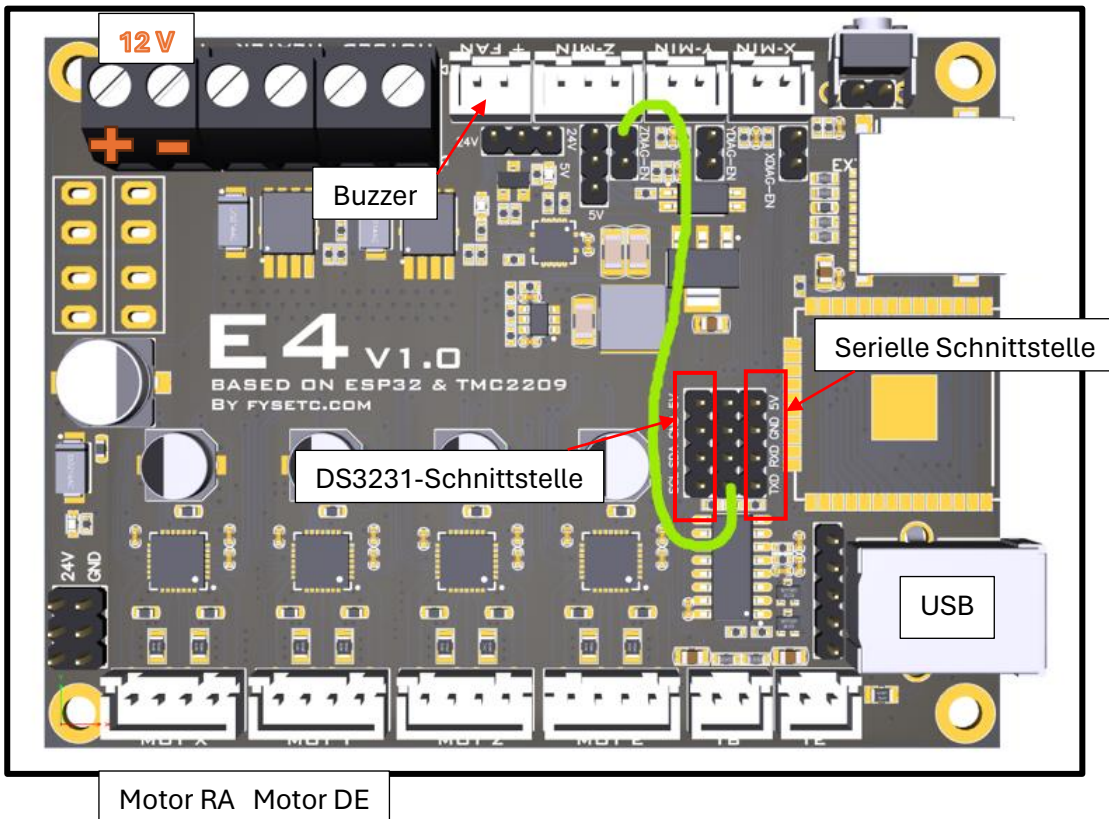


# Fysetc E4 und OnstepX

Alle Steckbrücken entfernen, nur die grüne Brücke setzen und 12V anschließen:

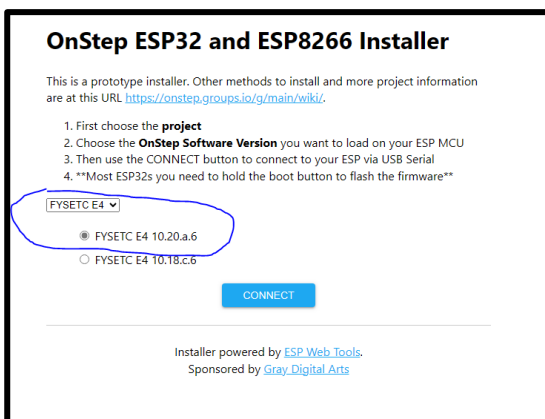


USB anschließen und folgenden Link aufrufen:

<https://graydigitalarts.com/OnStep-Web-Tools/>

(Wird das Board von Windows nicht erkannt, vorher CH340-Treiber für USB-Schnittstelle installieren)

Version auswählen, verbinden, Port auswählen und Software aufspielen



Es wird eine Standard-Installation aufgespielt, die mit dem Webbrowser später angepaßt werden kann. Die Arduino-Programmierungsumgebung wird hier nicht benötigt

