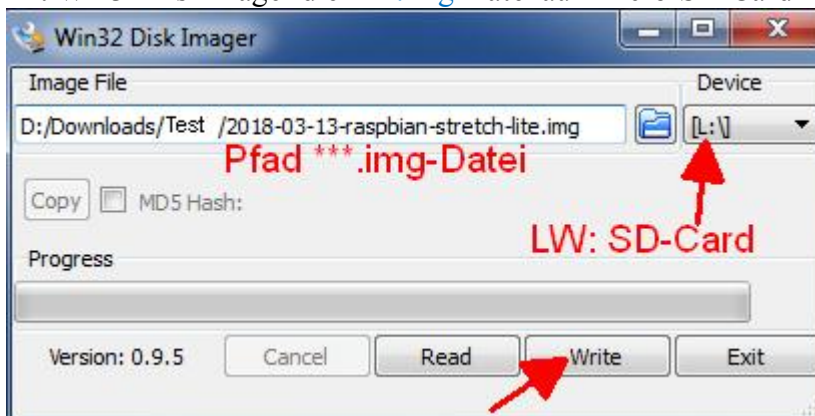


Raspbian installieren

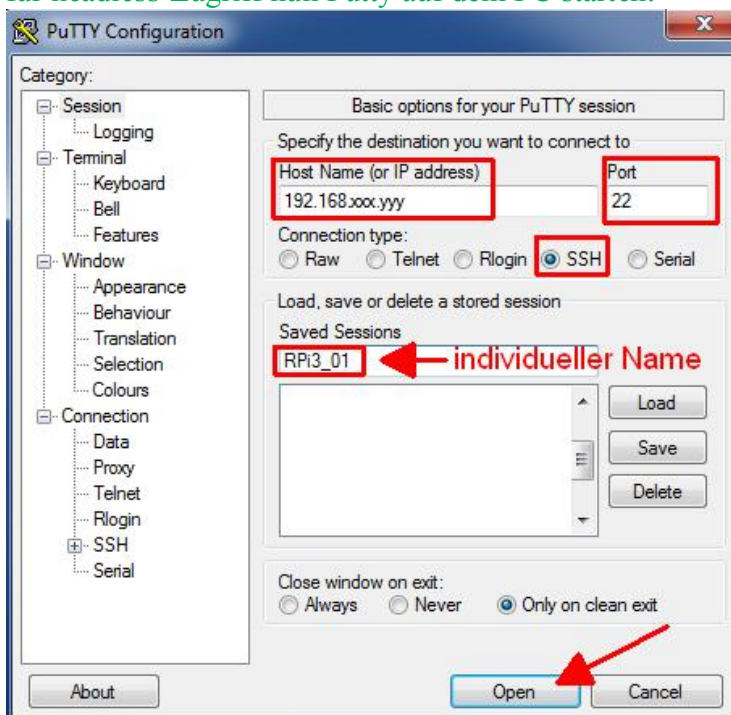
- Download hier: <https://www.raspberrypi.org/downloads/raspbian/>
- Für Systeme mit Monitor wird **DESKTOP** für Serversysteme ohne Monitor (headless) wird **LITE** empfohlen (Putty erforderlich)



- mit Winzip oder anderem Programm auf PC entpacken => xxx.img-Datei
- mit Win32Diskimager die xxx.img-Datei auf Micro-SD-Card >= 4GB schreiben



- nur bei SSH-Zugang (headless-System): Im Verzeichnis /boot eine leere Datei namens ssh anlegen (Neu => Textdokument => neues Textdokument .txt => in ssh ohne Endung umbenennen)
- SD Card auswerfen und danach in den RPi einsetzen
- RPi anschalten
- in Fritzbox (oder genutzten Router) einloggen => Heimnetz => Heimnetzübersicht suchen
Hier nach dem Eintrag raspberrypi suchen.
- Dem raspberrypi nun eine vorher definierte feste IP aus dem Home-LAN vorgeben.
s. „Diesem Netzwerkgerät immer die gleiche IPv4 –Adresse zuweisen.“ (z.B.192.168.xxx.yyy).
- für headless-Zugriff nun Putty auf dem PC starten:



