PiCAN-M User Guide

OpenPlotter / Signal K Installation Guide

January 2021

Product name   PiCAN-M with NMEA 0183 and NMEA 2000 Connection

Model number   RSP-PICAN-M

Manufacturer   SK Pang Electronics Ltd

Contents

Table of Contents

SK Pang electronics

1. Introduction

This is a PiCAN-M with NMEA 0183 and NMEA 2000 Connection. The NMEA 0183 (RS422) is via a 5 way screw terminal. The NMEA 2000 is via Micro-C connector.

Extra sensors can be connected via the Qwiic (I2C) connector.

Optional 3A SMPS. The 12v from the NMEA 2000 network can be use to power the PiCAN-M and the Raspberry Pi.

Easy to install SocketCAN driver. Programming can be done in C or Python.

1.1. Features

- CAN connection via Micro-C connector
- 120Ω terminator ready
- SocketCAN driver
    o appears as can0 to application
- NMEA 0183 (RS422) via 5-way screw terminal
    o appears as ttyS0 to application
- LED indicator (GPIO22)
- Qwiic (I2C) connector for extra sensors
- Optional 3A SMPS.
- Compatible with OpenCPN and Signal K
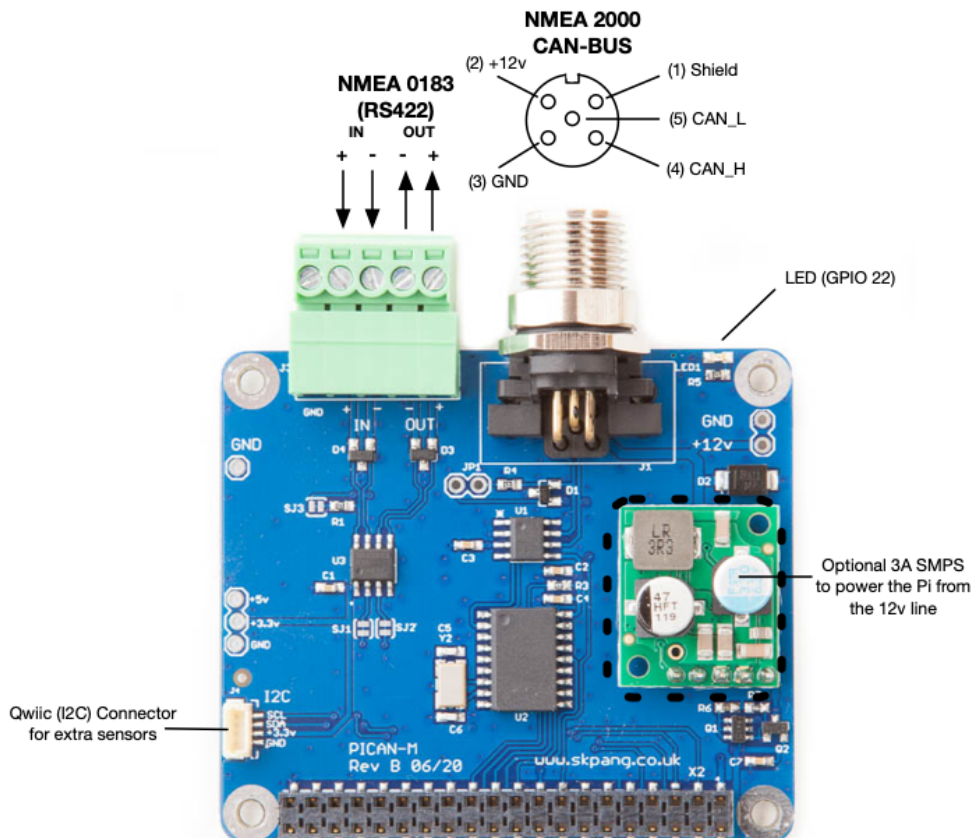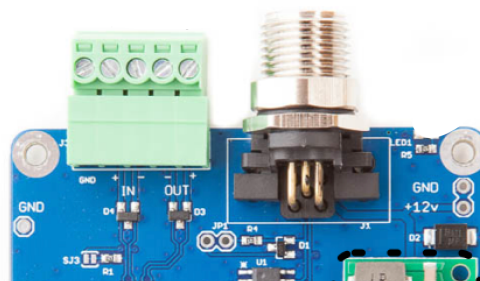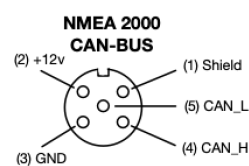
2. Hardware Installation

Before installing the board make sure the Raspberry is switched off. Carefully align the 40way connector on top of the Pi. Use spacer and screw (optional items) to secure the board.
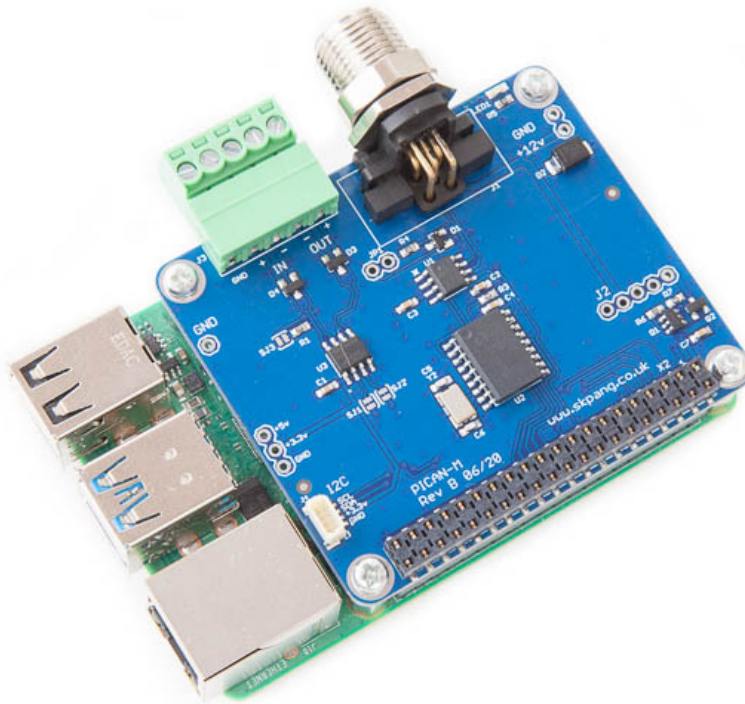
1.2. NMEA 2000 CAN-BUS Connection

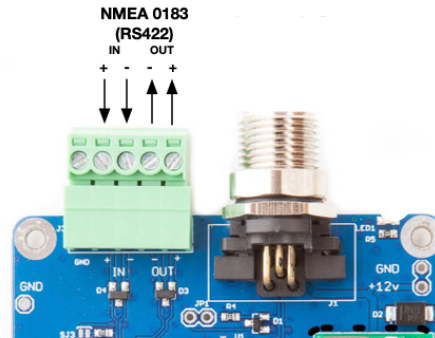The CAN-bus connections are made via J1 a 5 pin Micro-C connector.

J1	Function
1	Shield
2	+12v
3	GND
4	CAN_H
5	CAN_L

1.3. NMEA 0183 (RS422) Connection

The NNEA 0183 (RS422) connection are made via J3 a 5way screw terminal.

J3	Function
1	GND
2	IN +
3	IN -
4	OUT -
5	OUT +

1.4. 120Ω Terminator

There is a 120Ω resistor fitted to the board. To use the terminator solder a 2way header pin to JP1 then insert a jumper.
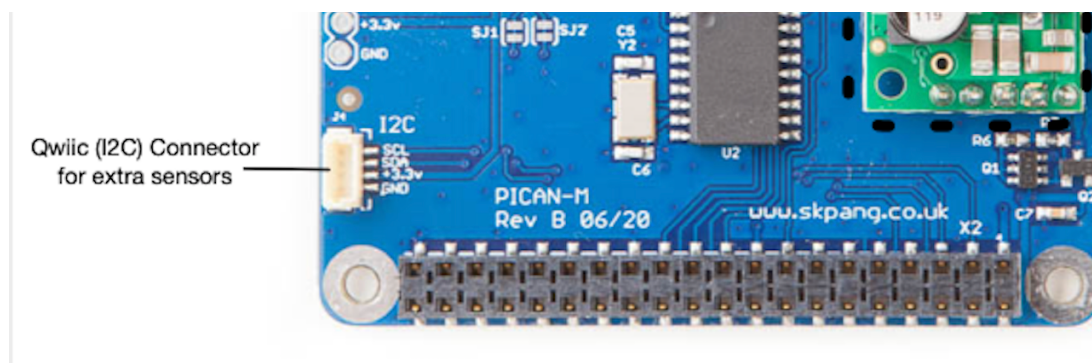
1.5. LED

There is a red LED fitted to the board. This is connected to GPIO 22.

1.6. Switch Mode Power Supply (optional)

This is an optional 5v 3A module that can power the Pi. It has an input voltage range of 7v to 24v.

1.7. Qwiic (I2C) Connector J4

J4 is a Qwiic (I2C) connector. The I2C bus can be use for extra sensors.

3. Software Installation

If you want a simple installation of OpenPlotter / Signal K then use a pre-configure SD image instead.

Download the 2020-12-16-OpenPlotter-v2-Starting-img.zip from

https://cloud.openmarine.net/s/sL9doDML7P4CQDo

Download and install Raspberry Pi Imager.

