

Open Source CCS

Implementierung
zur Nutzung der Schnellladeinfrastruktur

Linuxtag Augsburg
20 Apr 2024

Unser Hintergrund



 Johannes mit E-Touran in Greenwich 2023



Uwe mit FOCCI 2024



 Janosch, Johannes, Damian in Cork mit EV Taxi 2022

Motivation: Ladezeit für 30kWh

Type 2 AC
7kW



257 min



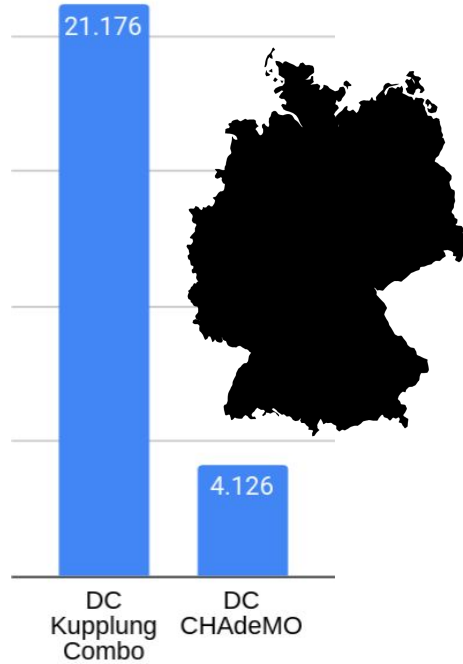
CCS Combo
60kW

30 min



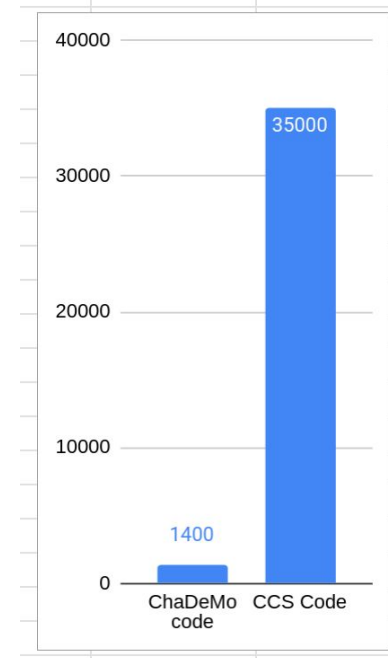
Anfahrt gestern: Kassel - Augsburg: 103 kWh

weit verbreitet



Bundesnetzagentur 11/2023

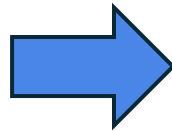
sehr komplex



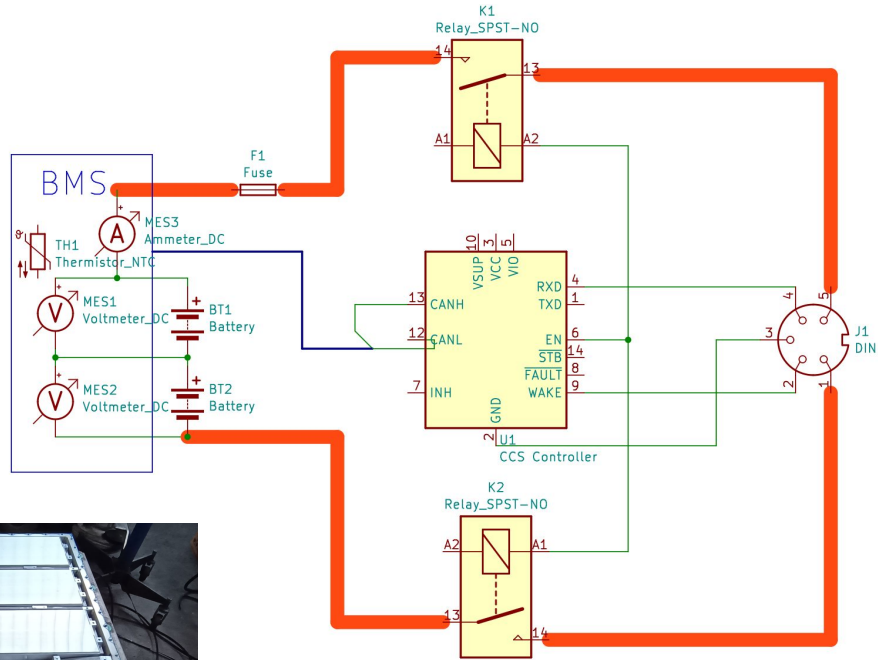
Eigene Messung 2024

Der Clou beim Schnellladen

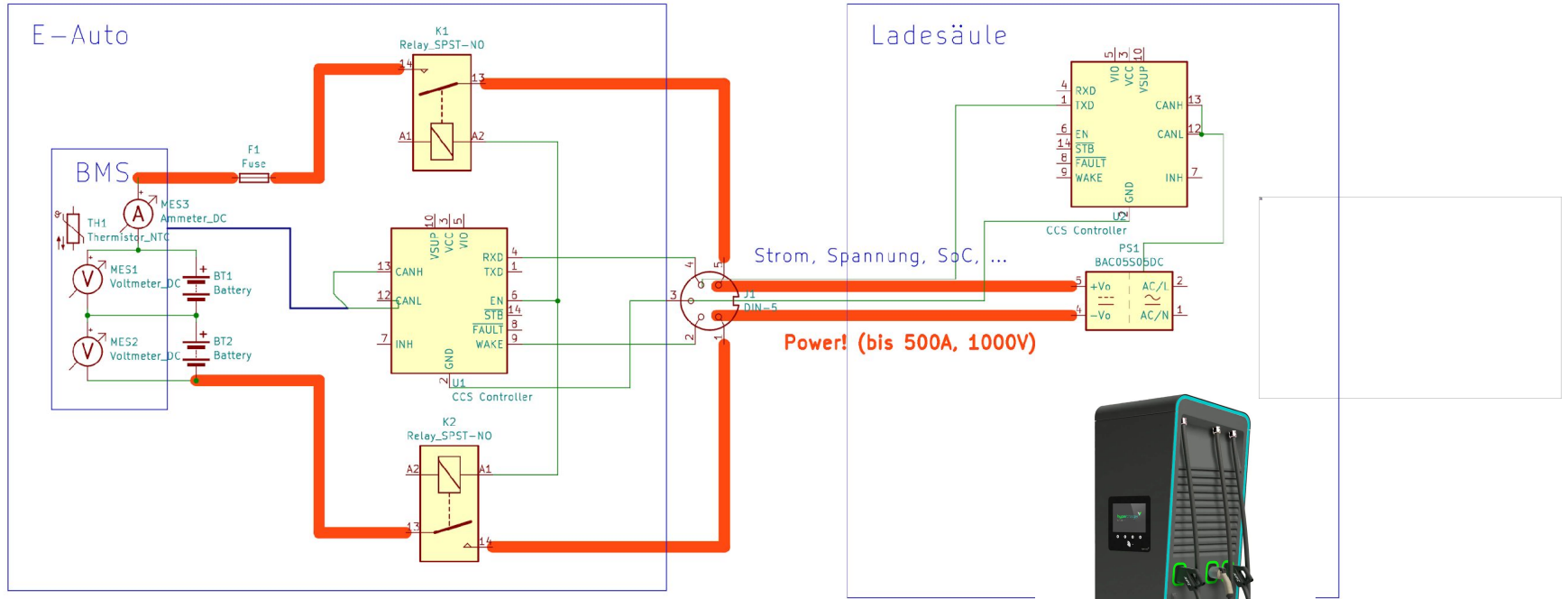
- Zum Laden muss geregelter Gleichstrom her
- Das Ladegerät wandert aus dem Auto heraus
- Dadurch kann es leistungsfähiger sein