- für headless Systeme SSH-key-Abfrage bestätigen und Passwort raspberry (user pi) eingeben
- für monitorbasierte Systeme am Desktop mit user pi und Passwort raspberry anmelden

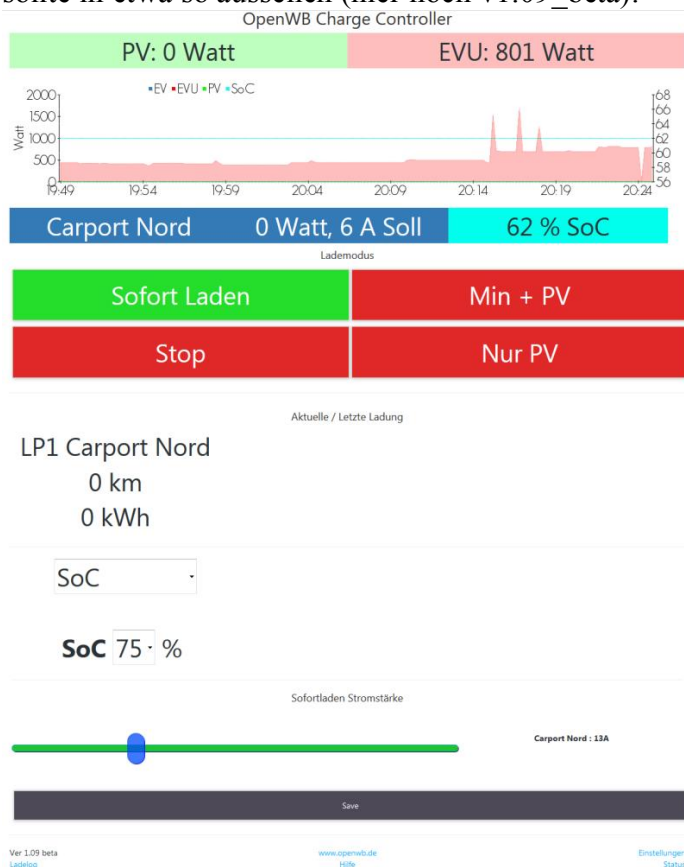
Raspbian für openWB konfigurieren

- auf der Konsole/Shell eingeben: `sudo raspi-config`
 - neues Passwort definieren
 - ev. Hostnamen ändern (z.B. `RPi3_01`)
 - language einstellen => `de_DE.UTF8` (dauert etwas)
 - timezone => `Berlin`
 - Interfacing: `SSH` aktivieren
 - für Modbus-Variante mit *digitaler* Ansteuerung der **EVSE-DIN**: kein `I2C` erforderlich
 - für DAC-Variante mit *analoger* Ansteuerung der **EVSE WB**: Interfacing: `I2C` aktivieren
 - Advanced: Expand filesystem => reboot mit OK erforderlich
- In der Shell folgendes eingeben:
`curl -s https://raw.githubusercontent.com/snaptec/openWB/master/openwb-install.sh | sudo sh`
=> Das Install-script installiert nun alles für openWB automatisch (dauert etwas).
- In der Shell folgendes eingeben: `crontab -e`
- Bei erstmaliger Nutzung – einen Editortyp bestätigen
In der crontab-Datei unter den #Anmerkung folgende Zeilen einfügen:

```
* * * * * /var/www/html/openWB/regel.sh >> /var/log/openWB.log 2>&1
* * * * * sleep 10 && /var/www/html/openWB/regel.sh >> /var/log/openWB.log 2>&1
* * * * * sleep 20 && /var/www/html/openWB/regel.sh >> /var/log/openWB.log 2>&1
* * * * * sleep 30 && /var/www/html/openWB/regel.sh >> /var/log/openWB.log 2>&1
* * * * * sleep 40 && /var/www/html/openWB/regel.sh >> /var/log/openWB.log 2>&1
* * * * * sleep 50 && /var/www/html/openWB/regel.sh >> /var/log/openWB.log 2>&1
```
- Crontab mit `Ctrl + X` verlassen und mit `J` (Ja) speichern
- RPi mit `sudo shutdown -r now` rebooten

openWB starten

- In der Browseradresseleiste folgendes eingeben und mit **Enter** bestätigen (xxx.yyy durch IP des RPi ersetzen): <http://192.168.xxx.yyy/openWB/web/index.php>
Es werden wegen fehlender Config vermutlich ein paar Fehlermeldungen ausgegeben. Die Anzeige sollte in etwa so aussehen (hier noch v1.09_beta).



openWB konfigurieren

- Auf der Hautseite des webinterfaces rechts unten auf **Einstellungen** klicken.
- Es erscheint eine umfangreich dokumentierte Konfigurationsseite.

