RUSOKU

Application Note. Signal K

About this Document

This manual provides a detailed overview and a step-by-step tutorial which will help adding Rusoku TouCAN converters to SignalK server while using Raspberry Pi.

This document consists of two parts. The first part will help you to install TouCAN SocketCAN driver on your Raspberry Pi. Second part of this manual will help you with installation of SignalK server.

Part 1 | TouCAN SocketCAN Driver on Raspberry Pi

Installation instructions listed in this document are based on the following Linux Kernel version:

```
pi@raspberrypi:~ $ uname -r
5.4.51-v7l+
```

Step 1. Plug TouCAN converter to your Raspberry Pi board. Type command “**dmesg**”. The following information should appear on your computer:

```
pi@raspberrypi:~ $ dmesg
[ 6875.801939] usb 1-1.1: new full-speed USB device number 6 using xhci_hcd
[ 6875.941529] usb 1-1.1: New USB device found, idVendor=16d0, idProduct=0eac, bcdDevice= 1.00
[ 6875.941549] usb 1-1.1: New USB device strings: Mfr=1, Product=2, SerialNumber=3
[ 6875.941565] usb 1-1.1: Product: TouCAN converter s/n: 00000002
[ 6875.941580] usb 1-1.1: Manufacturer: RUSOKU technologies
[ 6875.941594] usb 1-1.1: SerialNumber: 00000002
```

Step 2. Type command “**lsusb**” and check if USB device with **ID 16d0:0eac** is available:

```
pi@raspberrypi:~ $ lsusb
Bus 002 Device 001: ID 1d6b:0003 Linux Foundation 3.0 root hub
Bus 001 Device 004: ID 046d:c062 Logitech, Inc. M-UAS144 [LS1 Laser Mouse]
Bus 001 Device 003: ID 045e:0800 Microsoft Corp.
Bus 001 Device 005: ID 16d0:0eac MCS
Bus 001 Device 002: ID 2109:3431 VIA Labs, Inc. Hub
Bus 001 Device 001: ID 1d6b:0002 Linux Foundation 2.0 root hub
```

Step 3. Install build tools:

```
pi@raspberrypi:~ $ sudo apt install build-essential
Reading package lists... Done
Building dependency tree
Reading state information... Done
build-essential is already the newest version (12.6).
The following package was automatically installed and is no longer required:
  rpi-eeprom-images
Use 'sudo apt autoremove' to remove it.
0 upgraded, 0 newly installed, 0 to remove and 0 not upgraded.
```

Step 4. Install Linux Kernel Headers:

```
pi@raspberrypi:~ $ sudo apt install raspberrypi-kernel-headers
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following package was automatically installed and is no longer required:
  rpi-eeprom-images
Use 'sudo apt autoremove' to remove it.
The following NEW packages will be installed:
  raspberrypi-kernel-headers
0 upgraded, 1 newly installed, 0 to remove and 0 not upgraded.
Need to get 26.2 MB of archives.
After this operation, 171 MB of additional disk space will be used.
Get:1 http://archive.raspberrypi.org/debian buster/main armhf raspberrypi-kernel-headers armhf
1.20200723-1 [26.2 MB]
Fetched 26.2 MB in 26s (993 kB/s)
Selecting previously unselected package raspberrypi-kernel-headers.
(Reading database ... 154109 files and directories currently installed.)
Preparing to unpack .../raspberrypi-kernel-headers_1.20200723-1_armhf.deb ...
Unpacking raspberrypi-kernel-headers (1.20200723-1) ...
Setting up raspberrypi-kernel-headers (1.20200723-1) ...
```

Step 5. Install Git client software:

```
pi@raspberrypi:~ $ sudo apt install git
Reading package lists... Done
Building dependency tree
Reading state information... Done
git is already the newest version (1:2.20.1-2+deb10u3).
The following package was automatically installed and is no longer required:
  rpi-eeprom-images
Use 'sudo apt autoremove' to remove it.
0 upgraded, 0 newly installed, 0 to remove and 0 not upgraded.
```

Step 6. Download TouCAN SocketCAN driver source code from github:

```
pi@raspberrypi:~ $ sudo git clone https://github.com/rusoku/TouCAN-SocketCAN
Cloning into 'TouCAN-SocketCAN'...
remote: Enumerating objects: 5, done.
remote: Total 5 (delta 0), reused 0 (delta 0), pack-reused 5
Unpacking objects: 100% (5/5), done.
```

Step 7. Change directory to TouCAN SocketCAN source code:

```
pi@raspberrypi:~ $ cd ~/TouCAN-SocketCAN/
pi@raspberrypi:~/TouCAN-SocketCAN $
```

