Bareboat Necessities OS Documentation

mgrouch

Version 2024-03-03, Bareboat Necessities OS Documentation

https://bareboat-necessities.github.io

https://github.com/bareboat-necessities/lysmarine_gen

https://github.com/bareboat-necessities/lysmarine_gen/issues

https://bareboat-necessities.wixsite.com/my-bareboat

https://github.com/bareboat-necessities/bbn-m5stack-tough

https://github.com/bareboat-necessities/bbn-nmea200-m5atom

PDF version:

https://bareboat-necessities.github.io/my-bareboat/bareboat-os.pdf

Chapter 1. Getting Started

1.1. Why use BBN OS?

Why use BBN OS? Some people use just commercial marine electronics on board, some ones use smart tablets and marine Wi-Fi multiplexers. All these solutions lack one important component - an onboard computer capable of monitoring and controlling all aspects of the boat systems, and capable of storing data. Some people put an Intel-based computer for it. Intel-based solutions are more power consuming. With BBN OS you can build a central boat computer meeting your needs. All on low power consuming raspberry pi with flexibility of adding countless choices of sensors to talk to all boat systems, Internet, local Wi-Fi, cameras, NMEA network, and a system able to decode marine radio protocols. Initially this project started from experiences of using OpenPlotter to improve user experience. BBN OS is free and open source. It is based on commonly used community supported open source projects such as SignalK, PyPilot, OpenCPN, and others. BBN OS graphical user interface will let you build a cockpit front-end to all functionality of the OS from chartplotting, dashboards, weather information, to media player, etc.

1.1.1. Use Cases

- Cockpit Touchscreen Chartplotter
- Nav Station Computer
- Desktop Computer
- Pi Based Laptop
- Headless Server
- Media Player
- Autopilot
- SSB Radio Decoder
- Marine Data Server, Multiplexer
- Wi-Fi Access Point
- Cameras Monitoring
- Data Analytics Server
- Weather Station
- Boat Dashboards
- File Server
- Off-the-grid living

1.1.2. Smart Boat

Figure 1. Smart Boat Diagram

The diagram is color coded. Orange indicates what you (might) already have on your boat. Yellow are additions. Green is Pi (the central element).

1.2. Download Image

Download SD card image (~2.8Gb file):

https://github.com/bareboat-necessities/lysmarine_gen#download

Past major LTS releases:

https://github.com/bareboat-necessities/lysmarine_gen/releases/tag/v2024-03-03

https://github.com/bareboat-necessities/lysmarine_gen/releases/tag/v2023-08-17

https://github.com/bareboat-necessities/lysmarine_gen/releases/tag/v2022-08-03

1.2.1. 32-bit vs 64-bit images

> ℹ️ Default Bullseye image of BBN OS is fully arm64. Default Buster image of BBN OS is 64-bit arm kernel with 32-bit armhf user space programs.

- There are pros and cons to each

- 32-bit takes less memory but runs slower

- You still can run 64-bit kernel with 32-bit image, user space programs will be 32-bit. (Add arm_64bit=1 into /boot/config.txt)

- 32-bit is available on more hardware

- Most of the open (minor) issues are present in both 32 and 64-bit images

As of time of writing this the best performance/compatibility is the 64-bit image. Your preferred download image of BBN OS would be arm64 Bullseye.

- Install arm64 Bullseye image

- Proceed with all other configuration steps.

- After establishing Internet connectivity install official hot fixes:

```
cd /home/user/add-ons
./hot-fixes-install.sh
```

- Reboot (One reboot after installation required to pick up all settings)

1.3. Prepare SD Card

> ℹ️ Starting with Bookworm version BBN OS has two images to choose from: LITE and FULL.

Write the downloaded image to SD card. Minimum 32Gb SD card recommended. You can use Raspberry Pi Imager for that:

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

You do not need to explode .xz image file for it. You can load .xz file into RPI imager as is.

> ℹ️ Do not change default user on the image by RPI imager advanced settings. Do not use rpi-imager advanced tab at all. Burn the image to SD card as-is.

If your screen resolution is lower than 1024x600 you would need to manually set it in /boot/config.txt

file, by mounting /boot partition of your SD card (on Windows its done just by inserting it into SD card slot and editing in a plain text editor).

Quality of SD card is essential for performance of your system. BBN OS comes with disk benchmarking tool. Good SD card would show about 45 MB/sec (or higher on pi4) read i/o speed, 90 MB/sec or higher on pi5. SSD drives would show much better results: 178 MB/sec read and 147 MB/sec write i/o speed or higher.

> There were some cases of people just formatting SD card as fat32 file system and copying image file on it as-is, in form of .xz file or in some cases going extra length unpacking it into .img file and copying. Do not be like those guys. It can't possibly work for any OS image for Raspberry Pi. Read the docs.

> Also, there were cases of people trying to set up the system with VNC and not giving it full screen resolution. Those characters were trying to use the system even without realizing they do not see the whole screen. Obviously, they had really hard times, without succeeding. Again, read the docs.

1.4. First Boot from SD Card

Insert SD card into raspberry pi SD card slot, plug-in a mouse and a keyboard, and power on. Wait for boot process to run GUI (about 2 mins).

> Your touchscreen should be connected (HDMI and USB) and powered on during the first boot to be properly recognized and touchscreen calibrated.

1.5. Setting up Network

1.5.1. Wired Ethernet

If you have wired ethernet router you can just plug in your raspberry pi ethernet port into the router

1.5.2. Wi-Fi Client

Go into network settings menu and delete Wi-Fi wireless access point.

> You need to set up Wi-Fi country. In Bullseye version Wi-Fi may be disabled till you set Wi-Fi country.

Change /etc/wpa_supplicant/wpa_supplicant.conf to add line for your country (example):

```
country=US
```

Change /etc/default/crda to set your country (example):

```
REGDOMAIN=US
```

When raspberry pi Wi-Fi card discovers your Wi-Fi router network click to connect to Wi-Fi and enter the correct Wi-Fi password.

When the password storage manager pops up to enter the password for your password storage I just normally leave it blank, so I'm not prompted again.

1.5.3. Wi-Fi Access Point

Wi-Fi connections are managed by widely used Gnome NetworkManager. Look for 'nmcli' documentation (command line interface to NetworkManager). Or you can figure it out from Administration/Advanced Network Settings app menu. By default, OS image is set up to provide you with an access point.

> **ℹ** In Bullseye version Wi-Fi may be disabled till you set Wi-Fi country.

1.5.4. Using iPhone as Internet gateway

Enable Personal Hotspot on iphone. Plug it in into pi with BBN OS via USB and tell iPhone to trust the connected computer. You will have a tethered Internet connection from your pi via iPhone (using usbmuxd).

You do not have to be plugged in into USB if your phone, and a boat computer are on the same local Wi-Fi network.

1.5.5. Using Android phone as Internet gateway

Enable USB tethering in network settings on your Android phone. Plug it into your boat computer via USB.

The output of the command below should show a new route via usb0 interface:

```
netstat -nr
```

You can also do tethering via Wi-Fi instead of USB.

1.5.6. SpaceX Starlink

SpaceX Starlink Dishy is coming to boats soon and is easy to support. Plenty of boaters are already using RV version. Power usage might be relatively high for small boats.

1.5.7. Typical Setup on a Boat

- Raspberry Pi wired to OpenWrt LTE/4G/Wi-Fi router via ethernet port
- Raspberry Pi provides 5GHz Wi-Fi 802.11ac local access point for boat local Wi-Fi network
- OpenWrt LTE/4G/Wi-Fi router provides Wi-Fi connection to marinas
- OpenWrt LTE/4G/Wi-Fi router provides access to the Internet via LTE/4G cellular data network
- OpenWrt LTE/4G/Wi-Fi router provides 2.4GHz (Wi-Fi 802.11n) local access point for boat IoT devices
- OpenWrt LTE/4G/Wi-Fi router serves as firewall

If you use raspberry pi Wi-Fi it is better to disable Wi-Fi power management:

```
sudo systemctl unmask wifi_powersave@off.service
sudo systemctl enable wifi_powersave@off.service
sudo systemctl start wifi_powersave@off.service
```

> It is important to protect your installation from unauthorized access from the Internet. Make sure you put your raspberry pi behind a router which adds a firewall protection. It is also important to change default passwords.

1.6. Set Timezone / Locale

Open Terminal from GUI and on the terminal command line:

```
cd ~/add-ons
./timezone-setup.sh
```

For changing locale (ex: to en_US.UTF-8):

```
sudo su
perl -pi -e 's/# en_US.UTF-8 UTF-8/en_US.UTF-8 UTF-8/g' /etc/locale.gen
locale-gen en_US.UTF-8
update-locale en_US.UTF-8
```

1.7. Change Password

Default user and root passwords are changeme. You would want to modify it before your computer connects on-line to the Internet.

Open Terminal from GUI and on the terminal command line:

```
cd ~/add-ons
./change-password.sh
```

To change root password:

```
sudo su
passwd
exit
```

1.8. International Keyboard

Keyboard layout controlled by pre-installed ibus application. To add a language:

```
ibus-setup
```

1.9. SSD Boot

If you have an SSD drive, and you would like to boot from it (which would be a better way, and it would greatly improve the performance of the system) then you can follow the steps below:

The OS image comes with utility called 'rpi-clone' preinstalled. If you have a custom case for your raspberry pi (Ex. DeskPi Pro), then you would need to install vendor drivers for your case per vendor instructions.

Open Terminal from GUI and your command line for rpi-clone should look like (check usage https://github.com/billw2/rpi-clone as there might be nuances for your particular set up):

```
sudo rpi-clone sda
```

> ℹ️ Before doing it (if you had a previous installation), go into Disks tool from App Menu/Utilities/Disks and remove all partitions from your SSD drive.