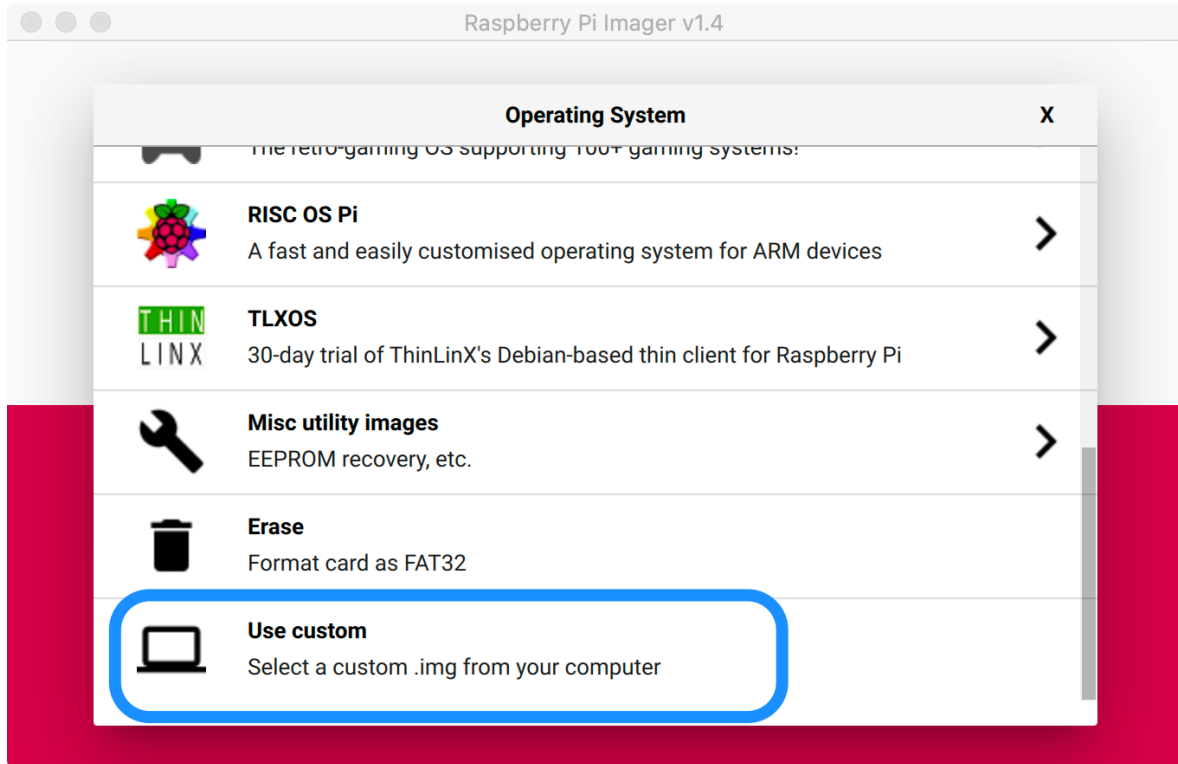https://www.raspberrypi.org/software/

Put a SD card in a reader and start Raspberry Pi Imager.

Click Choose OS.

Then click Use custom.

Select the zip file you have just downloaded and click CHOOSE SD CARD. Select your SD card and click WRITE.

Once the SD card write has finished remove it from your PC and insert into the Raspberry Pi and power up.

The first boot will take some time. You can set new password and system language.

Next we need to add entries to config.txt

Start a new terminal and add the overlays by:

`sudo nano /boot/config.txt`

Add these lines to the end of file:

`enable_uart=1`

`dtparam=i2c_arm=on`

`dtparam=spi=on`

`dtoverlay=mcp2515-can0,oscillator=16000000,interrupt=25`
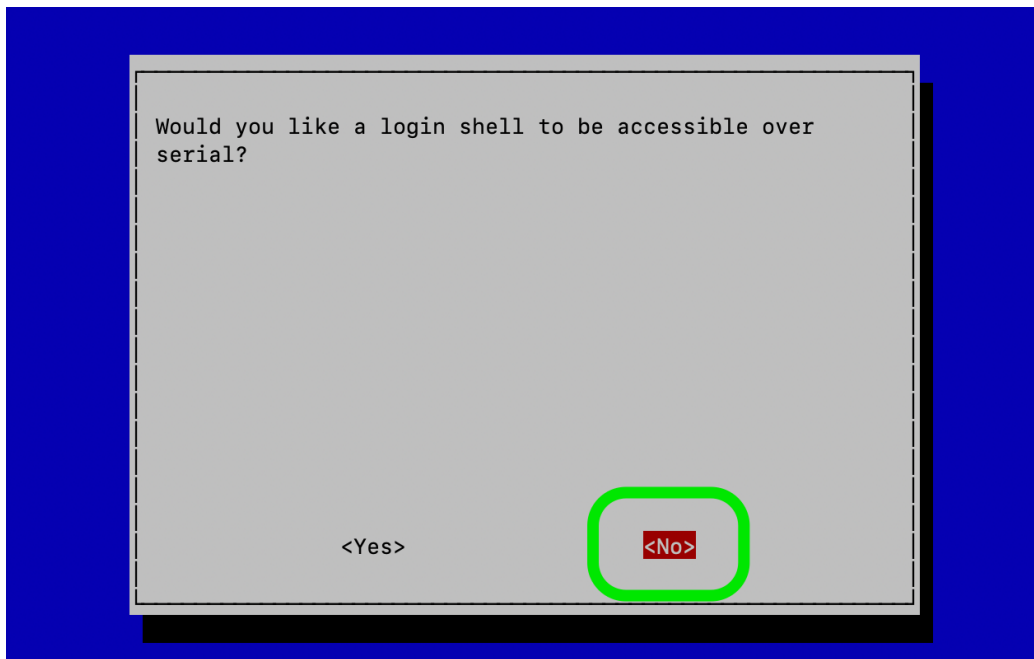
`dtoverlay=spi-bcm2835-overlay`

Press CTRL-x to save and exit.

Next we need to configure the UART by:

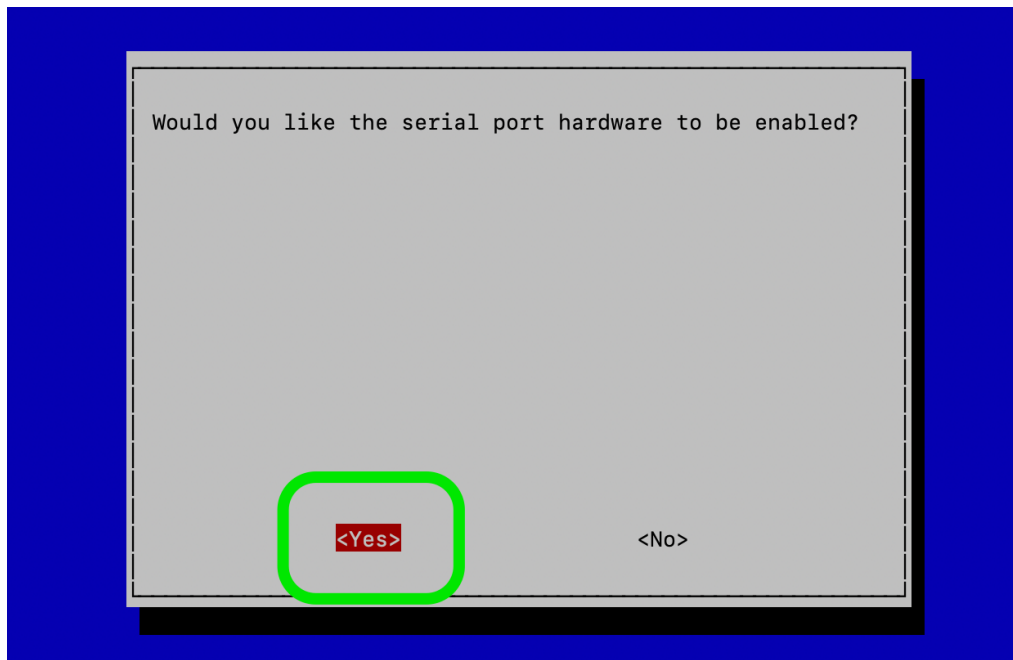`sudo raspi-config`

Select 5 Interfacing Options, P6 Serial

"Would you like a login shell to be accessible over serial ?" Select No.

"Would you like the serial port hardware to be enabled ?" Select Yes.

```
Would you like the serial port hardware to be enabled?



                        <Yes>              <No>
```

Select finish and reboot the Pi.