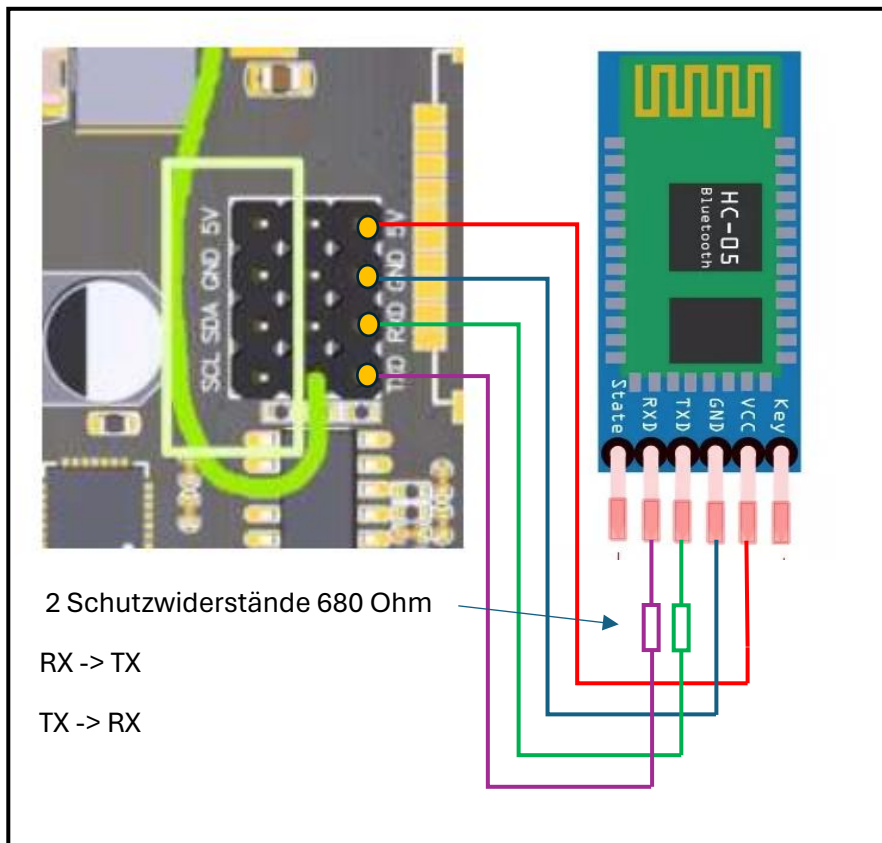
## Anschluß Bluetoothmodul HC05 oder Handbox mit serieller Schnittstelle

Es werden die RX und TX-Pins der USB-Schnittstelle genutzt

Entweder USB oder Bluetooth oder Handbox ist möglich

Bei Steuerung über PC (ASIAIR oder NINA etc.) darf Bluetooth-Modul bzw. Handbox nicht angeschlossen sein!

(Bluetoothmodul mit Level 3,3V an TX/RX und Power 3,6...6V auswählen, steht auf Rückseite)



Statt Bluetoothmodul kann auch Handbox mit serieller Schnittstelle angeschlossen werden

TX Fysetc -> Schutzwiderstand -> RX Handbox

RX Fysetc -> Schutzwiderstand -> TX Handbox

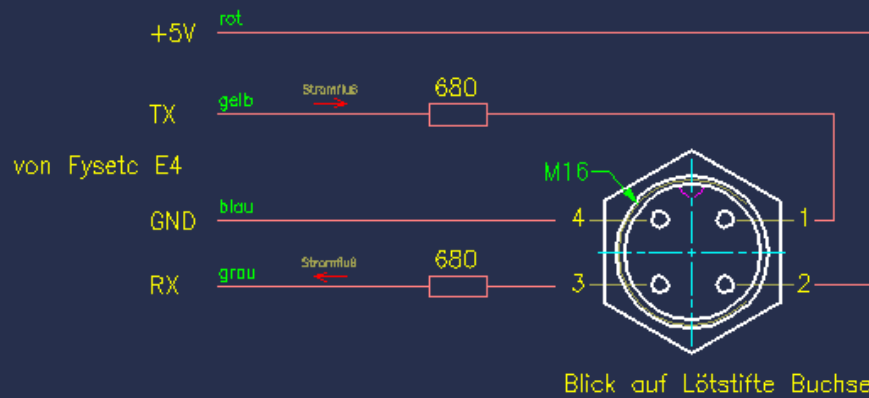
+5V / GND Fysetc -> +5V / GND Handbox

Die 680Ohm-Widerstände sollen einen Schutz darstellen, falls doch einmal USB und Bluetooth/Handbox gleichzeitig angeschlossen sind (allerdings Datenübertragung dann gestört, aber es geht nichts kaputt)

## Praktische Umsetzung:

### Buchse serielle Schnittstelle

(Aviator-Buchse mit Gewinde 16mm)