If you have NVMe disk, then instead of sda it will be named as nvme0n1 (or something like that, check /dev/ directory).

Follow the prompts.

1.10. Set up GPS

Plugin your GPS USB mouse and OS should recognize it. Check:

```
ls -l /dev/ttyLYS*
```

If your gps shows as /dev/ttyLYS_gps_0, then edit /etc/default/gpsd file

```
sudo nano /etc/default/gpsd
```

and set

```
DEVICES="/dev/ttyLYS_gps_0"
```

In many other cases you do not need to do anything.

1.11. Set up AIS

Plugin your dAISy AIS receiver into USB and OS should recognize it. Check:

```
ls -l /dev/ttyLYS*
```

1.12. Update OpenCPN Plugins

- Start OpenCPN
- Go to Tools/Options/Plugins
- Update Plugin Catalog
- Browse plugins list and update plugins when an update available (one by one)

OpenCPN Plugins Documentation: https://rasbats.github.io/opencpn-plugins-manual/

1.13. Set up Charts

OS image comes with several chartplotters:

- OpenCPN
- AvNav
- Freeboard-SK
- TukTuk

with extensive set of plugins as well as weather GRIB file viewers

- XyGrib

1.13.1. OpenCPN

- Start OpenCPN. Go into Tools/Options/Charts/Chart Downloader tab.

- Click 'Add Catalog'. For USA: click USA NOAA & Inland Charts / ENC / By Region.

- Pick your region, click (or touch) 'OK'

- Click 'Update' (to update the catalog)

- Click 'Download Charts...' tab

- Right-click (or long touch) in the charts list

- Click 'Select all' from the pop-up menu

- Press 'Download selected charts' button, and wait for it to finish

- Press 'Apply' button

- Click 'Chart Files' tab

- Press 'Prepare all ENC Charts' button

- Press 'OK' button when done.

O-Charts can be registered using preinstalled OpenCPN plugins (on arm32 user space make sure to update plugins on-line from OpenCPN catalog) with O-Charts USB dongle or key.

O-Charts

O-Charts are commercial charts compatible with OpenCPN. See: https://www.o-charts.org/

O-Charts can be shared between OpenCPN and AvNav via AvNav o-charts plugin.

Use of USD Key Dongle is recommended over key file, so you do not lose your charts access after upgrades.

Charts from Chartlocker

https://chartlocker.brucebalan.com/

Making own MbTiles charts

Making own MbTiles charts with SASPlanet: http://svocelot.com/Cruise_Info/Equipment/mbTiles.htm

1.13.2. AvNav

When you are online NOAA raster MB tiles should work out of the box. You can install offline charts by following AvNav documentation.

> **ℹ** NOAA is phasing out RNC charts in favor of ENC.

You can download NOAA MB Tiles from offline use here: https://distribution.charts.noaa.gov/ncds/index.html

O-Charts can be registered using AvNav plugin with O-Charts USB dongle or key.

1.13.3. SignalK, FreeBoard-SK, TukTuk

Follow SignalK documentation to install offline charts for these.

1.14. Set up your Ship Parameters

Do not forget to set up your ship parameters in SignalK and Vessel applications. They can come handy.

Also in OpenCPN Polar plugin pick a polar file for your boat from ~/Polars, /usr/share/opencpn/plugins/weather_routing_pi/data/polars/ or build it from your past voyages data recordings. It will be needed for weather routing, etc.

Polars are important for Weather Routing and Dashboard Tactics plugins.

1.15. SignalK

SignalK manages its own updates. Login into SignalK Marine Data Server web UI application and perform updates via its app store.

For SignalK support visit: https://discord.gg/uuZrwz4dCS

1.16. PyPilot

Starting PyPilot server:

```
sudo systemctl enable pypilot@pypilot
sudo systemctl start pypilot@pypilot
```

Apart from the official PyPilot documentation you will find this WiKi https://github.com/pypilot/workbook/wiki very useful as well.

Commercial PyPilot hardware: https://pcnautic.gitbook.io/autopilot/inhoud-autopilot-set

If you have pypilot arduino motor controller connected via GPIO pins you need to add

```
dtoverlay=disable-bt

enable_uart=1
```

in your /boot/config.txt to disable conflicting bluetooth UART. Your motor controller will use that on-board UART.

Check:

```
ls -l /dev/serial0
```

/dev/serial0 must point to /dev/ttyAMA0

On bookworm version you also need to disable the hciuart service, which initialises the Bluetooth modem:

```
sudo systemctl disable hciuart
```

Serial console must be disabled too. Check /boot/cmdline.txt

You will need to remove /dev/ttyAMA0 from /home/pypilot/.pypilot/blacklist_serial_ports to let pypilot probe that port. (otherwise, pypilot port-probing breaks bluetooth even if bluetooth is enabled)

```
sudo nano /home/pypilot/.pypilot/blacklist_serial_ports
```

PyPilot HAT is optional. It provides a remote IR control and a small LCD. Due to GPIO changes this hat is not supported on pi5. If you have pypilot hardware hat attached, then you will need to enable the following settings in /boot/config.txt

```
dtoverlay=gpio-ir,gpio_pin=4

dtparam=i2c_arm=on
dtparam=spi=on
#dtparam=i2s=on
```

and enable pypilot hat service to start at boot time:

```
sudo systemctl enable pypilot_hat

sudo systemctl enable lircd
```

For hat display to work you need to specify correct driver (ex: nokia5110) in /home/pypilot/.pypilot/hat.conf. Edit this file when pypilot_hat service is not running.

Later hardware of pypilot hat creates /proc/device-tree/hat/custom_0 file from its firmware. You can do

```
cat /proc/device-tree/hat/custom_0
```

to see hat configuration and its GPIO pins required to be set up and so on.

For steering by the wind set wind.sensors_height to your wind sensor height as pypilot tries to compensate for apparent wind created by boat motion.

1.16.1. PyPilot Motor Controller Wiring to Pi GPIO

For brushed DC motors: https://pypilot.org/schematics/hbridge_datasheet.htm

Hydraulic motor controller: https://pypilot.org/schematics/highpower_datasheet.htm

https://pypilot.org/wiki/doku.php?id=tinypilot

1.16.2. PyPilot Motor Controller Loading Firmware into MCU

> Building your own firmware should be considered experimental. It's always better to load tested firmware for a tested controller model/make.

Follow instructions: https://github.com/pypilot/pypilot/tree/master/arduino/motor

You might need to use updated Makefile from: https://raw.githubusercontent.com/bareboat-necessities/my-bareboat/master/pypilot/arduino/motor/Makefile

1.16.3. Typical Components of PyPilot Motor Controller

- IBT_2 H-bridge module
- Arduino nano with ATmega328P chipset
- 10k NTC (2 wires) temperature sensor for the motor
- Rudder feedback potentiometer

For compass IMU connected to Pi (that's not really a part of motor controller)

- icm20948 IMU
- mpu9255 IMU

1.16.4. IMU Calibration using PyPilot

3-axis IMU accelerometer and 3-axis magnetometer need to be calibrated initially to account for range and bias of readings on each axis.

PyPilot comes with a calibration utility. During calibration you need to rotate IMU along all axes to allow calibration process to assess ranges of readings and skew bias on each of them. Calibration process completes when it's able to optimally fit all these readings on a sphere. (two spheres: one for accelerometer, and one for magnetometer)

1.17. UI for controlling autopilots

There are three options:

- NMEA 2000 autopilots can be controlled by adding autopilot page in SignalK KIP dashboard or (experimental) in SignalK autopilot plugin.

- Raymarine Autohelm SeaTalk autopilots can be controlled by SignalK plugin. This UI works as a web application and uses authentication cookie from SignalK nativefier application, so make sure you log in (with 'remember me' option) to SignalK UI application and set SignalK session timeout (in SignalK security settings) to a large value. Otherwise, you will get authentications errors trying to control your autopilot during your cruise.

- DIY PyPilot plugins can be controlled 3 ways: by OpenCPN plugin, standalone PyPilot UI application, or PyPilot Web application.

1.18. Weather

You can add weather budgie desktop applet. Unfortunately it is linked to a fixed location which is fine for a day-sailor but doesn't work for others.

Offshore sailors or even coastal cruisers should focus on using XyGrib and GRIB plugin for OpenCPN.

For real blue water sailors OpenCPN Climatology and OpenCPN weather routing plugins are essential.

Mediterranean GRIB files: https://openskiron.org/en/

Marine Weather Center: https://mwxc.com/

1.18.1. GRIB

Launch xyGrib. Select an area on the map. Click to download Grib file. Choose Atmospheric Model and Wave Model. Pick Wind, Wind Gust options in Surface Data and Wave Height in Wave Data. Click Download.

Now you can overlay this GRIB data on OpenCPN chart. You can play Grib file forecast forward. Enable GRIB plugin in OpenCPN. Click on GRIB plugin in OpenCPN plugin toolbar. Load GRIB file into

OpenCPN. See weather data overlaid over your chart.

OpenCPN GRIB plugin can also prepare SailDoc email requests for Grib files. You can store them and email via Internet (if you have connection) or use WinLink Pat (WinLink client: https://github.com/la5nta/pat/wiki) via SSB radio, or using JPSKMail.

1.18.2. NOAA weather in SignalK

SignalK comes with NOAA weather plugin. After you configure it, weather alerts will show up in KIP dashboard.

1.18.3. NOAA weather in browser

https://www.nhc.noaa.gov/marine/

1.18.4. WeatherFax

WeatherFax OpenCPN plugin is capable of downloading weather faxes from Internet as well as capturing them from SSB radio via audio input. (Pi needs a sound card with audio input because it is sold without audio input card). Just as GRIB files they can be overlaid over charts.