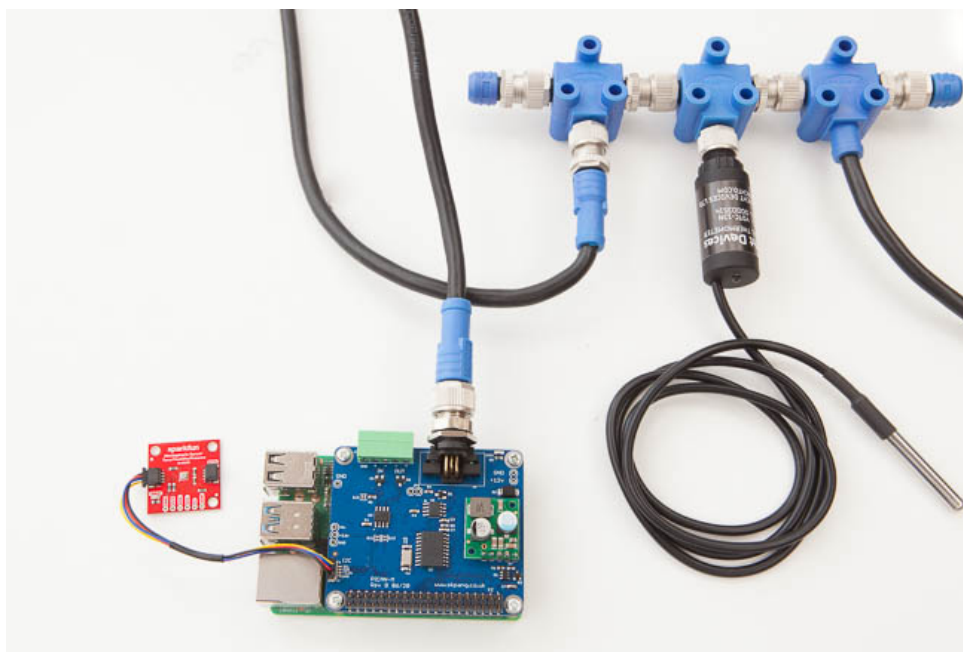### 1.8. Installing CAN Utils
Install the CAN utils by:

```
sudo apt-get install can-utils
```

1.9. Bring Up the Interface
You can now bring the CAN interface up with CAN 2.0B at 250kbps:

```
sudo /sbin/ip link set can0 up type can bitrate 250000
```

Connect a NMEA 2000 device to the network and start candump.

```
candump can0
```

You should see something like this:

```
pi@openplotter:~ $ sudo /sbin/ip link set can0 up type can bitrate 250000
pi@openplotter:~ $ candump can0
  can0  15FD0634   [8]  FF FF FF CE 73 FF FF FF
  can0  15FD0734   [8]  FF C1 CE 73 FF 7F FF FF
  can0  15FD0634   [8]  FF FF FF CE 73 FF FF FF
  can0  15FD0734   [8]  FF C1 CE 73 FF 7F FF FF
  can0  15FD0834   [8]  FF 00 01 CE 73 FF FF FF
  can0  15FD0C34   [8]  FF 00 01 0C 86 04 FF FF
  can0  15FD0634   [8]  FF FF FF CE 73 FF FF FF
  can0  15FD0734   [8]  FF C1 CE 73 FF 7F FF FF
  can0  15FD0634   [8]  FF FF FF CE 73 FF FF FF
  can0  15FD0734   [8]  FF C1 CE 73 FF 7F FF FF
  can0  15FD0634   [8]  FF FF FF CE 73 FF FF FF
  can0  15FD0734   [8]  FF C1 CE 73 FF 7F FF FF
  can0  15FD0634   [8]  FF FF FF CE 73 FF FF FF
  can0  15FD0734   [8]  FF C1 CE 73 FF 7F FF FF
  can0  15FD0834   [8]  FF 00 01 CE 73 FF FF FF
  can0  15FD0C34   [8]  FF 00 01 0C 86 04 FF FF
  can0  15FD0634   [8]  FF FF FF D4 73 FF FF FF
  can0  15FD0734   [8]  FF C1 D4 73 FF 7F FF FF
  can0  15FD0634   [8]  FF FF FF CE 73 FF FF FF
  can0  15FD0734   [8]  FF C1 CE 73 FF 7F FF FF
```

This capture is from a Yacht Devices Digital Thermometer NMEA 2000 YDTC-13N

To automatically bring up can0 at boot up we need to edit the interface by typing in:

```
sudo nano /etc/network/interfaces
```
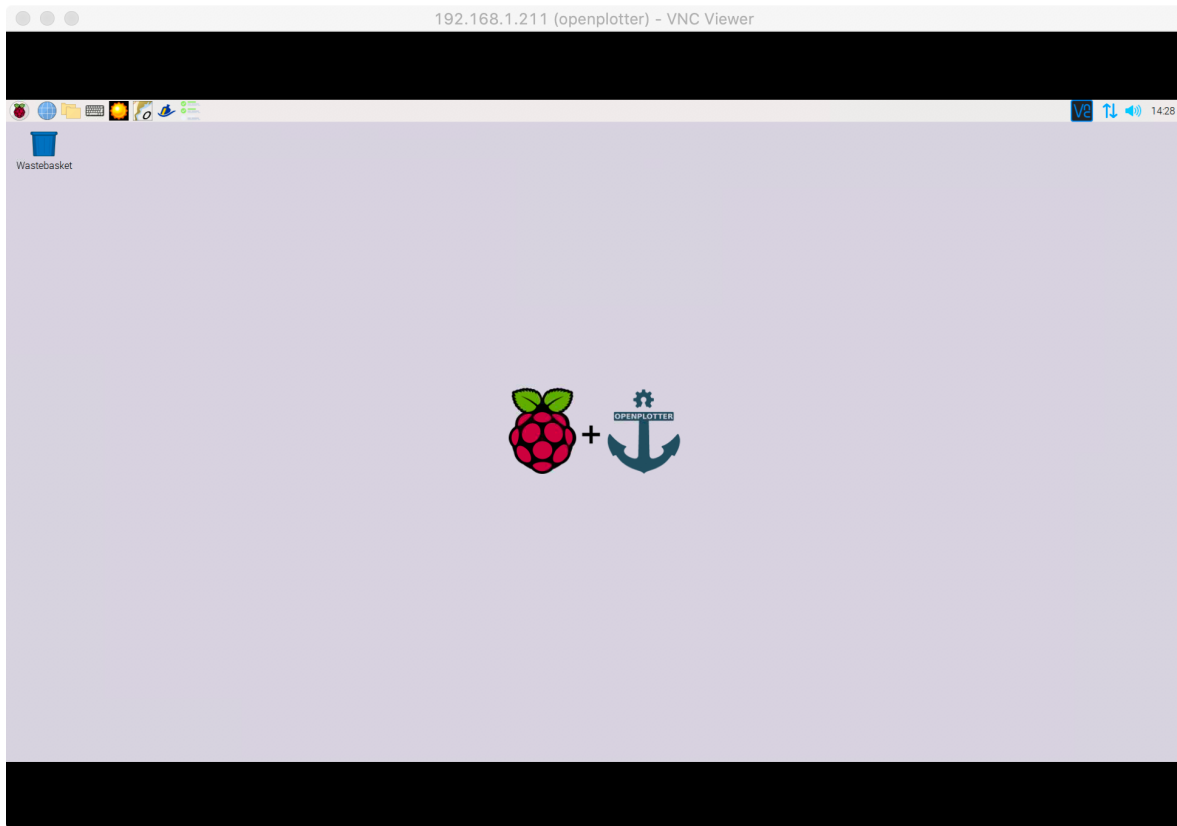
Add these lines to the end of file:

```
auto can0
iface can0 inet manual
    pre-up /sbin/ip link set can0 type can bitrate 250000
    up /sbin/ifconfig can0 up
    down /sbin/ifconfig can0 down
```
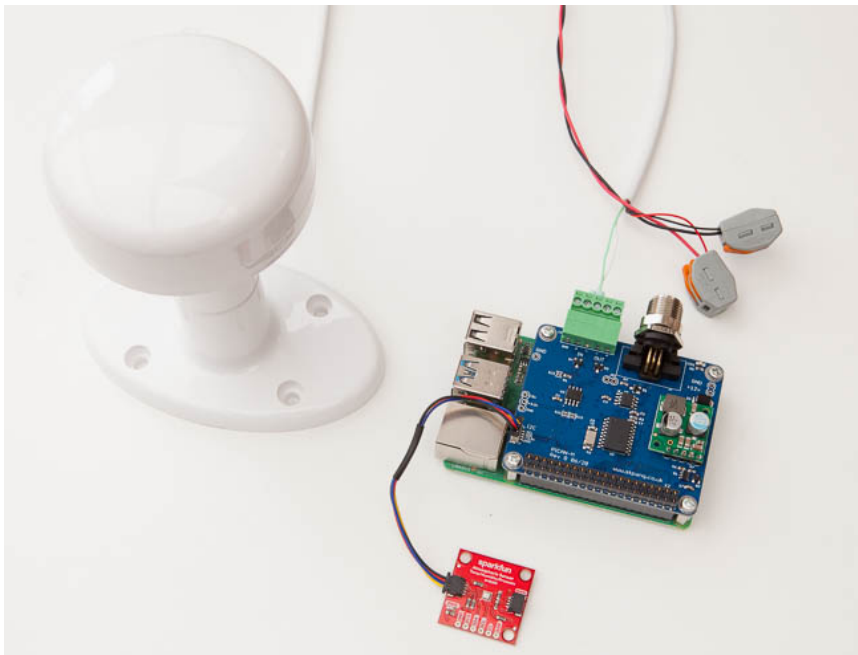
Press CTRL-x to save then exit.

