(*Ein Image – zu deutsch: Abbild – ist eine große Datei, die den vollständigen Inhalt eines Datenträgers als 1:1 Kopie enthält.)*

Seit der Version des Raspberry Pi OS vom 4.4.2022 (Debian 11, bullseye) gibt es keinen vorinstallierten Standardbenutzer 'Pi' mehr.
Insbesondere für eine 'headless' Installation muss also für den ersten Systemstart ein Benutzer angelegt werden, damit ein Zugang über das Netzwerk überhaupt möglich ist.
Eine manuelle Installation ist zwar noch weiterhin möglich, erfordert aber unter Anderem einen weiteren Linux-Rechner (z.B. bereits laufender Raspberry) zur Erzeugung des Passwort-Hash (Fingerprint des verschlüsselten Passwortes) – sprich: Es ist nicht mehr wirklich einfach.

Für eine einfache Vorbereitung empfiehlt sich deshalb die Verwendung des Hilfsprogramms 'Raspberry Pi Imager', das von der Raspberry Pi Foundation bereitgestellt wird.

2.3 Download Raspberry Pi Imager

Auf der Software-Seite von www.raspberrypi.org gibt es die Möglichkeit ein komfortables Programm (Tool) namens 'Raspberry Pi Imager' für 3 verschiedene Betriebssysteme herunterzuladen oder mittels Befehlszeile auf einem bereits laufenden Raspberry pi zu installieren :



Nach dem Download für das zutreffende System installieren wir die Software auf unserem Hilfsrechner.
Das hängt etwas von dem vorhandenen Betriebssystem ab, passiert aber in der Regel durch einen Doppelklick auf die gerade heruntergeladene Datei.

2.4 Raspberry Pi Imager verwenden

Wir starten den Raspberry Pi Imager und sehen dieses Startfenster:

Als Erstes müssen wir mit 'OS WÄHLEN' eine der Varianten des Betriebssystems auswählen.
Da hier im Weiteren ein System ohne die grafische Oberfläche benutzt werden soll, ist die Wahl hier nicht der empfohlene erste Eintrag, sondern aus dem Untermenu darunter der Eintrag 'Raspberry Pi OS Lite (32-bit)'.
Die 64 Bit Variante wäre auch möglich, braucht aber bei geringem Geschwindigkeitsvorteil mehr Arbeitsspeicher und etwas mehr Strom – Geschmackssache...

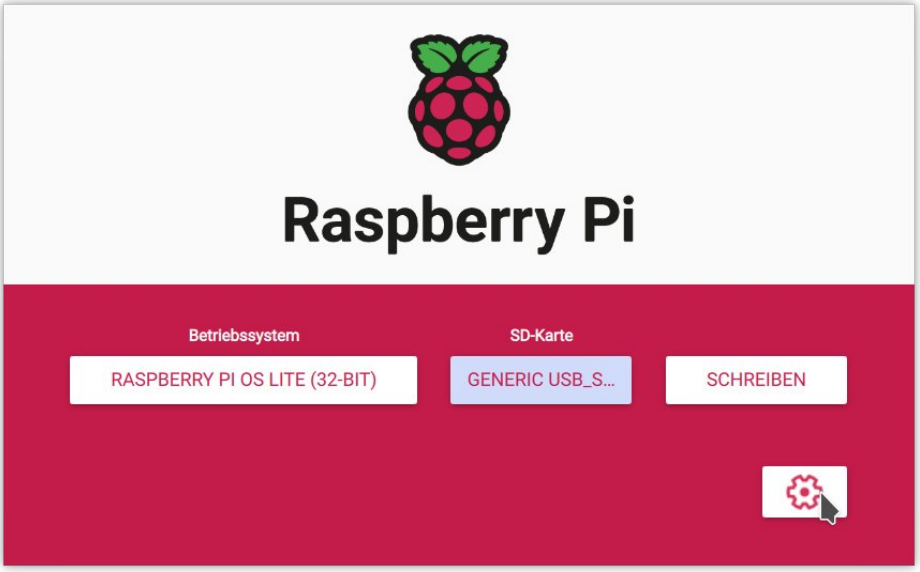
Natürlich lässt sich der PiLogger WebMonitor auch auf einem Desktop-System verwenden - also mit grafischer Oberfläche - dies ist aber nicht der Fokus dieser Anleitung.

Der Raspberry Pi Imager lädt die neueste Version live von raspberrypi.org herunter – es wird also eine Internetverbindung benötigt.

Das heruntergeladene Image wird für weitere zu flashende SD Karten auf dem Rechner gespeichert.

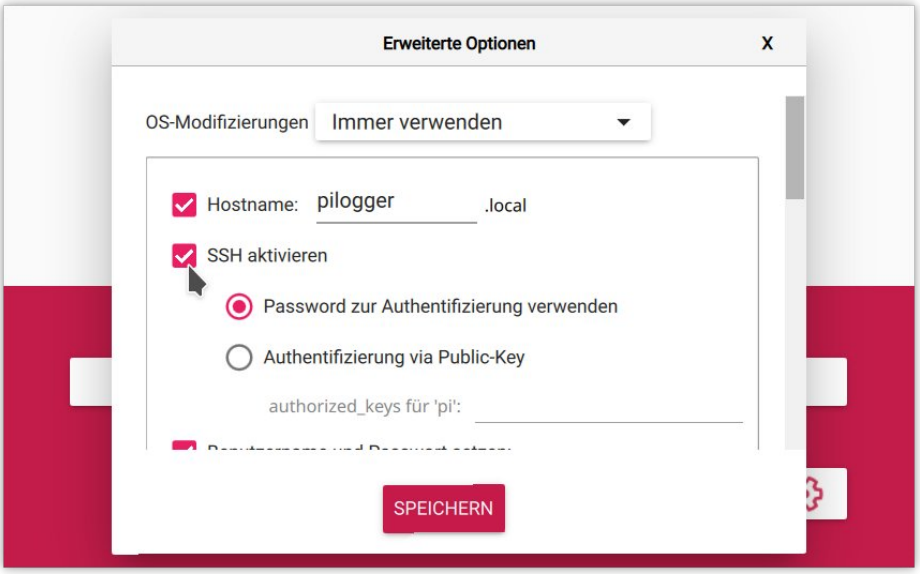
Nun müssen wir noch das *richtige* Laufwerk mit der Micro-SD-Karte für den Raspi unter 'SD-KARTE WÄHLEN'.

Danach sieht das Fenster dann so aus:



Vor dem Start müssen wir jetzt wichtige Änderungen in den Einstellungen (Zahnrad Symbol) vornehmen !

Nach dem Maus-Klick auf das Zahnrad-Symbol sehen wir das Menu 'Erweiterte Optionen':

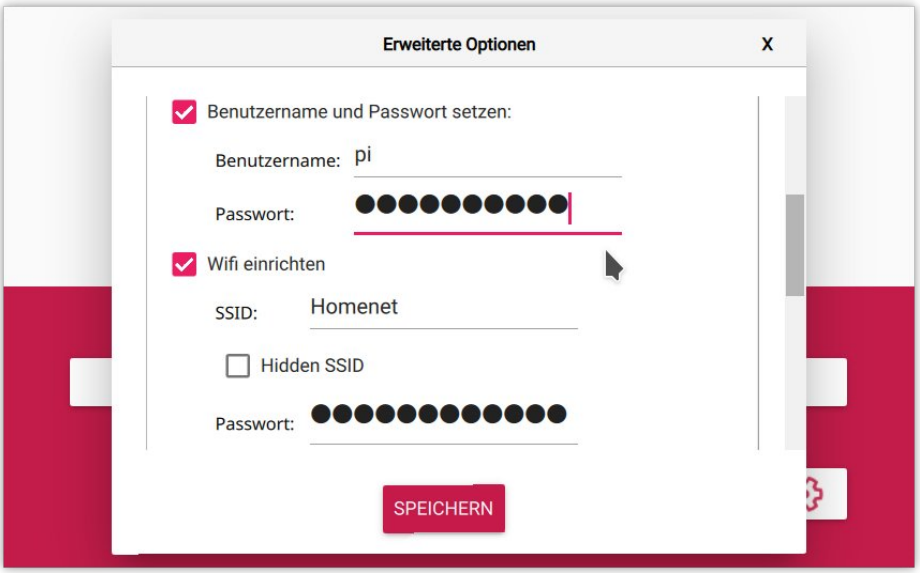


Zunächst setzen wir 'OS-Modifizierungen' auf 'Immer verwenden', das heißt die jetzt vorgenommenen Änderungen sollen für die nächste 'Imager' Sitzung gespeichert werden (merken).

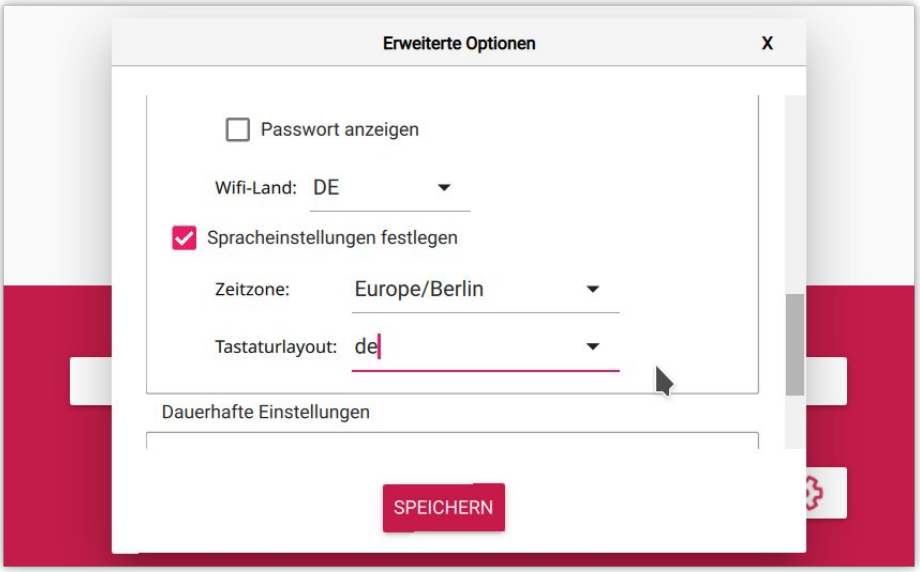
Unter 'Hostname' können wir dem neuen Raspberry einen anderen Netzwerknamen geben.

Die Checkbox darunter für 'SSH aktivieren' müssen wir einschalten und die Option 'Password zu Authentifizierung verwenden' auswählen.

Nun kommt das Anlegen des Benutzers. Hier kann nun auch ein anderer Benutzername als der alte Standard-Benutzer 'pi' gewählt werden – was einen eventuell Angriff erschwert, also sicherer ist.
Das gilt natürlich auch für das Passwort, das ja auch bisher schon so schnell wie möglich geändert werden musste. Hier also bitte ein gutes Passwort wählen :-)

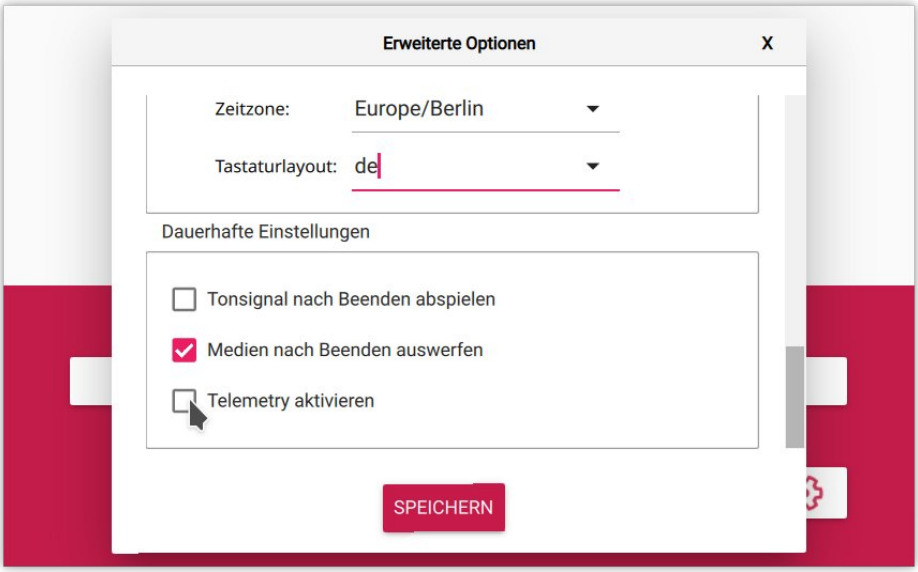


Der nächste Punkt ist die Wifi (WLAN) Einrichtung.
'SSID' (**S**ervice **S**et **Id**entifier) bedeutet hier der Name unseres lokalen WLAN Netzwerkes. Unter 'Passwort' tragen wir das zugehörige Zugangspasswort ein. Zur Kontrolle auf Tippfehler kann mit der Checkbox darunter das Passwort angezeigt werden (wenn niemand zuguckt :-)).



Als letzten Punkt für die WLAN-Einrichtung müssen wir noch das 'Wifi-Land' setzen, damit der Raspberry Pi die richtigen Funkkanäle mit der erlaubten Sendeleistung nutzt.

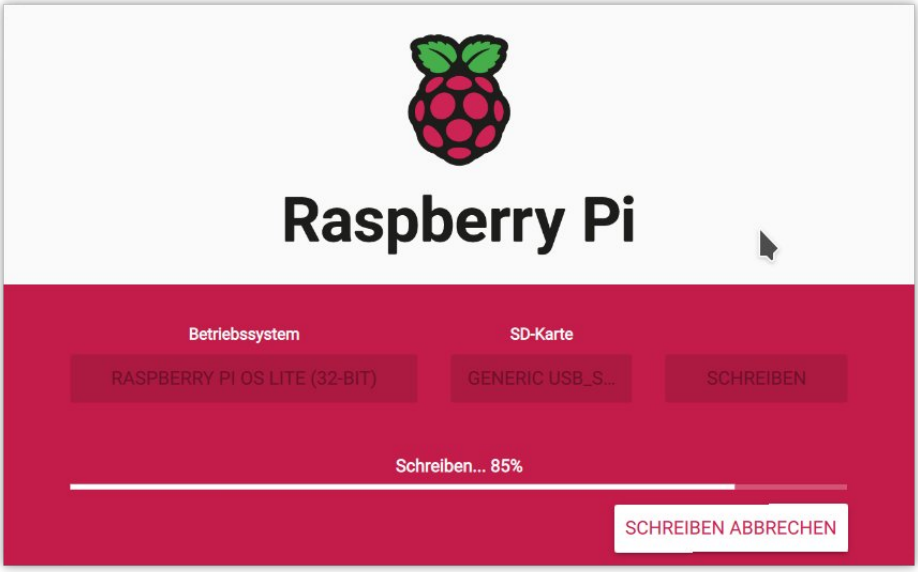
Nun sollten wir noch die 'Spracheinstellungen festlegen'.
Also 'Zeitzone' auf 'Europe/Berlin' und 'Tastaturlayout' auf 'de':



Der letzte Abschnitt heißt 'Dauerhafte Einstellungen' – das sind Einstellungen für den Imager selber:

- 'Tonsignal nach Beenden abspielen' ist Geschmacksache. Die Vollversion braucht schon eine ganze Weile zum Flashen, da sitzt man nicht die ganze Zeit daneben.
- 'Medien nach Beenden auswerfen' zu setzen ist sinnvoll. Muss sonst manuell nach dem Flashen gemacht werden.
- Interessant ist der Punkt 'Telemetry aktivieren': Imager schickt dann einige Daten zur Nutzung nach Hause...

Mit 'Speichern' sind diese besonderen Einstellungen erledigt. Wir können nun das eigentliche Flashen durch einen Klick auf 'Schreiben' im Hauptfenster starten.



Nach dem Schreiben der Daten wird automatisch die Überprüfung gestartet.

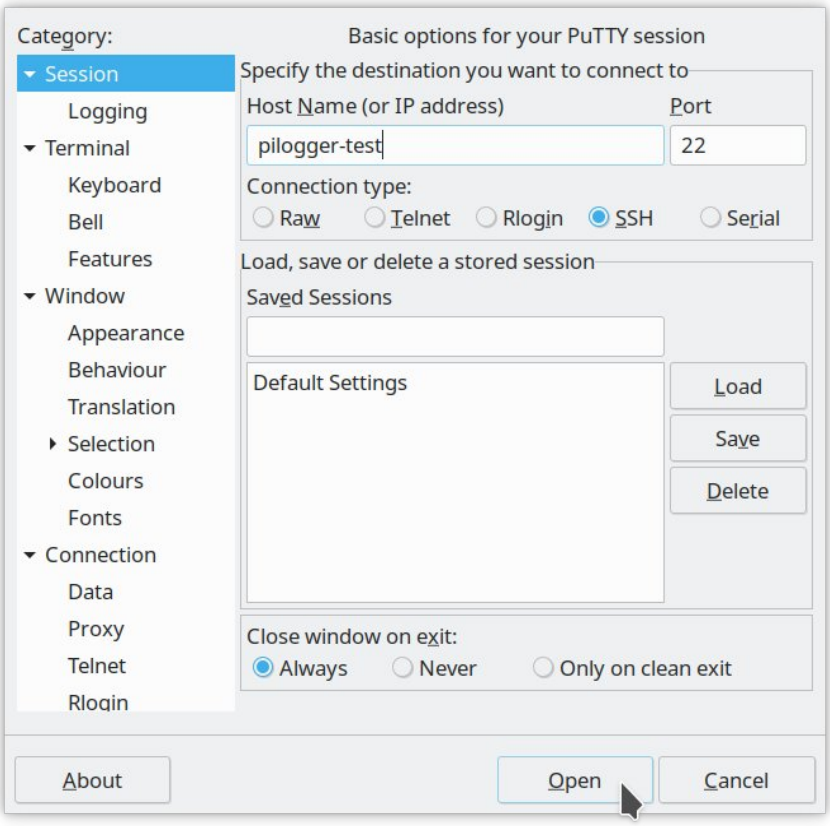
Wenn alles geklappt hat, ist die SD-Karte jetzt fertig vorbereitet und kann aus dem Laufwerk entnommen werden.

2.5 Erstes Booten, SSH Verbindung

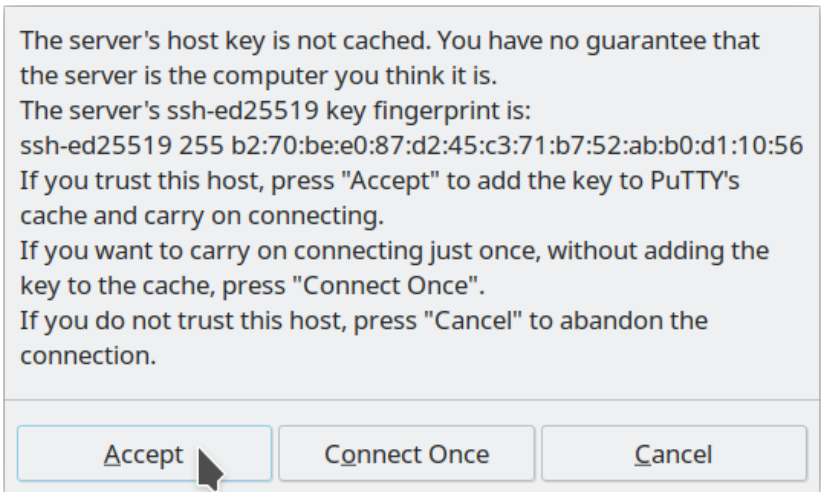
Nun ist es Zeit die SD-Karte in den Raspberry Pi einzusetzen und den Strom einzuschalten, um dem Raspberry Pi beim ersten Hochfahren (Booten) eine Weile zuzusehen – die SD-Karten-LED muss einige Zeit ganz hektisch blinken.
Der Raspberry meldet sich nun also beim ersten Booten am WLAN-Router (oder Access-Point) an. Dabei verwendet er den Netzwerk-Namen den wir in den besonderen Einstellungen im Imager angegeben haben.

Wir starten jetzt ein SSH-fähiges Terminal-Programm auf unserem anderen Rechner - z.B. das bewährte 'Putty'.

Dort geben wir für den zu kontaktierenden Raspberry Pi (jetzt von hier gesehen, der Host-Rechner) als Adresse den Netzwerknamen und Port 22 an.
Die IP-Adresse, die der Router dem Raspberry zugewiesen hat, wird automatisch ermittelt. Es macht aber durchaus Sinn im Router einzustellen, dass der Raspberry immer genau diese Adresse erhalten soll.