1.18.5. Weather routing

Weather routing OpenCPN plugin is able to plan your trip route. It uses your boat polar diagram, so make sure you have built it in advance using Polar plugin, or download it for your boat. Few places to check:

- SeaPilot https://www.seapilot.com/
- ORC-data https://github.com/jieter/orc-data

You also need to download climatology data using OpenCPN climatology plugin, and download a grib file. They are also needed and used in weather routing. Your request for a grib file should also ask for tide and currents data in order for them to be taken into the account by routing algorithms.

1.19. WinLink (SailMail analogue) for Raspberry Pi

To install:

```
cd ~/add-ons
./winlink-pat-install.sh
```

Documentation on setting it up and using it:

https://github.com/la5nta/pat/wiki https://getpat.io/

Video Series: Winlink on a Raspberry Pi
https://www.youtube.com/playlist?list=PL1QTYT4Qo9cY98NFmxrTvtGyWI9pgxtFq

ARDOP for pi: https://www.cantab.net/users/john.wiseman/Documents/ARDOPC.html

1.20. Music Players

The OS image comes with Mopidy, MPD server, MusicBox, Shairport-Sync (AirPlay) server. The default audio output set up to audio jack port.

> ℹ️ With Bullseye based versions it's possible to switch audio sources without modifying /boot/config.txt and without rebooting.

1.20.1. MusicBox

Start MusicBox web UI. Try pre-configured playlist, or you can search Tune-In or YouTube.

1.20.2. Iris

Iris is an alternative player to MusicBox. Iris is also included in the BBN OS image.

1.20.3. Playing from your iPhone (Spotify, etc)

Play music on iPhone. Select AirPlay on your iPhone and cast to 'lysmarine' airplay target (your phone must be on lysmarine-hotspot Wi-Fi).

1.20.4. Playing from mobile phones with MPD applications

Install MPD compatible media player on your mobile device, and from it, you can control playing your Mopidy library on your raspberry pi.

1.20.5. Playing Spotify

Start your Spotify app on your mobile device which is connected to boat Wi-Fi. Select 'Lysmarine' device as target to play on your pi via raspotify. You need to have a premium Spotify account.

1.21. Interfacing with ship systems

The first place to start configuring boat interfaces would be SignalK. SignalK comes with many plugins to talk to many boat devices with the support of various protocols.

1.21.1. NMEA 0183

If you use FTDI USB serial to USB sticks the OS should recognize them right away, and if they are wired correctly to NMEA devices (ex: wind/depth/speed/GPS) their reading should automatically show up in

instrument dashboards.

When wiring NMEA 0183 devices:

```
Transmit (Tx) (+) should connect to a receive (Rx) (+)
Tx (-) should connect to a Rx (-)
Rx (+) should connect to a Tx (+)
Rx (-) should connect to a Tx (-)
```

- In the event the sending device has a Tx (-), but there is not a corresponding Rx (-) on the receiving device, leave the sending device's Tx (-) disconnected. Failure to follow this guideline can damage the sending device.
- In the event that the receiving device has a Rx (-), but there is no corresponding Tx (-) on the sending device, bring the Rx (-) to ground.

What kind of messages travel via NMEA 0183

Sentence messages (max 82 characters in length) started with "$" or "!" and ending with "*" then two-characters checksum and a carriage return and line feed(<CR><LF>). Checksum can be optional.

- Regular NMEA 0183 (GNSS, Autopilot, Compass, Wind/Speed/Depth/Temperature/Pressure, Engine, Rudder, Chartplotter waypoints, routes, tracking)

  Example:

  ```
  $GPAAM,A,A,0.10,N,WPTNME*32
  $APHDM,256.740,M*31
  $APROT,0.004,A*33
  ```

- XDR Transducer Sentences NMEA 0183

  Example:

  ```
  $APXDR,A,30.900,D,PTCH*41
  $APXDR,A,9.239,D,ROLL*68
  ```

- Vendor extensions NMEA 0183

  Sentences start with "$P" followed by 3-letter manufacturer code.

  Example:

  ```
  $PASHR,085335.000,224.19,T,-01.26,+00.83,+00.00,0.101,0.113,0.267,1,0*06
  ```

- AIS

  Sentences start with "!"

  Example:

  ```
  !AIVDM,1,1,,A,14eG;o@034o8sd<L9i:a;WF>062D,0*7D
  ```

- DSC - Digital Selective Calling

  Example:

  ```
  $CDDSC,12,3380400790,12,06,00,1423108312,2019,,,S,E*6A
  $CDDSC,12,3381581370,12,06,00,1423108312,0236,3381581370,,S,*20
  $CDDSC,20,3381581370,00,21,26,1423108312,1902,,,B,E*7B
  $CDDSE,1,1,A,3380400790,00,45894494*1B
  ```

- STALK (SeaTalk1 over NMEA 0183)

  Example:

  ```
  $STALK,84,56,e,0,0,0,0,0,8*0F
  ```

- NavTex

  Example:

  ```
  $CRNRX,007,001,00,TD02,1,135600,27,06,2001,241,3,A,=========================*09
  $CRNRX,007,002,00,,,,,,,,,,=======^0D^0AISSUED ON SATURDAY 06 JANUARY 2001.*29
  $CRNRX,007,003,00,,,,,,,,,,^0D^0AINSHORE WATERS FORECAST TO 12 MILES^0D^0AOFF*0D
  $CRNRX,007,004,00,,,,,,,,,,SHORE FROM 1700 UTC TO 0500 UTC.^0D^0A^0D^0ANORT*70
  $CRNRX,007,005,00,,,,,,,,,,H FORELAND TO SELSEY BILL.^0D^0A12 HOURS FOREC*16
  $CRNRX,007,006,00,,,,,,,,,,AST:^0D^0A^0ASHOWERY WINDS, STRONGEST IN NORTH.^0D*15
  $CRNRX,007,007,00,,,,,,,,,,^0A^0A*79
  ```

- Radar NMEA Sentences

  Example:

```
$RARSD,,,,,,,,,0.000,000.2,0.125,N,N*50
$RARSD,,,,,,,,,0.145,244.5,0.125,N,N*55
$RARSD,0.883,099.3,0.200,047.6,,,,,0.145,245.2,0.1 25,N,N*54
$RARSD,0.150,233.5,0.177,086.4,,,,,0.150,233.5,0.1 25,N,N*59
$RARSD,0.251,243.9,0.177,087.5,0.376,131.3,0.167,0 11.5,0.376,131.3,0.25 ,N,N*5,N,N*5
$RATTM,01,0.438,018.6,T,0.00,180.0,T,0.438,0.0,N,T GT 01,Q,,180540.43,M*50
$RATTM,01,0.438,018.9,T,0.99,103.5,T,0.438,0.0,N,T GT 01,T,,180600.47,M*57
```

- Safety Messages (Inmarsat-C)

  Example:

```
/g:1-9-
1234,s:egcterm1,n:213,c:1333636200*hh/$CSSM3,123456,005213,798,0,3,14,00,2012,04,05,14
,30,3400,N,076,W,300*hh
/g:2-9-1234,n:214*hh/$CSSMB,008,001,0,123456,FROM:Maritime Rescue Coordination Centre
xxx^0D^0ATO:*hh
/g:3-9-1234,n:215*hh/$CSSMB,008,002,0,123456, ALL SHIPS IN xxxxxxx^0D^0ASAR SITREP NO:
02^0D^0AFIS*hh
/g:4-9-1234,n:216*hh/$CSSMB,008,003,0,123456,HING BOAT 'xxx' WITH THREE PERSONS ON
BOARD DEPARRTED*hh
/g:5-9-1234,n:217*hh/$CSSMB,008,004,0,123456, FROM xxx ISLAND ON^0D^0Axxx AT NOONTIME
AND SINCE TH*hh
/g:6-9-1234,n:218*hh/$CSSMB,008,005,0,123456,EN NO INFORMATION ABOUT HER. PARTICULARS
...^0D^0ASHI*hh
/g:7-9-1234,n:219*hh/$CSSMB,008,006,0,123456,PS SAILING IN VICINITY ARE KINDLY
REQUESTED TO KEEP A*hh
/g:8-9-1234,n:220*hh/$CSSMB,008,007,0,123456, SHARP LOOK OUT^0D^0AINFORMING
MRCC^0D^0AREGARDS^0D*hh
/g:9-9-1234,n:221*hh/$CSSMB,008,008,0,123456,^0A DUTY OFFICER*hh
```

- NMEA 0183 Amended Messages (MOB, SMV)

  Example:

```
$INMOB,ABCDE,V,123456.987,0,311219,132456.876,12.34,N,123.45,E,12.34,23.45,269104520,0
*53
$CSSMV,2,1,5,12,123456789,,1234.5678.N,12345.6789,W,2018,01,23,12,34,D*35
$CSSMV,2,2,5,12,123456789,MAXIMUM LENGTH FOR VESSEL NAME,,,,,,,,,D*25
```

1.21.2. NMEA 2000

Check SignalK plugin settings and SignalK documentation.

MCP2515 (Pican-M, etc)

For board with MCP2515 (Standalone Controller Area Network (CAN) controller) such as Pican-M, number of CM4 boards equipped with CAN Bus module, the setup procedure is usually:

Putting the line into /boot/config.txt:

```
dtoverlay=mcp2515-can0,oscillator=16000000,interrupt=25
```

That should create a CAN Bus network interface (socketcan) after reboot. It's same type of interface you would see with ifconfig command.

Then you would need to enable the service which starts can0 network interface on boot:

```
sudo systemctl enable socketcan-interface
sudo systemctl start socketcan-interface
```

After that you should be able to see some data from CAN Bus using:

```
candump can0
```

Use Ctrl-C to interrupt it.