Step 8. In order to compile TouCAN SocketCAN driver run “**sudo make**”:

```
pi@raspberrypi:~/TouCAN-SocketCAN $ sudo make
make -C /lib/modules/5.4.51-v7l+/build M=/home/pi/TouCAN-SocketCAN modules
make[1]: Entering directory '/usr/src/linux-headers-5.4.51-v7l+'
  CC [M]  /home/pi/TouCAN-SocketCAN/toucan.o
  Building modules, stage 2.
  MODPOST 1 modules
  CC [M]  /home/pi/TouCAN-SocketCAN/toucan.mod.o
  LD [M]  /home/pi/TouCAN-SocketCAN/toucan.ko
make[1]: Leaving directory '/usr/src/linux-headers-5.4.51-v7l+'
```

(Important) Make sure compiled driver with name “**toucan.ko**” appeared:

```
pi@raspberrypi:~/TouCAN-SocketCAN $ ls
Makefile  modules.order  Module.symvers  README.md  toucan.c  toucan.ko  toucan.mod  toucan.mod.c
toucan.o
```

Step 9. Now it is time to install TouCAN SocketCAN driver. Type command "**sudo make install**":

```
pi@raspberrypi:~/TouCAN-SocketCAN $ sudo make install
make -C /lib/modules/5.4.51-v7l+/build M=/home/pi/TouCAN-SocketCAN
modules_install
make[1]: Entering directory '/usr/src/linux-headers-5.4.51-v7l+'
  INSTALL /home/pi/TouCAN-SocketCAN/toucan.ko
  DEPMOD  5.4.51-v7l+
Warning: modules_install: missing 'System.map' file. Skipping depmod.
make[1]: Leaving directory '/usr/src/linux-headers-5.4.51-v7l+'
```

Create a list of module dependencies.Type command "**sudo depmod**":

```
pi@raspberrypi:~/TouCAN-SocketCAN $ sudo depmod
```

Step 10. Reboot your Raspberry Pi. Double check if TouCAN driver module is loaded after booting. Type command "**lsmod | grep toucan**":

```
pi@raspberrypi:~ $ lsmod | grep toucan
toucan             20480  0
can_dev             28672  1 toucan
```

Step 11. Using system administration utility “ifconfig” check if TouCAN device network interface appeared with name “**can0**”:

```
pi@raspberrypi:~/TouCAN-SocketCAN $ ifconfig -a
can0: flags=128<NOARP>  mtu 16
        unspec 00-00-00-00-00-00-00-00-00-00-00-00-00-00-00-00  txqueuelen 10 (UNSPEC)
        RX packets 0  bytes 0 (0.0 B)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 0  bytes 0 (0.0 B)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0
```

Congratulations! TouCAN SocketCAN driver has been installed!

Note Regarding NMEA 2000

In order to set-up a TouCAN device for NMEA 2000, you have to run the command:

```
pi@raspberrypi:~/TouCAN-SocketCAN $ sudo ip link set can0 up type can bitrate 250000
```

However, that only persists until the next reboot.

In order to enable the CAN bus interface at boot time you have to add the following string “**ip link set can0 up type can bitrate 250000**“ to file /etc/rc.local. Make sure you edit with root. For example:

```
pi@raspberrypi: sudo nano /etc/rc.local
```

```
!/bin/sh -e
#
# rc.local
#
# This script is executed at the end of each multiuser runlevel.
# Make sure that the script will "exit 0" on success or any other
# value on error.
#
# In order to enable or disable this script just change the execution
# bits.
#
# By default this script does nothing.

# Print the IP address
_IP=$(hostname -I) || true
if [ "$_IP" ]; then
  printf "My IP address is %s\n" "$_IP"
fi

ip link set can0 up type can bitrate 250000

exit 0
```

Part 2 | Installing the Signal K

Step 1. Install the Dependencies

Install nmp and nodejs:

```
pi@raspberrypi:~ $ sudo apt install nodejs npm
Reading package lists... Done
Building dependency tree
Reading state information... Done
nodejs is already the newest version (10.21.0~dfsg-1~deb10u1+rpi1).
nodejs set to manually installed.
npm is already the newest version (5.8.0+ds6-4+deb10u1).
npm set to manually installed.
The following package was automatically installed and is no longer required:
  rpi-eeprom-images
Use 'sudo apt autoremove' to remove it.
0 upgraded, 0 newly installed, 0 to remove and 0 not upgraded.
```