![SK Pang electronics]

4. Starting OpenPlotter

Once the Raspberry Pi has booted up you should see a screen like this:
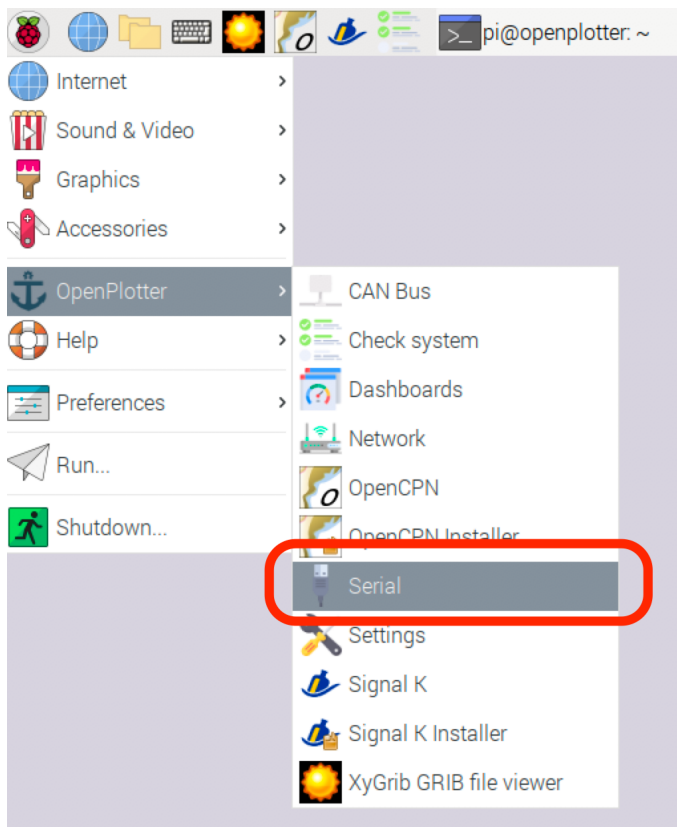

1.10. Configure GPS NMEA 0183 (RS422)

The NMEA 01832 (RS422) is connect to the Pi on UART port. This need to be enable first. Connect a NMEA 0183 GPS device to the 5 way green terminal plug like shown below:
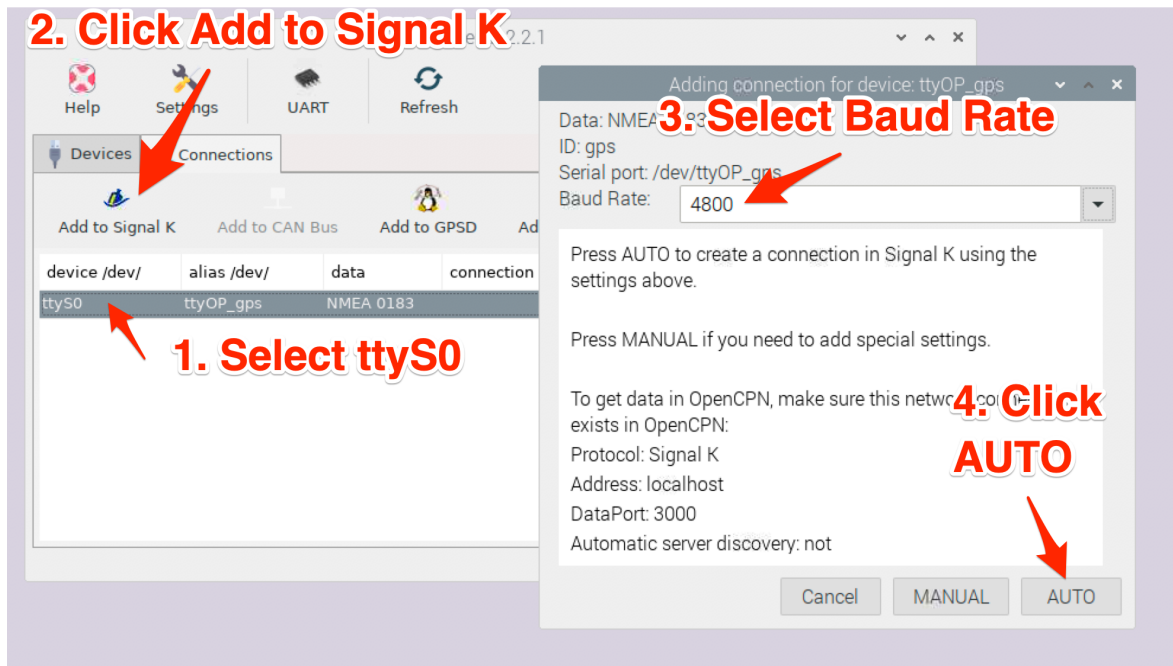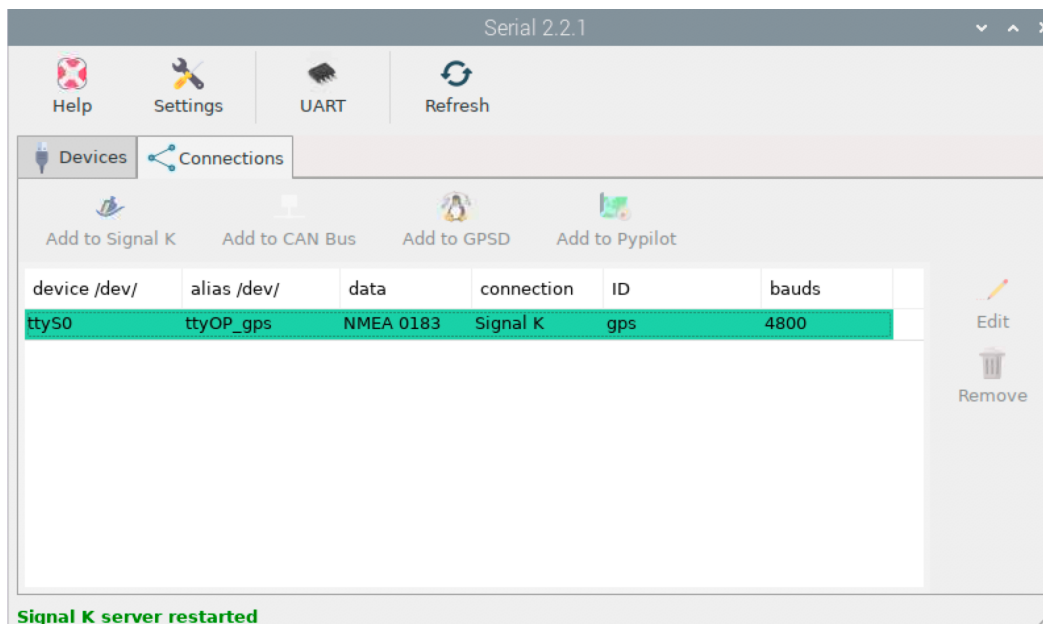
On the Pi click OpenPlotter, Serial:

1. Select ttyS0
2. Enter gps as alias
3. Select NMEA 0183 as data
4. Click Apply

Wait until the changes has been applied. The configure the connection:

1. Select Connections
2. Click Add to Signal K
3. Select Baud Rate
4. Click AUTO.

Wait until Signal K sever restart and it should look like this:

Now start OpenCPN:
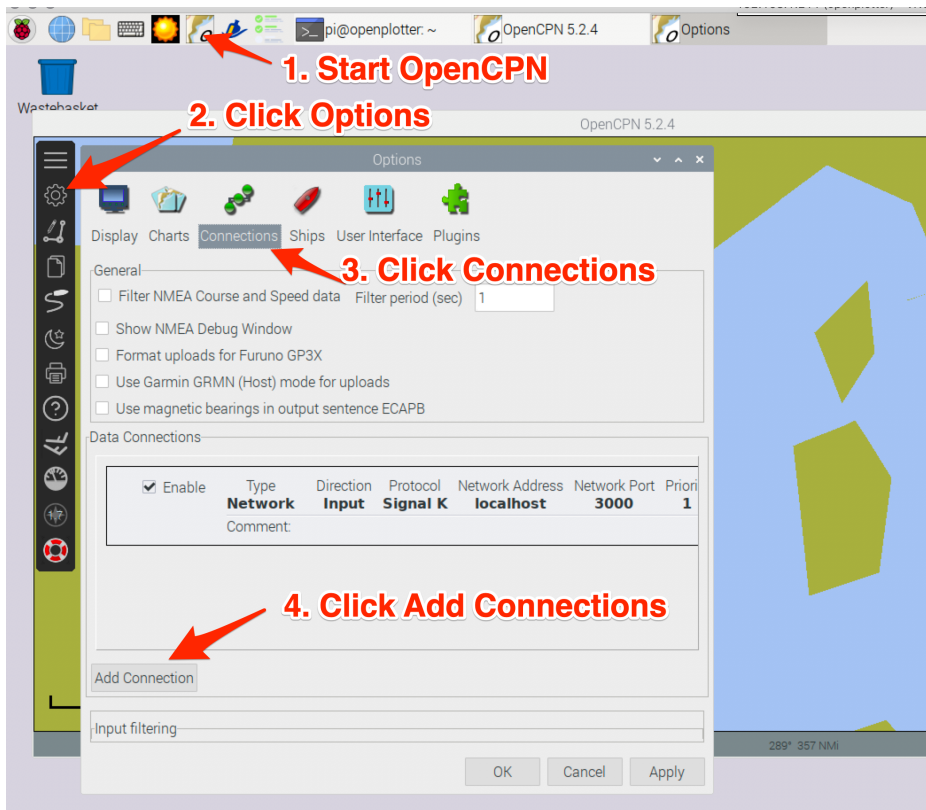
1.   Start OpenCPN
2.   Click Options
3.   Click Connections
4.   Click Add Connections

Next configure the new connections:
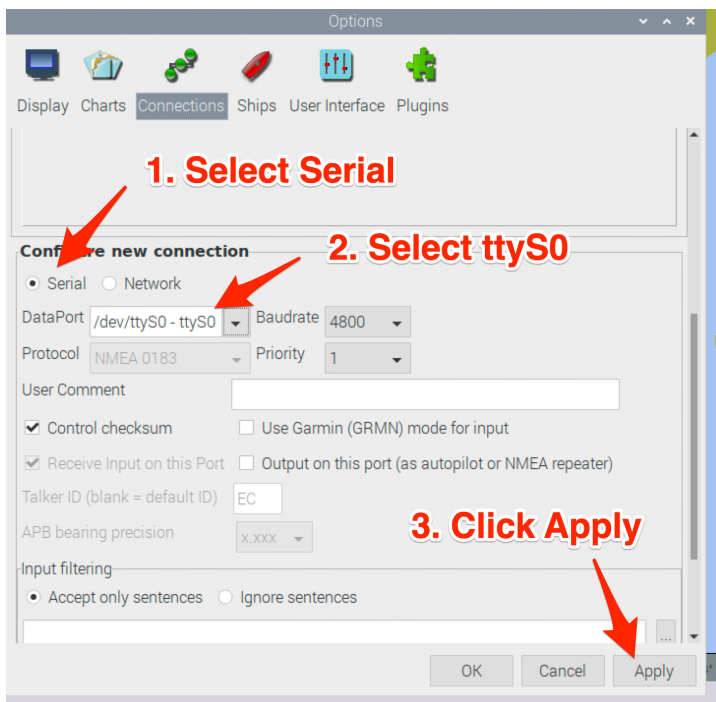
1.   Select Serial
2.   Select ttyS0
3.   Click Apply