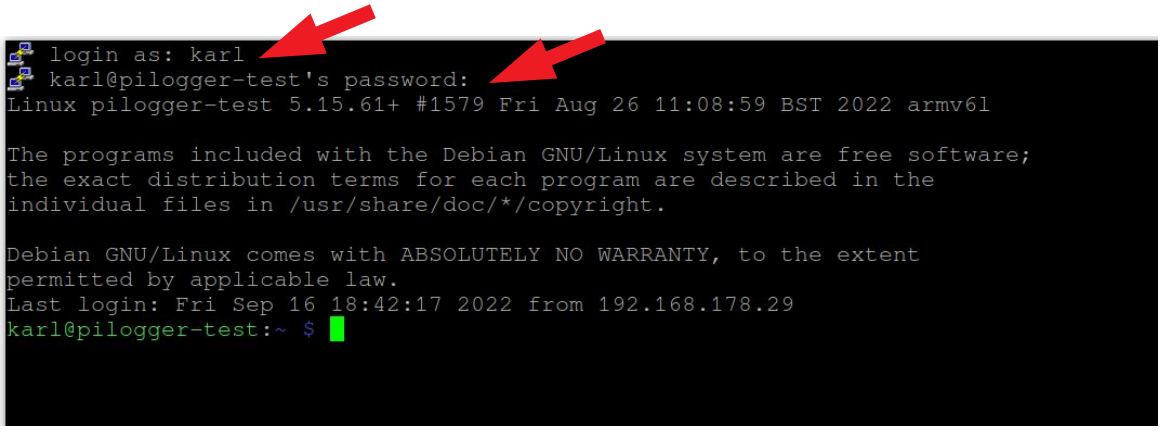


Der Raspberry Pi sendet nun ein Zertifikat, das wir in Putty jetzt annehmen müssen, um sicher zu stellen, dass der entfernte Rechner (in unserem Fall der Raspberry) wirklich der echte gewünschte Verbindungspartner ist:



Diese Nachfrage ist wie das Überprüfen eines Personalausweises, eine sogenannte Authentifizierung. Sie taucht immer auf, wenn ein neuer oder geänderter SSH-Server-Schlüssel unter einer bestimmten MAC-Adresse auftaucht (Netzwerkgeräte-Schnittstellen-ID – nicht zu verwechseln mit der IP-Adresse). Das passiert beim ersten Kontakt mit einem bestimmten Gerät oder wenn das System auf diesem Gerät neu installiert wurde – wie in unserem Fall. Wir akzeptieren also mit 'Accept' und die Verbindung wird aufgebaut. Wir bekommen jetzt ein Terminal-Fenster mit Kommando-Zeilen-Eingabe (CLI - command line interface).

Jetzt müssen wir uns einloggen:



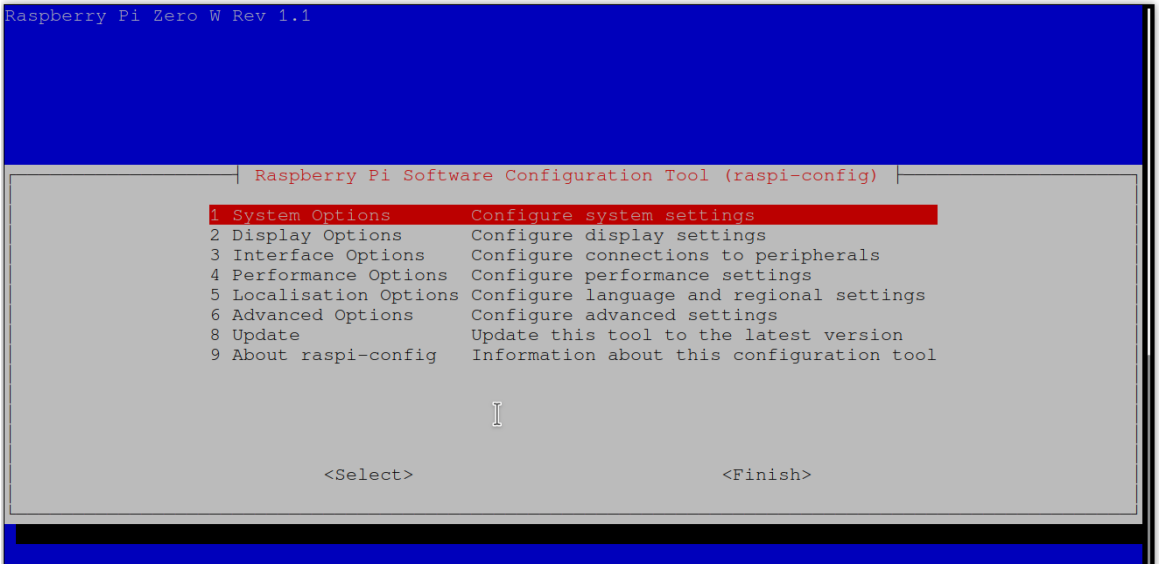
Hier verwenden wir nun natürlich Nutzername und Passwort, die wir bei der SD-Kartenkonfiguration angegeben haben.

2.6 System einrichten

Normalerweise muss ein frisches Raspberry Pi OS nun einmal grundlegend konfiguriert werden. Weil wir aber schon das meiste über die Konfiguration im 'Raspberry Pi Imager' erledigt haben, ist jetzt nicht mehr so viel zu tun.
Zum Beispiel ist die Ländereinstellung bereits in den besonderen Einstellungen gesetzt worden – hat aber die Linux 'locale' nicht vollständig korrekt gesetzt.
Für die letzten notwendigen Aktionen und um die Möglichkeiten zu zeigen, führen wir das Tool 'raspi-config' aus.
Dazu geben wir in der Kommando-Zeile folgendes ein:

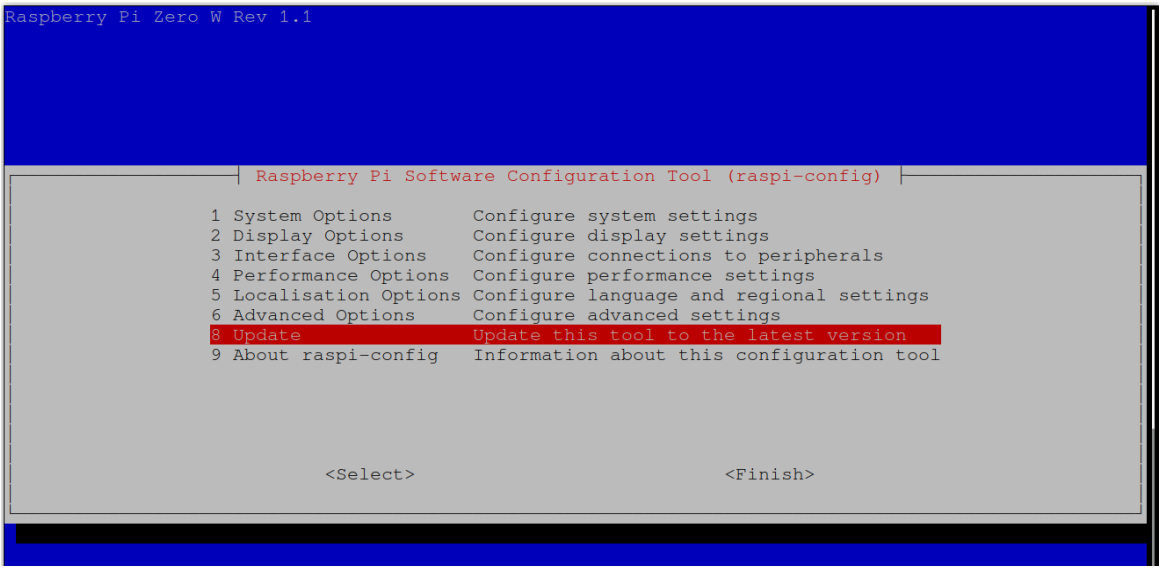
```
sudo raspi-config
```

Das Programm startet dann mit diesem Bildschirm:

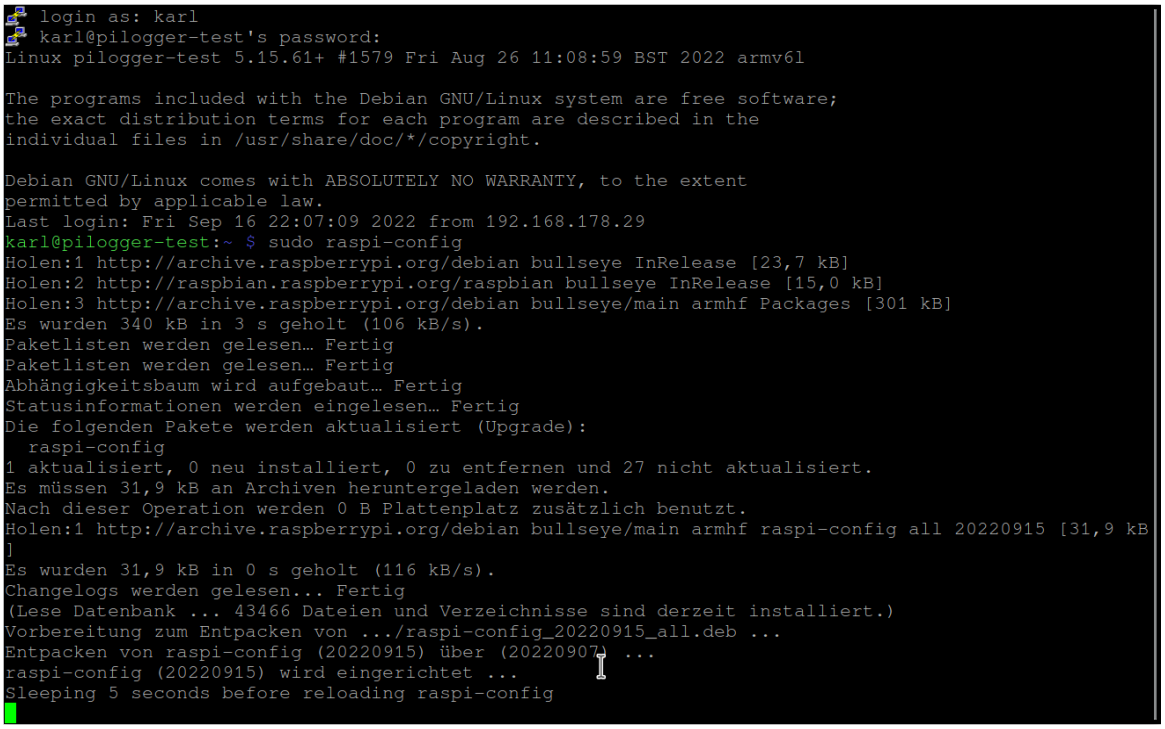


2.6.1 Raspi-Config updaten

Als Erstes machen wir einen Abstecher zur Option 8 und versuchen ein Update des Tools selber.
Dazu manövrieren wir mit der Taste 'Pfeil abwärts' auf die Zeile 8:



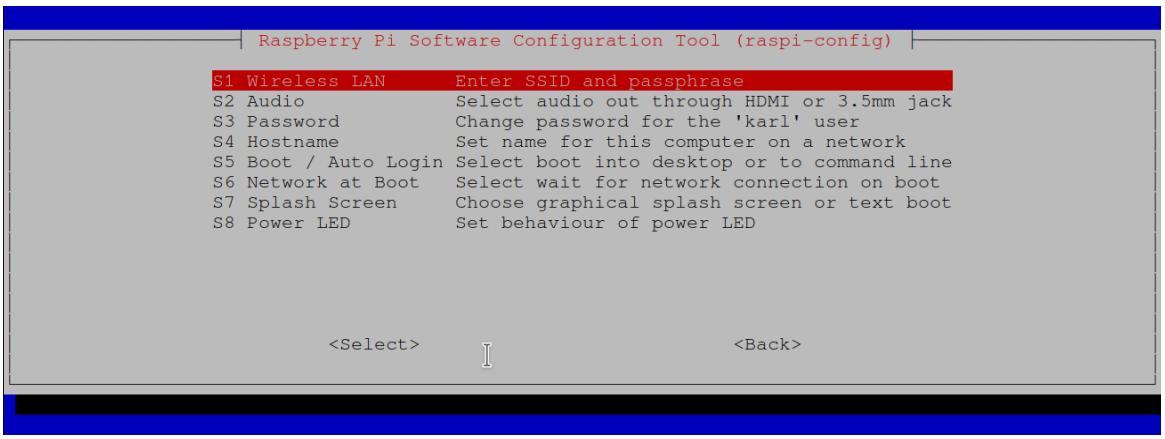
Hierfür ist eine funktionierende Internet-Verbindung notwendig.
Nach einem 'Enter' springt die Anzeige wieder auf die Kommandozeile und das Update wird durchgeführt:



Ist das Update beendet, wartet das Script 5 Sekunden und startet 'raspi-config' erneut, sodass wir wieder im Hauptmenü auf Punkt 1 stehen.

2.6.2 System Options

Da wir jetzt wieder auf dem Punkt 1 'System Options' im Hauptmenü stehen, gehen wir mit einem Druck auf die 'Enter'-Taste in das Untermenü:



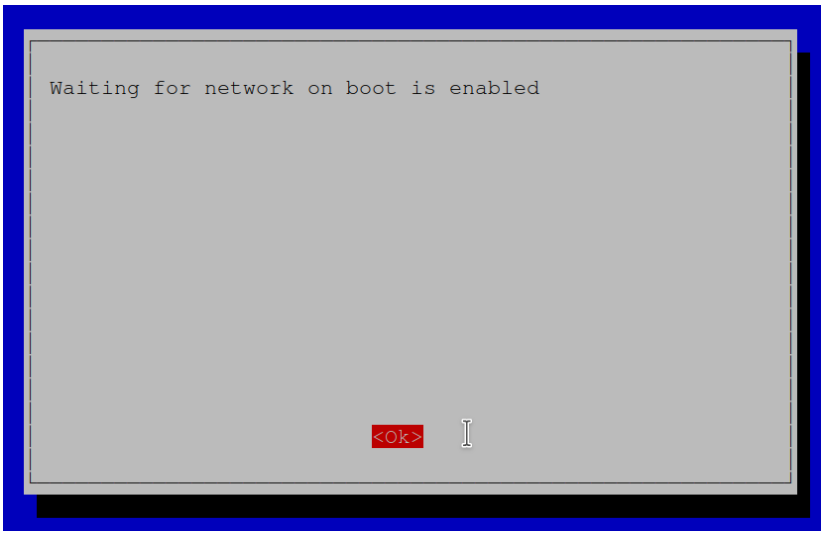
- S1 'Wireless LAN'
  gibt uns die Möglichkeit die Log-In-Daten für das WLAN zu ändern. Da wir offenbar erfolgreich eingeloggt sind, ist dies jetzt nicht notwendig.
- S2 'Audio'
  gibt die Möglichkeit bei Raspberry Pi Modellen mit Klinkenbuchse den Audioausgang zu wählen. Beim Modell Zero gibt es die nicht.
- S3 'Password'
  gibt die Möglichkeit für den aktuellen Benutzer das Passwort zu ändern. Sollte zur Zeit nicht notwendig sein.
- S4 'Hostname'
  ist der Netzwerkname des Raspi. Hatten wir ja auch frisch festgelegt.
- S5 'Boot / Auto Login'
  ist eigentlich nur für Systeme mit Display und Tastatur interessant.
- S6 'Network at Boot'
  ist eine Einstellung, die wir bearbeiten müssen…
- S7 'Splash Screen'
  ist eine Einstellmöglichkeit für Systeme mit Display.
- S8 'Power LED'
  ist eine Einstellung, die wir uns anschauen...

2.6.2.1 S6 Network at Boot

Diese Einstellung ist wichtig für den Logger-Betrieb. Die Log-Einträge erhalten ihren Zeitstempel vom Raspberry, der selber aber keine Echtzeituhr (Realtime Clock, RTC) hat. Der Raspberry synchronisiert seine Systemzeit über das Netzwerk mit einem Zeitserver (per NTP). Dies kann aber nur erfolgreich sein, wenn bereits eine Netzwerkverbindung besteht – solange steht die Systemzeit auf der zuletzt auf SD-Karte gespeicherten Zeit. Startet nun die Logger-Software vor der Zeitsynchronisation, können Log-Einträge mit falschen Zeiten erzeugt werden. Deshalb wählen wir jetzt die Möglichkeit beim Booten auf eine Netzwerkverbindung zu warten.

Wir gehen nun mit der 'Pfeil abwärts' Taste auf die Zeile S6 'Network at Boot' und drücken 'Enter'.

Daraufhin werden wir gefragt ob wir beim Booten warten wollen bis eine Netzwerkverbindung hergestellt ist.
Dabei ist bereits die Antwort markiert, die dem aktuellen Zustand dieser Einstellung entspricht. Wir wählen also 'Yes' - wenn nötig mit der 'Pfeil links'-Taste aus.
Der obige Screenshot zeigt übrigens schon ein <Ja>, weil die später folgende Lokalisierung auf diesem Raspi bereits ausgeführt wurde.
Nach dem Bestätigen mit der 'Enter'-Taste erscheint die Bestätigungsmeldung, dass diese Einstellung aktiviert ist:



Wir quittieren wieder mit 'Enter' und werden wie gewohnt in das Hauptmenü zurückgesetzt.

2.6.2.2 S8 Power LED

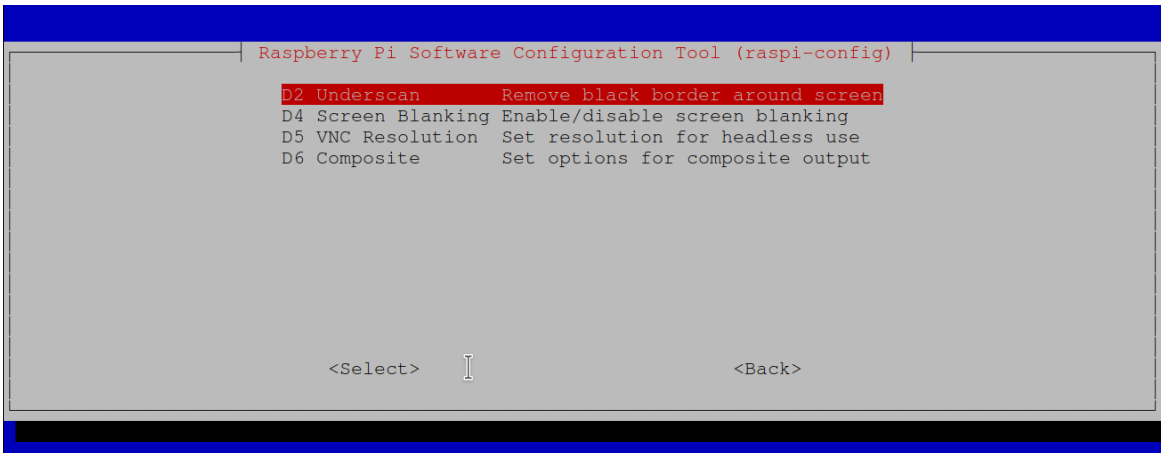
Als Nächstes gehen wir nochmal in das Menü 'System Options' und wählen die Zeile S8 'Power LED' aus. Nach dem Druck auf 'Enter' werden wir gefragt:

Also, ob die Power LED blinken soll um Zugriffe auf die SD-Karte zu signalisieren. Nun, das kann jeder für sich entscheiden – sonstige Auswirkungen hat es nicht.

Nach Beantwortung der Frage wird wieder ein Bestätigungsfenster angezeigt und nach dessen Quittierung mit 'Enter' werden wir wieder in das Hauptmenü zurückgeworfen.

2.6.3 Display Options

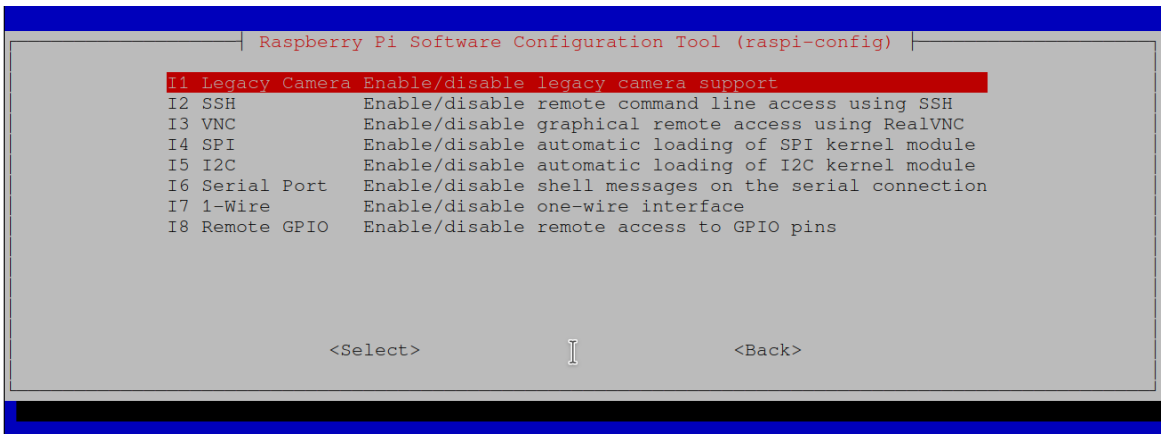
Im Hauptmenü wählen wir nun die Zeile 2 'Display Options' und bekommen dieses Untermenü:



Diese Optionen betreffen allesamt mögliche Display-Einstellungen.
Da wir hier eine 'Lite' Variante (ohne Desktop) aufgesetzt haben, sind diese Optionen hier nicht relevant.
Mit der 'Tab' Taste gehen wir auf <Back> und mit 'Enter' kehren wir in das Hauptmenü zurück.