Zurueck Ladeeinstellungen **Modulkonfiguration** Misc

Erster Ladepunkt

Name Ladepunkt 1: Carport Nord

Anbindung Ladepunkt 1: ModbusEVSE

Gültige Werte dac, modbusEVSE. Weitere Konfiguration je nach Anbindung erforderlich! Modbus nur mit EVSE DIN getestet. Auf der EVSE muss Register 2003 auf 1 gesetzt werden (Deaktivierung analog Eingang), sonst kein beschreiben möglich

EVSE Source: /dev/ttyUSB0
Gültige Werte /dev/ttyUSB0, /dev/virtualcom0. Serieller Port an dem der Modbus der EVSE angeschlossen ist.

EVSE ID: 1
Gültige Werte 1-254. Modbus ID der EVSE.

RS485/Lan-Konverter IP: 192.168.1.73 **nur nötig, wenn ein LAN-Konverter verwendet wird**
Gültige Werte IP. IP Adresse des Modbus/Lan Konverter. Vermutlich gleich der IP des SDM Zählers in der WB.

Ladeleistungsmodul: SDM 630 Modbus

Zähler Source: /dev/ttyUSB1
Gültige Werte /dev/ttyUSB0, /dev/virtualcomX. Serieller Port an dem der SDM in der Wallbox angeschlossen ist. Meist /dev/ttyUSB0
Nach ändern der Einstellung von ttyUSB auf virtualcom0 ist ein Neustart erforderlich

Zähler ID: 11
Gültige Werte 1-254. Modbus ID des SDM. Für SDM230 & SDM630v2.

RS485/Lan-Konverter IP: 192.168.1.73 **nur nötig, wenn ein LAN-Konverter verwendet wird**
Ist nur von belang wenn die Source auf /dev/virtualcomX steht. Ansonsten irrelevant
Gültige Werte IP. Wenn ein LAN Konverter genutzt wird muss die Source auf /dev/virtualcomx (z.B. /dev/virtualcom0) gesetzt werden.

SOC Modul: SoC BMW i3

Benutzername:
BMW Services Benutzername

Passwort:
BMW Services Passwort

VIN:
BMW i3 VIN

Zweiter Ladepunkte: Aus

Dritter Ladepunkt: Aus

- Zunächst wird der **erste Ladepunkt** vollständig konfiguriert.
 - **Ladepunkt 1 (LP1)**
 - a) Var. mit Modbus mit EVSE-DIN: wie oben dargestellt mit serieller Portnummer und Modbus-ID (RS485/LAN-Konverter nur bei Nutzung eines solchen erforderlich)
 - b) Var. mit analogem DAC-Modul 0-5V wie folgt:

DAC-Adresse ermitteln: Hierzu in der shell eingeben: `sudo i2cdetect -y 1`

```
pi@rpi3_01:~$ sudo i2cdetect -y 1
   0  1  2  3  4  5  6  7  8  9  a  b  c  d  e  f
00: -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
10: -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
20: -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
30: -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
40: -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
50: -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
60: 60 -- -- -- -- -- -- -- -- -- -- -- -- -- --
70: -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
pi@rpi3_01:~$
```

hier: DAC-Adresse 60. Diese nun in die Configseite unter „Anbindung LP1“ **DAC** eintragen.

Ladeleistungsmodul (zu LP1 zugehörig)

Modbus-Zähler mit serieller Portnummer und Modbus-ID
(RS485/LAN-Konverter nur bei Nutzung eines solchen erforderlich)

- **optional: Ladezustand des Fahrzeuges** (SoC = state of charge), welches an LP1 lädt (Hierzu gibt es verschiedene Methoden, diese spezielle Autoinformationen zu erhalten. Z.B. über einen Server des Autoherstellers, OBD-Dongles, uvm.)
- Nun wird das sehr wichtige **Strombezugsmessmodul** (EVU-Übergabepunkt) konfiguriert.
z.B. für SDM630-Modbus-Zähler: mit serieller Portnummer und Modbus-ID
(RS485/LAN-Konverter nur bei Nutzung eines solchen erforderlich)

Strombezugsmessmodul (EVU-Übergabepunkt)

Strombezugsmodul:

SDM 630 Zähler Source:

Gültige Werte /dev/ttyUSB0, /dev/virtualcom. Serieller Port an dem der SDM angeschlossen ist.