Simple Hardware im Auto



Gesamtsystem



Uwes Teil

Geschichte von pyPLC & Co
Etwas CCS-Theorie

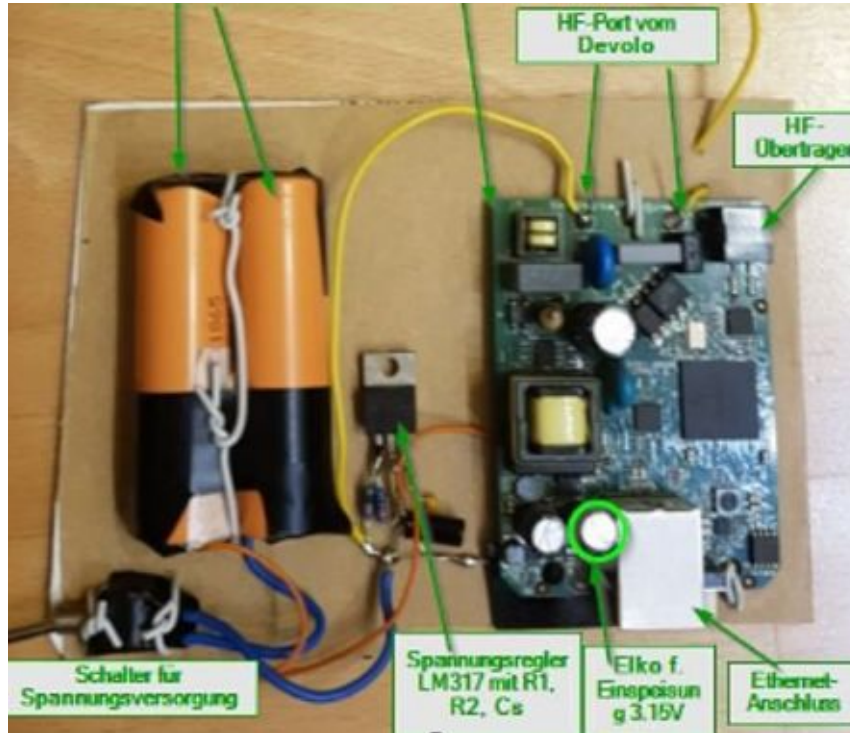
Und wie bekomme ich Strom?



Power-Line-Communication



Geschichte von pyPLC (2) – The Modem



Geschichte von pyPLC (3) - Wireshark

Time	Source	Destination	Protocol	Length	Info
6 66.72570	lonig 04:65:65:...	Broadcast	HomePlug	60	MAC Management, Unknown 0x6064
7 66.72922	Alpitronic 0a:19:4a:...	04:65:65:...	HomePlug	60	MAC Management, Unknown 0x6065
8 66.741380000	04:65:65:...	Broadcast	HomePlug	60	MAC Management, Unknown 0x6064
9 66.744374000	0a:19:4a:...	04:65:65:...	HomePlug	60	MAC Management, Unknown 0x6065
10 66.941248000	04:65:65:...	Broadcast	HomePlug	60	MAC Management, Unknown 0x606a
11 66.961388000	04:65:65:...	Broadcast	HomePlug	60	MAC Management, Unknown 0x606a
12 66.981418000	04:65:65:...	Broadcast	HomePlug	60	MAC Management, Unknown 0x606a
13 67.021358000	04:65:65:...	Broadcast	HomePlug	71	MAC Management, Unknown 0x6076
14 67.041403000	04:65:65:...	Broadcast	HomePlug	71	MAC Management, Unknown 0x6076
15 67.061248000	04:65:65:...	Broadcast	HomePlug	71	MAC Management, Unknown 0x6076
16 67.081317000	04:65:65:...	Broadcast	HomePlug	71	MAC Management, Unknown 0x6076
17 67.101239000	04:65:65:...	Broadcast	HomePlug	71	MAC Management, Unknown 0x6076
18 67.121348000	04:65:65:...	Broadcast	HomePlug	71	MAC Management, Unknown 0x6076
19 67.141246000	04:65:65:...	Broadcast	HomePlug	71	MAC Management, Unknown 0x6076
20 67.161254000	04:65:65:...	Broadcast	HomePlug	71	MAC Management, Unknown 0x6076
21 67.181388000	04:65:65:...	Broadcast	HomePlug	71	MAC Management, Unknown 0x6076
22 67.200992000	04:65:65:...	Broadcast	HomePlug	71	MAC Management, Unknown 0x6076
23 67.209336000	0a:19:4a:...	04:65:65:...	HomePlug	129	MAC Management, Unknown 0x606e
24 67.251303000	04:65:65:...	0a:19:4a:...	HomePlug	70	MAC Management, Unknown 0x606f
25 68.161289000	04:65:65:...	0a:19:4a:...	HomePlug	85	MAC Management, Unknown 0x607c
26 68.164399000	0a:19:4a:...	04:65:65:...	HomePlug	109	MAC Management, Unknown 0x607d

0x607C is CM_SLAC_MATCH, +1 die response dazu.