This will take you back to the main Options dialogue box:
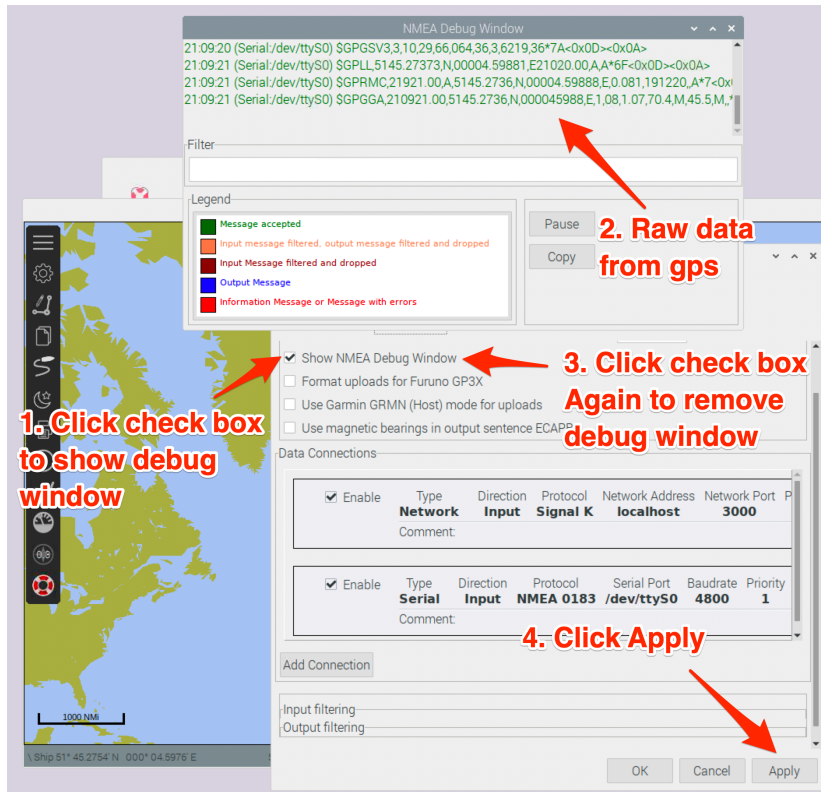
1. Click check box to bring up debug window
2. This will show the raw gps data received
3. Click check box again to remove debug window
4. Click Apply

Sometimes this does not work. If this is the case close OpenCPN go back to the Serial dialogue box:
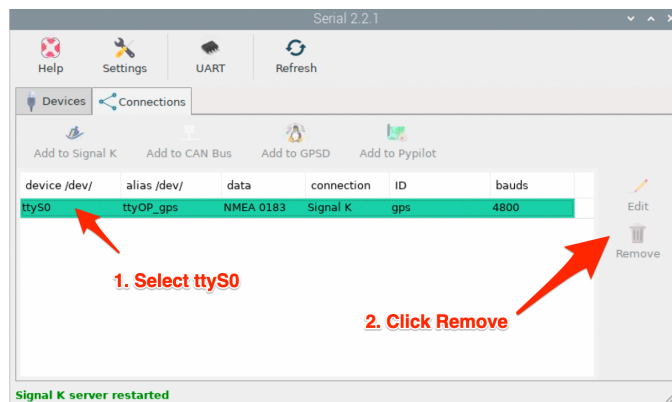
1. Select ttyS0
2. Click Remove

The close the Serial dialogue box and open it again. You should see:
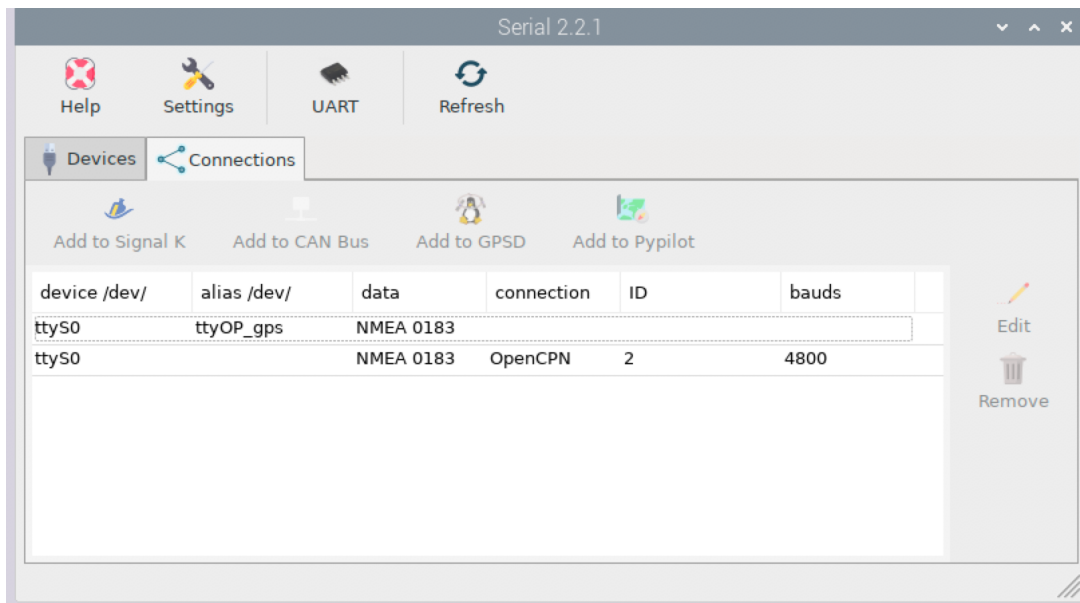
Start OpenCPN again, you should then see:

1. GPS lock
2. Click Dashboard

1.11. Configure I2C for BME280 Sensor

The Sparkfun BME280 sensor can provide barometric pressure, humidity, and temperature to the Raspberry Pi.

The default address on the board is 0x77 but this gives a problem to Signal K application. This can be solved by changing it to 0x76. To do this cut the track on the 0x77 address solder bridge then solder the bridge for address 0x76. This will change the board address to 0x76.

Now we need to install driver:
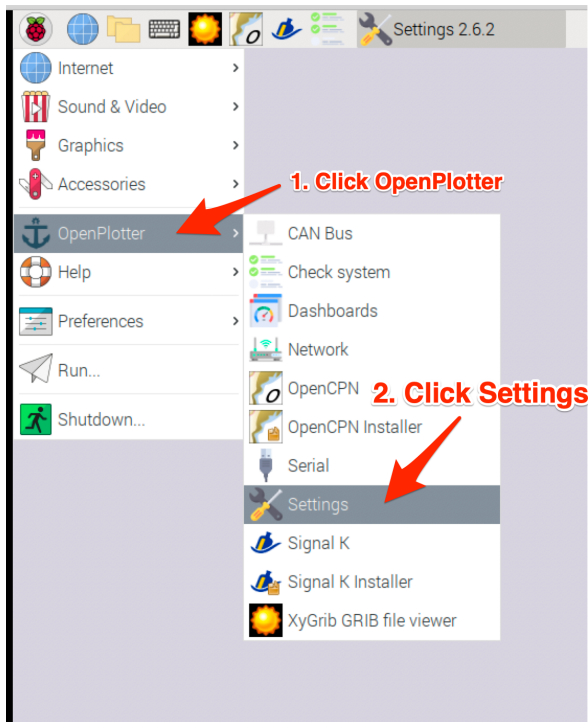
1. Click OpenPlotter
2. Click Settings