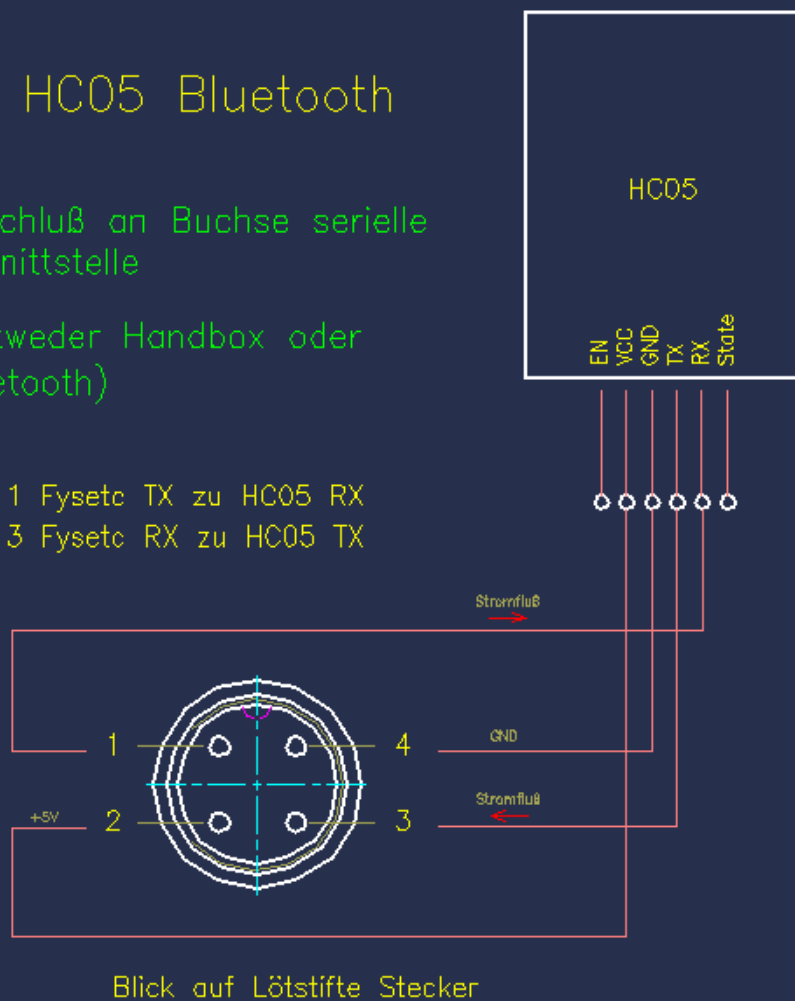


### HC05 Bluetooth

Anschluß an Buchse serielle Schnittstelle

(entweder Handbox oder Bluetooth)

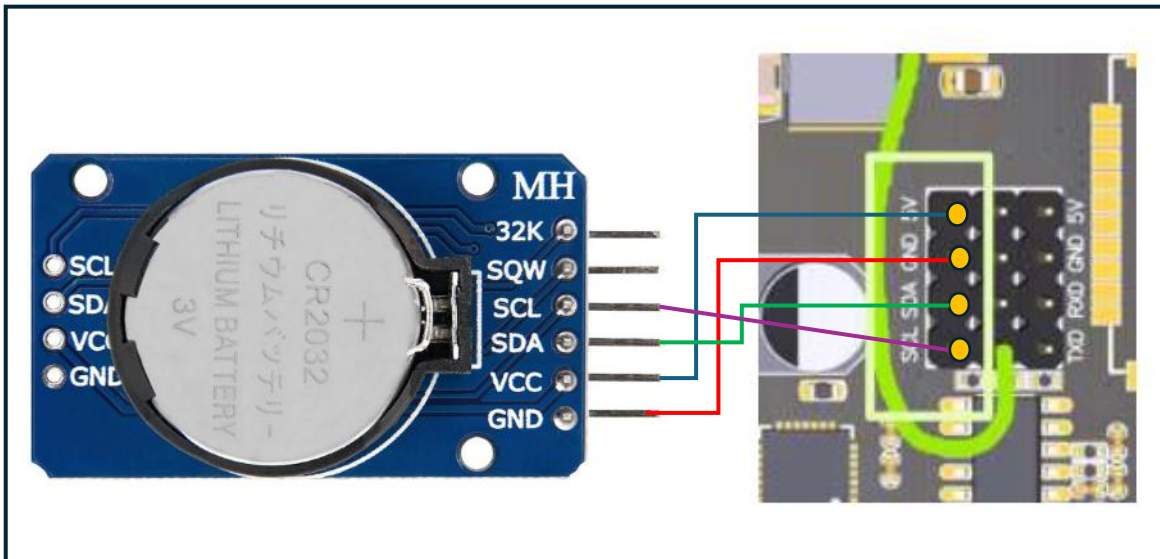
Pin 1 Fysetc TX zu HC05 RX  
Pin 3 Fysetc RX zu HC05 TX



(Handbox hat gleiche Belegung, siehe Schaltplan am Ende)