2.6.4 Interface Options

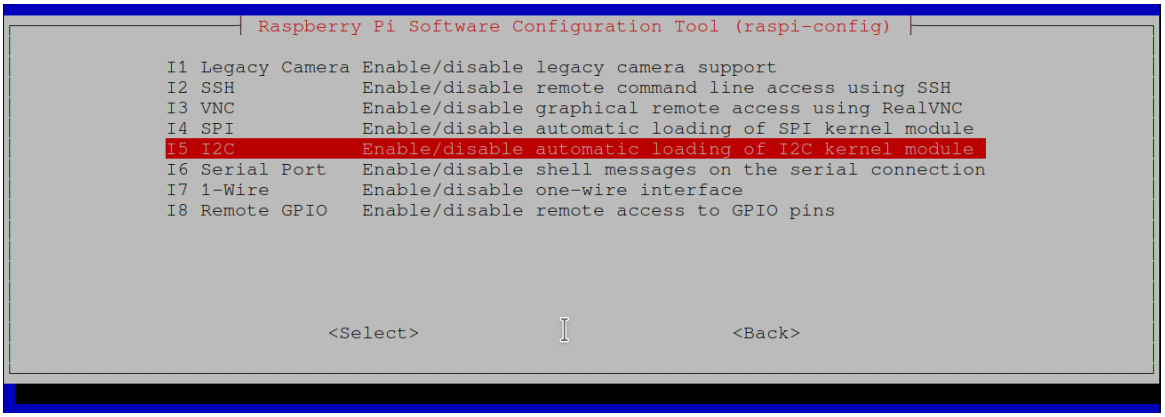
Im Hauptmenü wählen wir den Punkt 3 'Interface Options' aus.
Nach dem Bestätigen mit der 'Enter'-Taste öffnet sich das Untermenü:



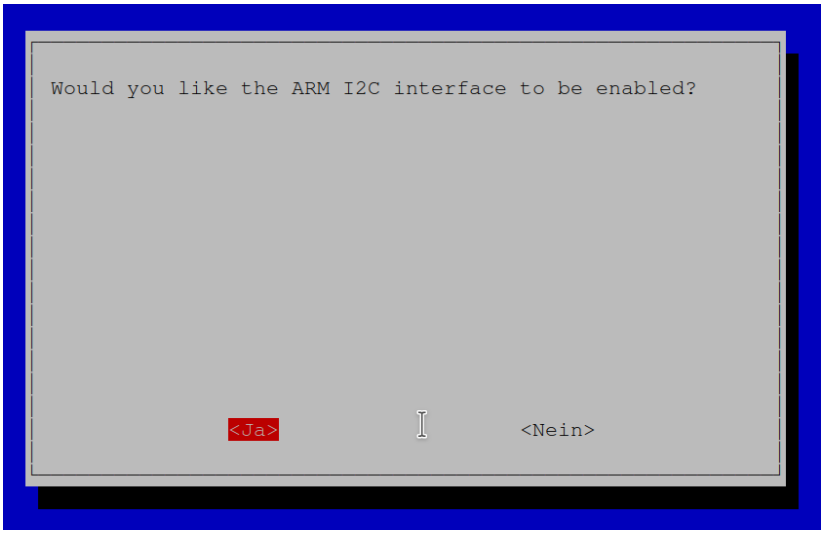
Von den vielen unterschiedlichen Optionen hier interessieren uns im Rahmen dieser Anleitung zunächst zwei. Unter I2 'SSH' kann der SSH Server aktiviert werden. Das haben wir aber bereits erfolgreich mit der SD-Karten-Vorbereitung gemacht.
Wir müssen aber noch mit I5 'I2C' die I²C Schnittstelle aktivieren. Diese ist für den PiLogger notwendig.

2.6.4.1 I5 I2C

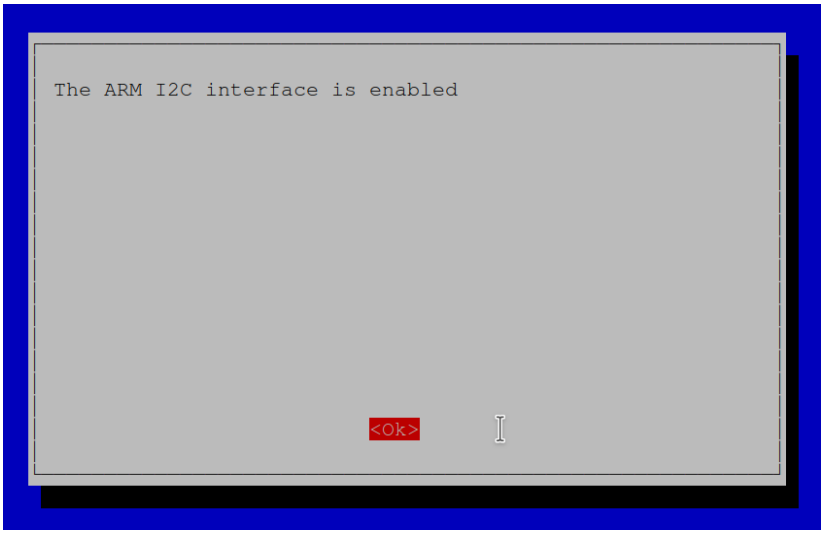
Zum Aktivieren der I²C Schnittstelle navigieren wir in die Zeile I5:



Nach einem Druck auf 'Enter' werden wir gefragt ob wir das 'ARM I2C interface' aktivieren wollen:



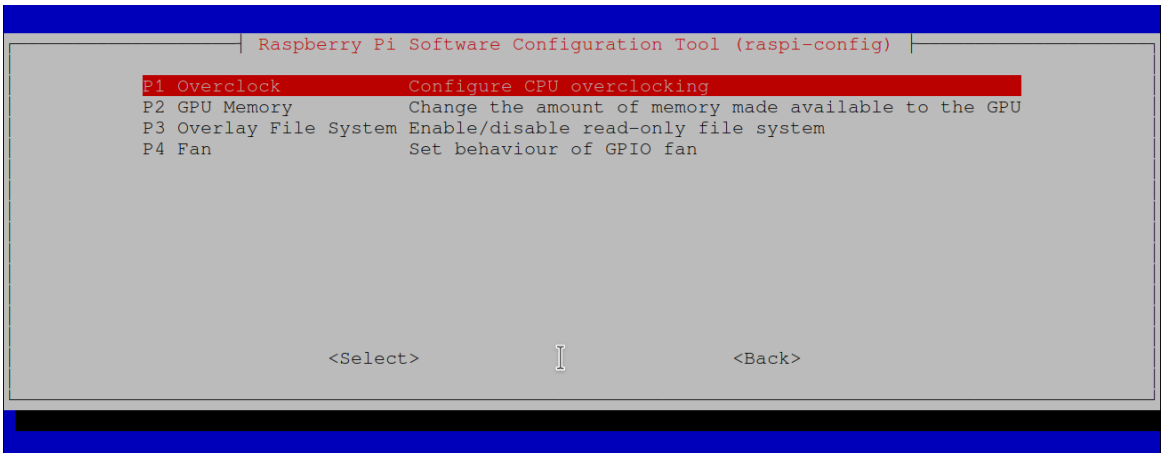
Die Voreinstellung ist <No>, weil diese Schnittstelle standardmäßig deaktiviert ist. Wir wechseln also mit der 'Pfeil links'-Taste auf <Yes> und drücken auf 'Enter'. Danach bekommen wir die Bestätigungsmeldung:



Nach einem Druck auf 'Enter' werden wir wieder in das Hauptmenü versetzt.

2.6.5 Performance Options

Unter dem Menüpunkt 4 können Einstellungen für Leistung des Raspberry vorgenommen werden.
Wir navigieren also auf die Zeile 4 und drücken die 'Enter'-Taste bekommen wir dieses Untermenü:



Dieses Menü ermöglicht verschiedene Einstellungen des Raspberry Pi rund um das Thema 'Leistung'. Tatsächlich haben wir hier im Moment nichts zu tun.
Mit zweimal 'Tab' springen wir zu <Back> und mit 'Enter' springen wir wieder zurück zum Hauptmenü.

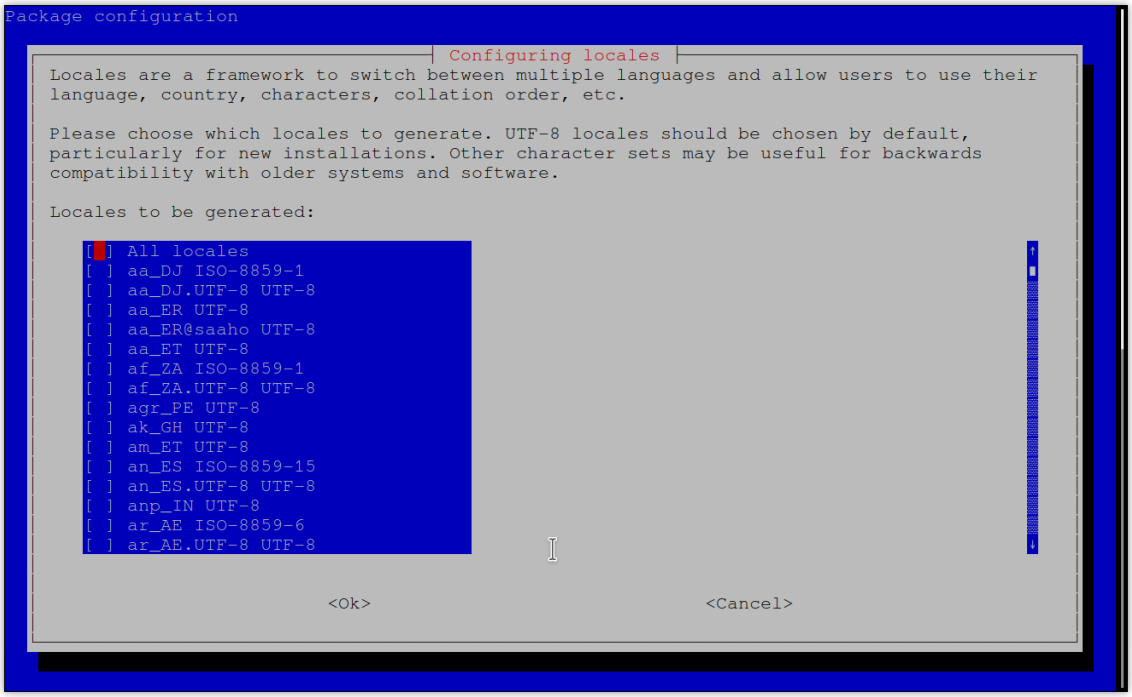
2.6.6 Localisation Options

Wir gehen nun auf die Zeile 5 'Localisation Options'. Nach dem Auswählen mit der 'Enter'-Taste bekommen wir dieses Untermenü:

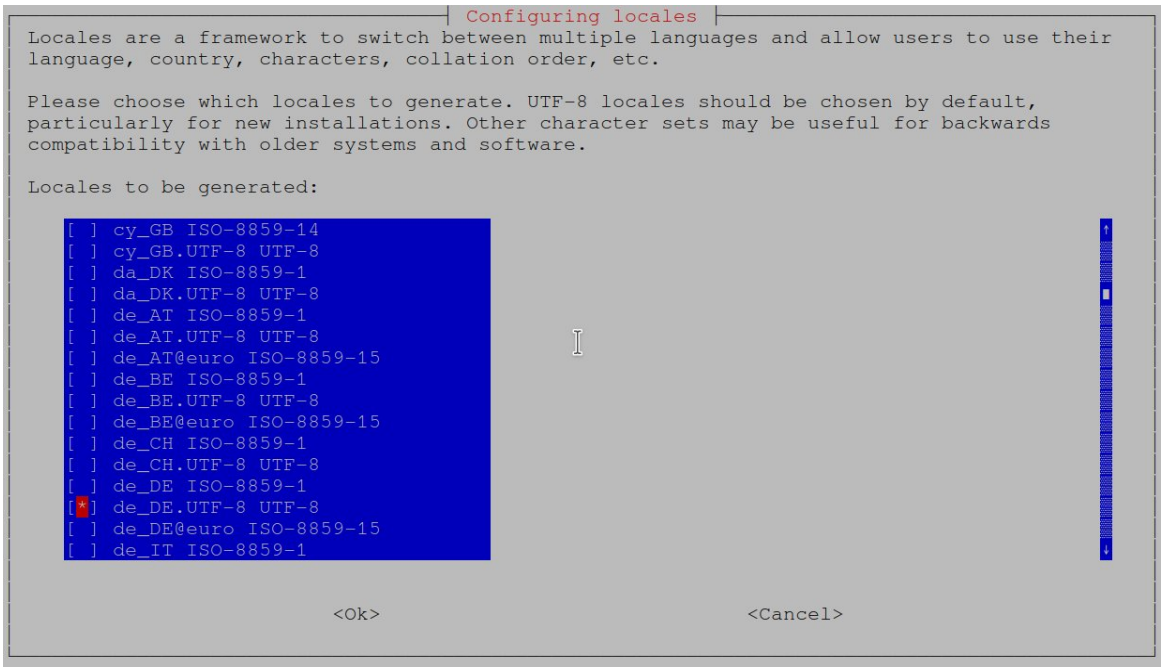
Wie oben bereits erwähnt, funktioniert die Länderauswahl im 'Imager' nicht vollständig. Die Punkte 'Timezone' und 'WLAN Country' sind allerdings erledigt. Wir müssen uns hier also um L1 und L3 kümmern.

2.6.6.1 L1 Locale

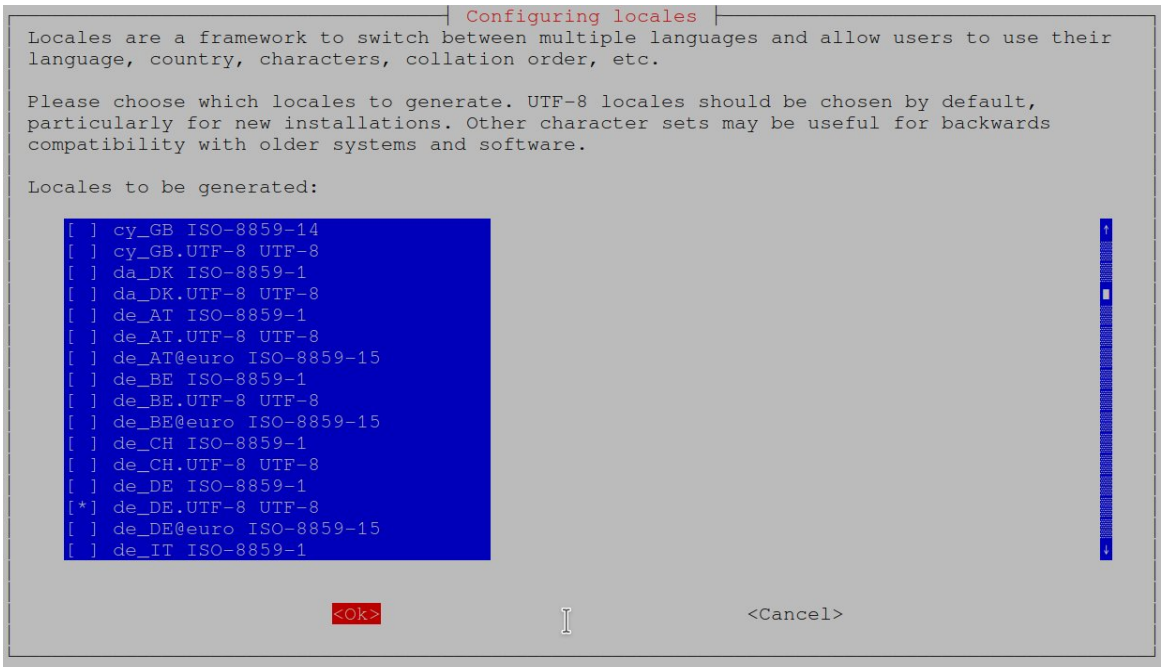
Die Option L1 ermöglicht das Einrichten der länderspezifischen Einstellungen des Systems, also Auswahl des Zeichensatzes, der Sprache und Anzeigenformate wie etwa Datum und Uhrzeit. Diese Ländereinstellungen werden 'Locale' genannt. Standardmäßig ist lediglich Englisch sowie ein Basissatz eingerichtet.
Nach einem Druck auf 'Enter' bekommen wir eine lange Liste von möglichen Ländereinstellungen zur Auswahl angeboten:



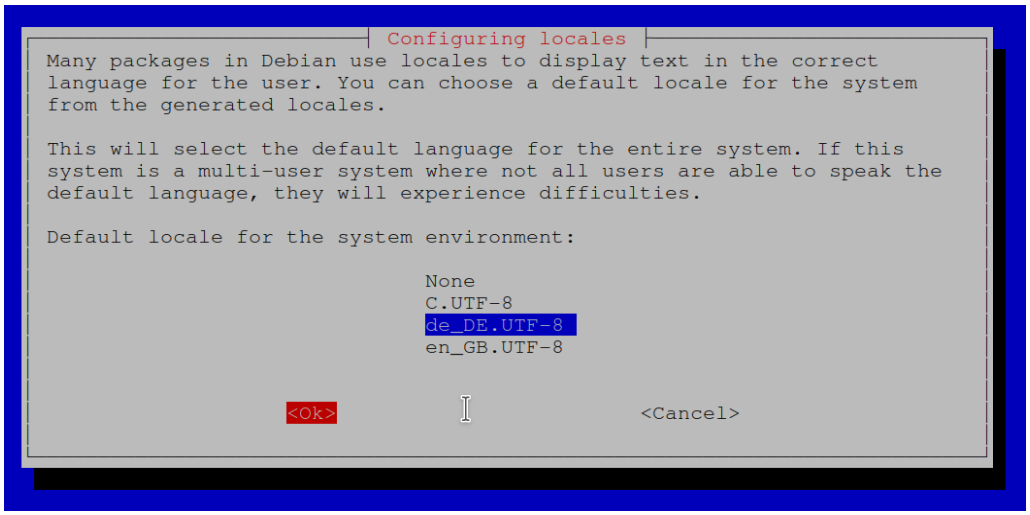
Grundsätzlich können entweder alle (All locales) oder einzelne zur weiteren Einrichtung ausgewählt werden. Alle auszuwählen würde nicht nur eine lange Einrichtungszeit bedeuten, es würde auch im Betrieb unnötig Ressourcen binden und einige Vorgänge verlangsamen.
Hier empfiehlt es sich genau eine weitere Landeseinstellung hinzuzufügen – in unserem Fall 'de_DE.UTF-8 UTF-8'.
Das ist die Sprache 'Deutsch' für das Land 'Deutschland' mit der Zeichensatz-Kodierung UTF-8.
Zu diesem Eintrag navigieren wir mit der 'Pfeil abwärts'-Taste:



Das Auswählen geschieht hier mit der Leertaste (Space), woraufhin der Eintrag ein Sternchen für den Status 'gewählt' bekommt:
Genauso können jetzt noch weitere Landessätze bei Bedarf ausgewählt werden. Dies ist, wie gesagt, in unserem Fall nicht notwendig.
Wir gehen also nun mit der **'Tab'-Taste** auf das Aktionsfeld <Ok>, welches dabei rot hinterlegt wird:



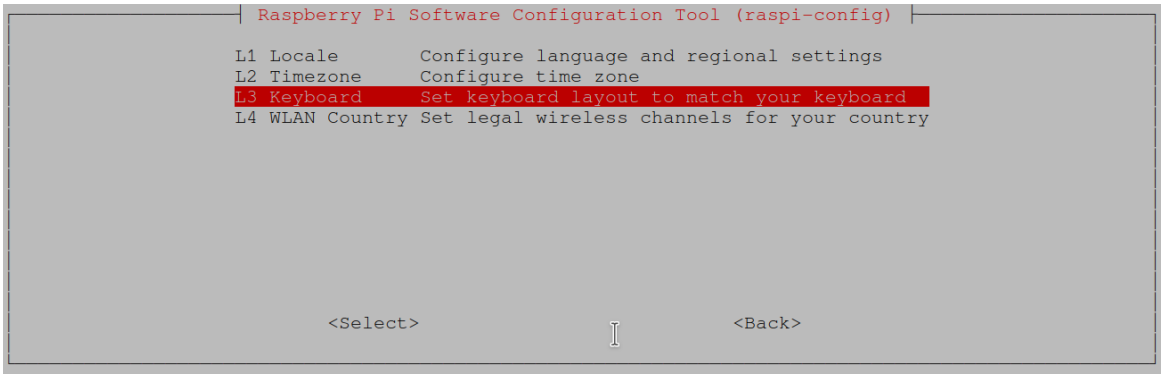
Mit einem Druck auf 'Enter' wird diese Eingabe nun abgeschlossen.
Das nächste Fenster fordert uns nun auf die bevorzugte Ländereinstellung unter den aktivierten auszuwählen:



Wir wählen also 'de_DE.UTF-8' mit der 'Pfeil abwärts'-Taste, gehen auf <Ok> mit der 'Pfeil rechts'-Taste und Drücken 'Enter'.
Jetzt wechselt die Anzeige wieder auf das Konsolenfenster und die Verarbeitung, das heißt das Einrichten, dauert eine kleine Weile.
Danach springt die Anzeige wieder in das Hauptmenü von 'Raspi-Config'.