Then you can create a data connection in SignalK:

- Data type: NMEA 2000
- NMEA 2000 Source: Canbus (canboatjs)
- Interface: can0

1.21.3. IMU

Check PyPilot settings and PyPilot documentation.

General steps are

- enable i2c (Interface options)

```
cd ~/add-ons
./os-settings.sh
```

- Enable pypilot service

```
sudo systemctl enable pypilot@pypilot
sudo systemctl start pypilot@pypilot
```

- At this point you should be able to see reading of pitch/roll, etc. and magnetic heading in pypilot control. Which you would need to calibrate.

- Shutdown SignalK. Start pypilot calibration. Press 'Boat Level' when the boat leveled. (Your IMU must be obviously mounted hard to the boat, can't be just hanging). For magnetic heading: IMU doesn't know how you oriented it inside (where bow is pointing), so you need to adjust it by filling magnetic heading adjustment field. Start SignalK.

> After calibration give pypilot 2-3 mins to run without restart to store settings. It is recommended to set heading offset to non-zero value (between -180 and 180). Setting it to zero will make pypilot to use GPS COG, POS, and magnetic variation to calculate heading offset dynamically (assuming no current or leeway) when boat is moving (i.e. assuming COG is true heading). Making a 360-degrees turn each time after leaving marinas or anchorages can help keeping your IMU compass stay calibrated for each voyage.

- Establish connection from PyPilot to SignalK (restart PyPilot just before doing it to get a fresh valid request)

- Go to SignalK web UI as admin and approve the access request from PyPilot for READ/WRITE access.

- IMU data should start flowing into SignalK

> What is considered as a pitch and what is a roll would depend on heading calibration offset. With 90 degrees heading offset pitch and roll swap and so on. So before hitting 'Boat is level' button in calibration screen make sure you put correct heading offset on same screen (your bow heading on the 3D screen needs to match heading from the other compass you are using for calibration). PyPilot allows offsets in range -180, 180. Other values are ignored.

Additional IMU and sensors settings adjustments are stored in /home/pypilot/RTIMULib.ini file. See more: https://github.com/seandepagnier/RTIMULib2/tree/master/Linux and https://github.com/seandepagnier/RTIMULib2

1.21.4. Barometer / Temperature / Humidity

Check SignalK plugin settings and SignalK documentation.

General steps are:

- enable i2c (Interface options)

```
cd ~/add-ons
./os-settings.sh
```

- To check if it's working:

```
lsmod | grep i2c-dev
i2cdetect 1
```

- Login into SignalK Marine Data Server

- Enable BMP or BME sensor plugin. Give it correct i2c address. Reduce poll timeout to 20 sec.

- Change 'salon.inside' to 'outside' (one word) for OpenCPN to display readings.

- Restart SignalK server.

- At this point you should be able to see barometric pressure and temperature (possibly humidity) in your data feed.

1.21.5. Other

Many other devices are supported (usually via SignalK)

1.22. Instrument Dashboards

1.22.1. OpenCPN

Enable OpenCPN 'Dashboard' plugin, add instruments. Dashboards are dockable to the right on bottom of OpenCPN canvas.

1.22.2. KIP / SignalK

- Load KIP demo. In setting of KIP dashboard change the URL to http://localhost:3000

- You should request KIP token to be registered in SignalK, then go into SignalK app and authorize it. After that edit instruments and layout in KIP dashboard settings.

https://48north.com/boats-and-gear/gear-reviews/supercharge-your-boats-information-gathering-with-signal-k/

1.23. Remote Access

1.23.1. VNC

BBN OS image comes with RealVNC server pre-installed and ready to be used on the local network. The

default host name to lookup in VNC client for connecting to your pi with BBN OS is 'lysmarine'.

OS image also provides VNC client app.

RealVNC with cloud connectivity

Create an account on realvnc.com website. Login from your pi's RealVNC server into realvnc.com cloud using the account you set up. Set up a password on your pi's VNC server for users to connect. While being logged in into realvnc.com website create a team and invite people you want to have remote desktop access to your pi. When they accept your invite let them know the password to connect to your local VNC server. They would need to download realVNC viewer (can be for PC, Linux, Mac, etc.) and follow the instructions from the invite.

1.23.2. ssh

OS image comes with ssh enabled. You can log in using ssh user: 'user'.

RaspController

RaspController is available in Google Play for your Android phone or tablet, and in App Store for Apple mobile devices. It allows remotely control your raspberry pi.

1.23.3. Android Devices

OS image comes with scrcpy pre-installed and pre-configured. You can view and control your Android devices. You need to enable USB debugging on your Android device or follow the instructions to enable controlling it via Wi-Fi. See: https://github.com/Genymobile/scrcpy In most cases you just need to enable USB debugging on your Android device and plug it in (thanks to autoadb).

Using this feature you can see your Android Navionics applications on BBN OS screen (as example).

1.23.4. Browsers

The URLs of the applications on your boat computer:

- http://lysmarine:8080 PyPilot
- http://lysmarine:3000 SignalK
- http://lysmarine:3000/@mxtommy/kip/ KIP
- http://lysmarine:3000/@signalk/signalk-node-red Node Red
- http://lysmarine:3000/admin/#/e/_signalk_vesselpositions Vessels Positions
- http://lysmarine:3000/tuktuk-chart-plotter/ TukTuk Chartplotter
- http://lysmarine:3000/@signalk/signalk-autopilot/ SK Autopilot
- http://lysmarine:3000/@signalk/freeboard-sk SK Freeboard

- http://lysmarine:8099 AvNav

- http://lysmarine:6680 Mopidy

- http://lysmarine:6680/musicbox_webclient/ Mopidy MusicBox

- http://lysmarine:6680/iris/ Mopidy Iris

- http://lysmarine:8765 MotionEye Cameras

Make sure your lysmarine computer is mapped to a static IP address on your network via MAC address mapping on your router.

1.23.5. Casting from Chromium to your TV

You can start Chromium and cast browser tab or whole desktop to your smart TV via ChromeCast protocol using Chromium menu 'Cast' option and selecting an appropriate source. Your TV and your boat computer need to be on the same network (LAN or Wi-Fi).

1.23.6. SmartPhone Applications

There are many applications for smartphones which will take live NMEA data stream from your boat computer host: lysmarine, port: 10110.

There are several ones which will even discover SignalK and use SignalK protocol. Examples: SignalK Monitor, SignalK.

Your phone and your boat computer need to be on the same Wi-Fi network.

1.24. Headless Operation

It's possible to disable booting into GUI by running:

```
sudo raspi-config
```

and selecting booting into the console under System Options / Boot - Auto Login.

1.25. Marine Radio, SSB, SDR, HAM

OS image comes with many HAM radio applications, decoders for many marine specific signals and protocols. Many SDR products should work. Decoding is also possible using external HAM receivers connected via sound input port (USB sound card required as raspberry pi doesn't have built-in sound input). Proper antennas required for correct reception.

HAM Radio menu contains a few applications useful especially for offshore sailors. If you do not have a cellular or a satellite connection, you still have SSB (and if you installed it SDR) radio. These applications will allow you (with some skill and set up):

- Receive weather GRIBs via WinLink

- Receive and decode WeatherFax, NavTex, NOAA and satellite weather images, Inmarsat Fleet messages

- Send and receive emails via WinLink (requires you to have a HAM amateur radio license)

- SDR can help you to decode AIS, and ADS-B (aviation analogue of AIS)

- Control your rig (SSB/SDR) with raspberry pi and monitor

- and more

1.25.1. Included Decoders

NavTex

JNX https://arachnoid.com/JNX/

WeatherFax

JWX https://arachnoid.com/JWX/

Satellite Weather

noaa-apt https://github.com/martinber/noaa-apt

Inmarsat STD-C

stdcdec https://github.com/cropinghigh/stdcdec

AIS

aisdecoder https://www.aishub.net/ais-decoder

rtl_ais https://github.com/dgiardini/rtl-ais

Starting with BBN Marine OS version 2022-11-10 AIS-Catcher https://github.com/jvde-github/AIS-catcher is included as well. AIS-Catcher can use many other SDRs, apart from RTL to decode AIS.

Other Decoders

Explore HAM Radio menu for more software.

1.25.2. WinLink

jpskmail

https://www.pskmail.org/

WinLink Pat

https://github.com/la5nta/pat

1.26. Iridium Phone

BBN OS will help you to use Iridium Phone as a modem for low bandwidth Internet access, or to send periodic short burst data with use of a specially written SignalK plugin.

More on using Iridium as modem: https://agile4life.blog/2018/07/16/sailing-with-linux-nmea-gpsd-iridium/

1.27. Cameras

1.27.1. IP Cameras

Should be easy to integrate using pre-installed VLC. See URL in /var/www/bbn-launcher/constants.js

IP cameras usually have some delay in video display.

1.27.2. RPI Cameras Interface

To enable cameras interface on raspberry pi4 run:

```
sudo raspi-config
```

and enable camera interface in there.

1.27.3. MotionEye

By default, motioneye service installed and enabled. To disable:

```
sudo systemctl disable motioneye
sudo systemctl stop motioneye
sudo systemctl status motioneye
```

Default user: admin

Password is empty.

1.28. Cruising within Cellular Phone Reception

Adding some OpenWrt LTE/4G router greatly improves your boat connectivity to the world near shore.

You should definitely do it to have internet access from your boat.

The OS image gives you internet applications for:

- Email

- Chat

- FB

- YouTube

- Browser

- On-Line Weather

- On-Line Charts

- Marina Booking

- Sailing Education

- SMS

- and much more

1.29. Offshore Features

For offshore sailors there are number of features preloaded into the OS image

- NavTex

- Inmarsat Fleet (receiving messages)

- Using Iridium as modem

- WeatherFax

- GRIB (could be over SSB)

- WinLink

- SDR / HAM Radio Apps

- AIS

- Weather Routing / Climatology

- Celestial Navigation

- Autopilot (PyPilot)

- Satellite Weather

- Radars (several supported)

- Location Reporting

They do require additional hardware, set up and dedication.