Geschichte von pyPLC (4) - Ziele

Use-Cases

1. Mitloggen der Kommunikation zwischen Fahrzeug und Ladesäule, um Ladeprobleme zu verstehen („ListenMode“)
2. pyPLC als Ladestation, um Fahrzeug zu laden oder entladen („EvseMode“)
3. pyPLC als Ladecontroller im Fahrzeug („PevMode“)

Das Puzzle fñgt sich zusammen

- **Dez 2021:** uhi's erster Wireshark-Mitschnitt an öffentlicher Ladestation
 - [EnBW bietet ab sofort AutoCharge an \(nicht Plug and Charge\) - Seite 22 - Öffentliche Lade-Infrastruktur - Elektroauto Forum \(goingelectric.de\)](#)
- **Feb 2022:** Catfish startet den AR7420-Thread mit modifizierten Homeplug-Modems auf openinverter.org
 - <https://www.goingelectric.de/forum/viewtopic.php?p=1754351#p1754351>
- **Mai 2022:** Johannes: „We really need a full open source implementation of CCS that lets everyone play with it until it works with all chargers. “
 - Entscheidung: python scripts □ Name „pyPLC“ (python power line communication)
- **Okt 2022:** Modem spricht erstmals mit Ioniq. Initial commit auf github.
- **Dez 2022:** Glühlampe am Alpitronic leuchtet