2.6.6.2 L3 Keyboard

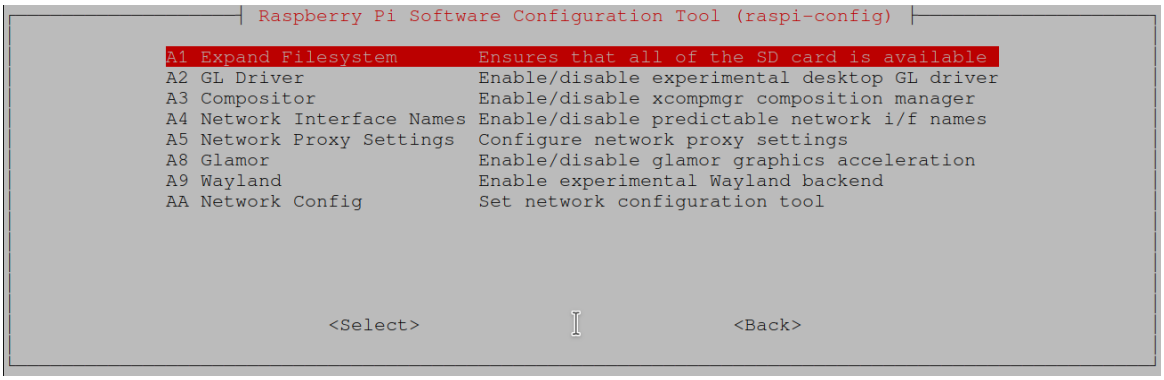
Wir gehen also noch einmal in das Menü 5 'Localisation Options'.
Mit dem Menüpunkt L3 wird das Tastatur-Layout den bisher gemachten Ländereinstellungen angepasst. Dazu müssen wir diese Option einmal ausführen.
Wir navigieren also wie gewohnt auf den Punkt L3:



Wir starten diese Aktion also mit einem 'Enter'-Tastendruck und werden auf die Konsolen-Anzeige geschaltet, wo die Meldung 'Reloading keymap. This may take a short while' um etwas Geduld bittet. Danach werden wir wieder auf das Hauptmenü zurückgesetzt.

2.6.7 Advanced Options

Auch wenn wir in diesem Untermenü derzeit keine Notwendigkeit haben etwas zu tun, werfen wir einen Blick auf die Möglichkeiten zur Information:



Der Punkt A1 'Expand Filesystem' war bisher sinnvoll – sollte aber jetzt durch die 'Raspberry Pi Imager' Konfiguration bereits erledigt sein.
Die anderen Optionen beschäftigen sich mit Desktop-Grafik und Netzwerk-Einstellungen, die in Spezialfällen wichtig sein können.
Wir sind hier fertig und wechseln mit 'Tab' auf <Back> und kehren mit 'Enter' zurück ins Hauptmenü.

2.6.8 Raspi-Config beenden

Da wir jetzt alle notwendigen Einstellungen vorgenommen haben, können wir das Konfigurationsprogramm beenden. Dazu wechseln wir mit der 'Pfeil rechts'-Taste auf das Aktionsfeld <Finish> und drücken 'Enter'.

Für die weiteren Aktionen ist es sinnvoll diesen Neustart jetzt durchzuführen.
Wir geben also im Konsolenfenster ein:

```
sudo reboot now
```

Die Eingabe schliessen wir mit 'Enter' ab. Da der Raspberry jetzt herunterfährt um neu zu starten, wird die Verbindung zu Putty getrennt:

Wir müssen jetzt eine Weile warten, bis der Raspberry neu gestartet ist und können uns dann mit Putty wieder neu verbinden.

2.7 Allgemeines Update

Um die neue Raspberry Pi OS-Installation komplett zu machen, sollte noch ein normales Update durchgeführt werden. Je nachdem, wie alt das Image ist, sind in der Zwischenzeit weitere Verbesserungen durchgeführt worden, die teilweise auch sicherheitsrelevant sein können. Diese werden so schnell wie möglich verteilt, also auch außer der Reihe von kompletten Installations-Images.

Die folgenden Aktionen setzen eine funktionierende Internet-Verbindung voraus.

2.7.1 Firmware Update

Ein Firmware Update wird bei den neueren Versionen von Raspberry Pi OS ***nicht*** mehr empfohlen.
Das Firmware Update soll nur noch in Ausnahmefällen durchgeführt werden:

```
Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Tue Sep 20 01:32:58 2022 from 192.168.178.29
karl@pilogger-test:~ $ sudo rpi-update
 *** Raspberry Pi firmware updater by Hexxeh, enhanced by AndrewS and Dom
 *** Performing self-update
 *** Relaunching after update
 *** Raspberry Pi firmware updater by Hexxeh, enhanced by AndrewS and Dom
 *** We're running for the first time
 *** Backing up files (this will take a few minutes)
 *** Backing up firmware
 *** Backing up modules 5.15.61+
#############################################################
WARNING: This update bumps to rpi-5.15.y linux tree
See: https://forums.raspberrypi.com/viewtopic.php?t=322879

'rpi-update' should only be used if there is a specific
reason to do so - for example, a request by a Raspberry Pi
engineer or if you want to help the testing effort
and are comfortable with restoring if there are regressions.

DO NOT use 'rpi-update' as part of a regular update process.
#############################################################
Would you like to proceed? (y/N)
```

Wir überspringen also diesen Schritt und fahren fort mit dem Distributions-Update.

2.7.2 Distributions-Update

Nun kann ein allgemeines Upgrade der Raspberry Pi OS-Distribution vorgenommen werden. Dazu muss die lokale Datenbank der installierten und verfügbaren Software-Pakete aufgefrischt werden.
Wir geben also folgende Zeile ein:

```
sudo apt-get update
```

Hier wird mit 'super user'-Rechten das Tool 'apt-get' aufgerufen und angewiesen ein Update der Paket-Datenbank auszuführen. Dieser Vorgang dauert nicht lange. Die Ausgabe sieht etwa so aus:

```
login as: karl
karl@pilogger-test's password:
Linux pilogger-test 5.15.61+ #1579 Fri Aug 26 11:08:59 BST 2022 armv6l

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Tue Sep 20 00:31:19 2022 from 192.168.178.29
karl@pilogger-test:~ $ sudo apt-get update
Holen:1 http://archive.raspberrypi.org/debian bullseye InRelease [23,7 kB]
Holen:2 http://raspbian.raspberrypi.org/raspbian bullseye InRelease [15,0 kB]
Holen:3 http://raspbian.raspberrypi.org/raspbian bullseye/main armhf Packages [13,2 MB]
Holen:4 http://archive.raspberrypi.org/debian bullseye/main armhf Packages [301 kB]
Es wurden 13,6 MB in 28 s geholt (491 kB/s).
Paketlisten werden gelesen… Fertig
karl@pilogger-test:~ $
```

Nun geben wir die folgende Textzeile ein:

```
sudo apt-get dist-upgrade
```

Damit wird jetzt das tatsächliche Update der Raspberry Pi OS-Installation gestartet. Zunächst wird ermittelt, welche Pakete geändert werden müssen. Dann wird abgefragt, ob diese Änderungen tatsächlich installiert werden sollen:

```
Es wurden 13,6 MB in 28 s geholt (491 kB/s).
Paketlisten werden gelesen… Fertig
karl@pilogger-test:~ $ sudo apt-get dist-upgrade
Paketlisten werden gelesen… Fertig
Abhängigkeitsbaum wird aufgebaut… Fertig
Statusinformationen werden eingelesen… Fertig
Paketaktualisierung (Upgrade) wird berechnet… Fertig
Die folgenden NEUEN Pakete werden installiert:
  python3-toml
Die folgenden Pakete werden aktualisiert (Upgrade):
  avahi-daemon base-files curl dpkg dpkg-dev libavahi-common-data libavahi-common3
  libavahi-core7 libcurl3-gnutls libcurl4 libdpkg-perl libgssapi-krb5-2 libk5crypto3
  libkrb5-3 libkrb5support0 libpam-systemd libpcre2-8-0 libpcre2-posix2 libsystemd0
  libudev1 publicsuffix raspberrypi-sys-mods systemd systemd-sysv systemd-timesyncd
  tzdata udev
27 aktualisiert, 1 neu installiert, 0 zu entfernen und 0 nicht aktualisiert.
Es müssen 15,7 MB an Archiven heruntergeladen werden.
Nach dieser Operation werden 73,7 kB Plattenplatz zusätzlich benutzt.
Möchten Sie fortfahren? [J/n] j
```

Nach der Eingabe von 'J' mit Abschluss durch 'return' beginnt der Hauptteil des Updates. Je nachdem wie frisch das verwendete Raspberry Pi OS Image ist, müssen eventuell etliche Pakete upgedatet werden. Das kann dann schon mal bis

zu einer Stunde dauern. Das gezeigte Beispiel ist mit einem 3 Monate alten Image gemacht, mit relativ überschaubarem Umfang (28 zu aktualisierende Pakete).

Das Update ist vollständig durchgeführt, wenn der Kommando-Zeilen-Prompt wieder erscheint. Es sollte nun einmal neu gestartet werden:

```
sudo reboot
```

Damit ist der Raspberry Pi grundsätzlich eingerichtet.

2.8 Montieren des PiLogger One

Alle bisherigen Schritte können gefahrlos auch mit bereits montiertem PiLogger One erfolgen. Falls der PiLogger One noch nicht auf dem Raspberry Pi montiert sein sollte, ist jetzt ein guter Zeitpunkt.
Für eine sichere Montage muss der Raspberry stromlos sein. Es ist ebenfalls ratsam die SD-Karte während der Handhabung nicht im Halter zu haben, um sie oder den Halter nicht zu beschädigen.

Für eine bebilderte Schritt-für-Schritt-Anleitung der Montage bitte im 'PiLogger One Handbuch' das Kapitel 5 nachlesen.

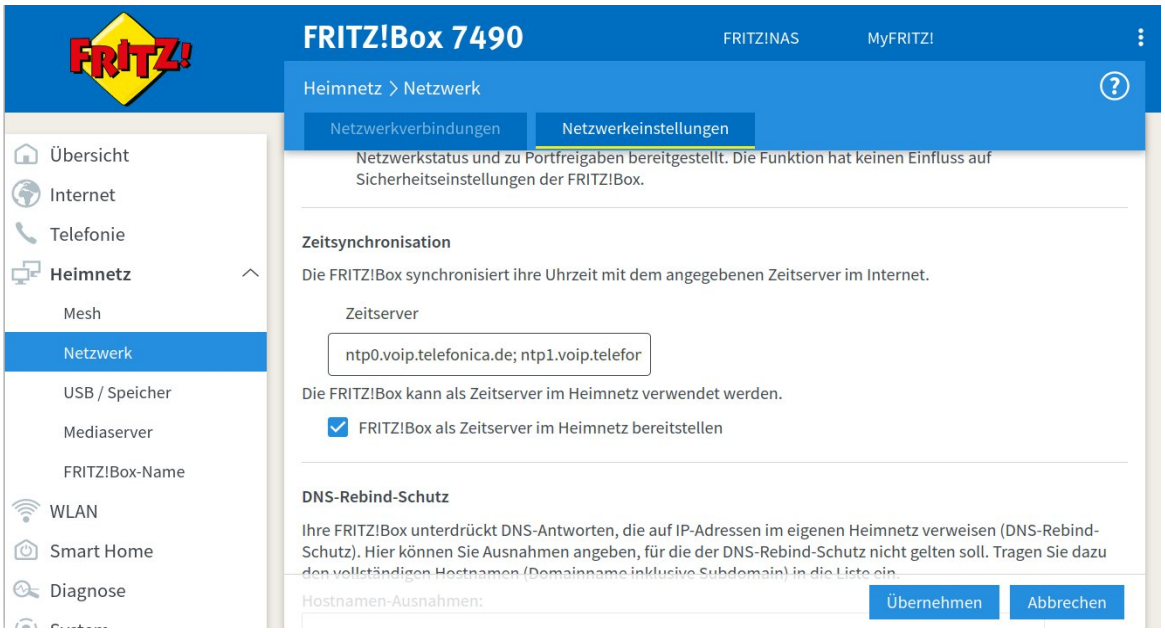
3 PiLogger WebMonitor installieren

Die Installation des PiLogger WebMonitor wird durch ein interaktives Installations-Programm (Shell-Skript) – den 'Installer' durchgeführt.
Der Installer lässt sich einfach, wie unten beschrieben, mit einer Zeile in der SSH-Konsole starten. Da dies eine ausführbare Datei ist, die aus dem Internet geladen wird, kann man den Installer auch separat von unserem Server herunterladen, begutachten und manuell starten:
https://www.pilogger.de/index.php/de/download-de/category/2-software

→ Eintrag : Installer PiLo-WebMon
Der Installer führt alle notwendigen Schritte aus und bietet optionale Einstellungen.
Dazu gehört auch die Möglichkeit die Uhrzeit-Synchronisation mit einem Internet-Zeitserver umzulenken auf den eigenen Heimnetzrouter. Diese Zeitsynchronisation ist notwendig, weil der Raspberry Pi keine eigene Echtzeituhr (RTC) besitzt.
Allerdings ist dies die einzige externe Internet-Verbindung die benötigt wird und so verhindert, dass dem Raspberry Pi der Zugriff auf das öffentliche Internet im Router gesperrt werden kann. Wenn der eigene Router die Möglichkeit bietet im Heimnetz einen NTP-Server bereitzustellen, kann auch diese Verbindung intern gehalten werden und der Internet-Zugang für den Raspberry kann gesperrt werden. Deshalb wird als nächstes diese optionale Vorbereitung beschrieben.

3.1 Router als NTP-Server einrichten

Internet-Router synchronisieren ihre eigene Zeit mit NTP-Servern im Internet. Bei Telefonie per VOIP stellt in der Regel der Anbieter seine eigenen Zeitserver im Router ein. Bei den meisten WLAN-Routern lässt sich zusätzlich einstellen, dass der Router selber als Zeitserver im eigenen Netzwerk zur Verfügung steht.
Als Beispiel zeigen wir hier die Einstellungsseite einer Fritz-Box. Hier meldet man sich auf der Konfigurationsseite des Routers an und navigiert über den Menüpunkt 'Heimnetz' in das Untermenü 'Netzwerk'. Dort wechselt man in das zweite 'Tab' (Karteikarte) 'Netzwerkeinstellungen'.
Hier gibt es, ziemlich weit unten, die Möglichkeit 'weitere Einstellungen' auszuklappen. Dort versteckt sich die Rubrik 'Zeitsynchronisation' und dort die Einstellmöglichkeit „Fritz!Box als Zeitserver im Heimnetz bereitstellen“.
Diese Option aktivieren wir und drücken anschließend auf den Knopf unten 'Übernehmen':



Das war's.

3.2 Installer starten

Um die Installation zu starten, nehmen wir mittels 'Putty' Verbindung zum Raspberry auf.
Für die Installation braucht der Raspberry eine funktionierende Internet-Verbindung.
In der Kommando-Zeile geben wir dann exakt diese Zeile ein:

```
curl -sSL https://www.pilogger.de/get/installer-webmon | bash
```

Die Eingabe schließen wir mit 'Enter' ab.

```
login as: karl
karl@pilogger-test's password:
Linux pilogger-test 5.15.68+ #1586 Fri Sep 16 12:04:27 BST 2022 armv6l

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Wed Sep 21 18:33:53 2022 from 192.168.178.29
karl@pilogger-test:~ $ curl -sSL https://www.pilogger.de/get/installer-webmon | bash

Dieses Script installiert die PiLogger WebMonitor Software
im Verzeichnis /home/karl/pilogger

Wollen Sie fortfahren ? [j/N]
```

Das Hilfsprogramm 'curl' lädt das Script von der PiLogger.de Website und startet seine Ausführung. Um einen Abbruch bei versehentlichem Aufruf zu ermöglichen, muss der Installationsstart bestätigt werden – mit der Taste 'j'.

Danach werden wir gefragt, ob wir den Heimnetz-Router als Zeitserver verwenden wollen:

```
login as: karl
karl@pilogger-test's password:
Linux pilogger-test 5.15.68+ #1586 Fri Sep 16 12:04:27 BST 2022 armv6l

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Wed Sep 21 18:33:53 2022 from 192.168.178.29
karl@pilogger-test:~ $ curl -sSL https://www.pilogger.de/get/installer-webmon | bash

Dieses Script installiert die PiLogger WebMonitor Software
im Verzeichnis /home/karl/pilogger

Wollen Sie fortfahren ? [j/N] j

Wenn Ihr Heimnetz-Router einen NTP-Zeitserver zur Verfügung stellt,
können Sie die Zeit des Raspberry Pi damit synchronisieren.
Dadurch wird hierfür kein externer Internetzugang mehr benötigt.

Heimnetz-Router als Zeitserver verwenden ? [j/N] j
```

Wenn wir den Router wie in 3.1 entsprechend eingerichtet haben, können wir hier 'j' antworten.

Als Nächstes wird ein alternativer I²C Treiber aktiviert.

```
login as: karl
karl@pilogger-test's password:
Linux pilogger-test 5.15.61+ #1579 Fri Aug 26 11:08:59 BST 2022 armv6l

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Wed Sep 21 23:57:06 2022 from 192.168.178.29
karl@pilogger-test:~ $ curl -sSL https://www.pilogger.de/get/installer-webmon | bash

Dieses Script installiert die PiLogger WebMonitor Software
im Verzeichnis /home/karl/pilogger

Wollen Sie fortfahren ? [j/N] j

Wenn Ihr Heimnetz-Router einen NTP-Zeitserver zur Verfügung stellt,
können Sie die Zeit des Raspberry Pi damit synchronisieren.
Dadurch wird hierfür kein externer Internetzugang mehr benötigt.

Heimnetz-Router als Zeitserver verwenden ? [j/N] j

192.168.178.1 als Zeitserver gesetzt


i2c_arm=on Parameter bereits aktiv

Overlay i2c-gpio,i2c_gpio_sda=2,i2c_gpio_scl=3,i2c_gpio_delay_us=2,bus=1 zu /boot/config.txt hinzuge
fügt
```

Die I²C Hardware-Module der Broadcom Prozessoren haben Fehler, weshalb mit den Standardtreibern das I²C Bus Verhalten 'Clock Stretching', (also der Busmaster Raspi wartet auf das Peripheriegerät, das um kurze Pause bittet) nicht funktioniert. Der PiLogger und andere I²C Module brauchen aber dieses Feature.

Deshalb wird hier ein I²C-Modul installiert, das den I²C Port in Software auf den richtigen Pins nachbildet. Das funktioniert für alle Raspberry Pi’s von Zero bis Modell 4 sehr gut.

Danach werden automatisch diese Aktionen durchgeführt:

- Installieren von 'python3-smbus'
  (Python Erweiterung für die I²C Schnittstelle)
- Installieren von 'python3-rpi.gpio'
  (Python Erweiterung für den direkten GPIO Zugriff)
- Installieren von 'python3-bottle'
  (WSGI Micro Web Framework für Python)
- Herunterladen und Entpacken der PiLogger WebMonitor Software-Komponenten
- Einrichten des Autostarts der WebMonitor Software

Im Anschluss wird noch eine Option angeboten, mit der die täglichen Kontakte zu den Raspberry Pi OS-Update-Servern abgeschaltet werden können:

```
Archive:  PiLo-WebMon.zip
   creating: templates/
  inflating: templates/.directory
  inflating: templates/index.html
   creating: static/
  inflating: static/Iconset.png
  inflating: static/favicon.ico
  inflating: static/einstell2.html
  inflating: static/diagramme.html
 extracting: static/.directory
  inflating: static/dygraph_p.css
  inflating: static/w3_pilo.css
  inflating: static/kalibration.html
  inflating: static/dygraph_p.min.js
  inflating: static/download.html
  inflating: static/einstell1.html
 extracting: Version.txt
  inflating: PiLogger-bottle.py
  inflating: pilo-webmon.service
  inflating: PiLo_Thermistor.py


Created symlink /etc/systemd/system/multi-user.target.wants/pilo-webmon.service → /lib/systemd/system
/pilo-webmon.service.

Autostart eingerichtet


Die täglichen Kontakte zum Raspbian-Update-Server (apt-daily)
sind externe Internetkontakte und können deaktiviert werden.

Möchten Sie Apt-daily deaktivieren ? [j/N]
```

Das Abschalten ist nicht unbedingt notwendig, wenn dem Raspberry der Internetzugang gesperrt wird, weil die Anfragen dann einfach ins Leere gehen. Wenn der Internetzugang nicht gesperrt wird, erzeugen diese Anfragen eine zunächst harmlose Nutzerstatistik, machen andererseits aber auch nur wirklich Sinn auf einem Raspberry der noch andere Internet-Aufgaben hat – zum Beispiel auf einem Desktop-System (auf dem dann automatische Updates eingerichtet werden sollten [Unattended Updates]).
Ansonsten kann der Raspberry völlig privat im eigenen Heimnetz als Logger arbeiten und bietet ohne diese Brotkrumen auch keine besondere Angriffsfläche und somit keinen dringenden Bedarf an Updates – einen guten und sicheren Router vorausgesetzt.