## Anschluß RTC-Modul DS3231

Echtzeituhr

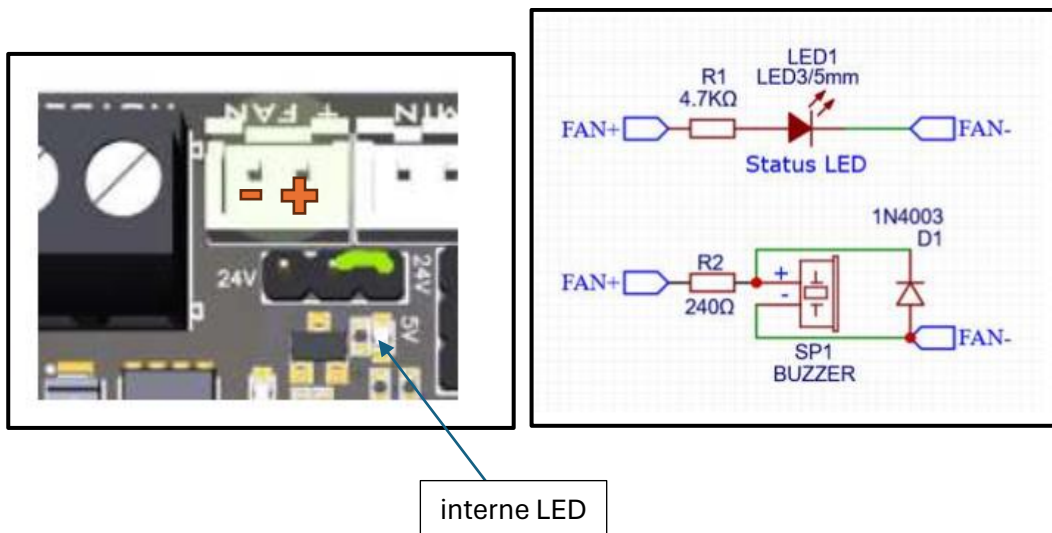


## Anschluß LED und Summer

Grünen Jumper setzen, um Schnittstelle und interne LED zu aktivieren

Buzzer (passiv) an FAN anschließen, bei Bedarf auch externe LED am gleichen Anschluß

externe LED und interne LED haben gleiche Funktion -> ist Buzzer aktiv, leuchten die LED



## **Erstkonfiguration mit Bluetooth**

Handy Einstellungen Bluetooth: verfügbare Geräte anzeigen lassen

Mac-Adresse wird kurz angezeigt -> merken (zumindest die letzten 4 Stellen) für OnStep-App

kurz danach ist die MAC-Adresse weg und es steht HC-05 in Anzeige -> Handy damit verbinden

Passwort ab Werk: 1234

GPS einschalten am Handy

Onstep-App öffnen, 3 Punkte oben anklicken, und auf Connection gehen: Mac-Adresse vom HC05 suchen und bestätigen

Nach Verbindungsaufbau wieder 3 Punkte oben anklicken

Beobachtungsort und Zeitzone eingeben, Daten vom Handy übernehmen (current Position)

Limits setzten: über Kopf +90, Horizont -10, Ost und West jeweils +15° (d.h. Montierung kann 15° über den Meridian laufen, bevor Flip notwendig wird)

Startmenü Initialize/Park: Uhr stellen

OnStep-App schließen

## **Weitere Konfiguration mit Handbox**

Mit Bluetooth ließ sich der Buzzer nicht aktivieren, das ging aber mit Handbox:

Settings -> Buzzer -> Buzzer on

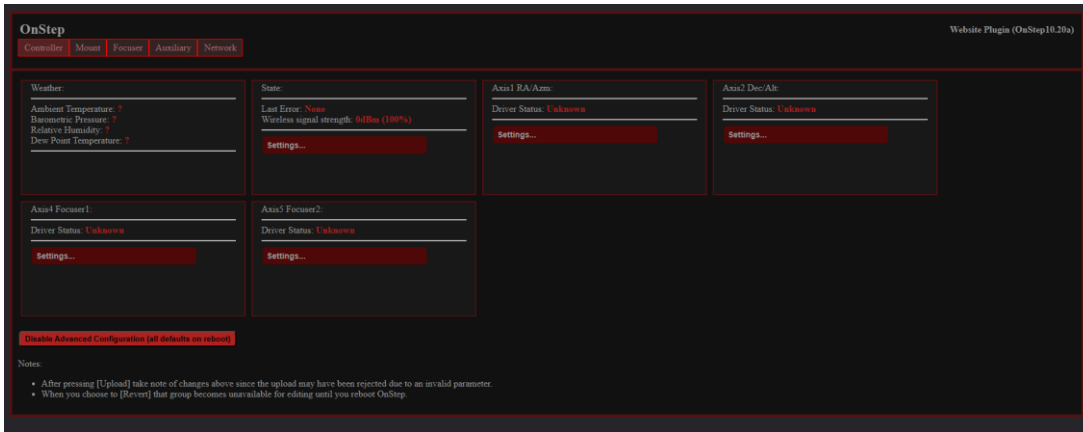
**Weiter mit Webbrowser (Motordaten anpassen, WLAN einrichten)**