Weitere Anwendungsbeispiele

Laptop laden am Schnelllader

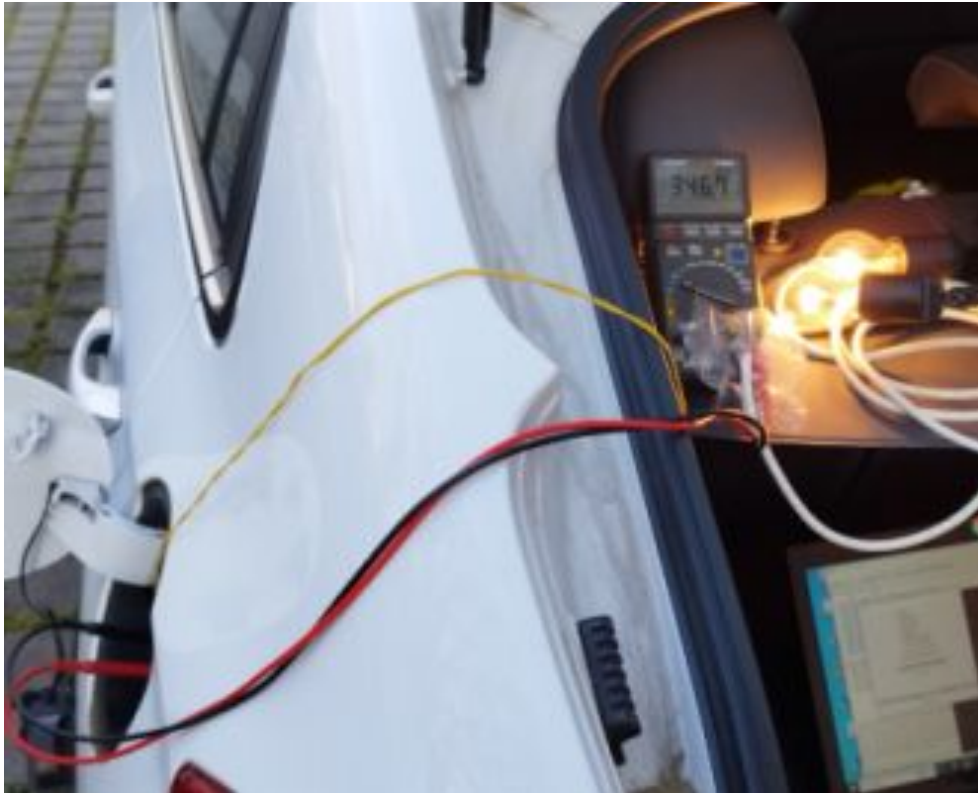


März 2023

Tee kochen am Schnelllader



Fahrzeug illuminiert Glühbirne



Elektro Umbau lädt am Schnelllader

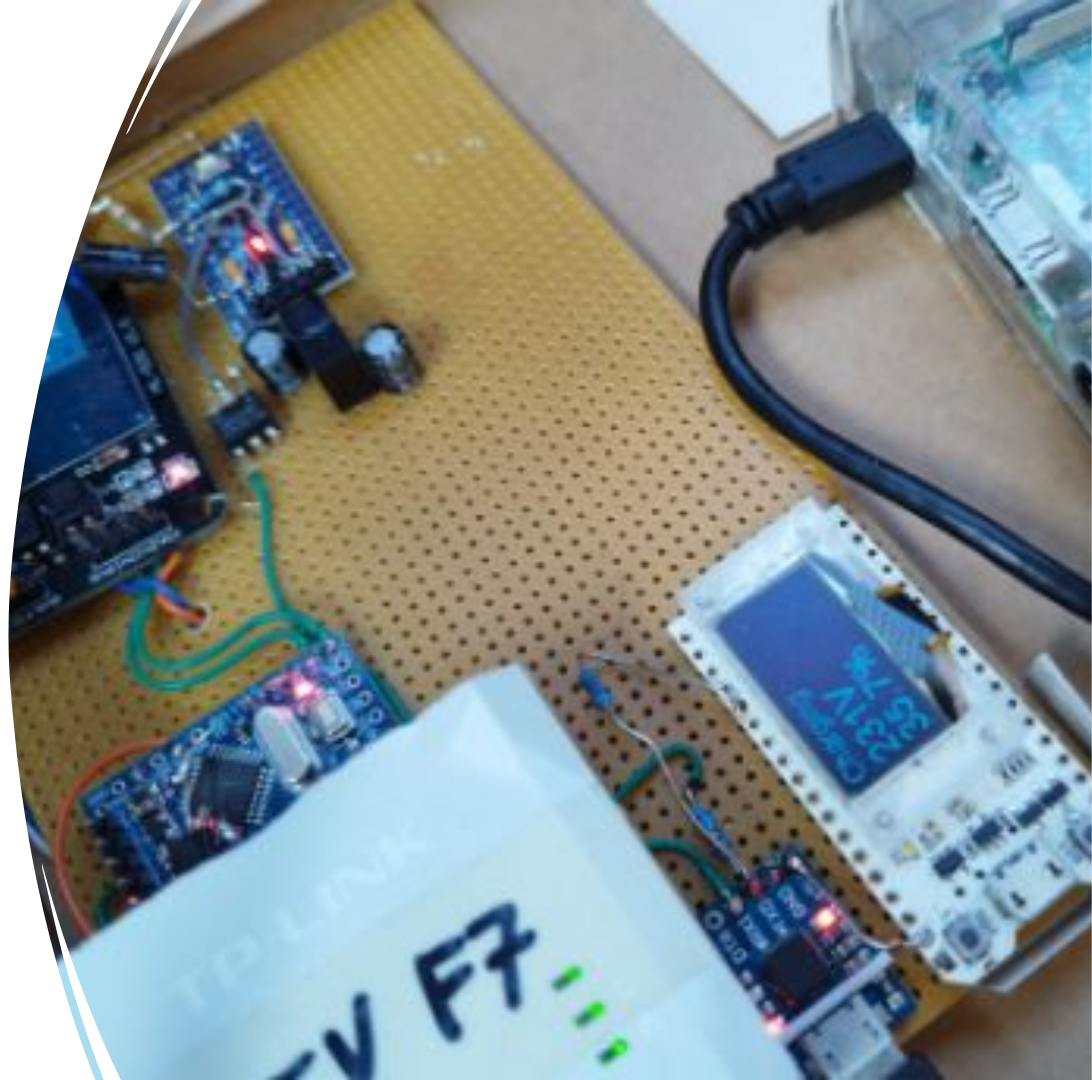


5000km Europa-Tour mit pyPLC

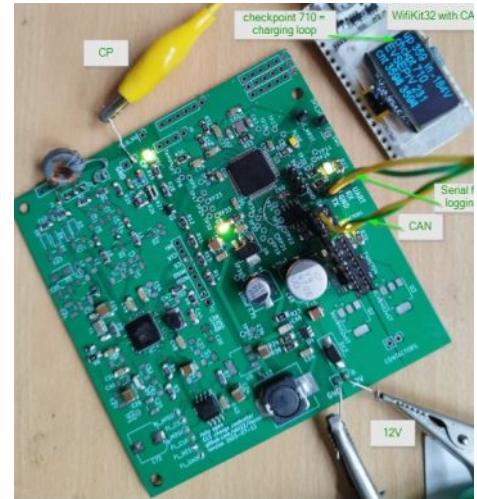
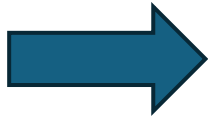


Geht's noch kleiner?

- „Sperrige“ Lösung mit
 - Homeplug-Modem
 - Raspberry o.ä.
 - Netzwirkkabel
 - Relais-Treiber
 - Spannungsversorgung
 - Interfacelogik



Der Wechsel zu Embedded



Juli –Okt 2023: embedded-Board entsteht

Das Kind braucht einen Namen

Foccci (Fully Open CCS
Charge Controller
Interface)



github.com/uhi22/foccci



Clara



github.com/uhi22/ccs32clara



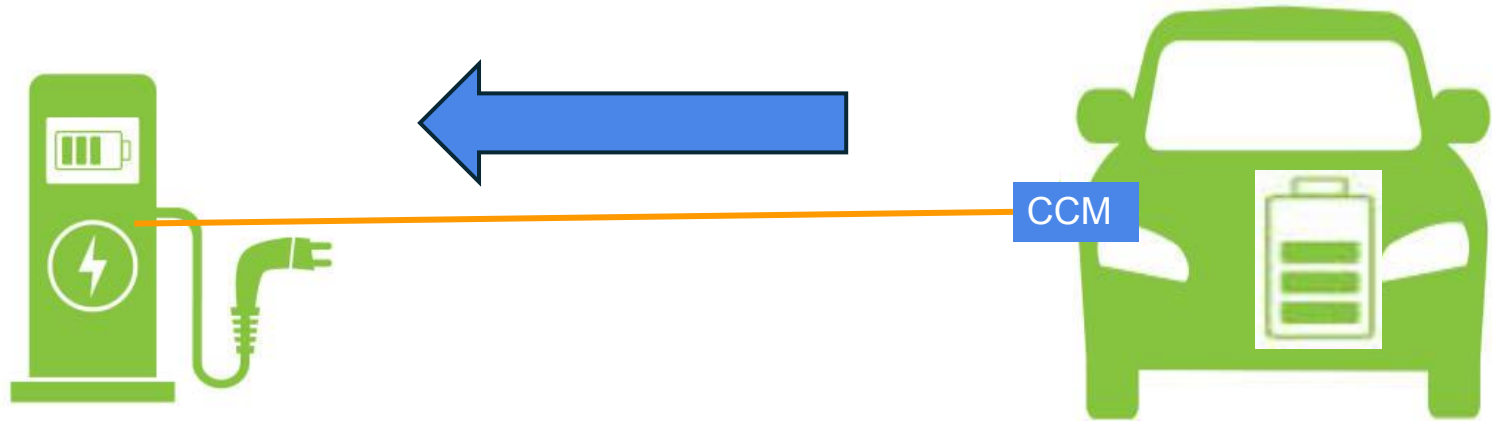
Status quo und Ausblick

- Foccci im Shop erhältlich
- In zwei Fahrzeugen erfolgreich im Einsatz, weitere im Aufbau
- Software stabil
- Verbesserungen in Richtung Stromsparen (sleep/wakeup)
- Feature-Ergänzungen je nach Bedarf
- Komplettes Kit für den Shop
- Docu verbessern