Wenn diese letzte Wahl mit 'j' beantwortet wird, sieht das Ergebnis etwa so aus:

```
Möchten Sie Apt-daily deaktivieren ? [j/N] j
Removed /etc/systemd/system/timers.target.wants/apt-daily.timer.
● apt-daily.timer - Daily apt download activities
     Loaded: loaded (/lib/systemd/system/apt-daily.timer; disabled; vendor preset: enabled)
     Active: inactive (dead) since Thu 2022-09-22 00:10:22 CEST; 4s ago
    Trigger: n/a
   Triggers: ● apt-daily.service

Sep 06 21:08:24 pilogger-test systemd[1]: Started Daily apt download activities.
Sep 22 00:10:22 pilogger-test systemd[1]: apt-daily.timer: Succeeded.
Sep 22 00:10:22 pilogger-test systemd[1]: Stopped Daily apt download activities.
Created symlink /etc/systemd/system/apt-daily.service → /dev/null.
● apt-daily.service
     Loaded: masked (Reason: Unit apt-daily.service is masked.)
     Active: inactive (dead)
TriggeredBy: ● apt-daily.timer
Removed /etc/systemd/system/timers.target.wants/apt-daily-upgrade.timer.
● apt-daily-upgrade.timer - Daily apt upgrade and clean activities
     Loaded: loaded (/lib/systemd/system/apt-daily-upgrade.timer; disabled; vendor preset: enabled)
     Active: inactive (dead)
    Trigger: n/a
   Triggers: ● apt-daily-upgrade.service

Sep 06 21:08:24 pilogger-test systemd[1]: Started Daily apt upgrade and clean activities.
Sep 22 00:10:36 pilogger-test systemd[1]: apt-daily-upgrade.timer: Succeeded.
Sep 22 00:10:36 pilogger-test systemd[1]: Stopped Daily apt upgrade and clean activities.

Die Installation ist jetzt durchgeführt.
Einige Änderungen am System erfordern einen Neustart.
Nach dem Neustart wird automatisch PiLogger WebMon ausgeführt.

Bereit zum Neustart ? [j/N]
```

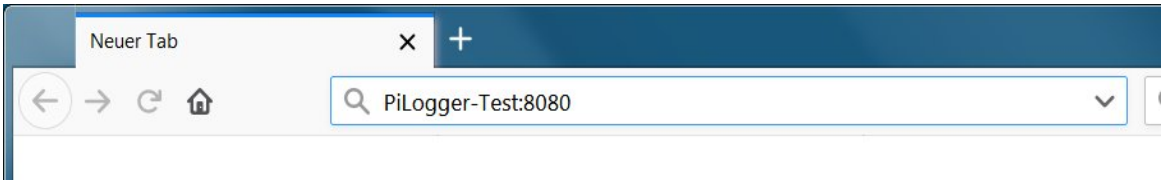
Damit ist die Installation komplett und mit einem Neustart des Raspberry kann der PiLogger WebMonitor gestartet werden.

Wir bestätigen also mit einem 'j' – und der Raspberry beendet die Verbindung und startet neu. Nach diesem Neustart können wir dann mit einem Web-Browser die erste Verbindung zum WebMonitor aufnehmen.

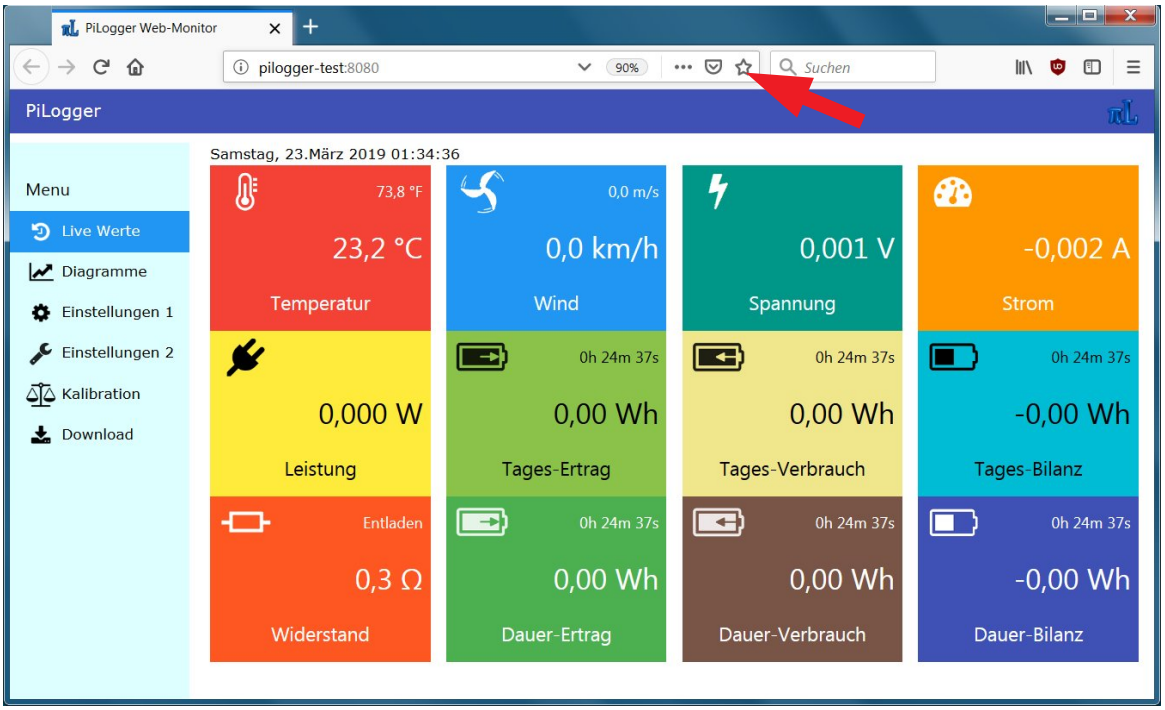
4 Bedienungsanleitung PiLogger WebMonitor

4.1 Aufruf der Web-Seite mit dem Browser

Wenn alle Vorbereitungen aus den vorherigen Kapiteln erfolgreich abgeschlossen sind, kann die Web-Seite des PiLogger WebMonitor auf einem beliebigen Gerät im Heimnetzwerk mit einem Web-Browser aufgerufen werden.
Dazu geben wir den Netzwerknamen des Raspberry gefolgt von ':8080' in die Adressleiste des Browsers ein:



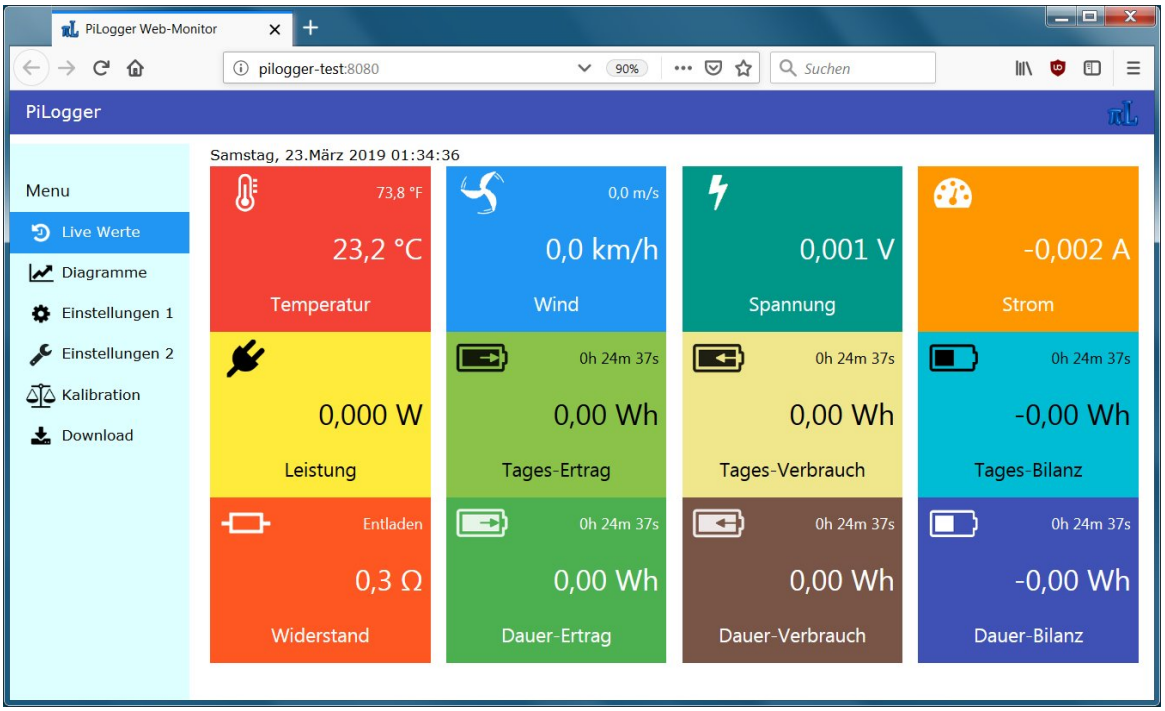
Die Eingabe müssen wir mit 'Enter' abschließen.
Die meisten Web-Browser halten sich aber strenger an die Regeln für Netzwerknamen und machen im gezeigten Beispiel automatisch eine kleingeschriebene Variante daraus und zeigen die gewünschte Seite trotzdem an:

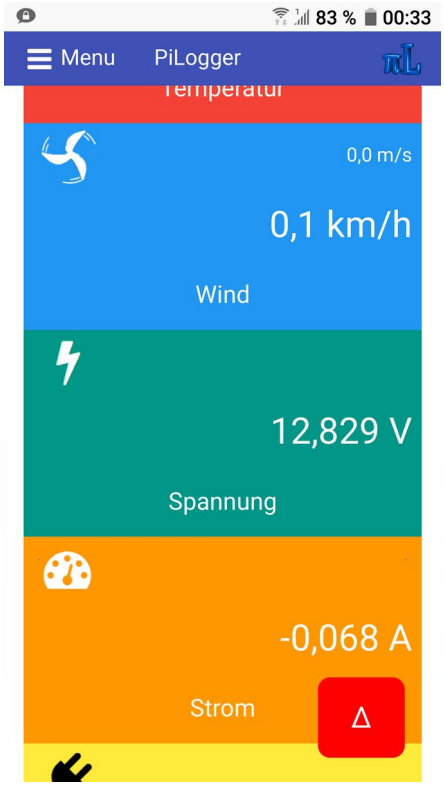


Das ist die Startseite unseres frisch aufgesetzten Servers, wenn alles wie geplant gelaufen ist. Wir können nun für den Browser einen Favoriten anlegen – in diesem Beispiel bei 'Firefox' mit einem Klick auf den Stern ganz rechts in der Adressleiste. Damit sind die weiteren Aufrufe dieser Seite mit einem Klick möglich.
Sollte es mit dem Netzwerknamen ein Problem beim Aufruf geben, kann alternativ die IP-Adresse eingegeben werden, die der Router dem Raspberry zugewiesen hat. Diese lässt sich im Router-Verwaltungsmenü nachschauen. Die Eingabe wäre dann beispielsweise:
'192.168.178.21:8080'.

4.2 Die Startseite – *Live Werte*

Die Startseite zeigt die aktuellen Mess- und Zählerwerte des PiLogger One an und ruft diese automatisch alle 2 Sekunden frisch ab.
Auf der linken Seite gibt es ein Navigationsmenü, das es ermöglicht die weiteren Seiten aufzurufen. Die jeweils aktuelle Seite ist dabei blau unterlegt. Dieses Menü verschwindet bei Geräten mit zu kleinem Bildschirm und muss dort mit einem Tippen (Tap) auf das stattdessen auftauchenden Menü-Icon extra aufgerufen werden.



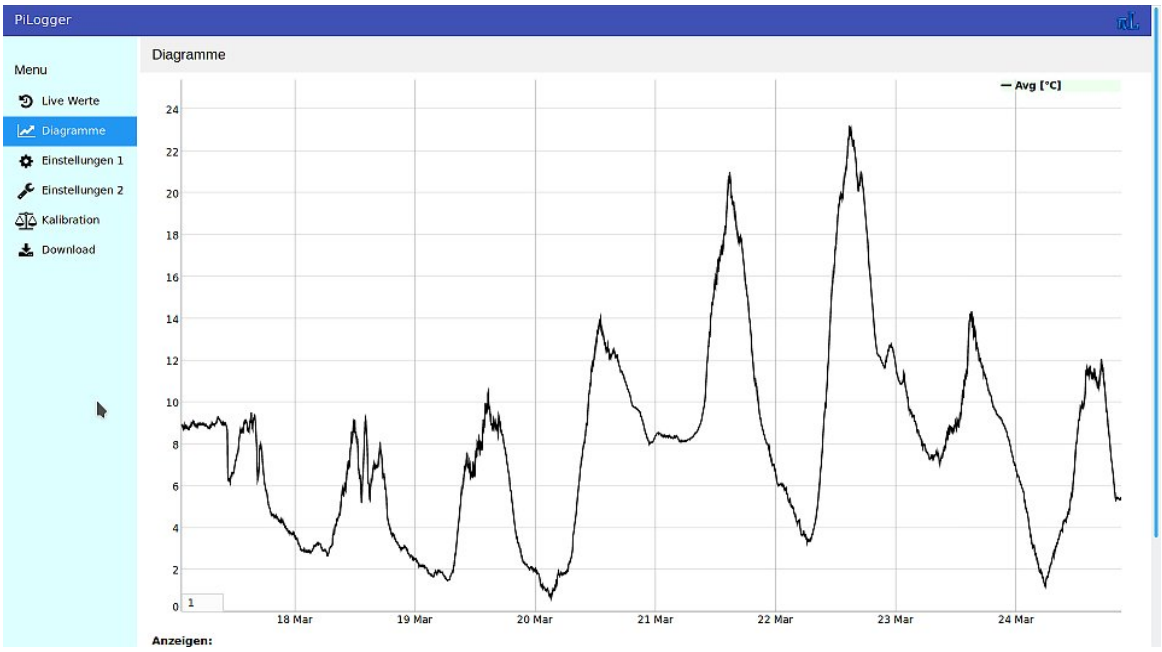


Auf kleineren Bildschirmen, wie etwa Smartphones, erscheinen die Messwertfelder als Liste. Wird in dieser Liste gescrollt (hoch gewischt), erscheint ein rotes Quadrat, das den schnellen Rücksprung zum Anfang der Liste ermöglicht.

Jedes Anzeigefeld für einen Messwert oder Zählerstand dient außerdem als Schaltfläche für einen Link auf die Zeitdiagrammdarstellung des jeweiligen Wertes.
Ein Klick oder Tap auf das Temperaturfeld ruft also die Diagramme-Seite auf und hat dabei die Anzeige der Minimum-, Mittelwert- und Maximum-Werte der Temperaturmessungen aktiviert.
Gleiches gilt sinngemäß für die anderen Messwertfelder.

4.3 Die Seite *Diagramme*

Wird die Seite 'Diagramme' über das Navigationsmenü aufgerufen, startet diese Seite mit der Darstellung der Temperatur-Mittelwerte über der Zeit:



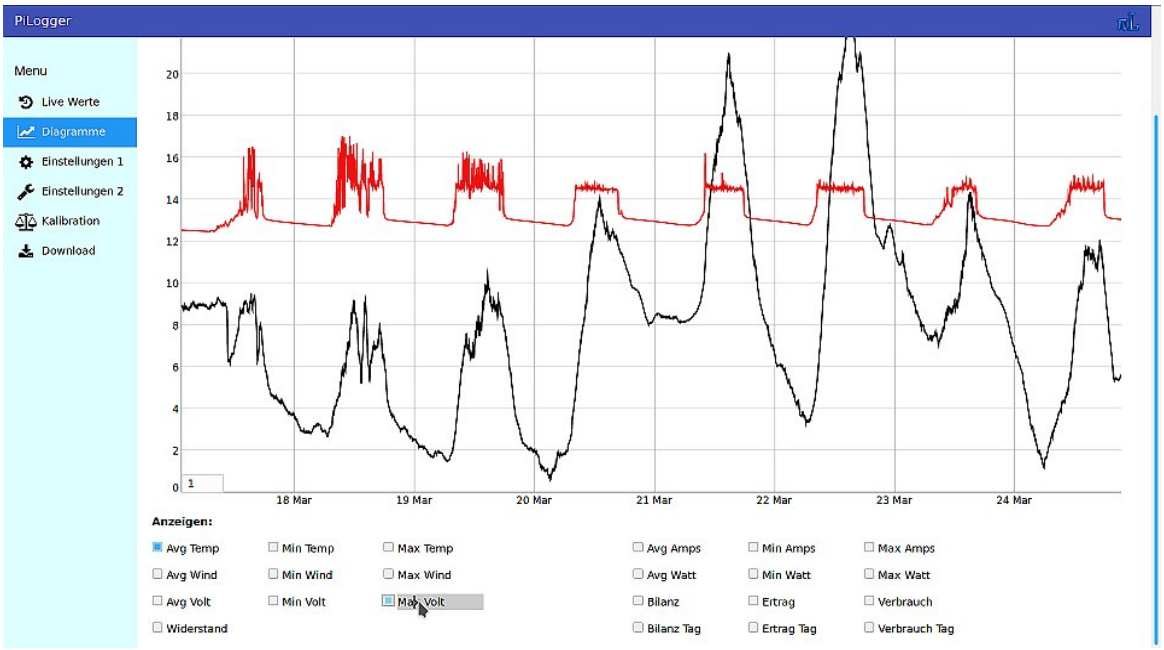
Der Diagrammbereich passt sich automatisch der verfügbaren Bildschirmfläche an. Die Achsen werden automatisch den Datenbereichen angepasst.
Die Daten werden dabei aus der aktuellen Logdatei plus dem vorherigen Logzeitraum bezogen. Dazu wird die Datei 'showdata.csv' jeweils beim Aufruf der Seite frisch vom Raspberry heruntergeladen.
Oben rechts im Diagrammbereich wird eine Legende der aktiven Kurven angezeigt und bei aktivem Marker auch die jeweiligen Einzelwerte zu einem gewählten Zeitpunkt (siehe 4.3.2).
Unten links, nahe dem Achsenschnittpunkt wird der aktuelle gleitende Mittelwertfaktor angezeigt (siehe 4.3.4).

4.3.1 Anzuzeigende Messwertreihen auswählen

Je nach Bildschirmgröße ist unterhalb der Zeitachse gerade noch der Titel 'Anzeigen:' des unterhalb folgenden Bereiches zu sehen. Durch die etwas kleinere Darstellung des Diagrammbereiches lässt sich auf Touch-Displays an der rechten Kante in einem kleinen Bereich der Seiteninhalt nach oben schieben (scroll down). Der Diagrammbereich selber ist ebenfalls berührungsempfindlich, verschiebt dort aber den Diagramminhalt selber.
Für den Fall, dass rechts kein leerer Streifen zum Schieben vorhanden ist, eignet sich der Menü-Bereich am besten – wenn das Menü eingeblendet ist.

Wir scrollen also zum unteren, verdeckten Seitenbereich und erhalten folgende

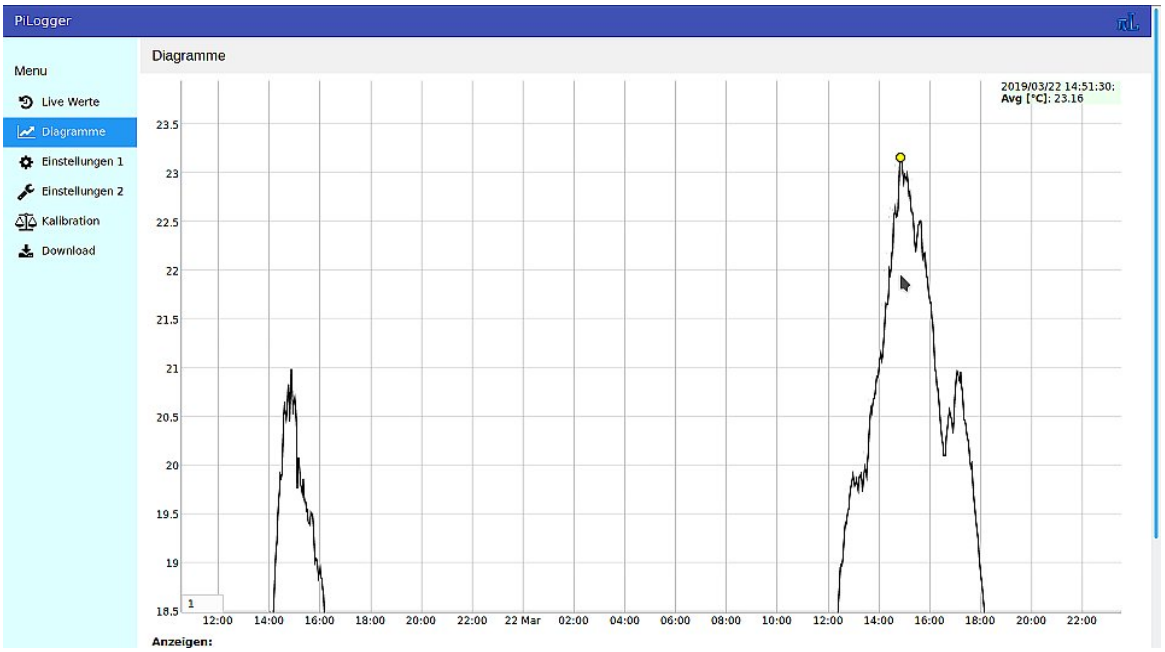
Anzeige:



Der Bereich 'Anzeigen:' bietet in 2 Spalten alle verfügbaren Messwertreihen an. Durch Klicken auf das Auswahlfeld oder den zugehörigen Text kann die Anzeige dieser Reihe ein- oder ausgeschaltet werden. Der Screenshot zeigt das zusätzliche Einschalten der Messwertreihe 'Max Volt', also der maximalen Spannungen. Damit wird das Auswahlquadrat gefüllt angezeigt und eine zusätzliche rote Kurve wird angezeigt.
Die jeweiligen Minimum-Werte werden in blau angezeigt, die Mittelwerte in schwarz und die Maximum-Werte in rot.
Die senkrechte Achse wird dabei jeweils automatisch so angepasst, dass alle Werte auch angezeigt werden. Wie in dieser Gesamtanzeige in interessante Bereich hineingezoomt werden kann wird weiter unten beschrieben (4.3.3).