PyPilot based autopilots:

- https://pypilot.org/
- https://pcnautic.nl/nl/autopilot

These features would be particularly interesting for offshore sailors:

- AirMail Install AirMail on Raspberry Pi
- SASPlanet Install SAS.Planet on Raspberry Pi

1.30. Watching Movies

Watching on-line (or listening) prepaid copyrighted content (Netflix, Amazon PrimeVideo, Google, Spotify, etc.) in a web browser as Chromium requires closed-source DRM libraries. On arm32 version of the OS you can install it from add-ons folder ~/add-ons/ by running:

```
./widevine-lib-install.sh
```

> As of moment of writing this procedure doesn't work on arm64. It does work on arm32, and even on arm32 with 64-bit kernel.

1.31. Drones

Late versions of BBN OS include software for controlling drones. So if you have a drone (or a fleet of drones) you might be able to control them from the same BBN display.

The software included is APM Planner 2.0. See:

- https://github.com/ArduPilot/apm_planner
- https://ardupilot.org/planner2/

It supports MAVlink based autopilots including APM and PX4/Pixhawk:

- Copters
- Helicopters
- Planes
- VTOL/QuadPlanes
- Rovers
- Subs
- Vehicles

List of supported drones:

https://ardupilot.org/planner2/docs/common-rtf.html

1.32. Shutting Down / Rebooting

On the desktop click on the 'Commands' icon. You will see a menu from where you can perform restart/shutdown, and more.

1.32.1. Safe Power-Down

Raspberry pi doesn't have a safe power-off feature. I.e. it doesn't perform OS shutdown before powering off with a button. There are numerous third-party solutions with raspberry pi hats or custom cases. Make sure you do not forget to install required software for them per vendor documentation.

1.33. Customizing Desktop

Desktop can be customized by editing the JavaScript files in /var/www/bbn-launcher (i.e. constants.js).

PyPilot web client looks better in dark skin. Switch to the dark theme if it wasn't done for you automatically.

1.33.1. Customizing Applications Menu

Applications menu can be customized by editing gnome-applications.menu in ~/.config/menus.

1.34. Customizing On First Boot

You can add additional customizations which will be performed on system first boot by mounting OS image and editing /boot/first-boot.sh script. That script as its name suggests executes only once on the first boot.

1.35. Known Issues and Workarounds

See: https://github.com/bareboat-necessities/lysmarine_gen/issues

If GPS fix is lost in OpenCPN the first thing to try is to restart SignalK. You can do it from the touchscreen via desktop 'Commands' icon.

Do not create data loops with your data flows between OpenCPN, AvNav, SignalK, GPSd, Kplex, PyPilot.

1.35.1. Touchscreen

BBN Open Source Marine Linux OS for Raspberry Pi by Bareboat Necessities is customized for best touchscreen support (due to need to support cockpit touchscreen chartplotter).

On screen keyboard, Two finger zoom, Long touch, Double finger tap, Three finger scroll are supported. No finger rotate (so far), Copy/paste is via on-screen keyboard.

There are sometimes timing issues with USB ports initialization during first boot. If you see touchscreen not being calibrated then try (after establishing Internet connectivity and with your touchscreen properly connected):

```
cd /home/user/add-ons
./hot-fixes-install.sh
```

Use the same procedure if you change your touchscreen to another one.

Check your /usr/share/X11/xorg.conf.d/90-touchinput.conf file:

```
cat /usr/share/X11/xorg.conf.d/90-touchinput.conf
```

If MatchProduct string in it is empty you will need to put it in manually. First find the product:

```
cat /var/log/Xorg.0.log | grep TOUCH
```

Then copy and paste product name into MatchProduct section

```
sudo nano /usr/share/X11/xorg.conf.d/90-touchinput.conf
```

and save the file.

Calibration issues can be resolved by running:

```
xinput_calibrator
```

and following on-screen instructions.

> The touchscreen issues mentioned below fixed since 2021-09-10 release.

- Some applications (namely OpenCPN and gtk2 based as well as some Qt) sometimes stop responding to touch events. There is a workaround. With your finger you can toggle a maximized mode via window frame icon, then you MOVE the window frame by dragging window header few pixels, and switch back to maximized mode if needed. This should restore touch events in that app.
- Some gtk3 applications menus (ex: terminal) have issues handling touch events. You can select a menu item with touch but to perform a click on it, you would actually need to perform simulated right click by holding finger a bit longer and letting it go.

1.35.2. OpenCPN

OpenCPN feature to auto hide toolbar on timer seems causing OpenCPN to hang under some conditions. Avoid it and keep it turned off.

1.35.3. Wi-Fi DNS

If you use your devices on LAN or Wi-Fi networks they might get assigned different IP addresses from DHCP servers. However, DNS servers tend to cache (remember) name to IP address mappings for quite some time. Thus, you can run into DNS name resolution issues if you do not configure DHCP servers to assign consistent IP addresses (by MAC address for example) to all your devices.

Another situation which can cause name resolution issues in DNS is when one device configured to be able to choose multiple Wi-Fi networks to connect to.

1.35.4. Under-Voltage Detected

If you see this message it means you have a poor power supply for your Pi and some of your connected devices will not work. In that case you need higher amps power supply and raspberry Pi with many devices connected needs 5.2 volts power input.

1.35.5. Issues during first boot

- Did you wait long enough for desktop to boot?
- Are you having too many devices plugged in? You just need a keyboard, mouse, touchscreen (video and USB), and possibly ethernet.
- Do you see under-voltage message when you run 'dmesg' from command line?
- Was SD card burned and verified successfully?
- Are you using pi4, CM4 (with Wi-Fi), or pi400?

1.36. Add-ons

Check /home/user/add-ons directory. It contains number of scripts for installing many additional programs which for one or another reason couldn't be a part of the distribution image.

Few notable add-ons:

1.36.1. Text-to-speech

```
cd /home/user/add-ons
./text-to-speech-install.sh
```

1.36.2. Navionics

```
cd /home/user/add-ons
./navionics-demo-install.sh
```

1.36.3. QtVlm

```
cd /home/user/add-ons
./qtvlm-install.sh
```

1.37. Default Ports

See:

```
sudo netstat -tulpen
```

```
Proto   Port      Transport          Program/Service
tcp     22        ssh                sshd Secure Remote Shell
tcp     25        smtp               exim4 e-mail
tcp     139       netbios            smbd
tcp     445       smb                smbd
tcp     631       CUPS               cupsd Printing
tcp     2947      gpsd               gpsd
tcp     3000      http/WS            SignalK
tcp     4713      pulseaudio         pulseaudio audio
tcp     4997      http               bbn-desktop
tcp     5000      airplay            shairport-sync music
tcp     5037      adb                android adb
tcp     5900      vnc                x11vnc
tcp     6600      mpd                Mopidy MPD Music
tcp     6680      http               Mopidy Music
tcp     8080      http               PyPilot_web
tcp     8082      http               AvNav Ocharts
tcp     8085      http               AvNav Updater
tcp     8099      http               AvNav
tcp     8375      http/json          SignalK Deltas
tcp     8765      http               motioneye
tcp     10110     NMEA 0183          SignalK
tcp     20220     NMEA 0183          PyPilot
tcp     23322     json               PyPilot
tcp     34567     NMEA 0183          AvNav NMEA out
tcp     dynamic   spotify            librespot
udp     68        DHCP               dhclient
udp     137       nmb                nmbd
udp     138       nmb                nmbd
udp     323       ntp                chronyd time service
udp     631       IPP                CUPS Internet Printing Protocol
udp     5353      service discovery  avahi-daemon, chromium, others (bonjour, zeroconf,
mDNS)
udp     10116     NMEA 0183          AvNav, KPlex
```

1.38. Firewall

Starting with BBN OS version 2021-10-02 you will have a firewall running on your system, and it's enabled by default. The default firewall rules are:

```
sudo ufw status

Status: active

To                      Action      From
--                      ------      ----
Anywhere                ALLOW       192.168.0.0/16
Anywhere                ALLOW       169.254.0.0/16
Anywhere                ALLOW       10.0.0.0/8
Anywhere                ALLOW       100.64.0.0/10
Anywhere                ALLOW       172.16.0.0/12
Anywhere                ALLOW       224.0.0.0/4/udp
Anywhere on can0        ALLOW       Anywhere
Anywhere (v6)           ALLOW       fd00::/8
Anywhere (v6)           ALLOW       fe80::/10
Anywhere (v6)           ALLOW       ff00::/8/udp
Anywhere (v6) on can0   ALLOW       Anywhere (v6)
```

1.39. Bluetooth

As of 2021-10-04 Bluetooth seems good only for low speed devices. You likely to experience skipping playing music via on-board pi bluetooth.

> Versions 2021-11-29 and after have this issue fixed. Make sure you close bluetooth setting window before playing from a bluetooth device.

1.40. Amazon Alexa, etc

Set it up to be on your boat Wi-Fi. You can also pair it with your pi via Bluetooth and play music via Amazon EchoDot, etc. Mopidy seems needs a restart to switch to Bluetooth playing.

Say "Alexa, pair a device". Proceed to device pairing. Make sure you close bluetooth settings window in BBN OS before playing music to avoid skipping.

1.41. QtVlm Chartplotter

QtVlm can be installed via add-ons:

```
cd /home/user/add-ons
./qtvlm-install.sh
```

1.42. Maritime Library

You can install your Maritime Publications On-Board Library via add-ons:

```
cd /home/user/add-ons
./maritime-lib-install.sh
```

1.43. Offline Wikipedia

Using preinstalled Kiwix application you can store offline Wikipedia and more on your boat computer.