Now install the I2C Sensors:
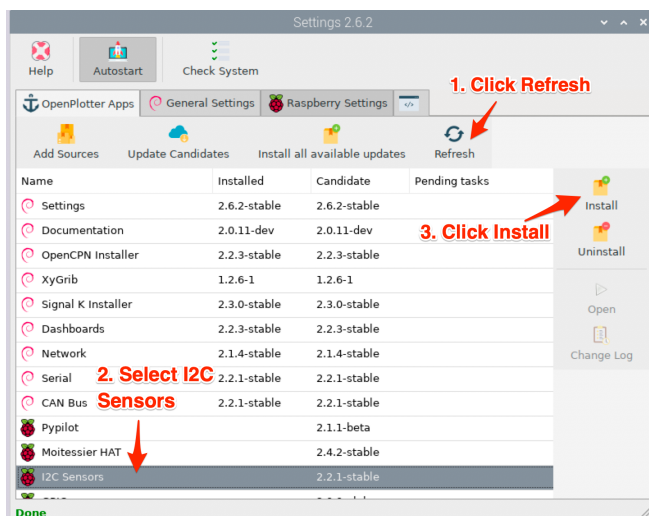
1. Click Refresh
2. Select I2C Sensors
3. Click Install, then yes
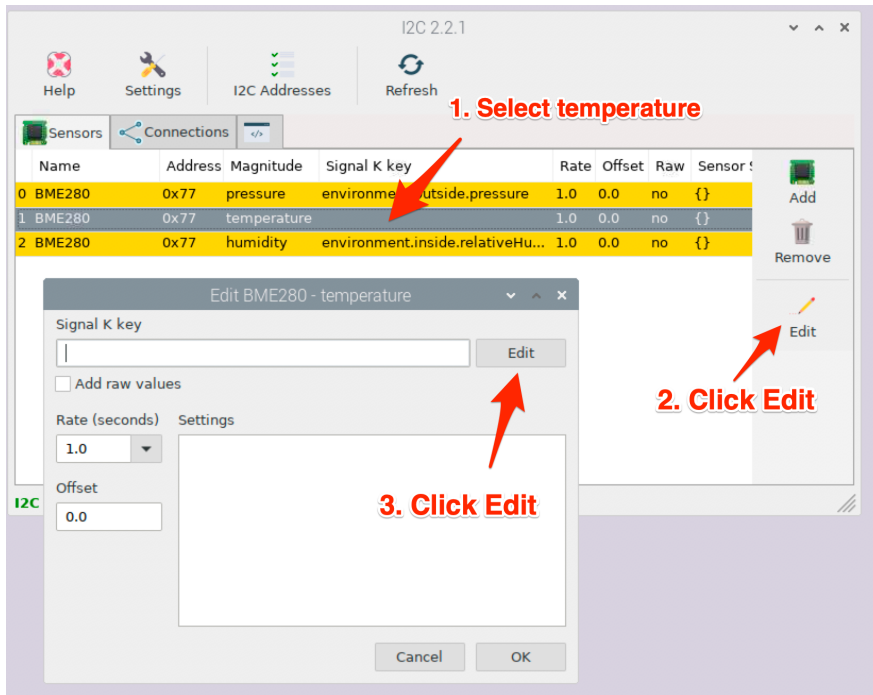
Now start the I2C dialogue box:
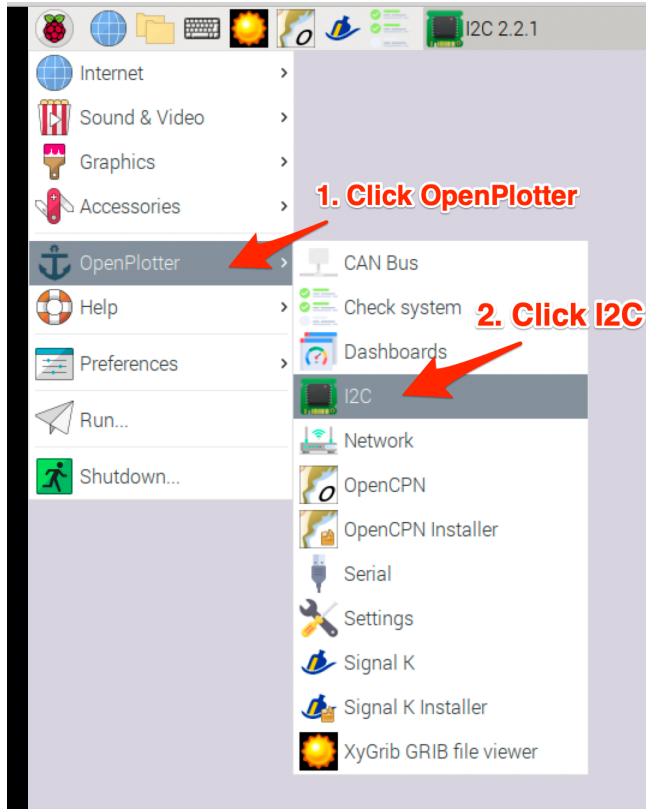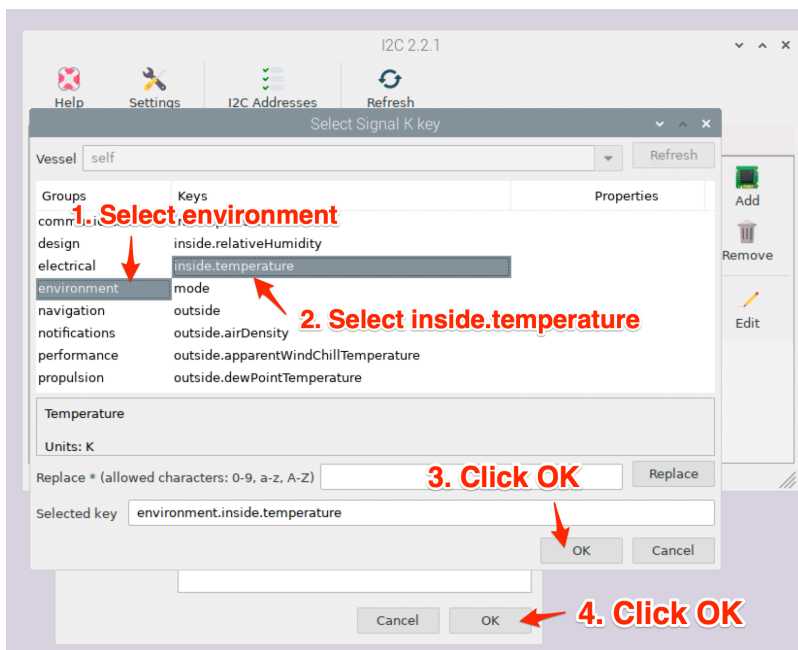

Now configure the BME280 sensor:
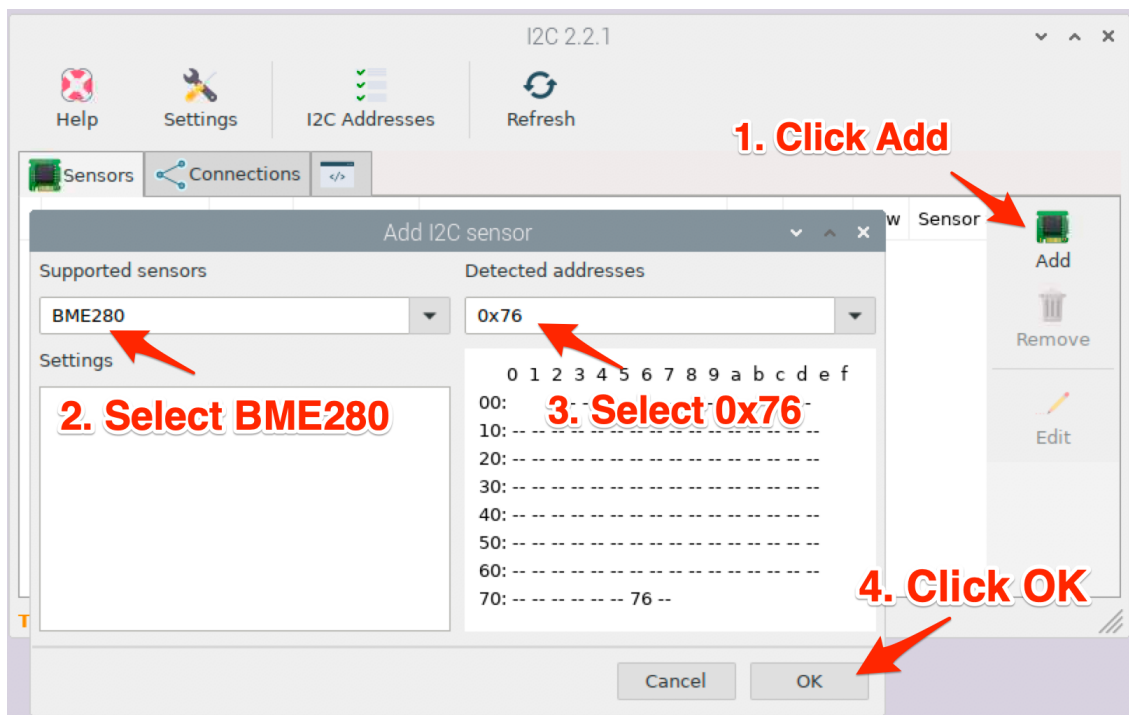
1. Click Add
2. Select BME280
3. Select 0x77
4. Click OK

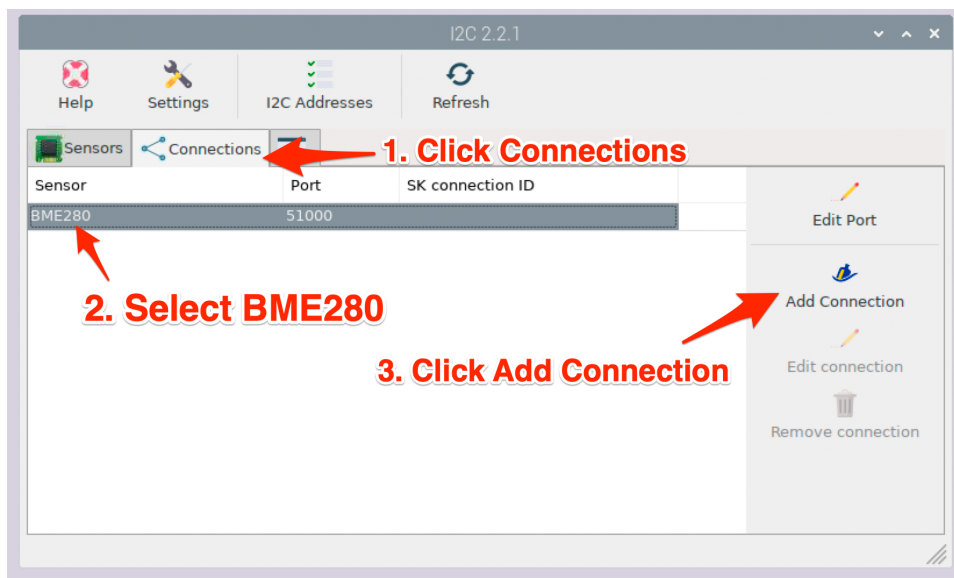Now we need to add the missing temperature reading:

1. Select temperature
2. Click Edit
3. Click Edit
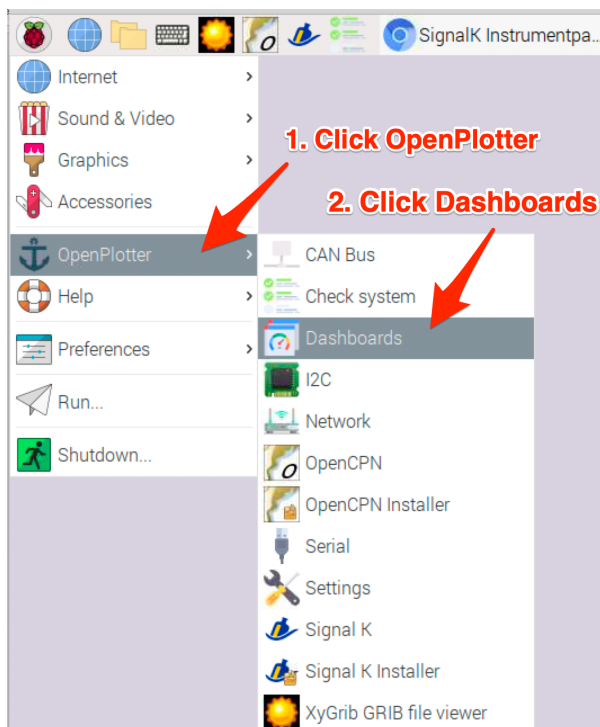
Now add the temperature key:

1. Select environment
2. Select inside.temperature
3. Click OK
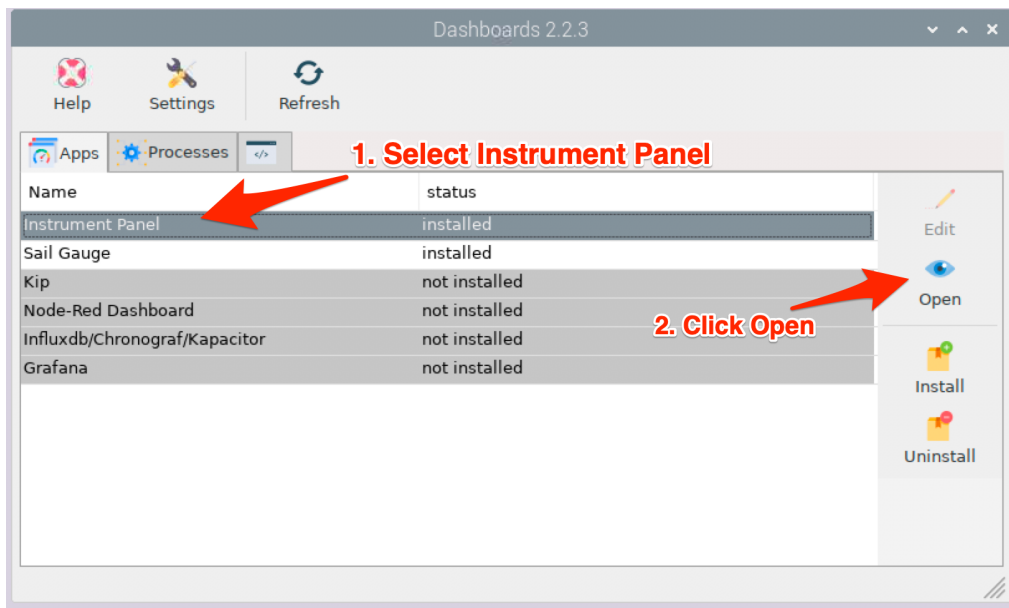4. Click OK

Now connect BME280 to Signal K:

1. Click Connections
2. Select BME280
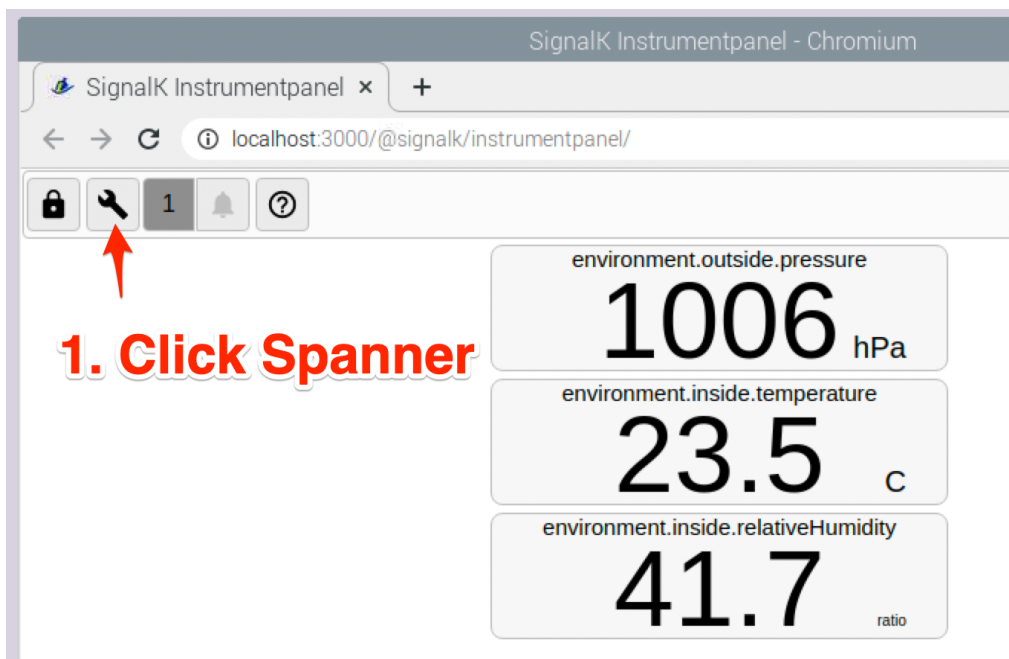3. Click Add Connection

Now open Signal K Dashboards:
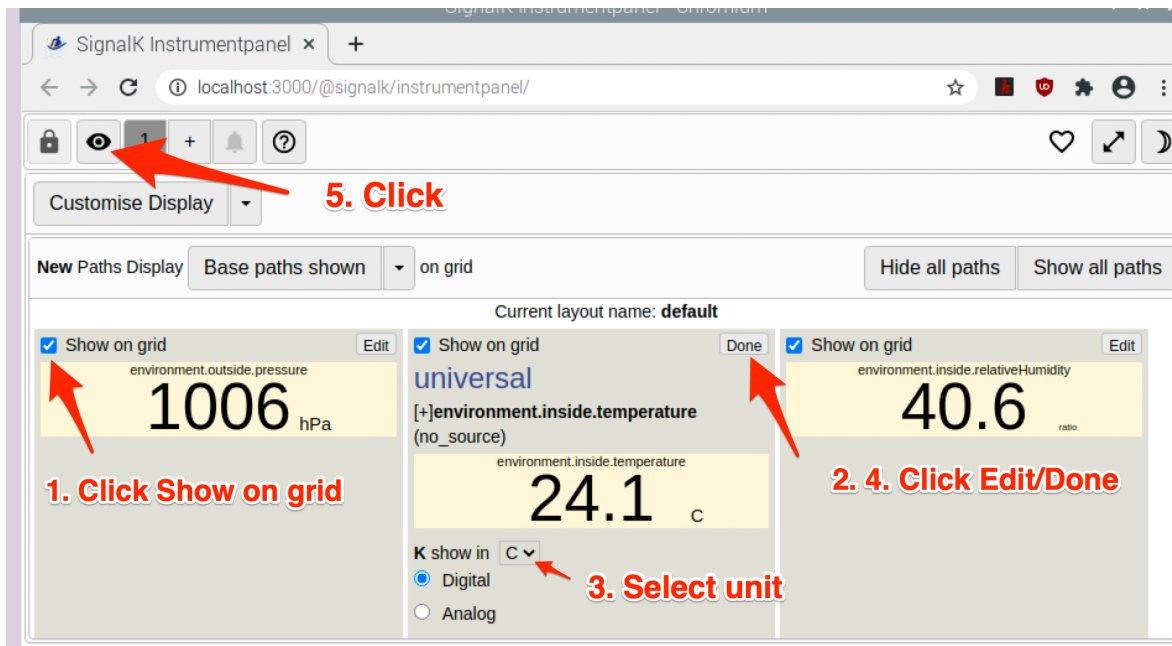
Now open Instrument Panel:

1. Select Instrument Panel
2. Click Open

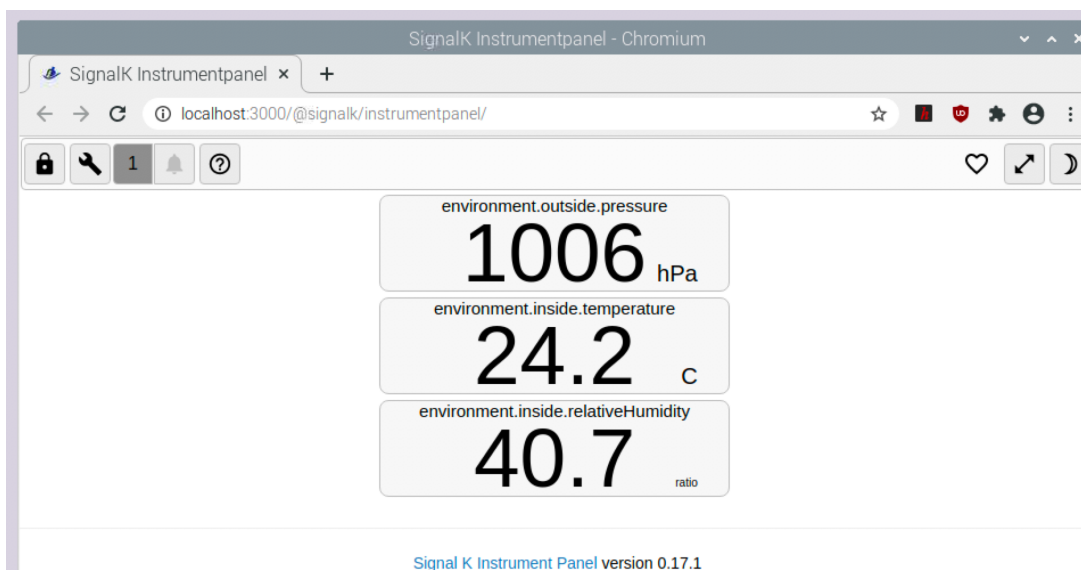In Signal K open Customise Display (spanner symbol) :

1. Click Spanner
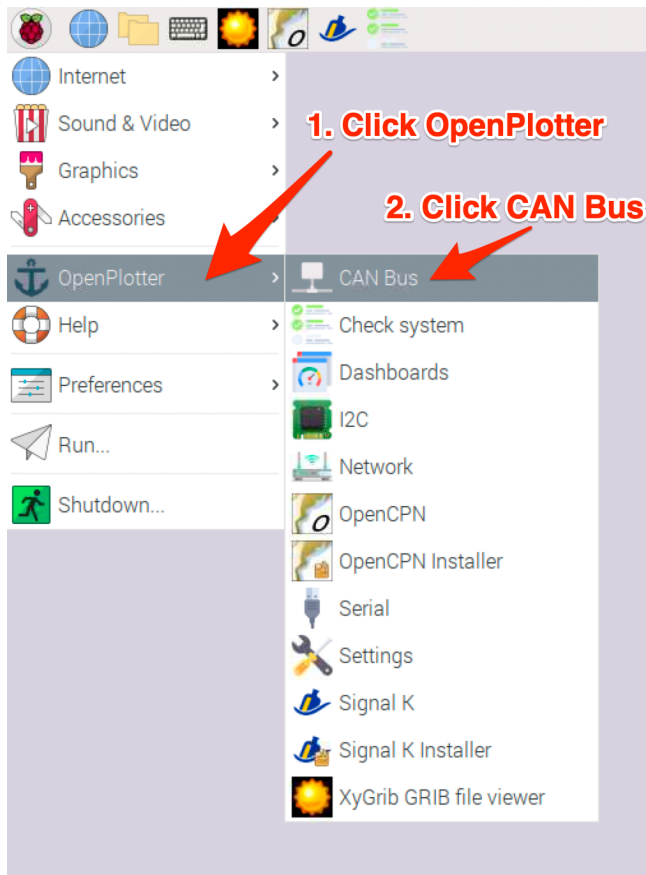
Now configure the signals:

1. Click Show on grid
2. Click Edit
3. Select unit to suit
4. Click Done. Repeat for all 3 readings
5. Click eye symbol

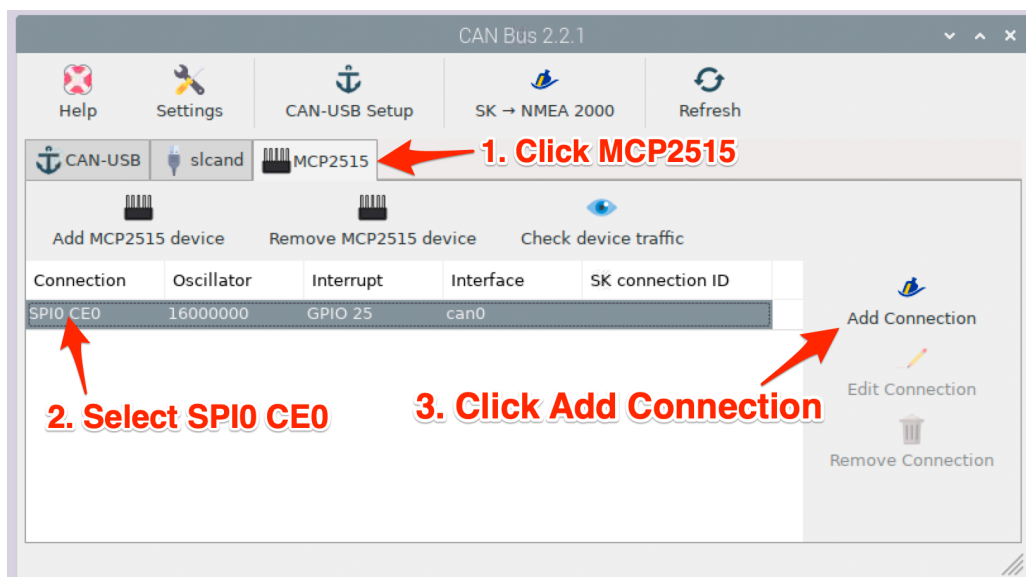Now you should see all 3 readings on the panel:

1.12. Configure NEMA 2000

We will a wind speed sensor and a temperature sensor from Yacht Devices Digital Thermometer NMEA 2000 YDTC-13N.
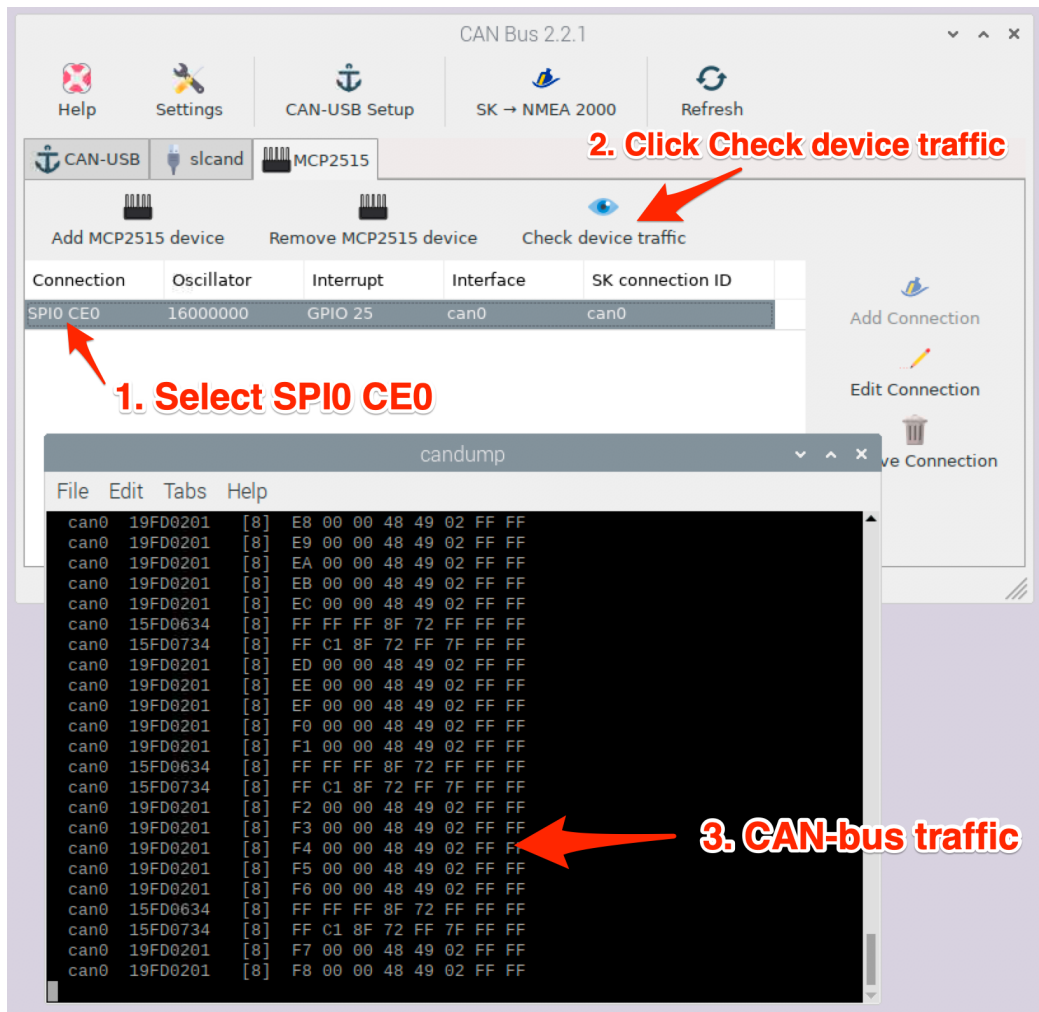

1. Click OpenPlotter
2. Click CAN Bus

Now configure CAN bus:

1. Click MCP2515
2. Select SPI0 CE0
3. Click Add Connection

Wait for Signal K to restart.

Check CAN-bus traffic:

1. Select SPI0 CE0
2. Click Check device traffic
3. Check there are CAN-bus traffice

If you see CAN-bus traffic that means the board has been configure correctly.

Close candump.

Open Signal K Instrument Panel and you should see the NMEA 2000 devices.
Configure the units as required.