Zähler ID:

Gültige Werte 1-254. Modbus ID des SDM. Getestet SDM230 & SDM630v2.

RS485/Lan-Konverter IP:

nur nötig, wenn ein LAN-Konverter verwendet wird.

Gültige Werte IP. Ist die source "virtualcomX" wird automatisch ein Lan Konverter genutzt.

- Nun wird das *optionale PV-Modul* (Messung PV-Leistung) konfiguriert.
z.B. für MPM3PM-Modbus-Zähler: mit serieller Portnummer und Modbus-ID
(RS485/LAN-Konverter nur bei Nutzung eines solchen erforderlich)

PV Modul

PV Modul:

MPM3PM Wechselrichterleistung Source:

Gültige Werte /dev/ttyUSB0, /dev/virtualcomX. Serieller Port an dem der MPM3PM angeschlossen ist. Meist /dev/ttyUSB0
Nach ändern der Einstellung von ttyUSB auf virtualcom0 ist ein Neustart erforderlich

MPM3PM Wechselrichterleistung ID:

Gültige Werte 1-254. Modbus ID des MPM3PM.

IP des Modbus/Lan Konverter:

nur nötig, wenn ein LAN-Konverter verwendet wird.

Gültige Werte IP. Ist die source "virtualcomX" wird automatisch ein Lan Konverter genutzt.

- Nun wird das *optionale Speichermodul* (Einbindung eines Hausspeichers) konfiguriert.
z.B. mittels http-Abfrage

Speicher Modul

Speicher Modul:

Speicherleistung URL:

Gültige Werte URL. Vollständige URL die den aktuellen Leistungswert in Watt wiedergibt. Erwartet wird eine Ganzzahl. Positiv heißt Speicher wird geladen und eine negative Zahl bedeutet das der Speicher entladen wird. Das Modul dient dazu bei NurPV Ladung eine Entladung des Speichers zu verhindern.

SpeicherSoC URL:

Gültige Werte URL. Vollständige URL die den aktuellen SoC wiedergibt.

Speicher Import Wh URL:

Gültige Werte URL. Wenn nicht vorhanden, none eintragen. Vollständige URL die den Zählerstand der Batterieladung in WattStunden wiedergibt. Erwartet wird eine Ganzzahl.

Speicher Export Wh URL:

Gültige Werte URL. Wenn nicht vorhanden, none eintragen. Vollständige URL die den Zählerstand der Batterieladung in WattStunden wiedergibt. Erwartet wird eine Ganzzahl.

LP2 und LP3 werden bei Bedarf nach den gleichen Prinzipien konfiguriert.

openWB logging nutzen

- Es ist ein Live-, Daily-, Monthly- und Yearly-Logging in openWB integriert.
- Start auf der Hauptseite unten rechts:



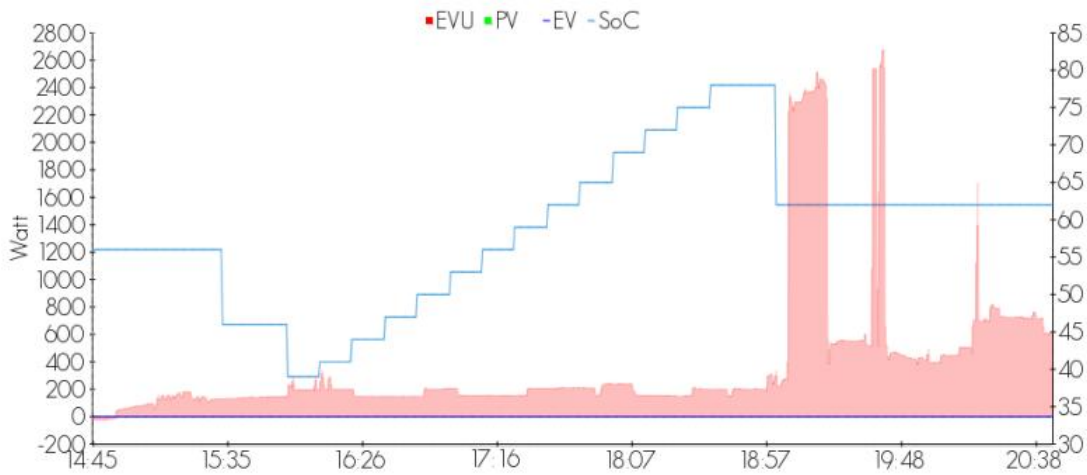
Ver 1.09 beta
Ladeleg

www.openwb.de
Hilfe

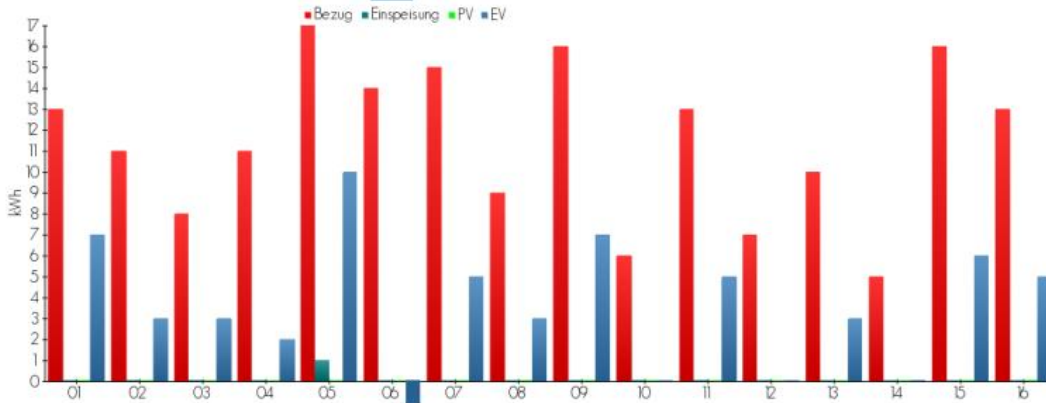
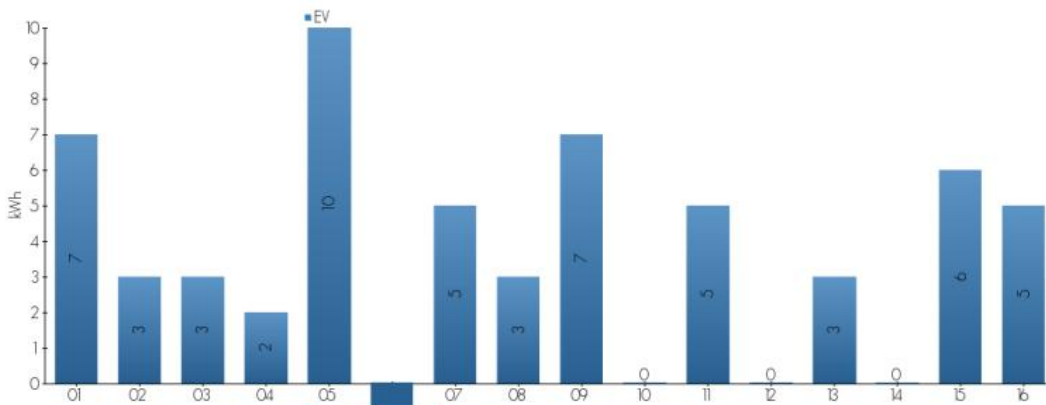
[Einstellungen](#)
[Status](#)
[Logging](#)



OpenWB Logging



Live



2018-12

Go

Monthly

openWB updaten

- auf Einstellungen -> Misc wechseln und unten links **UPDATE openWB** klicken.
Das Update dauert ca. 1 bis 2min (**bitte warten!**). Nach dem update wird die openWB-Webseite mit neuer Version automatisch neu geladen.