1.44. Media Library

You can install JellyFin via add-ons:

```
cd /home/user/add-ons
./jellyfin-install.sh
```

1.45. HomeAssistant

You can install HomeAssistant (for many BBN users their boat is their home) via add-ons. You need to be connected to Internet, and give 15 mins or so after the below commands finish:

```
cd /home/user/add-ons
./homeassistant-install.sh
```

It will also install EspHome. Url to access: http://localhost:6052

Examples of use cases:

https://boathackers.com/boat-energy-management-using-victron-and-home-assistant/

https://boathackers.com/dynamically-set-location-in-home-assistant-for-boats/

ESP32 devices can be managed using https://esphome.io

1.45.1. HomeAssistant with SignalK

- Enable "Signal K - MQTT Home Assistant Bridge" plugin in SignalK (with run local server option, port 2883)

- In Home assistant add MQTT integration Broker: lysmarine Port: 2883 Subscribe Topic:

```
+/+/#
```

- Edit /home/homeassistant/.homeassistant/configuration.yaml

```
mqtt: # SignalK MQTT HA Bridge
  # https://developers.home-assistant.io/docs/core/entity/sensor/
  sensor:
    - name: "Wind App Speed"
      unique_id: "speedApparent"
      #device_class: "wind_speed"
      state_topic: "vessels/self/environment/wind/speedApparent"
      unit_of_measurement: "kn"
      value_template: "{{ (value_json * 1.9438452) | round(1) }}"

    - name: "Wind App Angle"
      unique_id: "angleApparent"
      state_topic: "vessels/self/environment/wind/angleApparent"
      unit_of_measurement: "deg"
      value_template: "{{ (value_json * 180 / 3.14159265359) | round(1) }}"
```

- Edit Dashboard, Click "Add Card" and you can select to draw gauges, meters, chart, etc views for your sensors

1.46. SolarThing (Solar Power Monitoring)

Yor system includes installed SolarThing. See more at https://github.com/wildmountainfarms/solarthing

1.47. Maiana AIS

Maiana AIS supported via add-ons

```
cd /home/user/add-ons
./maiana-ais-install.sh
```

1.48. Victron

To monitor your Victron devices you can access

http://localhost:8000/app?host=venus.local&port=9001

Change the host and port in that URL to your Victron Venus OS device name and port.

1.49. Marine Species Identification

When Internet is available you can use Menu link in App Menus→Utilities→Marine Life

1.50. Knots

There is an application to display basic knots. App Menus→Utilities→Knots

1.51. ColReg Regulations

English version of ColReg is available under: App Menus→Utilities→ColReg

1.52. Sailing Education and Certifications

Sailing education links available under: App Menus→Internet→Nauticed Sailing Training

1.53. Find a Crew

'Find a Crew' link available under: App Menus→Internet→Find a Crew

1.54. Suggestions

The beauty of Linux is that you can customize it for your needs in infinite ways. While this distro aimed to strike common need, you will find that number of post-install customization steps would be required. The key is to script those steps, make them non-interactive, make the steps requite NO GUI. In that case your set-up becomes REPRODUCIBLE in case of new OS image releases. You can share your post-install scripts, so the system can be improved and even more fine-tuned.

While the system supports touchscreens during set-up phase you would still want to have a regular wired keyboard and mouse attached to it as there is plenty of activities involved on the shell command line.

You do not have to be a software engineer to install the system. A mechanic, electrician, paralegal professional, civil engineer, money manager are few examples of people with different backgrounds who were able to install and set it up for their boats.

1.55. Hardware

This is not my first build of the boat computer with raspberry pi. A lot of ideas can be taken from my older (2020 build which was based on OpenPlotter). For up-to-date build I would change few things:

- Instead of expensive Argonaut M7 I would have used TouchThink TPC-CD-M101, Cost under $300: https://www.touchtecs.com/

- My waterproofing technique would be cheaper and better. Instead of costly connectors at the back of the computer (even if it is below deck) I would use waterproof glands for exits from the enclosure and pigtail connectors. I would cover the point of connection with heat shrink tubing.

- I would have used some kind of safe power-off solution and SSD instead of just SD card. SSD gives HUGE performance gain.

- I would use this OS image (instead of OpenPlotter image)

- dAISy AIS is better solution than SDR

- Use USB 2.0 hub where USB 3.0 not required

- SD card should be easily replaceable for boot. There shouldn't be many screws to get access to it.

For older hardware solutions (lot of it is still valid) see:

https://bareboat-necessities.github.io/my-bareboat/

1.56. Printing

Printing from Chromium is a bit inconvenient. You need to choose 'Print' menu, then scroll down and choose 'More Settings' and select 'Print Using System Dialog' which will let you choose a printer.

1.57. Social Networks and Messaging

BBN OS has several programs to help you stay connected with friends via social networks and messaging applications via Internet.

1.58. Location Reporting

SignalK comes with preinstalled plugin for saillogger.com

1.59. Most useful features for average short cruises

- GPS, OpenCPN off-line charts for your sailing area

- IMU Compass

- Tides / Currents in OpenCPN

- Waypoints, routes in OpenCPN, tracking

- AIS in OpenCPN

- Weather Windy, etc. (if using LTE internet)

- Music players

- Dashboards Wind, Speed, Depth, GPS, Local Sunset, etc

- Barograph

- Autopilot (if equipped)
- Cameras for docking and night light
- Local boat Wi-Fi hotspot and LTE gateway

1.60. IoT

1.60.1. Mosquitto MQTT Server

BBN OS image comes with Mosquitto clients and server preinstalled but disabled. To enable and start it:

1. Enable and start Mosquitto server (on default port 1883)

   ```
   sudo systemctl enable mosquitto
   sudo systemctl start mosquitto
   ```

2. Subscribe to topics

   ```
   mosquitto_sub -t '#'
   ```

3. Publish from SignalK: Enable MQTT Gateway plugin. Pick 'Send data to remote server' with URL mqtt://localhost. Add topic (Ex: navigation.headingMagnetic). Restart SignalK and observe the data stream in the subscriber.

1.60.2. NodeRed

NodeRed is installed into SignalK. To use NodeRed login into SignalK by visiting http://localhost:3000 clicking 'login', choosing 'remember me' session and storing the password in a browser.

More: https://github.com/SignalK/node-red-embedded

Example flows: https://flows.nodered.org/node/@signalk/node-red-embedded

SenseHat into SignalK: https://forum.openmarine.net/showthread.php?tid=3899&pid=21758#pid21758

Solar Controller: https://forum.openmarine.net/showthread.php?tid=3397

You can search for ready flows and additional nodes on https://flows.nodered.org/

To install additional nodes using npm, make sure you run npm under 'signalk' user and from /home/signalk/.signalk directory.

1.61. Data Analytics

1.61.1. Grafana

BBN OS image comes with grafana preinstalled but disabled. To enable and start it:

1. Edit the /etc/grafana/grafana.ini and /usr/share/grafana/conf/defaults.ini files and change line ';http_port = 3000' to 'http_port = 3080' (to avoid conflict with SignalK)

```
sudo nano /etc/grafana/grafana.ini
sudo nano /usr/share/grafana/conf/defaults.ini
```

2. Enable and start Grafana server

```
sudo systemctl enable grafana-server
sudo systemctl start grafana-server
```

3. Access http://localhost:3080 with user and password 'admin'.

1.61.2. InfluxDB

1. Enable and start InfluxDB server (on default ports 8086 (client-server), 8088 (RPC))

```
sudo systemctl unmask influxdb.service
sudo systemctl enable influxdb
sudo systemctl start influxdb
```

2. Initialize database and connect to it (for example) from SignalK barograph plugin.

Chronograf and Kapacitor also come pre-installed. To enable:

```
sudo systemctl enable kapacitor
sudo systemctl start kapacitor

sudo systemctl enable chronograf
sudo systemctl start chronograf
```

1.62. Power consumption

With Raspberry Pi4 power consumption of your system should be around under 4.5 watts without a monitor. A monitor depending on it's size and backlight brightness (needed for sunlight readability)

can add another 5 watts or even more. Human brain runs on about 12-25 watts for comparison.

1.63. Upgrading your System

1.63.1. Verified Updates

Fixes verified by the development team can be installed by running:

```
cd /home/user/add-ons
./hot-fixes-install.sh
```

1.63.2. All Other Updates

Your system packages can be upgraded following standard Debian packaging system procedure:

```
sudo apt update
sudo apt upgrade
```

SignalK server and modules can be upgraded using command line: Example for signalk-hwt901b-imu:

```
cd /home/signalk/.signalk
sudo su signalk -c "rm ../.signalk/pnpm-lock.yaml; pnpm config set auto-install-peers
true"

sudo su signalk -c "pnpm install signalk-hwt901b-imu --unsafe-perm --save-peer"
```

OpenCPN charts can be updated using OpenCPN chart downloader plugin.