CCS – Theorie

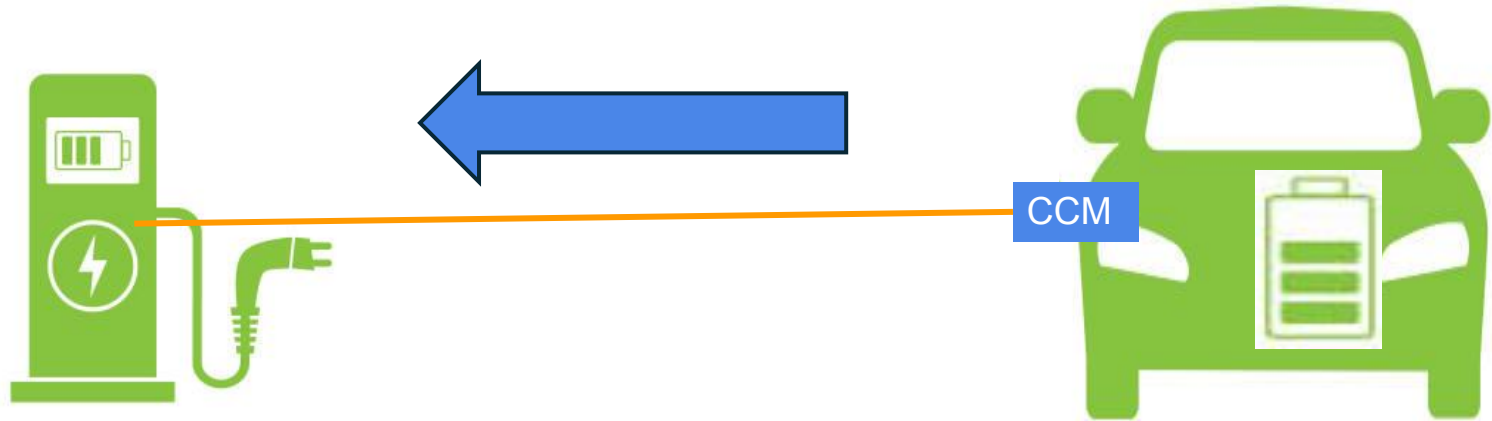
Theorie (1) - Verbindungsaufbau

SLAC, SetKey, SDP, TCP
connect



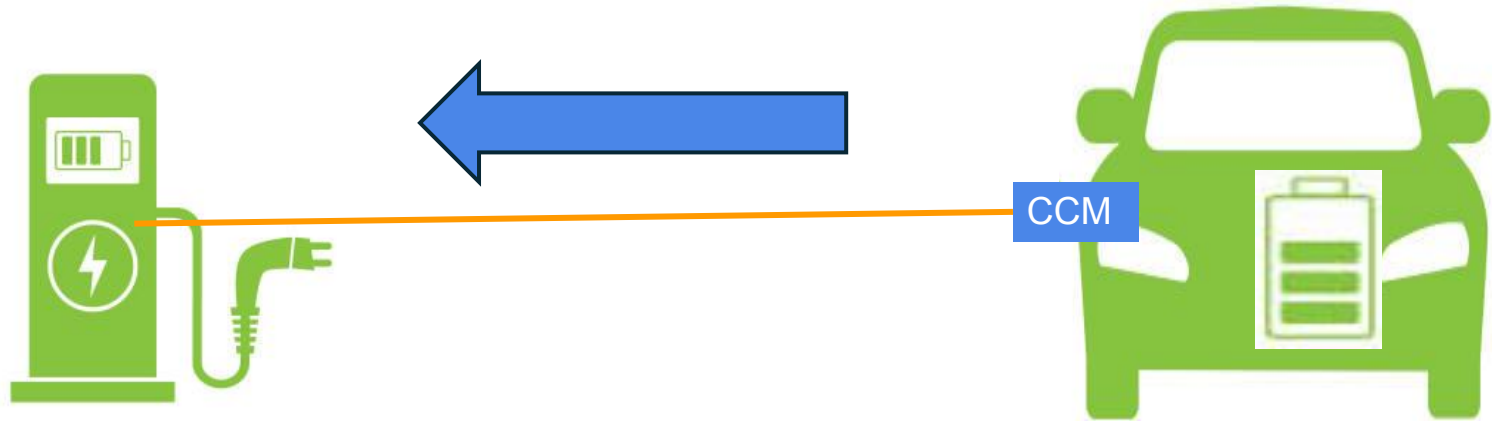
Theorie (2) – Session Setup

- Protokoll wird ausgehandelt (DIN70121, ISO15118 oder Tesla)
- Ladeparameter werden ausgehandelt (minimale und maximale Spannung, verfügbare Leistung, ...)