The screenshot shows the 'Misc' settings page in the openWB interface. At the top, there is a navigation bar with 'Zurück', 'Ladeeinstellungen', 'Modulkonfiguration', and 'Misc' (highlighted with a pink box and arrow). Below the navigation bar, there are several settings sections:

- Debugmodus:** 0 (dropdown menu). Description: 0=Debug aus, 1=Schreibe Regelwerte in das log, 2= Schreibe die Berechnungsgrundlage in das log. Das Debug Log ist [HIER](#) zu finden.
- Geschwindigkeit Regelintervall:** Normal (dropdown menu). Description: Durch verdoppeln wird das Regelintervall von 10Sek auf 55Sek gesetzt. Voraussetzung ist das alle Module schnell genug Antworten. Ebenso müssen die BEVs die geladenen werden schnell genug auf die Ladestromänderung reagieren. Sollten Probleme oder Fehlermeldungen auftauchen zunächst das Regelintervall auf Normal stellen.
- Description: Werden Module genutzt welche z.B. eine Online API zur Abfrage nutzen oder möchte man weniger regeln kann man das Regelintervall auf langsam(=20Sekunden) herabsetzen. !Bitte beachten! Nicht nur die Regelung der PV geführten Ladung sondern auch Ladestromänderung, Stop, etc.. werden dann nur noch alle 20 Sekunden ausgeführt. Die Regelung wird träger.

Logging Einstellungen

- Anzeige Daily Graph in Watt oder Wh:** Watt (dropdown menu)
- Einspeisung im Daily Graph positiv oder negativ anzeigen:** Negativ (dropdown menu)
- Zeitintervall für den Live Graphen der Hauptseite:** 30 Min (dropdown menu)

Releasechannel

- Releasechannel:** Beta (dropdown menu)
- Description: Der Stable train ist der empfohlene. Im Betazweig befinden sich die Änderungen für künftige Releases. Nightly ist der aktuelle Entwicklungszeitweig. Man kann grundsätzlich immer zwischen den Zweigen wechseln. Hierfür den gewünschten Zweig auswählen, Speichern und ein Update durchführen.

Save (green button)

Das Backup stellt im Falle eines Hardwaredefektes die Einstellungen und Ladelogfiles wieder her

- Backup erstellen** (red button)
- Backup wiederherstellen** (orange button)

Auf die neuste Version updaten, Einstellungen bleiben erhalten.
Der Update Prozess kann bis zu einer Minute dauern, je nach Internetverbindung!
Zur Sicherheit vorher ein Backup erstellen.

- UPDATE openWB** (red button, highlighted with a pink box and arrow)
- SMA Support** (red button)

Open Source made with love!
Spenden (yellow button)

Bei Nutzung von einem **Modbus(RS485)/LAN-Konverter** (z.B. USR-TCP232-410):



folgende Konfiguration verwenden:

```
Baud Rate: 9600
Data Size: 8 Bit
Parity: None
Stop Bits: 1
Flow Control and RS485: RS485
Local Port Number: 26
Remote Port Number: 26
Work Mode: TCP Server None
TCP Server detail: default Type
Timeout: 0
UART packet Time: 10 ms
UART packet length: 512 chars
```

A screenshot of the USR IOT web configuration interface. The top bar shows 'firmware revision : v3005' and '中文 logout'. The main header includes the USR IOT logo, 'Be Honest, Do Best!', and '-IOT Experts-'. A left sidebar contains navigation options: 'Current Status', 'Local IP Config', 'RS232', 'RS485' (highlighted), 'Web to Serial', 'Misc Config', and 'Reboot'. The main content area is titled 'parameter' and contains the following settings:

- Baud Rate: 9600 bps
- Data Size: 8 bit
- Parity: None
- Stop Bits: 1 bit
- Flow Control and RS485: RS485
- Local Port Number: 26
- Remote Port Number: 26
- Work Mode: TCP Server, None
- TCP Server detail: default type
- Remote Server Addr: 192.168.110.21 [N/A]
- Timeout: 0 seconds (< 256, 0 for no timeout)
- UART packet Time: 10 ms (< 256)
- UART packet length: 512 chars (<= 1460, 0 for no use)
- Sync Baudrate(RF2217 similar):

At the bottom are 'Save' and 'Cancel' buttons. A right sidebar titled 'help' provides details for 'baud', 'flowcontrol and RS485', 'local port', 'remote port', and 'packet time'.