4.3.2 Einzelmesswert anzeigen

Auf Bildschirmen mit Mausbedienung taucht sofort, wenn der Mauszeiger im Diagrammbereich ist, ein Marker in Form eines gelb gefüllten Kreises auf, der der horizontalen Mauszeigerposition folgt, aber auf der Messwertkurve bleibt. Werden mehrere Messwertkurven angezeigt, hat jede Kurve ihren eigenen Marker.
Die jeweiligen Einzelmesswerte am Markerpunkt werden oben rechts in der Legende angezeigt zusammen mit Datum und Zeit.
Auf Touch-Bildschirmen wird dieser Marker mit einem kurzen, leichten Tipp an eine gewünschte Stelle im Diagramm aktiviert. Der Tipp darf nicht zu lang sein und keine Bewegung enthalten, da er sonst als Geste interpretiert wird. Auch das kräftige Aufdrücken kann als Bewegung missverstanden werden.
Hier ein Beispiel für eine Mauszeiger-gesteuerte Messwertanzeige:



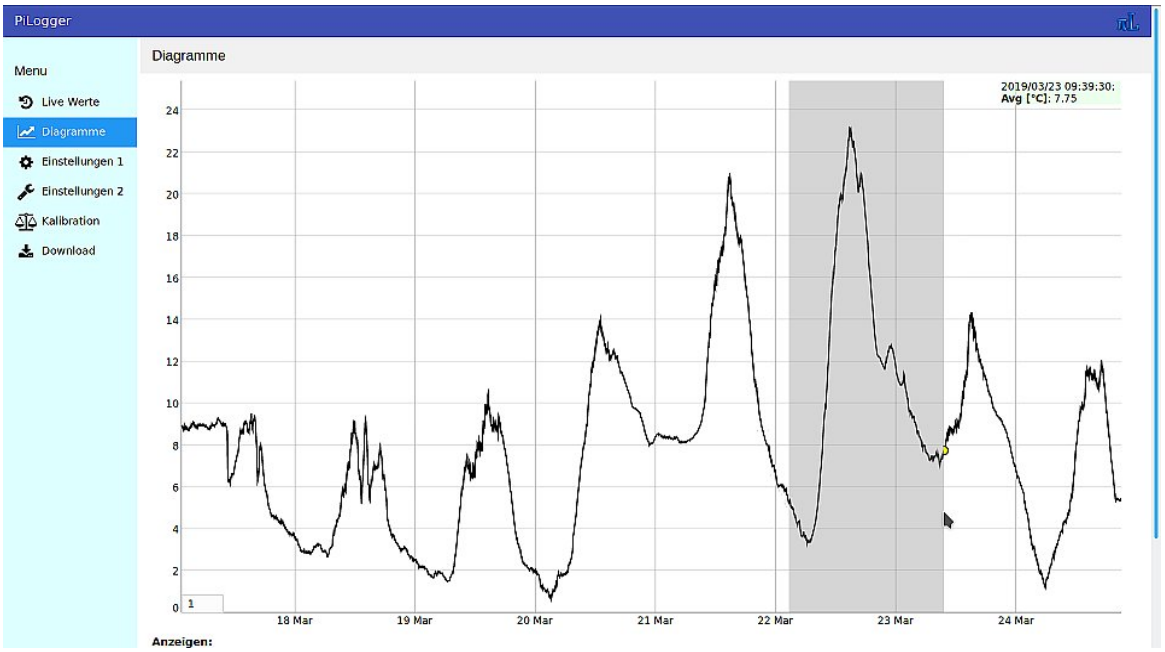
4.3.3 Verschieben und Zoomen

Der angezeigte Kurvenbereich innerhalb des Diagramms lässt sich verschieben und zoomen. Das ist sowohl mit Maus als auch mit Touch möglich, aber sehr unterschiedlich.

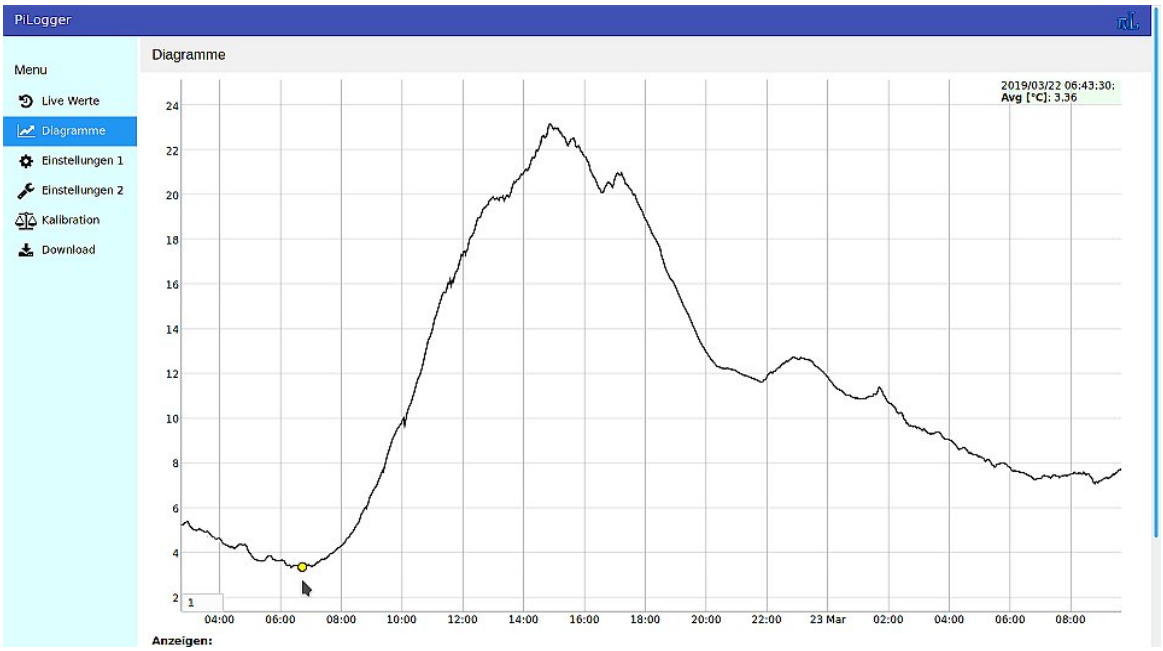
4.3.3.1 Maus-Bedienung

- <u>Zoomen</u>

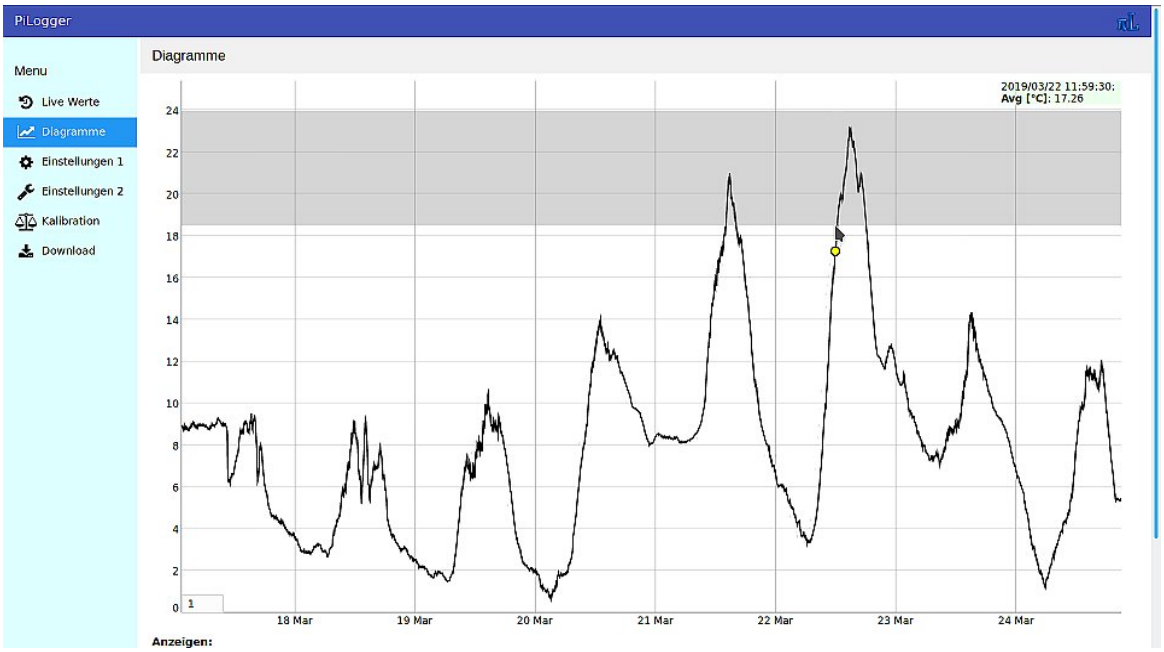
  Um mit der Maus einen Bereich zu vergrößern, fahren wir an eine Stelle für den gewünschten Start des zu vergrößernden Bereiches – drücken und halten die linke Maus-Taste und ziehen den Mauszeiger zum gewünschten Ende des Zoom-Bereiches. Dabei wird der Bereich farblich markiert. Dann lassen wir die linke Maus-Taste los und das Diagramm wir auf den gewünschten Bereich vergrößert.



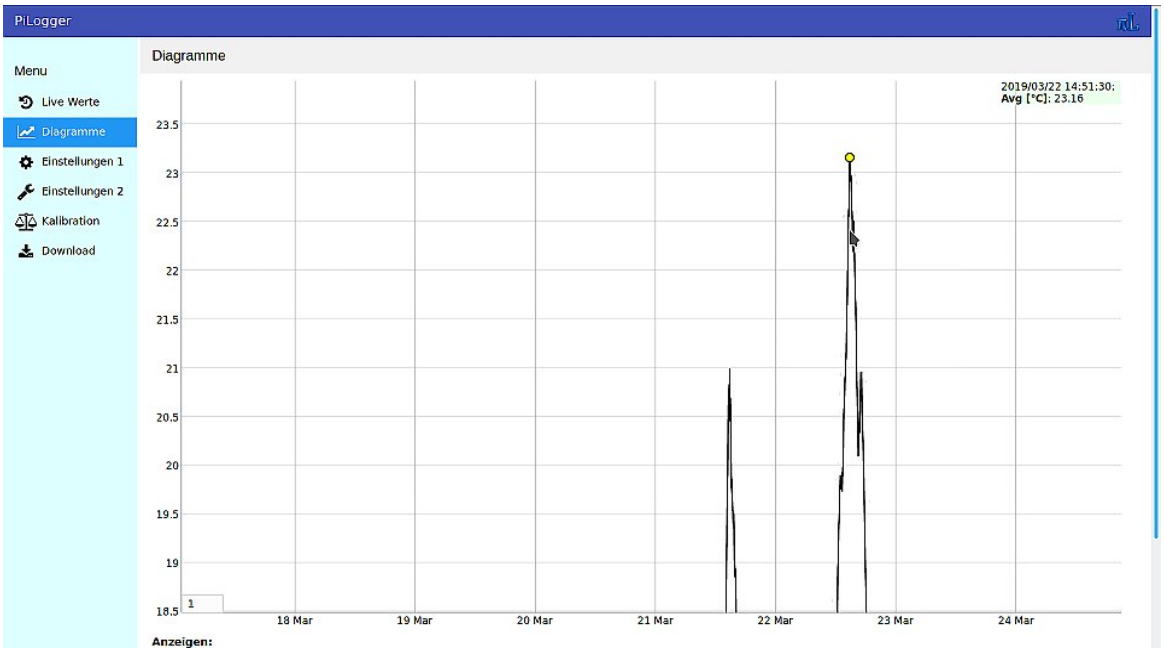
Der Screenshot zeigt das Markieren eines Zeitbereiches. Nach dem Zoomen sieht das so aus:



Analog lässt sich auch ein vertikaler Bereich, also ein Wertebereich, zoomen. Hier das vertikale Markieren des Bereiches:



Mit diesem Ergebnis:



- <u>Zoom zurücksetzen</u>

  Mit einem Doppel-Klick im Diagrammbereich wird der Zoom wieder zurückgesetzt. Es wird wieder die automatische Vollsicht über alle Daten angezeigt.

- <u>Verschieben</u>

  Um den Diagrammbereich zu verschieben (Pan), setzen wir den Mauszeiger irgendwo in den Diagrammbereich und drücke und halten die ,'Shift'-Taste. Wenn wir nun den Mauszeiger bewegen, verschieben wir damit den Diagrammbereich. Nach dem Loslassen der 'Shift'-Taste, haben wir wieder die normale Bewegung des Markers mit dem Mauszeiger.

Ist das Diagramm im nicht gezoomten Zustand, können wir nur waagerecht verschieben. Im gezoomten Zustand ist die Bewegung auch senkrecht möglich oder in Kombination.

- Verschieben zurücksetzen

  Auch hier führt ein Doppel-Klick irgendwo im Diagrammbereich zurück zur Standard-Anzeige.

4.3.3.2 Touch-Bedienung

Bei der Touch-Bedienung ist die Flüssigkeit der Steuerung stark vom verwendeten Browser abhängig. So reagiert zum Beispiel der Firefox-Browser für Android eigentlich immer auch selber auf die Zwei-Finger-Geste mit einer Zoom-Änderung – die hier aber nicht gemeint ist – das führt zu heftigen Verzögerungen oder auch zum ungewollten Zoom des gesamten Seiten-Inhalts. Das lässt sich am besten im Navigationsmenü-Bereich wieder rückgängig machen.

- Zoomen

  Auf einem Touch-Bildschirm wird mit einer Zwei-Finger-Geste gezoomt. Einfach zwei Finger gleichzeitig im Diagrammbereich aufsetzen und anschließend die Finger auf dem Bildschirm voneinander entfernen, um den Bereich unter den Fingern zu vergrößern – oder aufeinander zu bewegen um den Bereich zu verkleinern (Pinch). Das funktioniert gleichzeitig in waagerechter, wie in senkrechter Richtung.

- Verschieben

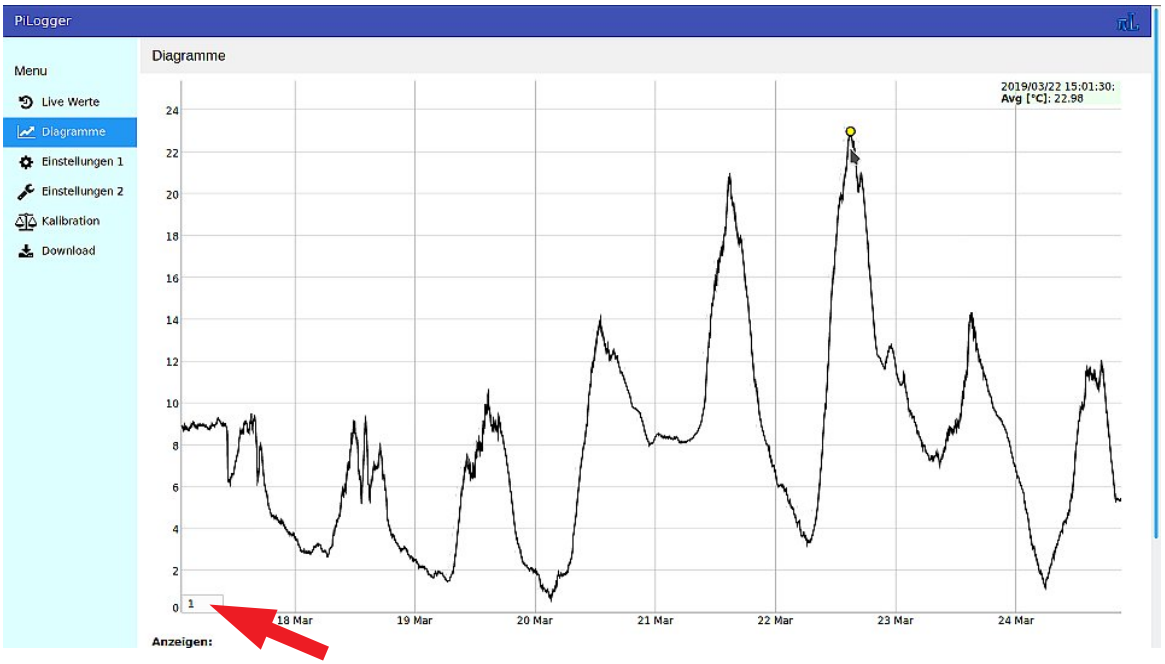
  Verschieben funktioniert mittels dauerhaft aufgesetztem Finger und anschließendem Bewegen des aufgesetztem Fingers in beliebiger Richtung – im Diagrammbereich. Es können auch zwei Finger dafür benutzt werden. Es lässt sich also auch kombinieren mit dem Zoomen.

- Zurücksetzen

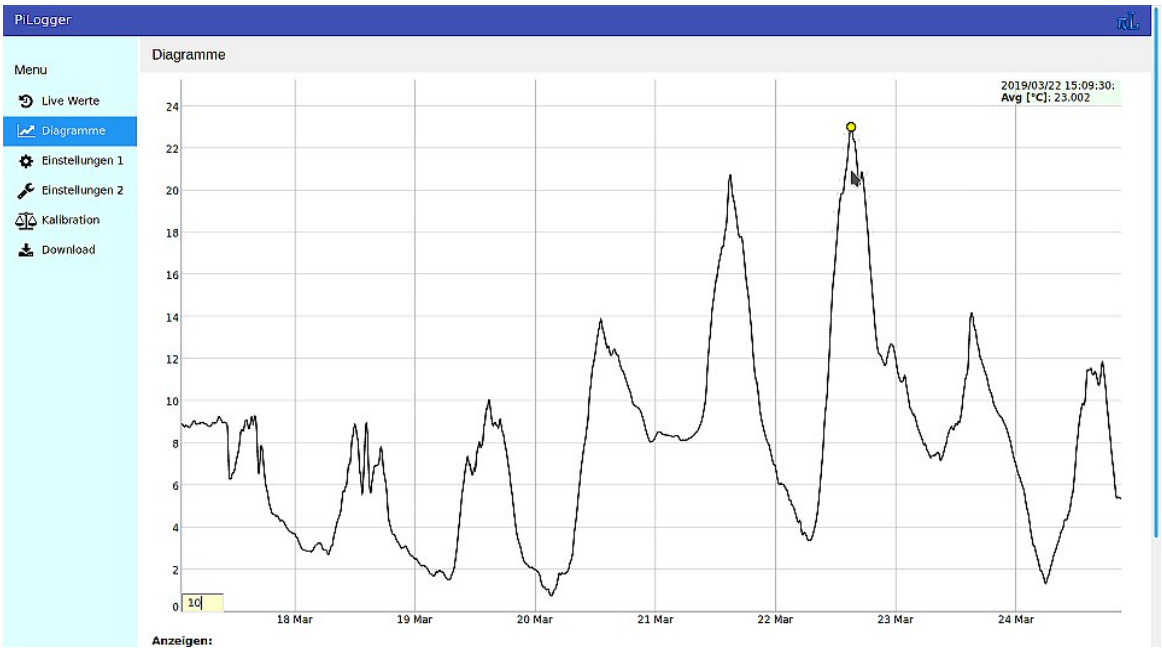
  Auch bei Touch-Bedienung werden Zoom und Verschiebung durch einen Doppel-Tipp im Diagrammbereich zurückgesetzt.