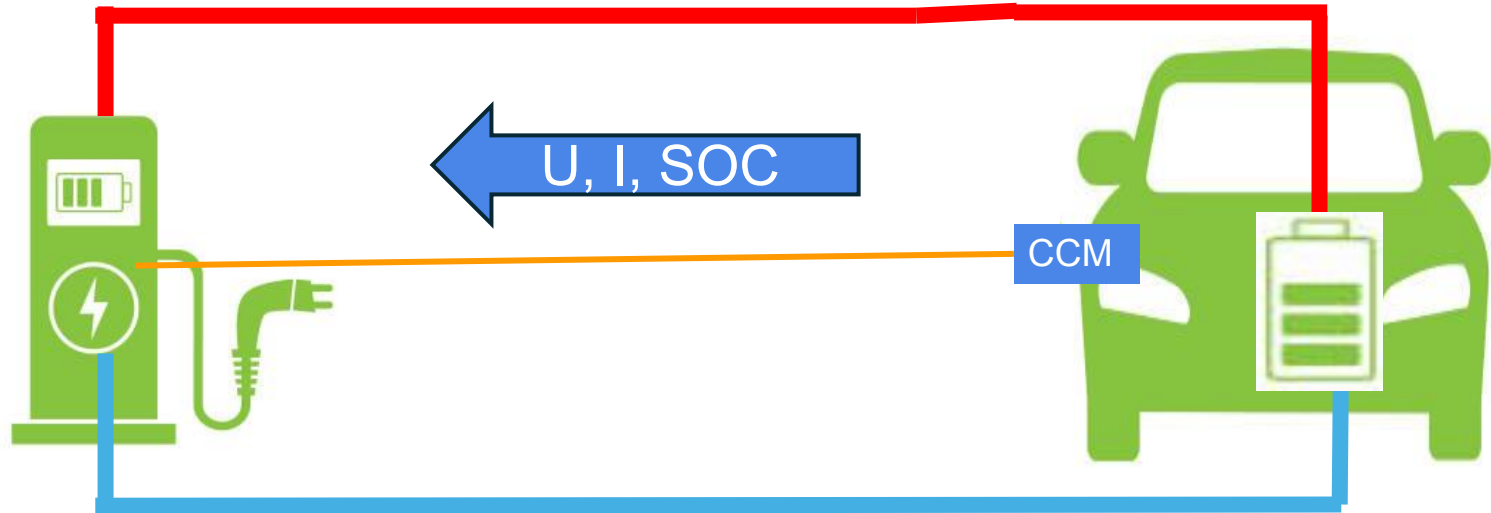
Theorie (3) – Ladevorbereitung

- Stecker wird verriegelt
- Isolationsprüfung
- PreCharge



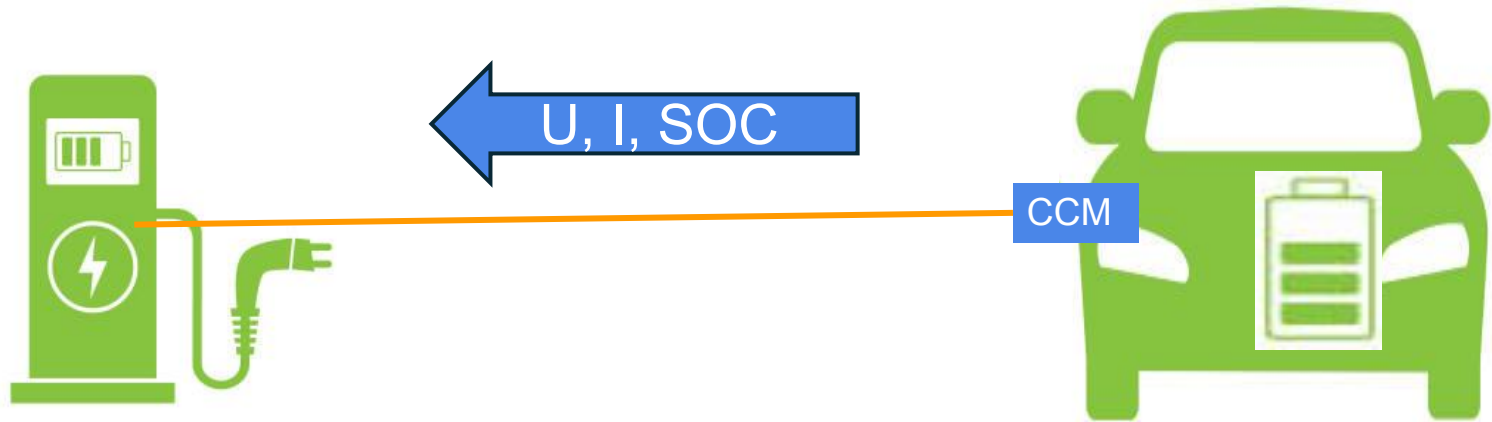
Theorie (4) – Laden

- Fahrzeug sendet gewünschten Strom und Spannung
- Ladesäule liefert Energie entsprechend der Vorgabe



Theorie (5) – Ladung beenden

- Nutzerwunsch an Säule oder Fahrzeug, oder Akku voll
- Prüfung, ob Abschaltung funktioniert
- Stecker entriegeln



Theorie (11) – Protokollstack

EXI Komprimierte XML-Datenstrukturen

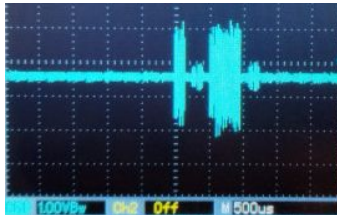
V2GTP Botschaftstyp und -größe

TCP Zuverlässige Übertragung, Wiederholung verlorengangender Nachrichten

IPv6 16-Byte IP-Adressen

Ethernet MAC-Adressen je 6 Byte

PLC



Homeplug GreenPHY:

- 1155 Träger im Bereich von 2MHz bis 30MHz
- Phasenmodulation (QPSK)
- Forward Error Correction (FEC)
- 10 Mbit/s
- Security: AES 128 und SHA-256

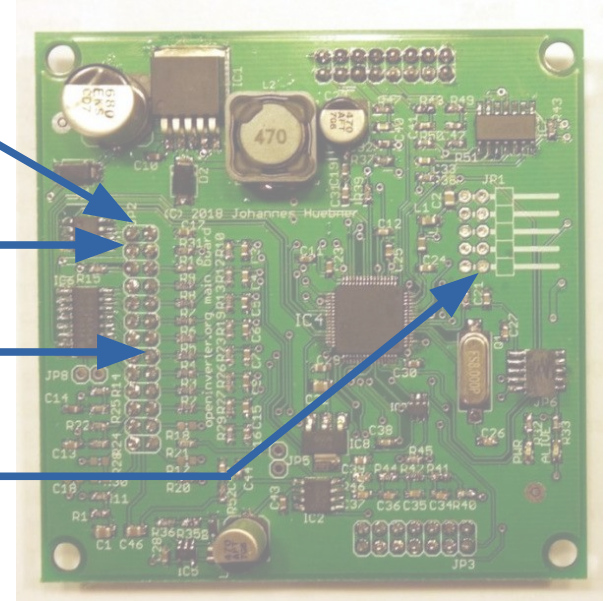
Openinverter Framework C++

- Etwas HAL für den STM32
- Bootloader
- Standardisierte Kommunikation

```
//Next param id (increase when adding new parameter!): 159
//Next value Id: 2053
/*          category      name          unit      min      max      default id */
#define MOTOR_PARAMETERS_COMMON \
PARAM_ENTRY(CAT_MOTOR, polepairs, "", 1, 16, 2, 32 ) \
PARAM_ENTRY(CAT_MOTOR, respolepairs, "", 1, 16, 1, 93 ) \
PARAM_ENTRY(CAT_MOTOR, sincosofs, "dig", 1, 4096, 2048, 131 ) \
PARAM_ENTRY(CAT_MOTOR, encmode, ENCMODES, 0, 5, 0, 75 ) \
PARAM_ENTRY(CAT_MOTOR, fmax, "Hz", 21, 1000, 200, 9 ) \
PARAM_ENTRY(CAT_MOTOR, numimp, "ppr", 8, 8192, 60, 15 ) \
```