# Konfiguration mit Webbrowser

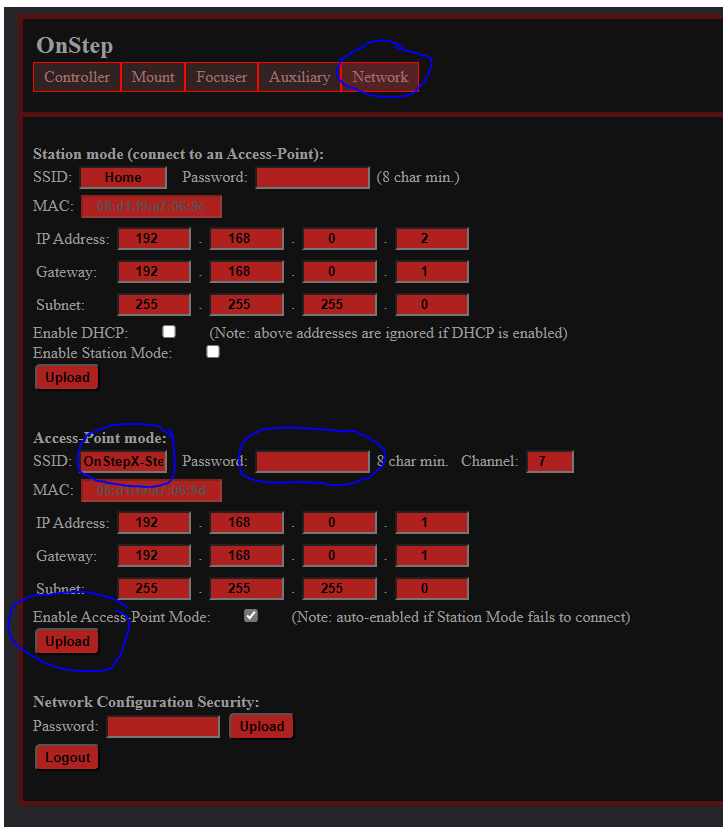
Fysetc baut WLAN-Netzwerk mit Kennung OnStepX auf, damit verbinden

Passwort: password

Im Browser 192.168.0.1 aufrufen, es erscheint die Homepage des Webrowsers



bei Network SSID und Passwort ändern (damit man auf Teleskoptreffen sein eigenes Gerät finden kann und kein Fremder sich einloggen kann)



Mit Upload bestätigen, Fysetc ausschalten, kurz warten, wieder einschalten und mit PC mit den neuen Daten verbinden

Montierungstyp und Motordaten eingeben (aus Excel-Sheet)

**OnStep**

Controller Mount **Focuser** Auxiliary Network

**Weather:**

Ambient Temperature: ?  
 Barometric Pressure: ?  
 Relative Humidity: ?  
 Dew Point Temperature: ?

**State:**

Last Error: **None**  
 Wireless signal strength: 0dBm (100%)

**Settings...**

Mount select 1 GEM, 2 EQ Fork, or 3 Alt/Azm:

**Upload** **Revert to Defaults**

**OnStep Reboot Control:**

**Reboot OnStep!**  
**Wipe all settings and Reboot OnStep!**

**Notes:**

- OnStep must be At Home or Parked, otherwise the reboot request is ignored
- If you just rebooted OnStep please wait for several seconds for everything to startup again, then check to be sure you're still connected to WiFi.

**Axis1 RA/Azm:**

Driver Status: **Unknown**

**Settings...**

Steps per worm rotation  
 Steps per degree  
 Rev. direction 0 false or 1 true  
 °, Minimum position  
 °, Maximum position

**Step/Dir Driver:**

A value of -1 is equivalent to Config.h OFF. Use caution with these!

Microsteps  
 Microsteps Goto  
 mA Hold Current  
 mA Tracking Current  
 mA Slewing Current

**Upload** **Revert to Defaults**

**Axis2 Dec/Alt:**

Driver Status: **Unknown**

**Settings...**

Steps per degree  
 Rev. direction 0 false or 1 true  
 °, Minimum position  
 °, Maximum position

**Step/Dir Driver:**

A value of -1 is equivalent to Config.h OFF. Use caution with these!

Microsteps  
 Microsteps Goto  
 mA Hold Current  
 mA Tracking Current  
 mA Slewing Current

**Upload** **Revert to Defaults**

**Axis4 Focuser1:**

Driver Status: **Unknown**

**Settings...**

Steps per micron  
 Rev. direction 0 false or 1 true  
 mm, Minimum position  
 mm, Maximum position

Einzeln durchklicken und mit Upload bestätigen

Motorströme: siehe Config.h im Arduino-Code

// for TMC2130, TMC5160, TMC2209, TMC2226 STEP/DIR driver models:

```
#define AXIS1_DRIVER_IHOLD      OFF //  OFF, n, (mA.) Current during standstill. OFF uses IRUN/2.0
Option
```

```
#define AXIS1_DRIVER_IRUN      500 //  OFF, n, (mA.) Current during tracking, appropriate for
stepper/driver/etc.  Option
```

```
#define AXIS1_DRIVER_IGOTO      OFF //  OFF, n, (mA.) Current during slews. OFF uses IRUN.
Option
```

Das -1 im Webbrowser entspricht OFF im Code

Ich setzte für RA und DE als Startwert und teste damit erstmal:

DRIVER\_IHOLD        200mA

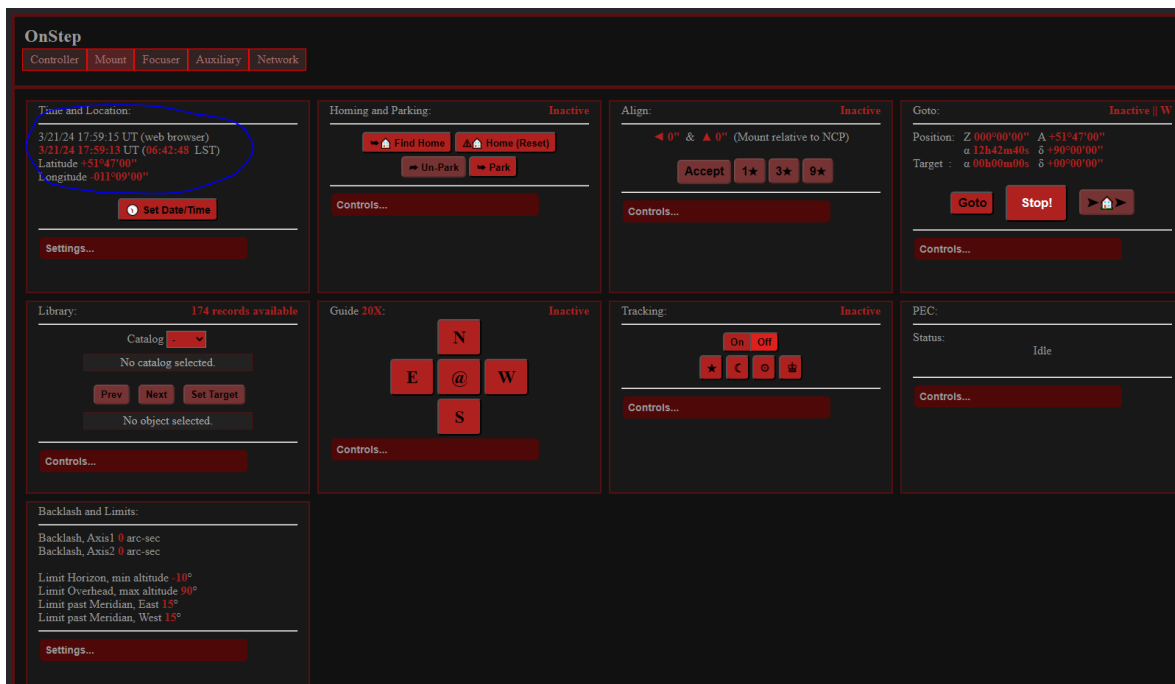
DRIVER\_IRUN        500mA

DRIVER\_IGOTO       800mA

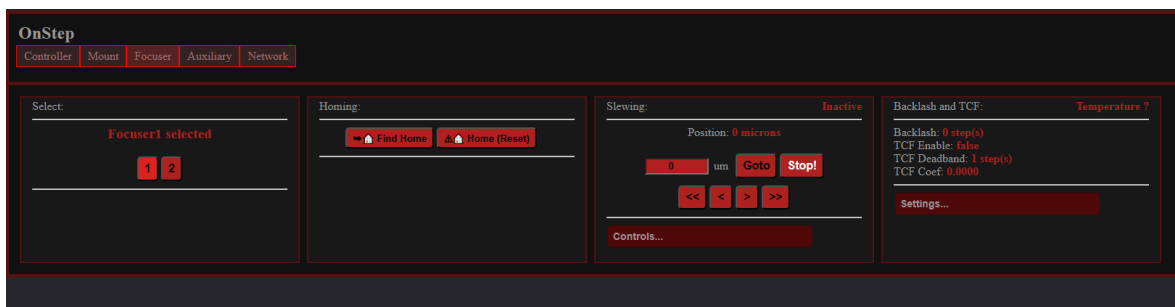
(Hat für alle meiner Montierungen mit alter Steuerung bisher gereicht)

Mount-Seite: Beobachtungsort und Zeit sind schon von Handy-App vorhanden, die anderen Knöpfe zum Testen, ob sich alles bewegt. Ist die Handbox angeschlossen, sieht man das dort auch in der Anzeige.

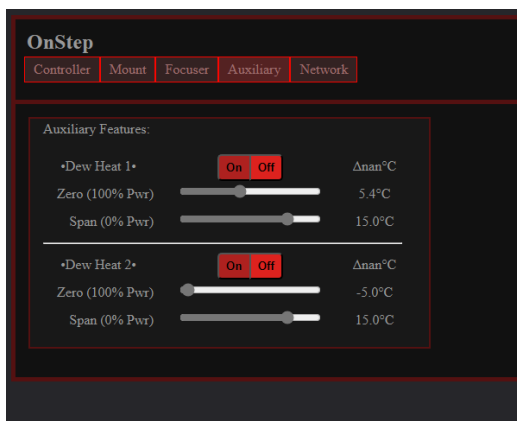
Umgekehrt sieht man im Webbrowser an der Anzeige, wenn man an der Handbox etwas macht.