Make sure that you are using the latest version of npm:

```
pi@raspberrypi:~ $ sudo npm install -g npm@latest
npm WARN npm npm does not support Node.js v10.21.0
npm WARN npm You should probably upgrade to a newer version of node as we
npm WARN npm can't make any promises that npm will work with this version.
npm WARN npm Supported releases of Node.js are the latest release of 4, 6, 7, 8, 9.
npm WARN npm You can find the latest version at https://nodejs.org/
/usr/local/bin/npm -> /usr/local/lib/node_modules/npm/bin/npm-cli.js
/usr/local/bin/npx -> /usr/local/lib/node_modules/npm/bin/npx-cli.js
+ npm@6.14.7
added 434 packages from 880 contributors in 23.562s
```

Step 2. Install Signal K Node Server and Consumers

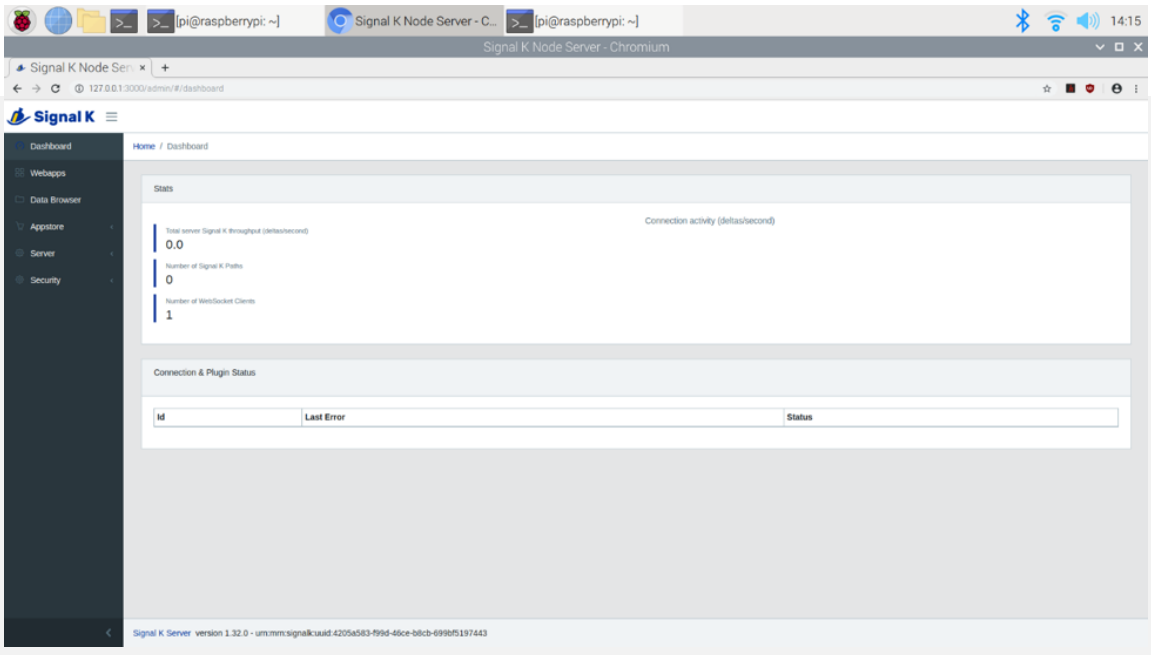
Install the Node Server using npm:

```
pi@raspberrypi:~ $ sudo npm install -g --unsafe-perm signalk-server
```

Run Signal K server. Type the following command:

```
pi@raspberrypi:~ $ signalk-server
```

Run your favorite web browser and open "**http://127.0.0.1:3000**"
This will open the dashboard and provide you some info about the SignalK server:



Adding TouCAN SocketCAN interface

Go to Dashboard -> Connections-> Add

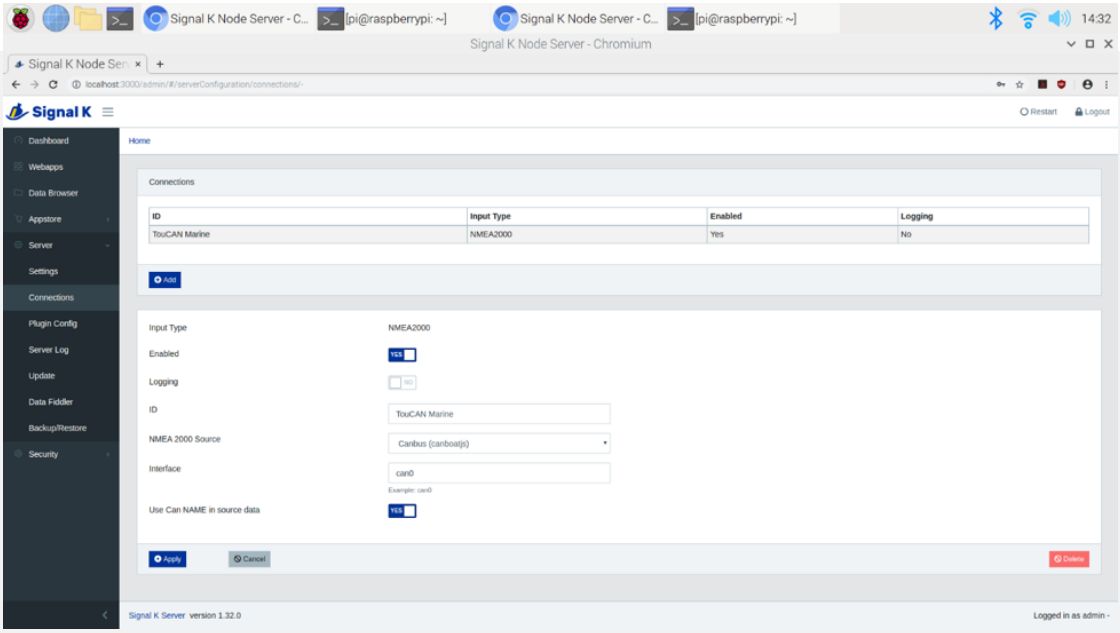
“Input Type” – NMEA 2000,

“ID” – TouCAN Marine,

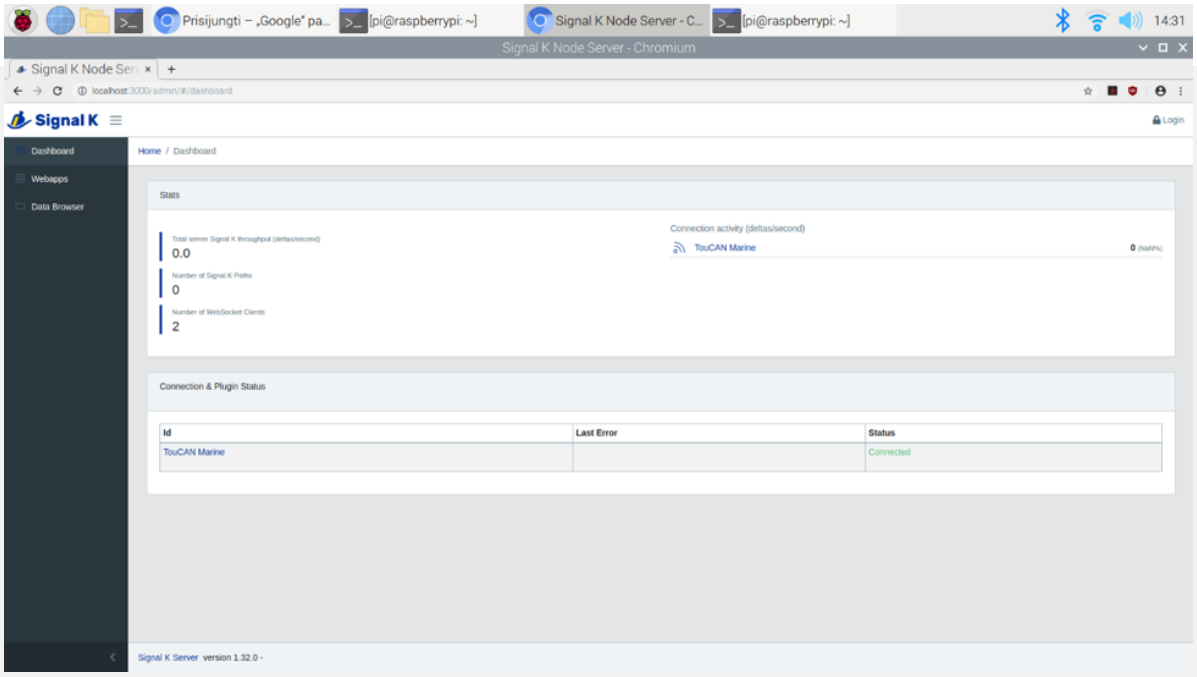
“NMEA 2000 Source” – canbus (canboatjs),

“Interface” – can0

Finally type “Apply” button.



The following screen should appear if you've followed the installation steps correctly.



Find more information about SignalK installation by following the links below:

1. https://github.com/signalk/signalk-server-node/blob/master/raspberry_pi_installation.md
2. https://github.com/rusoku/TouCAN-SocketCAN