Hardwarefunktionen

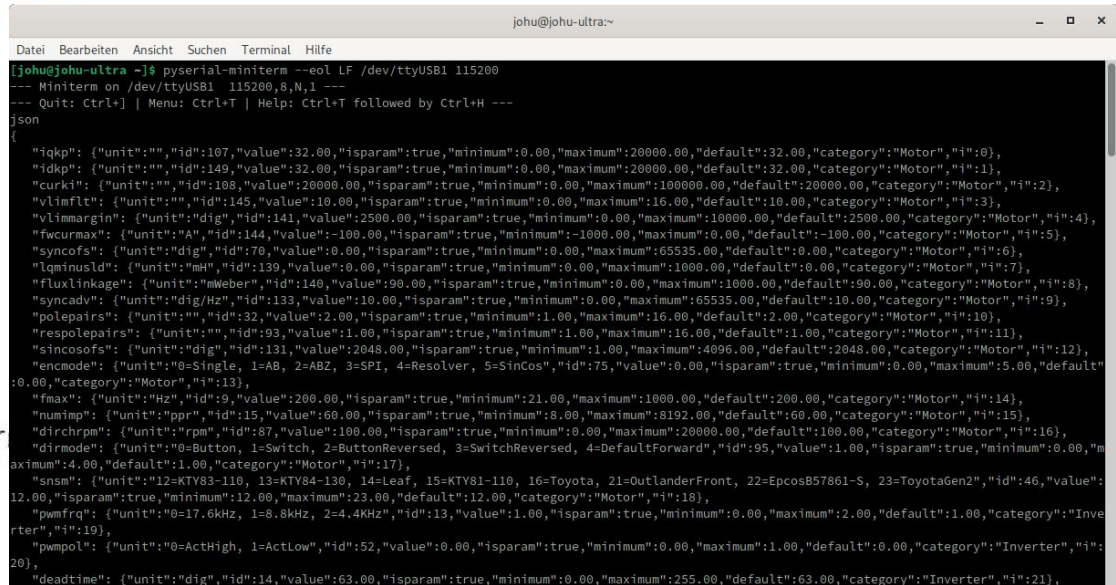
- CAN (Open) Kommunikation
- Digitale IO Pins
- Analoge Eingänge
- UART Textterminal
- LIN Kommunikation
- Effizienter Hardware Scheduler



Steuerung via Terminal

```
extern "C" const TERM_CMD TermCmds[] =
{
    { "set", TerminalCommands::ParamSet },
    { "get", TerminalCommands::ParamGet },
    { "flag", TerminalCommands::ParamFlag },
    { "stream", TerminalCommands::ParamStream },
    { "binstream", TerminalCommands::ParamStreamBinary },
    { "json", TerminalCommands::PrintParamsJson },
    { "can", TerminalCommands::MapCan },
    { "save", TerminalCommands::SaveParameters },
    { "load", TerminalCommands::LoadParameters },
    { "reset", TerminalCommands::Reset },
    { "defaults", LoadDefaults },
    { "stop", StopInverter },
    { "start", StartInverter },
    { "serial", PrintSerial },
    { "errors", PrintErrors },
    { NULL, NULL }
};
```

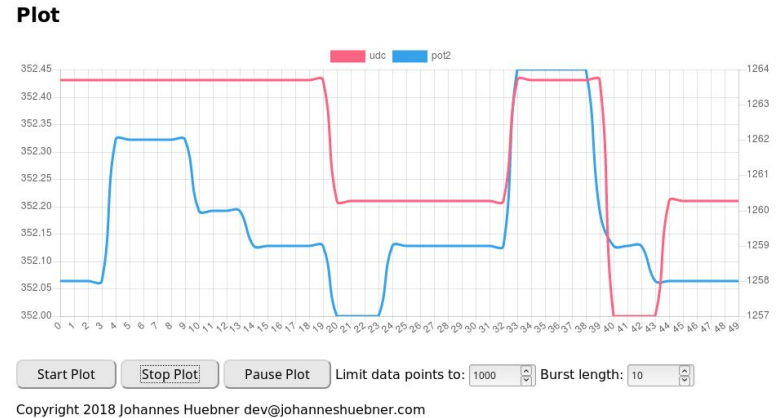
```
static void LoadDefaults(Terminal* term, char *arg)
{
    arg = arg;
    term = term;
    Param::LoadDefaults();
    printf("Defaults loaded\r\n");
}
```



```
johu@johu-ultra:~$ ./pyserial-miniterm --eol LF /dev/ttyUSB1 115200
--- Miniterm on /dev/ttyUSB1 115200,8,N,1 ---
--- Quit: Ctrl+] | Menu: Ctrl+T | Help: Ctrl+T followed by Ctrl+H ---
json
{
  "iqkp": {"unit":"","id":107,"value":32.00,"isparam":true,"minimum":0.00,"maximum":20000.00,"default":32.00,"category":"Motor","i":0},
  "idkp": {"unit":"","id":149,"value":32.00,"isparam":true,"minimum":0.00,"maximum":20000.00,"default":32.00,"category":"Motor","i":1},
  "curki": {"unit":"","id":108,"value":20000.00,"isparam":true,"minimum":0.00,"maximum":100000.00,"default":20000.00,"category":"Motor","i":2},
  "vlinflt": {"unit":"","id":145,"value":10.00,"isparam":true,"minimum":0.00,"maximum":16.00,"default":10.00,"category":"Motor","i":3},
  "vlinmargin": {"unit":"dig","id":141,"value":2500.00,"isparam":true,"minimum":0.00,"maximum":10000.00,"default":2500.00,"category":"Motor","i":4},
  "fwcurmax": {"unit":"A","id":144,"value":-100.00,"isparam":true,"minimum":-1000.00,"maximum":0.00,"default":-100.00,"category":"Motor","i":5},
  "syncofs": {"unit":"dig","id":70,"value":0.00,"isparam":true,"minimum":0.00,"maximum":65535.00,"default":0.00,"category":"Motor","i":6},
  "lqminuid": {"unit":"mH","id":139,"value":0.00,"isparam":true,"minimum":0.00,"maximum":1000.00,"default":0.00,"category":"Motor","i":7},
  "vlinlinkage": {"unit":"mWeber","id":140,"value":190.00,"isparam":true,"minimum":0.00,"maximum":1000.00,"default":190.00,"category":"Motor","i":8},
  "syncadv": {"unit":"dig/Hz","id":133,"value":10.00,"isparam":true,"minimum":0.00,"maximum":65535.00,"default":10.00,"category":"Motor","i":9},
  "polepairs": {"unit":"","id":32,"value":2.00,"isparam":true,"minimum":1.00,"maximum":16.00,"default":2.00,"category":"Motor","i":10},
  "respolepairs": {"unit":"","id":93,"value":1.00,"isparam":true,"minimum":1.00,"maximum":16.00,"default":1.00,"category":"Motor","i":11},
  "sincosofs": {"unit":"dig","id":131,"value":2048.00,"isparam":true,"minimum":1.00,"maximum":4096.00,"default":2048.00,"category":"Motor","i":12},
  "encmode": {"unit":"","id":75,"value":0.00,"isparam":true,"minimum":0.00,"maximum":5.00,"default":0.00,"category":"Motor","i":13},
  "fmax": {"unit":"Hz","id":9,"value":200.00,"isparam":true,"minimum":21.00,"maximum":1000.00,"default":200.00,"category":"Motor","i":14},
  "numimp": {"unit":"ppr","id":15,"value":60.00,"isparam":true,"minimum":8.00,"maximum":8192.00,"default":60.00,"category":"Motor","i":15},
  "dirchrpm": {"unit":"rpm","id":87,"value":100.00,"isparam":true,"minimum":0.00,"maximum":20000.00,"default":100.00,"category":"Motor","i":16},
  "dirmode": {"unit":"","id":1=Button, 1=Switch, 2=ButtonReversed, 3=SwitchReversed, 4=DefaultForward","id":95,"value":1.00,"isparam":true,"minimum":0.00,"maximum":4.00,"default":1.00,"category":"Motor","i":17},
  "snsm": {"unit":"","id=KTY83-110, 13=KTY84-130, 14=Leaf, 15=KTY81-110, 16=Toyota, 21=OutlanderFront, 22=EpcosB57861-S, 23=ToyotaGen2","id":46,"value":12.00,"isparam":true,"minimum":12.00,"maximum":23.00,"default":12.00,"category":"Motor","i":18},
  "pwmfrq": {"unit":"","id=17.6kHz, 1=8.8kHz, 2=4.4kHz","id":13,"value":1.00,"isparam":true,"minimum":0.00,"maximum":2.00,"default":1.00,"category":"Inverter","i":19},
  "pwmpl": {"unit":"","id=ActHigh, 1=ActLow","id":52,"value":0.00,"isparam":true,"minimum":0.00,"maximum":1.00,"default":0.00,"category":"Inverter","i":20},
  "deadtime": {"unit":"dig","id":14,"value":63.00,"isparam":true,"minimum":0.00,"maximum":255.00,"default":63.00,"category":"Inverter","i":21},
}
```