Wer Fokussierer hat, hier durchklicken:



Wer Heizbänder steuern will, ist hier richtig (setzt allerdings Temperatursensoren voraus)





Um auf die Network-Seite zu kommen, ist als Passwort password einzugeben (hat nichts mit dem geänderten WLAN-Passwort zu tun)

Um das auch zu ändern, unten neu eingeben und Upload klicken:

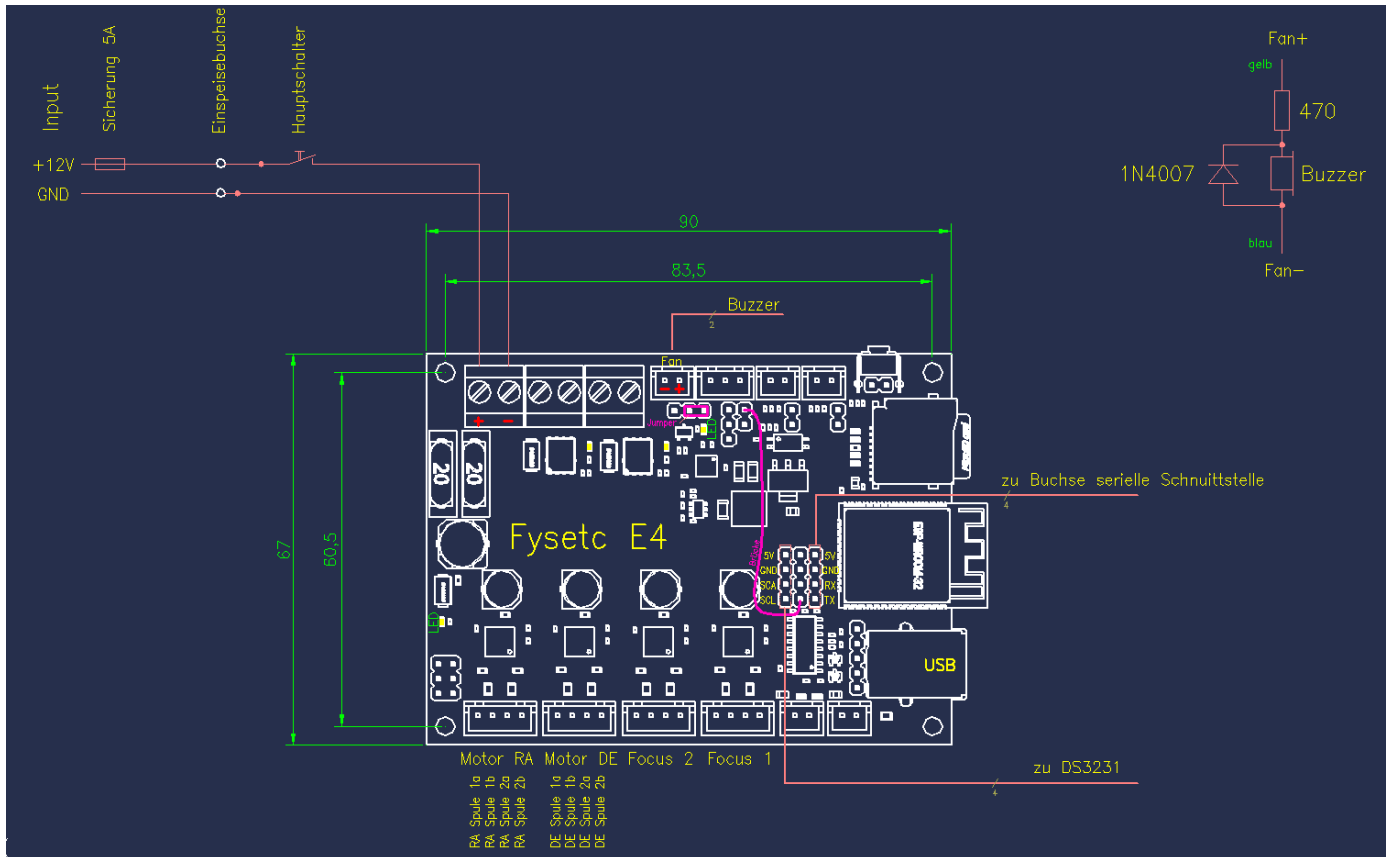
(habe WLAN und Network Configuration Security Passwort identisch gemacht, um nicht durcheinander zu kommen)

Zum Testen Logout drücken

Ich brauche den Webbrowser nur, um die Motordaten anzupassen, den Rest kann ich mit Handbox oder Bluetooth machen.

21.03.2024 STEKU

## Schaltpläne:



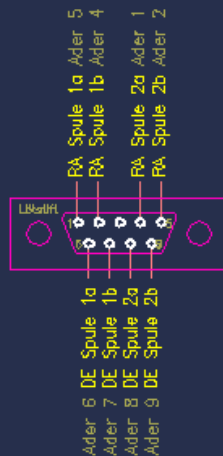
## Motorkabelanschluß

Sub-D 9-polig

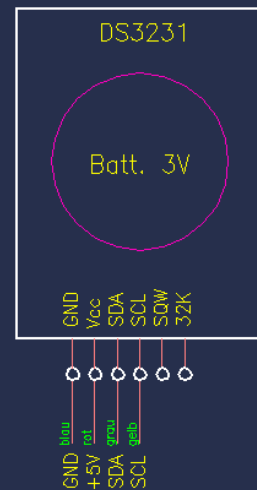
Buchse im Steuerungsgehäuse  
Stecker am Motorkabel

Blick auf Lötstifte Buchse

Motorkabel Ölflex 8+PE x 0,5  
(Adernummer vom Ölflex-Kabel)  
(Ader 3 und gelb/grün frei)