> ℹ️ As result of software upgrades you might end up with a configuration which wasn't previously tested. Though you might be able to get some support via community forums at: https://github.com/bareboat-necessities/lysmarine_gen/discussions

> ⚠️ Any attempts to upgrade underlying OS (Buster) to Bullseye will lead to a broken system. Until Bullseye based version is available you can't have this system running on bullseye.
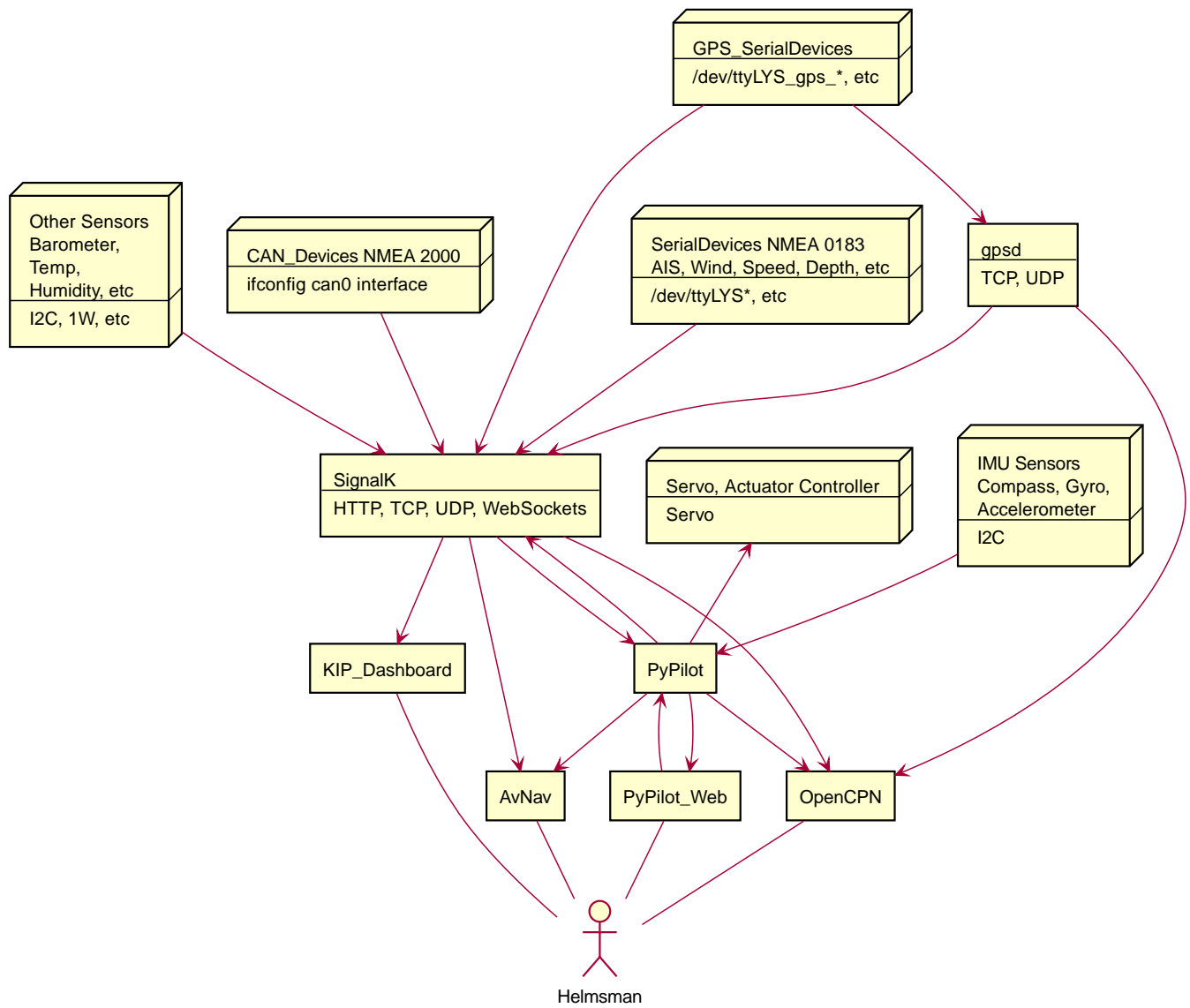
1.64. Data Flow

Figure 2. Data Flow

Chapter 2. Supported Hardware

2.1. Monitors

For a cabin monitor many people do not use special marine grade ones. If you have enough space on your nav station desk for a keyboard and a mouse, you might consider getting a screen without touch functionality.

For a cockpit display you need >1000 nit brightness for sunlight readability, waterproof (IP67 or more), touchscreen, with accessible backlight (brightness) control buttons for night sailing. You would need long HDMI and USB (for touch) cables to connect to raspberry pi below deck. There are even fiber optic solutions for that. They allow thinner cables (at additional cost). Bonus, if your touchscreen display comes with built-in speakers for alarms. Check Sihovision product line, if one of their products will suit you. Some examples:

SL7W 7" https://www.alibaba.com/product-detail/sunlight-readable-7-inch-touchscreen-waterproof_60543830769.html

SL10W 10" https://www.alibaba.com/product-detail/Outdoor-display-9-7-inch-IPS_1600251157854.html

SL12W 12" https://www.alibaba.com/product-detail/daylight-viewable-monitor-ip67-touch-marine_60718449450.html

SL113 13" https://www.alibaba.com/product-detail/Marine-displays-IP67-waterproof-13-3_62066633231.html

SL100W 10" https://www.sihovision.com/full-ip65-high-brightness-touch-monitor/waterproof-monitor-sl100w.html

More ideas from TouchThink: https://www.touchtecs.com/marine-monitor/ip65-sunlight-readable-marine-display-15-inch.html

Pick HDMI screen over DSI as they do not take extra GPIO pins which you might have other plans for. Most touchscreens nowadays will be capacitive multitouch due to better durability and clearness.

2.1.1. Connecting monitors

Instead of laying long USB and HDMI cables from a cockpit to the boat computer below deck, you could use HDMI USB over Cat6 Ethernet extender. Given the high cost of longer USB and HDMI cables, this solution can be even cheaper overall. Plus instead of two thick cables, you would need to lay only one and thinner Cat6 Ethernet cable. However, double check with the vendor that the extender supports touchscreen functionality over USB.

This might also work for setting dual (mirrored) monitors.

2.1.2. Brightness

LED is LCD with LED backlight. IPS (In-plane-switching) is a type of LED with better picture quality and more power consumption than TFT (thin-film-transistor) LCD. OLED doesn't have backlight.

Sunlight readability is >= 1000 nits.

Software brightness control is possible on monitors which support DDC/CI (Display Data Channel) with raspberry pi via ddcutil program. It works as an i2c channel over HDMI.

Software brightness control is also possible with monitors which create entry in /sys/class/backlight/<entry> on raspberry pi. xbacklight utility can be used to control backlight brightness in such cases.

Choose monitors with less than 1 watt usage per one diagonal inch.

2.1.3. Round touchscreens for gauges

Pimoroni HyperPixel 2.1 Round

2.2. Rpi4 Case

DeskPi Pro v2 is a nice enclosure for your below deck Raspberry Pi4 with fast SSD interface and 12v input.

https://github.com/DeskPi-Team/deskpi

Advantech Uno 220 for Raspberry Pi4 looks good, however doesn't come with SSD support.

2.3. CM4 Based Solutions

One of the promising CM4 boards and computers is Waveshare Industrial IoT Mini-Computer Based on Raspberry Pi Compute Module 4:

https://www.waveshare.com/product/raspberry-pi/boards-kits/compute-module-4-cat/cm4-io-poe-4g-box.htm

Looks like can be a much better base for a boat computer than just regular Raspberry Pi4.

It comes with (and more):

- 12v power input (7v-36v)
- PCIe slot
- External antennas connectable to CM4 Wi-Fi antenna port, and to GPS or LTE antenna port on LTE modem

- CanBus (for NMEA 2000)

- RS-485 (for NMEA 0183)

- GPIO exposed via header with screws

- ADC interface

- I2C interface

- RTC clock with a place for a battery

- SIM card slot on a panel

- Full size double HDMI ports and two DSI ports

- Screw mountable case

- Dual camera inputs (CSI)

- USB ports

- USB TO UART, for serial debugging

- One Gigabit Ethernet port with PoE enabled

- More status lights

- Panel SD card slot (for CM4 without eMMC)

- RS-232 port

- Slot for LTE Modem

- Programming Slot

- Buzzer

Or Waveshare Mini-Computer Based on Raspberry Pi Compute Module 4, Mini IO Board Version B

https://www.waveshare.com/product/raspberry-pi/boards-kits/compute-module-4-cat/cm4-io-base-box-b.htm

with a space for SSD.

To enable RTC clock if your Waveshare board has it you need to add into /boot/config.txt something like:

```
# Enable RTC
dtparam=i2c_vc=on
dtoverlay=i2c-rtc,pcf85063a,i2c_csi_dsi
```

Another option is the board and a case from McuZone. The advantage of it is that it has no moving parts (just a cooling radiator).

For McuZone board you will need to add:

```
dtoverlay=dwc2,dr_mode=host
```

into /boot/config.txt before first boot to enable USB ports.

List of the boards: https://pipci.jeffgeerling.com/boards_cm.html

Several of these products could be interesting for boating applications:

- Waveshare Mini IO Board Computer (ver B): https://www.waveshare.com/product/cm4-io-base-box-b.htm

- Waveshare IoT CM4 Computer https://www.waveshare.com/product/raspberry-pi/boards-kits/compute-module-4-cat/cm4-io-poe-4g-box.htm

- MCUzone Carrier Grade CM4 Computers (fanless) https://www.aliexpress.com/item/1005001972265702.html

- EDATEC CM4 Industrial Computer https://www.edatec.cn/en/Product/Camera_Modules/2019/0826/76.html

- PiGear Nano https://www.uugear.com/product/pigear-nano/

- SeeedStudio reTerminal DM https://www.seeedstudio.com/reTerminal-DM-p-5616.html

> Check internal SSD drive size (2242, 2260 usually are ok, 2280 is usually too long) and compatibility with your CM4 board before purchasing. If m.2 slot on your board is NVME you need NVME SSD (SATA will not work). Some boards m.2 slot designed to work only with 4G/LTE modems with PCIE M.2 KEY-M interface (will not work with SSDs).