4.3.4 Gleitenden Mittelwert verwenden

Die Diagramm-Kurven lassen sich temporär mit einem gleitendem Mittelwert geglättet darstellen (die Original-Daten bleiben unverändert). Dazu klicken wir in das kleine Eingabefeld unten links im Diagramm:

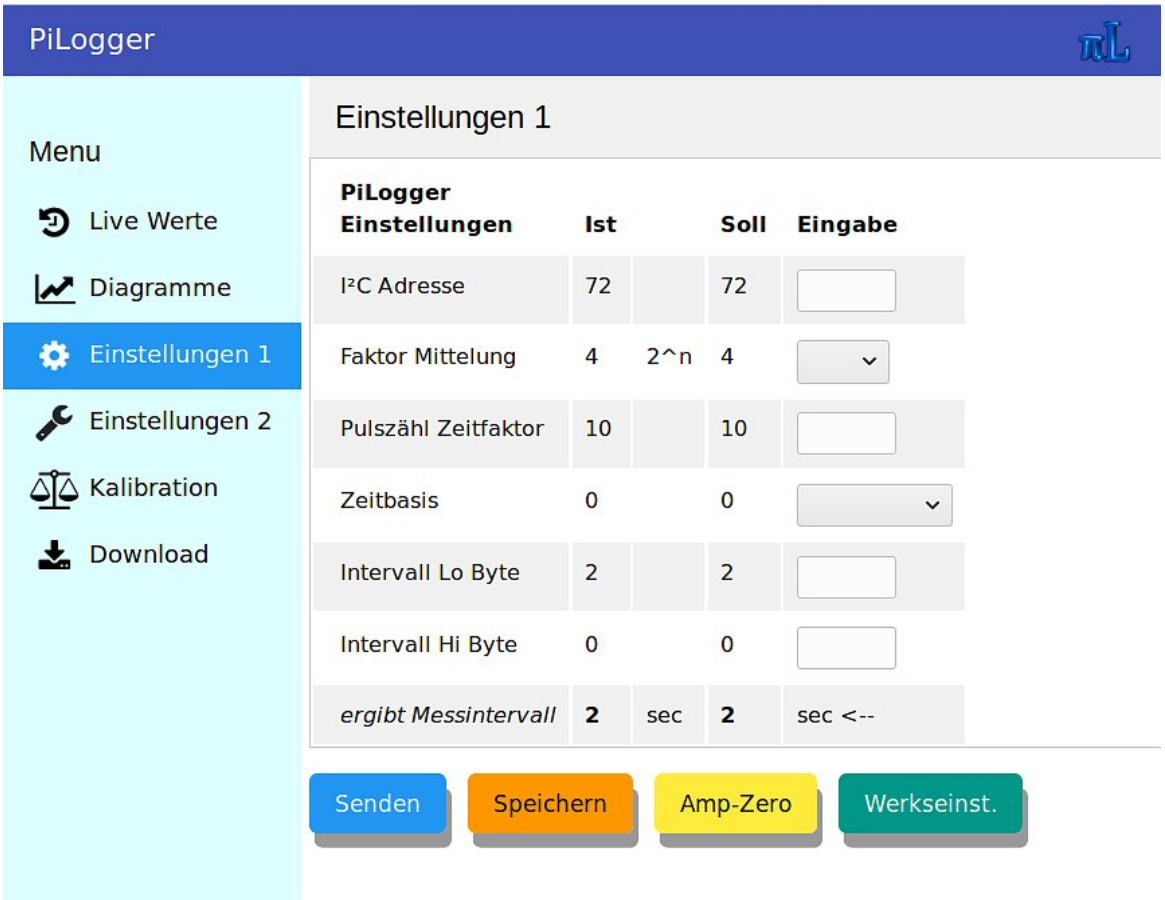


Danach geben wir einen größeren Faktor als '1' (keine Mittelung) ein, zum Beispiel '10', und schließen die Eingabe mit 'Enter' ab. Das ergibt dann diese Darstellung:



Wir können natürlich den Wert genauso auch wieder auf '1' zurücksetzen. Das ist aber nicht unbedingt nötig, da beim nächsten Aufruf der 'Diagramme'-Seite der Wert ohnehin zurückgesetzt ist.

4.4 Die Seite *Einstellungen 1*



Auf dieser Seite werden alle Einstellungen verwaltet, die direkt mit dem PiLogger selber zusammenhängen.

Sobald das Fenster aufgerufen wird, werden die Werte der linken Spalte 'Ist' automatisch einmal vom PiLogger gelesen.
In der rechten Spalte 'Eingabe' können für die einzelnen Parameter neue Werte eingegeben werden. Die Spalte 'Soll' listet die Werte auf, die beim nächsten 'Senden' oder 'Speichern' verwendet werden.
Diese sind im Einzelnen:

- **I²C Adresse**
  Dieser Parameter wird als Dezimalwert angegeben und ist die Bus-Adresse auf die der PiLogger reagiert. Diese Adresse darf im Bereich 4 bis 124 (0x04 bis 0x7C) liegen.
  Die Standard-Adresse ist die 72 (0x48). Nur in Ausnahmefällen sollte es einen Grund geben diese Adresse zu ändern – wenn zum Beispiel zwei PiLogger an einem Raspberry Pi betrieben werden sollen.
  Damit eine geänderte I²C Adresse dauerhaft im PiLogger gespeichert wird, muss anschließend einmal auf 'Speichern' gedrückt werden – siehe unten.
- **Faktor Mittelung**

Der PiLogger berechnet für jeden der abrufbaren 5 Messwerte fortlaufend einen gleitenden Mittelwert. Der dafür benutzte Gewichtungsfaktor kann hier gewählt werden.
Gültige Werte für diesen Parameter M liegen zwischen 1 und 7, wobei der Mittelungsfaktor dann 1/(2^M) beträgt. Somit lässt sich der Mittelungsfaktor zwischen ½ und 1/128 einstellen. Der aktuelle Messwert wird mit diesem Faktor gewichtet und zum bisherigen Mittelwert addiert.

- **Pulszähl Zeitfaktor**
  Hier kann ein Vervielfachungsfaktor für die Messzeit am Impulszähleingang gesetzt werden. Gerade bei kleinen Messintervallen würden sonst viele Messungen mit Null Impulsen erzeugt werden, die keinen sinnvollen Wert für z.B. die Windgeschwindigkeit ergeben. Wenn beispielsweise das Messintervall für Spannung, Strom und Leistung auf ¼ Sekunde (250 msec) eingestellt wird, kann hier durch einen Faktor 40 dafür gesorgt werden, dass die Werte für die Windgeschwindigkeit alle 10 Sekunden ermittelt werden.

- **Zeitbasis**
  Der PiLogger arbeitet mit einer internen Zeitbasis die für das Messintervall verwendet wird. Hier kann zwischen 4 Basis-Zeiteinheiten gewählt werden:
  0 : 1 sec
  1 : 250 msec
  2 : 15,625 msec
  3 : 1,953125 msec
  Damit wird der insgesamt einstellbare Zeitbereich für das Messintervall und die Auflösung darin bestimmt. Mit den nachfolgenden 2 Werten für den Faktor kann ein Vielfaches dieser Zeitbasis von 1x … 65535x als Messintervall eingestellt werden.
  Als kleinstes Messintervall sollten 99 msec nicht unterschritten werden, da eine Messung etwa 84 msec dauert, also 51 x 1,953125 msec = 99,609375 msec. Als größtes Zeitintervall können 65535 sec eingestellt werden, was 18 Stunden, 12 Minuten und 15 Sekunden entspricht.
  Die WebMonitor Software berechnet aus den aktuell eingestellten Werten jedes Mal das resultierende Messintervall in Sekunden. Man kann also durch Ändern der Werte und senden an den PiLogger das neue Messintervall live testen und den Wert in Sekunden sehen. Die LED am PiLogger blitzt für jede Messung einmal auf (ca. 84 msec an).

- **Intervall Low Byte**
  Niederwertiges Byte des Faktors für die Messintervall-Einstellung.

- **Intervall High Byte**
  Höherwertiges Byte des Faktors für die Messintervall-Einstellung.

Der Gesamtfaktor berechnet sich zu H x 256 + L. Mindestwert ist 1.

- **Knopf 'Senden'**
  Neu eingegebene Werte werden durch einen Klick auf den Knopf 'Senden' an den PiLogger geschickt und sind sofort wirksam. Bei einer Änderung des Messintervalls sieht man das sehr schön an der Aktivität der PiLogger-LED.

- **Knopf 'Speichern'**
  Sollen die geänderten Einstellungswerte dauerhaft übernommen werden, so dass nach einem Strom-Ausschalten und wieder Einschalten (power cycle) der PiLogger auch wieder mit diesen Einstellungen arbeitet, müssen diese Werte im internen Flash-Speicher des PiLoggers abgelegt werden. Dazu wird einmal der Knopf 'Speichern' angeklickt.

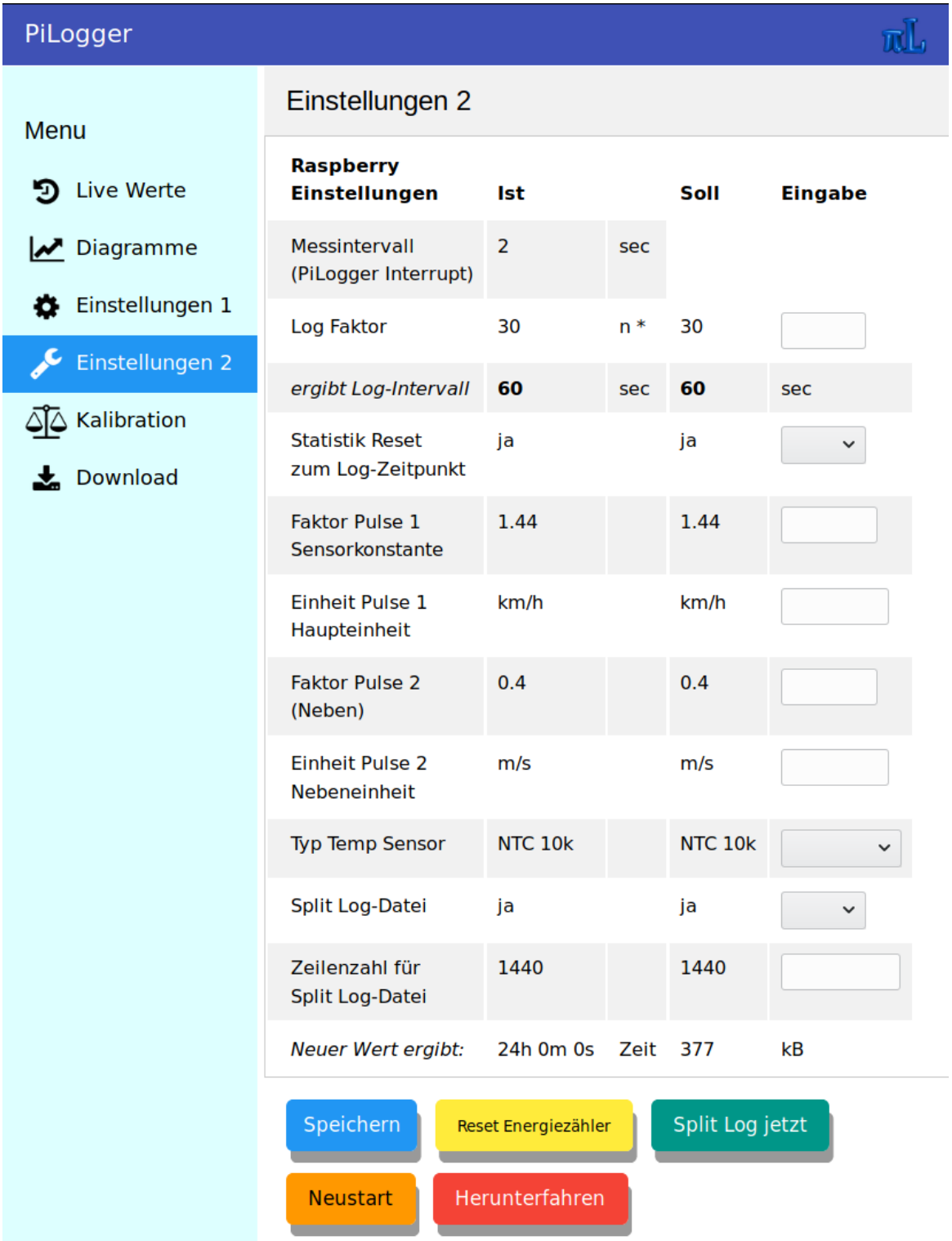
- **Knopf 'Amp-Zero'**
  Der Knopf in der unteren Zeile mit der Aufschrift 'Amp-Zero' sendet einen Befehl an den PiLogger um den aktuellen Mittelwert für den Strom als Nullwert zu setzen.

  Achtung : Dieser Befehl darf nur angewandt werden, wenn zur Zeit wirklich kein Strom durch den Sensor fliesst, also zum Beispiel an den Schraubklemmen nichts angeschlossen ist.

  Dadurch werden Exemplarstreuungen und andere Toleranzen im Strommesszweig heraus kalibriert. Vor Anwendung sollte der PiLogger etwa 20 Minuten aktiv gewesen sein, damit er thermisch eingeschwungen ist.
  Das Drücken des Knopfes 'AmpZero' kann bei Bedarf beliebig oft wiederholt werden.
  Ist das Ergebnis nach einigen Minuten stabil nahe Null, muss diese Einstellung durch das Drücken des Knopfes 'Speichern' permanent im PiLogger gespeichert werden, damit auch nach einem Neustart diese Korrektur aktiv ist.
  Diese Kalibration ist einmal bei Erstinbetriebnahme sinnvoll und gegebenenfalls wenn nach einiger Zeit im Betrieb eine deutliche Missweisung festgestellt wird, oder der Aufbau sich verändert hat.

- **Knopf 'Werkseinst.'**
  Für den Fall, dass der Zustand des PiLogger unklar ist, zum Beispiel durch extreme Messintervalle, kann mit diesem Knopf ein Befehl an den PiLogger gesendet werden, der die Standard-Einstellungen wiederherstellt.
  Diese Standard-Einstellungen lädt der PiLogger aber erst bei einem Neustart – das heißt, der PiLogger muss einmal für einige Sekunden stromlos geschaltet werden und dann wieder eingeschaltet werden. Dafür kann unter 'Einstellungen 2' der Knopf 'Herunterfahren' genutzt werden, um den Raspberry mitsamt dem PiLogger kurz von der Versorgung zu trennen.

4.5 Die Seite *Einstellungen 2*



Diese Seite enthält Einstellungen für das Python-Programm selber, ist also die Raspberry Pi Seite der Einstellungen.

Diese Einstellungen werden in einer separaten Datei 'PiLogger_Config.json' im Programmverzeichnis gespeichert.
Einige der PiLogger-Einstellungen werden dort ebenfalls gespeichert, damit zum Beispiel bei geänderter I²C Adresse, das Programm den PiLogger auch

ansprechen kann.

Auch hier werden die aktuellen Werte in der linken Spalte 'Ist' mit dem Aufruf der Seite aktuell vom Raspberry abgerufen. In der rechten Spalte 'Soll' werden die neu abzuspeichernden Werte aufgelistet und in der Spalte 'Eingabe' werden entsprechend die Werte geändert.
Die erste Zeile gibt die aktuell eingestellte Zeit für das Messintervall des PiLoggers an. Das ist die Basis für die Logger-Einstellungen.
Die Einstellungen im Einzelnen:

- **Log Faktor / Log-Intervall**
  Mit dem hier eingestelltem 'Log Faktor' wird das Messintervall multipliziert um das Log-Intervall einzustellen. Das resultierende Log-Intervall wird in der Zeile darunter angezeigt.
  Dieses Intervall ist der zeitliche Messpunktabstand der mitgeschriebenen Messwerte.
  Die Messwerte werden mit Zeitstempel in der Datei 'logdata.csv' im Verzeichnis '/home/pi' gespeichert.
- **Statistik-Reset zum Log-Zeitpunkt**
  Dieses Feld ist eine Ja/Nein-Entscheidung mittels einer Aufklappliste.
  Diese Funktion bewirkt, dass jedes Mal, wenn ein Eintrag in die Log-Datei gemacht wird, auch ein Befehl an den PiLogger gesendet wird um die Statistik-Werte 'Mittel', 'Minimum' und 'Maximum' auf die aktuellen Messwerte zurückzusetzen.
  Damit werden diese Werte zu den Statistik-Werten die für genau ein Log-Intervall gelten. Das ist insbesondere wichtig für die Energiezähler, die das Produkt von Leistung mal Zeit (-Intervall) pro Log-Intervall aufaddieren.
  Aus diesem Grund ist die Standard-Einstellung für diesen Punkt 'aktiv' - also 'ja'.
- **Faktor Pulse 1**
  Hier wird der Umrechnungsfaktor für den Impulszähleingang eingestellt.
  Die Grundeinheit ist Impulse pro Sekunde. Das Messzeitintervall wird zusätzlich von der PiLogger WebMonitor Software mit berücksichtigt. Hier muss die Fühlerkonstante unter Einbeziehung der Einheit, die im nächsten Feld eingegeben wird, angegeben werden.
  Also beispielsweise für einen Windgeber die Wegstrecke pro Impuls multipliziert mit dem Zeitfaktor der angestrebten Anzeige-Einheit.
  In dem gezeigten Beispiel also 1,44 km/h – das entspricht 0,4 m pro Impuls pro 3600 Sekunden.
  Dieses Feld gilt für die Hauptanzeige (groß) auf der 'Live Werte'-Seite.
- **Einheit Pulse 1**
  Dies ist die zugehörige Maßeinheit für die Umrechnung der Impulszähler-

Messwerte.
Dieses Feld gilt für die Hauptanzeige (groß) auf der 'Live Werte'-Seite.

- **Faktor Pulse 2**
  Wie unter 'Faktor Pulse 1', hier aber für die Nebenanzeige (klein) auf der 'Live Werte'-Seite. In der Log-Datei wird nur der Wert für die Haupteinheit gespeichert.

- **Einheit Pulse 2**
  Dieses Feld gilt für die Nebenanzeige (klein) auf der 'Live Werte'-Seite.

- **Typ Temp Sensor**
  Dies ist eine Aufklappliste mit zur Zeit 6 Einträgen:
  NTC 10k B3928
  NTC 10k B3477
  NTC Tabelle
  PTC Pt1000
  PTC KTY81-110
  PTC Tabelle
  Je nach angeschlossenem Temperaturfühler wird hier die notwendige Umrechnung der Messwerte in Temperaturwerte festgelegt.
  Diese Umrechnungsalgorithmen stehen in der WebMonitor Software zur Verfügung.

  Achtung : Der Jumper (Steckbrücke) auf dem PiLogger One muss entsprechend der Klasse des angeschlossenen Fühlers gesteckt werden. Werkseitig ist dieser in der Position ‚NTC‘ gesteckt.

  Als NTC eignet sich ein Fühler mit 10 kOhm bei 25°C, als PTC ist ein Platinfühler mit 1000 Ohm bei 0°C vorgesehen.
  Siehe hierzu auch ‚PiLogger One Handbuch‘, Kapitel 7.3 ‚Temperaturfühlereingang‘.

- **Split Log-Datei**
  Dieses Feld ist eine Ja/Nein-Entscheidung mittels einer Aufklappliste.
  Ist diese Funktion aktiviert, wird automatisch nach der Anzahl der geloggten Zeilen, die im nächsten Feld angegeben werden, eine neue Log-Datei angelegt.
  Ist diese Funktion nicht aktiviert, wird die Log-Datei unendlich fortgeführt. Die Log-Datei kann trotzdem manuell geteilt werden mit dem Knopf 'Split Log jetzt' - siehe unten.

- **Zeilenzahl für Split Log-Datei**
  Hier wird die Anzahl der Zeilen angegeben, bei der eine neue Log-Datei begonnen wird. Jede neu angelegte Log-Datei bekommt im Namen den Zeitstempel der Neuanlage.

Je nach Log-Intervall können schnell sehr große Dateien entstehen, deren Übertragung für die Diagramm-Anzeige dann entsprechend lange dauern. Dabei wird für die Diagramm-Anzeige tatsächlich eine noch größere Datei übertragen, die zusätzlich noch die Daten der vorherigen Log-Datei enthält. Damit ist direkt nach der automatischen Anlage einer neuen Log-Datei zumindest der Zeitraum davor als Kurve anzeigbar.
Die Zeile darunter zeigt an welcher Zeitraum und welche Dateigröße je Log-Datei mit den Eingabewerten entsteht.

- **Knopf 'Speichern'**
  Mit Betätigung diese Knopfes werden die Werte aus der Spalte 'Soll' in der Konfigurationsdatei 'PiLogger_Config.json' gespeichert.
  Zusätzlich wird das Programm mit diesen Werten neu initialisiert.