## DS3231 RTC



# Handbox V1

Anschluß an serielle Schnittstelle  
(entweder Handbox oder Bluetooth oder PC per USB anschließbar)

- statt Teeny 4.0 kann auch Teeny 3.2 benutzt werden
- wegen Speichermangel sind dann aber manche Kataloge gekürzt
- RX2/TX2 wird dann zu RX3/TX3 (Serial3 im Programmcode eintragen)

Buchse/Stecker Handbox

von Fysete E4

Handbokkabel 1,6m  
Ölflex 4x0,5

Ader 1 4 gr/gr  
Ader 2 3 gel/gel

Black auf Lötfläche Stecker  
(Adernummer vom Ölflex-Kabel)

OLED 1,3"

Teeny 4.0

+5V

100nF

100µF

R-Netzwerk 7x4,7k

Steuertasten  
(beide gegen GND)

F2 rechte H. Taste  
F1 linke H. Taste  
F (Shift)  
W Rv+  
D Rv-  
S RS-  
N RS+  
DE+

Shift weiße Taste

GND

LED

Kartensteck-LED für Bestückungstest

Bestückungsseite

Leitersseite

F1/F2: Taster T113A RT oder T113A GR oder T113A SW  
Shift/H/S/O/W: RAFI 200.507 (grün mit Schraubenschluß)  
oder RAFI 300.507 (grün mit Lötlanschluß)

Handbox  
1591CGY

Höhe: 40

Dimensions (mm):

- Overall width: 65
- Overall height: 120
- Top edge: 120, 108, 88, 82, 67, 51, 35, 31
- Bottom edge: 18 von unten, 15, 0
- Left edge: 32,5, 18, 0
- Right edge: 32,5, 18, 0

Internal dimensions and features:

- Top edge: 17,5, 15,25, 15,25, 17,5
- Center: 0
- Top edge: 0, 15,25, 17,5
- Center: 0
- Bottom edge: 0, 18, 32,5
- Right edge: 18, 32,5

Component locations and labels:

- Ø3 auf Halteschnur halber Höhe
- OLED: 1,3"
- 35
- 20
- 16
- N
- D
- W
- S
- Markierungs LED
- F1
- F2
- F (Shift)
- 18 von unten
- 15
- 0
- Ø10 (Verschraubung)
- 15 von unten

## Anpassung Config.h für Smart Hand Controller Version 4.00c

Config.h Originalcode:

```
// SERIAL PORTS -----
#define SERIAL_ONSTEP SERIAL_ST4 // .._ST4, SERIAL_ST4 for ST4 port sync comms, you can also use any other available Infreq
// serial port (if pins are unused,) Serial3 on the Teensy3.2 or SERIAL_IP
// for example. This is the serial interface connected to OnStep.
#define SERIAL_ONSTEP_BAUD_DEFAULT 9600 // 9600, Common baud rates for these parameters are 9600,19200,57600,115200. Infreq
// Only used for async serial communication with OnStep.
```

Geänderter Code:

```
Config.h - Editor
Datei Bearbeiten Format Ansicht Hilfe
// -----
// Configuration for OnStep Smart Hand Controller (SHC)
/*
 * For more information on setting OnStep up see http://www.stellarjourney.com/index.php?r=site/equipment_onstep
 * and join the OnStep Groups.io at https://groups.io/g/onstep
 *
 * *** Read the compiler warnings and errors, they are there to help guard against invalid configurations ***
 */
// -----
// ADJUST THE FOLLOWING TO CONFIGURE YOUR ADD-ON'S FEATURES -----
// <-Req'd = always must set, <-Often = usually must set, Option = optional, Adjust = adjust as req'd, Infreq = infrequently changed

// Parameter Name Value Default Notes Hint
// DISPLAY -----
#define DISPLAY_LANGUAGE L_en // L_en, English. Specify language with two letter country code, if supported. Adjust
#define DISPLAY_OLED SH1106 // SH1106, OLED 1.3" I2C display commonly used. SSD1306 is a 0.96" OLED display. Infreq
// The SSD1309 is often used on 1.54" or 2.3" I2C displays.
// SH1106 hat I2C-Adresse 0x78
// Anschlüsse am Display beachten, verschiedene Ausführungen beim gleichen Typ

// SERIAL PORTS -----
#define SERIAL_ONSTEP Serial2 // Anpassung KBSK 21.07.2023
// Serial2 für Teensy 4.0 Serial3 für Teensy 3.2
// Teensy 4.0 hat mehr Katalogdaten als Teensy 3.2 (siehe Catalogs.config.h)
// beide Versionen getestet -- > ok
// Teensy 3.2: --> CPU-Speed 72MHz
// Optimize: Smallest Code
// Teensy 4.0: --> CPU-Speed 600MHz
// Optimize: Faster
#define SERIAL_ONSTEP_BAUD_DEFAULT 9600 // 9600, Common baud rates for these parameters are 9600,19200,57600,115200. Infreq
// Only used for async serial communication with OnStep.
```

Schnittstelleneinstellung Arduino-IDE 1.8.19 mit Teensy-Umgebung