Web UI

- Über UART oder CANOpen
- Selbe Weboberfläche für alle openinverter Projekte
- Parameter, Anzeigewerte, Plots, Datenlogger, Firmware Upgrade
- Läuft auf ESP8266 oder ESP32














CAN Mapping

CAN Mapping

On this page you can configure the CAN mapping settings for your OpenInverter board. CAN mapping allows you to send and receive data via CAN bus. You can specify spot values that you would like to transmit on the CAN bus. Additionally you can specify spot values that you would like to set based on data received on the CAN bus.

A maximum of 8 items per CAN message can be mapped.

Existing CAN Mappings

Spot Value	Transmit or Receive	ID	Position	Length	Gain	Offset	Delete Mapping
udcdivider	Transmit	265	0	8	0	10	 Delete mapping
evsevtg	Transmit	265	8	16	1	0	 Delete mapping
evsecur	Transmit	265	24	8	1	0	 Delete mapping
opmode	Transmit	265	40	8	0.289999992	0	 Delete mapping
evsemaxvtg	Transmit	264	8	16	1	0	 Delete mapping
evsemaxcur	Transmit	264	24	8	1	0	 Delete mapping
batvtg	Receive	256	0	16	1	0	 Delete mapping
targetvtg	Receive	258	8	16	1	0	 Delete mapping
chargecur	Receive	258	24	8	1	0	 Delete mapping
soc	Receive	258	48	8	0.5	0	 Delete mapping
enable	Receive	258	40	1	1	0	 Delete mapping



DASHBOARD

UPDATE

PARAMETERS

SPOT VALUES

PLOT & GAUGE

DATA LOGGER

CAN MAPPING

WIFI SETTINGS

FILES

SUPPORT

Node ID: 22

Auto reload

firmware : 0.38.B
web : v2.2

I	Name	Value	Unit	Min	Max	Default
- Charge parameters						
13	MaxPower	100	kW	0	1000	100
14	MaxVoltage	410	V	0	1000	410
15	MaxCurrent	255	A	0	500	125
16	TargetVoltage	250	V	0	1000	0
17	ChargeCurrent	10	A	0	500	0
18	soc	0	%	0	100	0
19	BatteryVoltage	250	V	0	1000	0
20	enable	On		0	1	1
21	AcObsState	0		0	15	0
- Testing						
22	DemoVoltage	250	V	0	500	0
23	DemoControl	StandAlone		0	511	0
24	ActuatorTest	None		0	7	0
25	logging	<ul style="list-style-type: none"> • ConnMgr • Homeplug • StateMachine • Tcp • TcpTraffic • SDP • AllButTraffic • EthTraffic • All 		0	2047	1645

Save & Load

Save parameters to flash

Restore parameters from flash

Download parameters file

Load parameters from file

Parameter Database

Submit parameters to OpenInverter

Subscribe to parameter set

Stop subscription

Misc

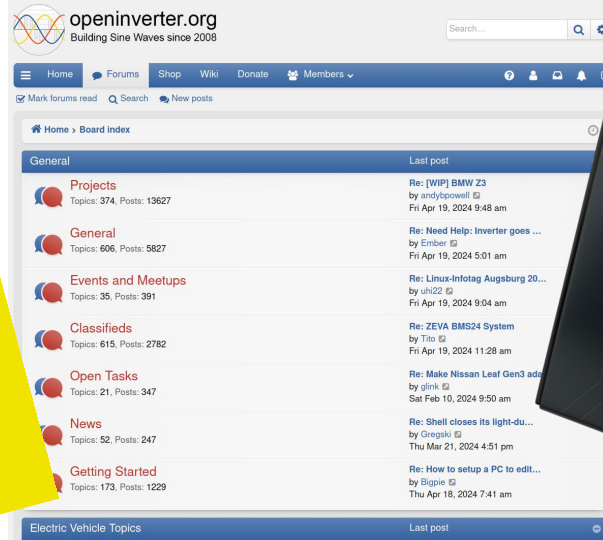
Launch syncofs tuner

Toggle category visibility

Parameter reference

Resourcen (online)

- <https://openinverter.org/>
- <https://github.com/uhi22/foccci>
- <https://foxev.io/ev-mastermind-kit>



Ressourcen (heute)

- E-Touran am Parkplatz
- OpenInverter Org CCS Stand