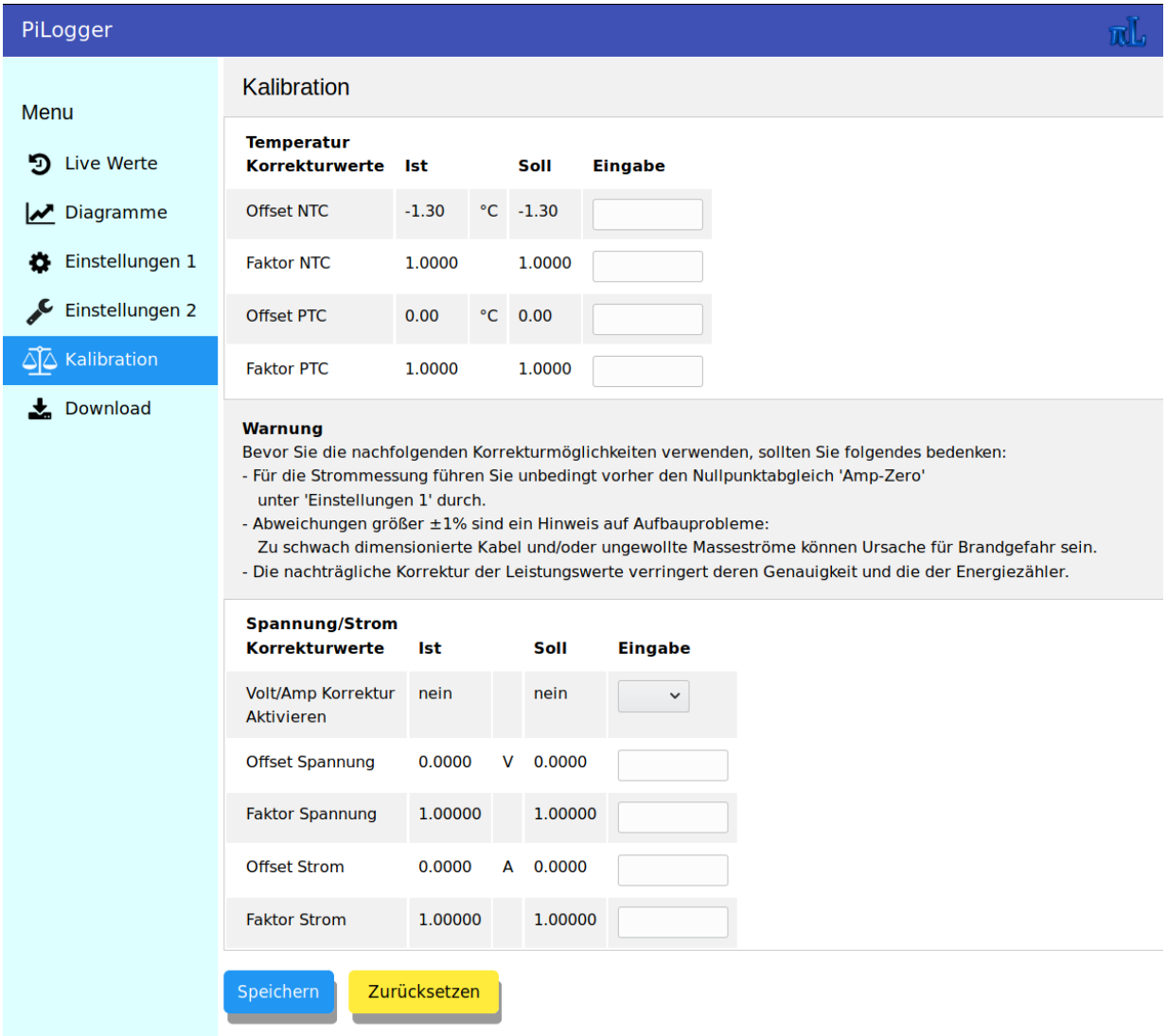
- **Knopf 'Reset Energiezähler'**
  Dieser Knopf setzt sofort alle Energiezähler auf Null, also sowohl die Dauerzähler als auch die Tageszähler. Inklusive der zugehörigen Zeitzähler.

- **Knopf 'Split Log jetzt'**
  Wird dieser Knopf betätigt, wird sofort eine neue Log-Datei angelegt. Die bisherige Datendatei ist dabei weiterhin herunterladbar und auch anzeigbar – siehe 4.7 Die Seite Download.
  Die Daten der bisherigen Log-Datei sind nun der Anzeigen-Vorlauf für die Zeitdiagramme.
  Soll auch dieser Teil nicht mehr angezeigt werden, zum Beispiel, weil die Einstellungen stark geändert wurden (andere Impuls-Einheiten, andere Kalibrierwerte etc.), muss der Knopf 'Split Log jetzt' noch einmal gedrückt werden. Dabei entsteht eine unnütze Log-Datei ohne Messwerte oder nur sehr wenigen Messwertzeilen. Sie lässt sich gegebenenfalls auf der Seite 'Download' löschen.

- **Knopf 'Neustart'**
  Dieser Knopf bietet die Möglichkeit den Raspberry kontrolliert neu zu starten.
  Im Gegensatz zum Trennen der Stromversorgung wird mit dieser Methode ein Datenverlust oder gar Unbrauchbar machen der SD-Karte vermieden.

- **Knopf 'Herunterfahren'**
  Dieser Knopf bietet die Möglichkeit den Raspberry geordnet herunterzufahren.
  Im Gegensatz zum Trennen der Stromversorgung wird mit dieser Methode ein Datenverlust oder gar Unbrauchbar machen der SD-Karte vermieden.
  Nach den üblichen letzten Blink-Zeichen (4 mal gleichmäßig) der Raspberry Pi LED kann die Stromversorgung gefahrlos getrennt werden.

4.6 Die Seite *Kalibration*



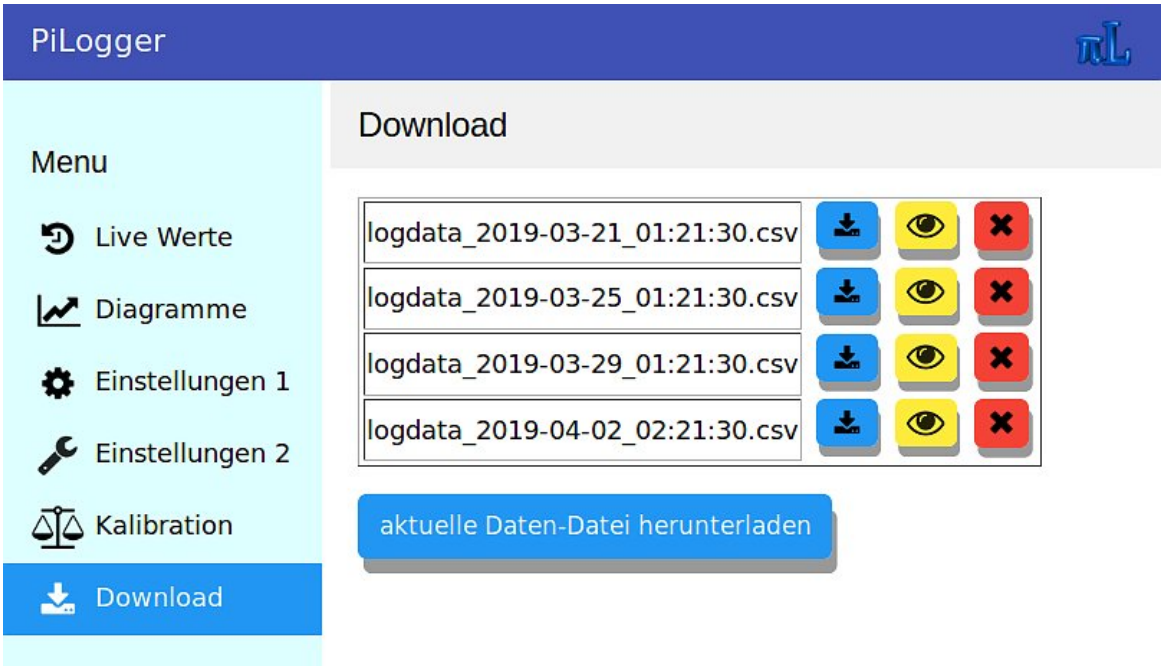
Die Seite 'Kalibration' bietet die Möglichkeit Korrekturwerte für die Temperaturmessung und auch für die Spannungs- und Strommessung einzustellen.
Dabei handelt es sich grundsätzlich um eine einfache lineare Umrechnung mit einer Verschiebung (Offset) und einer Steigung (Faktor).
Für die Temperaturmessung ist diese Korrekturrechnung immer sinnvoll und ermöglicht etwa eine klassische 2-Punkt-Kalibration entsprechend der Definition der Celsius-Skala:
Dafür setzt man zuerst den Temperaturfühler einem Eiswasserbad aus (eine gute, wasserdichte Isolation der Anschlüsse vorausgesetzt) und nimmt den erhaltenen Anzeigewert negativ (mal -1) als 'Offset'. Anschließend wird der Temperaturfühler siedendem Wasser ausgesetzt und der 'Faktor' wird auf '100/Anzeigewert' gesetzt.

Die Korrektur der Spannungs- und Stromwerte dagegen sollte nur aktiviert werden, wenn folgendes beachtet wird:

- Bei Abweichungen der Stromwerte immer zuerst den Nullpunkt-Abgleich 'Amp-Zero' unter 'Einstellungen 1' durchführen – siehe Seite 58.
- Wenn die Abweichungen der Spannungsmessung größer ±1% des Messwertes sind, ist das ein Hinweis auf ein mögliches Aufbauproblem:
  - zu schwach dimensionierte Anschlusskabel führen zu Spannungsabfällen und können durch Eigenerwärmung ein Sicherheitsproblem darstellen. Der PiLogger misst die Spannung zwischen seinen 'OUT+' und 'OUT-' Klemmen. Durch einen Spannungsabfall am Zuleitungskabel kann diese Spannung von der Spannung am Generator oder Akku abweichen. Stellen Sie sicher, dass die Kabel nicht überlastet sind und auch die Kontakte gut verbunden sind (keinen nennenswerten Spannungsabfall erzeugen). Ebenso sollten die Kabel immer so kurz wie möglich gehalten werden. Soll also, zum Beispiel, die Akkuspannung als Referenz genommen werden, ist es sinnvoll den PiLogger näher am Akku zu montieren als am Generator.
  - zusätzliche Ströme auf dem Minus-Kabel des Akkus, die nicht auch auf der vom PiLogger gemessenen Plus-Leitung fließen, führen zu einer Spannungsanhebung an der Messstelle des PiLoggers – einem Masseversatz.
- Wird die Spannungs- und Stromwert-Korrektur aktiviert, werden auch die Leistungswerte, die der PiLogger direkt berechnet hat, entsprechend nachträglich korrigiert. Das bedeutet zusätzliche Divisions- und Multiplikationsoperationen, die die Genauigkeit der Leistungswerte verringern. Und damit natürlich auch die Genauigkeit der Energiezähler.

  Generell ist ein optimierter Aufbau einer nachträglichen Korrekturrechnung vorzuziehen.

4.7 Die Seite *Download*



Diese Seite ermöglicht es die Logdaten vom Raspberry herunterzuladen, anzusehen und auch zu löschen.

Alle Logdateien sind '.csv'-Dateien. Um sie in ein deutschsprachiges Tabellen-kalkulationsprogramm laden zu können, muss jedoch mit einem Textprogramm eine Ersetzung von '.' zu ',' durchgeführt werden. Die '.csv'-Dateien verwenden für die Zahlendarstellung die amerikanisch-englische Darstellung, um mit der Diagramm-Bibliothek kompatibel zu sein. Das führt ohne diese Ersetzung zu einer Fehlinterpretation beim Import. Als Feld-Trennzeichen wird das Semikolon (;) verwendet, was in der Regel vom Importfilter erkannt wird, manchmal aber extra gesetzt werden muss.

Mit Aufruf der Seite wird das Verzeichnis aller verfügbaren 'logdata_xxx.csv' Dateien frisch vom Raspberry abgefragt.
Hinter jeder dieser Dateien gibt es 3 Schaltflächen:
Die erste blaue Fläche mit dem Download-Symbol startet einen normalen Datei-Download für diese Datei. Je nach Konfiguration des aufrufenden Browsers erscheint dabei ein Dialog-Fenster mit der Frage ob und wohin die Datei gespeichert werden soll.
Die zweite gelbe Fläche mit dem Augen-Symbol ermöglicht es, diese Datei mit der Diagramm-Funktion anzuschauen. Dabei wird, wie gewohnt, die Seite Diagramme aufgerufen und standardmäßig die Durchschnittstemperatur dargestellt. Damit können Daten aus zurückliegenden Zeiträumen durchgesehen werden.
Die dritte rote Fläche mit dem X-Symbol löscht diese Datendatei von der SD-Karte des Raspberry Pi. Da dieser Vorgang endgültig ist, wird zunächst eine Sicherheits-Rückfrage per Overlay-Fenster zwischengeschaltet, um einen

Abbruch bei versehentlicher Betätigung zu ermöglichen:

Unter der Datei-Verzeichnis gibt es noch die Schaltfläche 'aktuelle Daten-Datei herunterladen'. Damit lässt sich die aktuell verwendete Log-Datei 'logdata.csv' herunterladen. Da diese Datei zur Zeit zum Loggen benutzt wird, kann man sie natürlich nicht löschen, und auch das Anzeigen ist an dieser Stelle nicht sinnvoll, da dies über den normalen Aufruf der Diagramme-Seite erfolgt.

5 Konzept PiLogger WebMonitor

In diesem Kapitel beschreiben wir die Arbeitsweise und den Ansatz der WebMonitor Software. Das soll dem Verständnis dienen und auch als Anregung für eigene Anpassungen oder Projekte – PiLogger WebMonitor ist freie offene Software (FOS).

Der PiLogger WebMonitor ist eine Netzwerk-Anwendung (WebApplication) die die Fähigkeiten eines Raspberry Pi sowohl eine Netzwerk-Verbindung als auch den Zugriff auf die GPIOs (General Purpose Input/Output; Universeller Eingangs- und Ausgangs-Anschluss) bereitzustellen nutzt.
Kernstück dieser Webanwendung ist dabei das Python Web Framework 'Bottle' von Marcel Hellkamp (https://bottlepy.org/).
„Bottle ist ein schnelles, einfaches und leichtgewichtiges WSGI (Web Server Gateway Interface) micro web-framework für Python. Es wird als ein Eine-Datei-Modul verteilt und hat keine weiteren Abhängigkeiten als die zur Python Standard Bibliothek.“
Es wird unter MIT Lizenz angeboten und kann aus den Raspberry Pi OS-Software-Ablagen (Repositories) installiert werden. Der eingebaute Server für Entwicklungszwecke reicht völlig für eine private (nicht Internet-verbundene) Web-Seite. Bottle lässt sich mit Erweiterungsmodulen (Plugins) ausbauen – was bezüglich Zugriffssteuerung mit beispielsweise 'Bottle-Cork' noch folgen soll. Statt des eingebauten Servers lassen sich auch alternativ viel andere Server nutzen – siehe Bottle-Dokumentation.

Ausgehend von diesem Grundbaustein liegt es nahe auch den Rest direkt in Python zu lösen. Das Hauptprogramm auf dem Raspberry ist demzufolge ein Python-Programm (üblicherweise Script genannt – obwohl Python eine ausgewachsene Hochsprache ist).
Die Programmdatei heißt 'PiLogger-bottle.py' und ist als Teil des Archivs 'PiLo-WebMon.zip' herunterladbar (https://www.pilogger.de/index.php/de/download-de/send/2-software/8-pilo-webmon).

Voraussetzung ist ein aktiviertes I²C Interface mit aktiviertem alternativen I²C Treiber (siehe Kapitel 3.2 Seite 43), sowie die Installation von 'python3-smbus', dem Modul für Python um auf den I²C Bus zuzugreifen. Außerdem das Installieren von 'python3-rpi.gpio', einer Python-Erweiterung für den direkten Zugriff auf die GPIOs. Damit kann die Interrupt-Leitung des PiLogger One auch als solche ausgewertet werden.
Das ist die Grundausstattung für alle I²C Sensoren, die mit dem Raspberry verwendet werden sollen. Der PiLogger One ist sozusagen ein Multi-Sensor. Die

Anwendungs-Programmier-Schnittstelle (API) ist im Handbuch für den PiLogger One ausführlich im Kapitel 9 (Seite 65) beschrieben.

Das Python-Script erledigt also nun die Hauptaufgabe – das Loggen, indem es die Messwerte vom PiLogger One ausliest, nachbearbeitet und auf der SD-Karte des Raspberry Pi speichert. Dabei wird keine große Datenbank benutzt, sondern die Daten werden in einer reinen Textdatei im Anhängen-Modus gespeichert. Das ist eine klassische CSV-Datei (comma separated values), die Zeile für Zeile die Messwerte aufzeichnet.
Das Python-Script bietet nun zusätzlich über das Modul 'bottle' die Möglichkeit Webseiten an einen Browser auszugeben. Was diese HTML Dateien ermöglichen ist im Kapitel 4 ausführlich beschrieben worden.

Damit wird eine plattformübergreifende Benutzeroberfläche mit allen modernen Darstellungsmöglichkeiten bereitgestellt.

Um den Datenverkehr zu minimieren und insbesondere keine externen Internet-Server ansprechen zu müssen, werden keine Dritt-Webschriften (Fonts) und keine riesigen Universal-Frameworks verwendet. Tatsächlich wird für alle Seiten eine gemeinsame CSS-Datei verwendet um den Darstellungsstil festzulegen. Darin enthalten ist auch die Verwendung einer einzigen Bilddatei, die alle benötigten Icons enthält, als sogenannte Sprites. Ausnahme ist hier die zusätzliche Datei für das sogenannte favicon – also das kleine Icon zur Seitenerkennung. So ist der Aufruf einer Seite (außer Diagramme) mit 4 kleinen Dateien beantwortet (1 document, 1 stylesheet, 1 img iconset & 1 img favicon). Diese Ressourcen sind statisch und werden in der Regel vom Browser zwischengespeichert (ge-cashed). Das heißt, der nächste Seitenaufruf geht nochmal schneller.

Die eigentlich anzuzeigenden Daten werden mit XHR dynamisch vom Raspberry geladen. Diese Technik nennt sich Ajax und XHR steht für 'XMLHttpRequest'. Kern ist hierbei, dass im Browser ein JavaScript-Script ausgeführt wird, das in unserem Fall direkt in die HTML-Seite eingebettet ist, um nicht noch eine weitere Datei anfordern zu müssen.
Dieses JavaScript-Script führt die erwähnten Requests (Anforderungen) für die anzuzeigenden Daten aus, die dann als kleine JSON-Datei vom Raspberry gesendet werden.
In der Gegenrichtung werden die Daten in Richtung Server mit der übertragenen Navigationsadresse als sogenannte Query-Parameter gesendet (url-data). Auch dies ist ein XHR – nur das hier die Antwort des Servers lediglich ok oder nok ist – also eine reine Erfolgsrückmeldung (Acknowledge).

Ein weiterer wichtiger Baustein der PiLogger WebMonitor Software ist die JavaScript-Bibliothek 'dygraphs' von Dan Vanderkam (http://dygraphs.com/). Diese sogenannte Charting-Library (Diagramm-Bibliothek) wird auf der Seite

'Diagramme' zum Erzeugen der Messwert-Zeitdiagramme verwendet.
Dafür werden auf dieser Seite 2 zusätzliche Dateien vom Server angefordert : 'dygraph_p.min.js' - die Charting-Library in minimierter Form und 'dygraph_p.css' mit den Diagramm-spezifischen Stilvorgaben.
Die Datei 'dygraph_p.min.js' basiert dabei auf der Original-Version 2.1.0 und beinhaltet Patches (Reparaturen) für die Touch-Bedienung.
Dieses JavaScript-Modul ruft beim Aufruf der Seite 'Diagramme' die Logdatendatei 'showdata.csv' vom Raspberry ab, zerlegt und analysiert (parsed) die Daten und erzeugt ein Zeitdiagramm der Messwertreihen als Grafik auf einer zugewiesen Fläche der Seite.
Diese Charting-Library kommt dabei auch mit sehr umfangreichen Messreihen in erstaunlich kurzer Zeit zurecht. Die längste Zeit nimmt in der Regel die Übertragung der Datendatei in Anspruch. Das hängt in erster Linie von der Verbindungsqualität und der Größe der Datei ab. Dabei ist zu bedenken, dass diese Datendatei bis zu doppelt so groß werden kann, wie die eingestellte Datendateigröße, da sie aus der bisherigen Logdatei plus der aktuellen Datendatei gebildet wird – um immer einen sinnvollen Betrachtungszeitraum darstellen zu können.

Bisher sind alle Daten und die Kommunikation rein nur zwischen dem Raspberry Pi als Server und einem im lokalen Netz angemeldeten Benutzer-Rechner als Client über den eigenen Router gelaufen.
Damit der Raspberry beim Loggen einen korrekten Zeitstempel zu den Messdaten hinzuzufügen kann, muss er seine interne Zeit mit einem Zeitserver synchronisieren. Das geschieht standardmäßig mit einem Internet-Zeitserver per NTP (Network Time Protocol). Mit einer einfachen Maßnahme lässt sich auch dieser Internet-Zugriff vermeiden:
Der Router synchronisiert selber seine interne Zeit mit einem Internet-Zeitserver, meistens mit einem Zeitserver des Netzanbieters. Die meisten Router bieten die Möglichkeit diese Zeit für das lokale interne Netz als Zeitserver zur Verfügung zu stellen – siehe Kapitel 3.1 .
Wenn diese Voraussetzung erfüllt ist, kann der Raspberry sich nun lokal mit dem Router synchronisieren.
Nun bleiben nur noch die regelmäßigen (täglichen) Anfragen bei den Raspberry Pi OS-Update-Servern. Ist der Raspberry ausschließlich als Logger vorgesehen und strikt hinter der Firewall des Routers, so ist ein Update in der Regel nicht sicherheitsrelevant. Zumal diese Update-Anfragen ohne ein eingerichtetes Auto-Update sowieso nur die Datenbank aktualisieren – und andernfalls ein unbeaufsichtigtes Update den Logger-Betrieb unterbrechen kann. Diese Update-Anfragen lassen sich wahlweise abschalten – siehe Kapitel 3.2 Seite 43.

Zur Zeit ist der Zugang zu allen Funktionen des PiLogger WebMonitor allen

Benutzern des heimischen Netzwerk offen. Dies lässt sich mit einem Erweiterungsmodul für 'Bottle' wie etwa 'Bottle-Cork' ändern – todo...

Wer den PiLogger WebMonitor auch weit entfernt über das Internet benutzen möchte, sollte dies über ein VPN (virtuell privates Netzwerk) tun – also über eine gesicherte Punkt-zu-Punkt-Verbindung.

PiLogger WebMonitor zeigt somit auch, dass 'Smart Home' und 'Internet der Dinge' auch ohne Cloud und Fremdserver möglich sind.