> If you use a Waveshare board, and your CM4 module comes with eMMC then you can't boot your from SD card, and you have to burn eMMC. To burn eMMC on Windows you have to install latest rpiboot_setup.exe from usbboot project on GitHub. You would need to switch jumpers (switches) on your board to use OTG USB slave cable to be able to burn eMMC. It's a slow process as it is done via USB 2.0 interface. Moreover, stock Raspberry Pi OS image wouldn't have Waveshare USB ports enabled. So before burning an image into eMMC you would need to modify it and add dtoverlay=dwc2,dr_mode=host line into /boot/config.txt on the image. For that (and other) reasons, BBN OS has a separate image for Waveshare board support. The image is based on the base BBN OS image with additional tweaks to support the Waveshare board.

2.3.1. Home Assistant Yellow

Home Assistant Yellow includes also ZigBee interface for IoT.

https://www.crowdsupply.com/nabu-casa/home-assistant-yellow

2.4. Industrial Pi5 Solutions

- Pi5 EDATEC ED-IPC3020 https://www.edatec.cn/en/elpc/ipc3020.html

- Pi5 EDATEC HMI3010-101C https://www.tomshardware.com/news/raspberry-pi-5-industrial-pc

2.5. IMU, compass, accelerometer, gyroscope

PyPilot supports these via RTIMULib2. Here is a list of supported units by RTIMULib2:

- InvenSense MPU-9150

- InvenSense MPU-9250

- InvenSense MPU-9255

- STM LSM6DS33/LIS3MDL

- STM L3GD20H/LSM303D

- STM L3GD20/LSM303DLHC

- STM LSM9DS0

- STM LSM9DS1

- STM LSM6DSL

- InvenSense ICM-20948

- Honeywell HMC5883L with ADXL345 and L3G4200D

- STM ISM330DHCX

- Bosch BMX055 (experimental magnetometer support)

- Bosch BNO055 (doesn't work reliably on Pi, clock issues)

PyPilot Documentation recommends using 9 Dof (9 Degrees of freedom) IMU with icm20948.

Even without autopilot motor controller these sensors allow displaying roll, pitch, and yaw.

SignalK plugin for electronic compass WitMotion: https://github.com/W-Geronius/signalk-hwt901b-imu

See also: https://open-boat-projects.org/en/magne

2.6. Environmental Sensors

2.6.1. Barometer, Temperature, Humidity I2C Sensors

These are handled by Signal-K plugins. Supported ones are:

- Bosch BME280

- Bosch BME680

There is also barograph plugin in SignalK.

2.7. AIS

There are several options:

- If you have AIS transducer you just use it via NMEA bus.

- AIS built in into NMEA multiplexer used via NMEA network

- dAISy USB AIS or Raspberry Pi dAISy AIS Hat. USB one is easier to set up.

- Decode AIS radio signals using SDR. Probably the worst option as SDR is quite energy consuming for that task to be run constantly.

- Open Source (receiver/transducer): https://github.com/peterantypas/maiana

2.8. Radars

OpenCPN Radar plugin reports support of the following radars:

- Navico/B&G/Lowrance/Simrad: BR24, 3G, 4G, and HALO20, 20+, 24, 4, 6, and 8

- Garmin: HD and xHD

- Raymarine: Ethernet-Analog: RD218, RD418 / Ethernet-Digital: RD418D, RD424D

Quantum/Quantum2 Radars from Raymarine seems are not supported by OpenCPN, nor are the Garmin Fantom radars, or Furuno units.

We didn't have hardware to all these, so reports of successful installations and encountered issues are welcome.

2.9. USB Hubs

Choose powered USB hub which doesn't backfeed power into pi4, to avoid issue of pi4 not being able to reboot when USB hub is still 'on'.

If you do not plan to connect to your USB hub storage devices, cameras, network cards (high speed devices), then USB 2.0 hub will do the job. No need for USB 3.0 hub.

USB ports follow color convention:

Type	Color
USB 1.0	**White**
USB 2.0	**Black**

Type	Color
USB 3.0	Blue
USB 3.1	Teal
Charge & Sleep or High Current	Yellow
	Red
	Orange

The hub we like: https://www.amazon.com/dp/B07BBLL3MJ/ Aiibe 6 Ports Super High Speed USB 3.0 Hub Splitter

2.10. LTE/Wi-Fi Routers

The Wi-Fi router you need is somewhat different from your home router. Your home router connects wired to Internet WAN, and it gives you local Wi-Fi. On a boat the situation reverses. Your WAN is the marina's Wi-Fi and your Pi should (preferably) be wired into the router. These Wi-Fi routers are being sold as Travel Wi-Fi routers. Not all cheap Wi-Fi routers will give you the ability to connect to the marina Wi-Fi and share it on a boat. It's better to choose the ones running OpenWrt operating system, which is Unix based and open-source. LTE is a very valuable feature.

Few recommended would be: GL.iNet (GL-X750 V2) Spitz router, Teltonika RUTX12 (dual SIM).

Others to consider: InHand Networks IR300 Compact Industrial Router, YeaComm Industrial or YeaComm Outdoor LTE Routers.

2.11. SD Cards

In SD card you will be looking at speed (i/o reads, i/o writes), durability, size. >32Gb is recommended.

2.12. SeaTalk1

SeaTalk1 is Raymarine proprietary protocol. Many popular Autohelm Raymarine autopilots use it. SeaTalk1 unlike NMEA 0183 is a bus allowing multiple talkers and multiple listeners of the same bus. To integrate it into your system you have a number of choices:

- Use bidirectional NMEA 0183 multiplexer which supports one SeaTalk1 port. First vendors to check are YakBitz and Quark-elec Marine. There are others as well. You want bidirectional multiplexer to be able to control autopilot from your pi via NMEA. There is a dedicated Autohelm autopilot plugin in SignalK.

- Buy a dedicated Seatalk1 to NMEA converter.

- For read/only Seatalk1 connection you can use this solution: https://github.com/SignalK/signalk-server/blob/master/docs/src/setup/seatalk/seatalk.md

SeaTalk1 had been reverse engineered and a converter into NMEA is available at: https://github.com/MatsA/seatalk1-to-NMEA0183

https://github.com/Thomas-GeDaD/Seatalk1-Raspi-reader

https://github.com/yOyOeK1/seatalk4

https://github.com/AK-Homberger/Seatalk-Autopilot-Remote-Control

2.13. B&G Fastnet

https://github.com/trlafleur/Fastnet

https://www.oppedijk.com/bandg.html

2.14. NMEA 0183

Widely supported. Many options to connect to pi.

USB/Serial:

- with FTDI FT232 chipset https://www.amazon.com/dp/B07B416CPK
- with CP210x chipset RS422 to USB converter
- with CH340 chipset RS422 to USB converter
- with Prolific PL2303 chipset RS422 to USB converter
- NMEA 0183 Multiplexers working over USB (YakBitz, Quark-elec, etc)

> ℹ️ FTDI is preferable choice between FTDI, CP210x, CH340, or PL2303. FTDI seems always have unique serial ID which might be missing on others, and it will make harder to write udev rules to uniquely name the device on USB bus.

Board:

- PiCAN-M Raspberry Pi Hat by https://www.skpang.co.uk
- Sailor Hat for Raspberry Pi by https://hatlabs.fi
- GeDaD MCS, Marine Control Server Board by https://www.gedad.de/

TCP/IP:

NMEA 0183 Multiplexers working over Wi-Fi or Ethernet

Vendors:

- YakBitz http://www.yakbitz.com/

- Quark-elec https://www.quark-elec.com/

- OceNav https://ocenav.com/

- ShipModul http://www.shipmodul.com/

- Yacht Devices https://www.yachtd.com/

- DigitalYacht https://digitalyachtamerica.com/

- Actisense https://actisense.com/

- NoLand Engineering https://nolandeng.com/

- Veinland https://veinland.net/

- Zinnos http://zinnos.com/

- Maretron https://www.maretron.com/

- Encoded Solutions https://www.encodedsolutions.com/

- Comar Systems https://comarsystems.com/

- gadgetpool.de https://www.gadgetpool.eu/

- Star Tracking https://www.star-tracking.com/

- Vela-Navega https://vela-navega.com/

- Pitufino https://www.pitufa.at/pitufino/

Yacht Devices multiplexers stand out from others by also providing a web UI with a dashboard with gauges.

2.15. NMEA 2000

NMEA 2000 protocol was reverse engineered as a part of this project: https://github.com/canboat/canboat

Since then NMEA 2000 had been natively incorporated into SignalK as well.

Options to connect Pi to NMEA 2000 bus:

PCB boards/hats:

- PiCAN-M Raspberry Pi Hat by https://www.skpang.co.uk

- Sailor Hat for Raspberry Pi by https://hatlabs.fi

- GeDaD MCS, Marine Control Server Board by https://www.gedad.de/

USB/Serial:

- Actisense NGT-1 NMEA 2000® to PC https://actisense.com/products/ngt-1-nmea-2000-to-pc-interface/

- CANable and CANable Pro by https://canable.io/

- iKonvert NMEA 2000-USB Converter

- ShipModul MiniPlex-3USB-N2K

- Yacht Devices NMEA 2000 USB Gateway YDNU-02

- CSS Electronics https://www.csselectronics.com/pages/can-bus-hardware-products

- TouCAN Marine https://www.rusoku.com/products/toucan-marine

TCP/IP gateways:

- Bi-directional NMEA 2000 Multiplexer Wi-Fi (Quark-elec, etc)

- ShipModul MiniPlex-3E-N2K

- Yacht Devices NMEA 2000 Wi-Fi Gateway YDWG-02

- Yacht Devices NMEA 2000 Ethernet Gateway YDEN-02

- NavLink2 digitalyachtamerica.com

- Pitufino https://www.pitufa.at/pitufino/

Monitoring devices: - https://simarine.net/

OpenCPN TwoCan plugin has a list of compatible with it devices as well.

Stay away from devices which just convert NMEA 2000 into NMEA 0183. You need to feed NMEA 2000 into SignalK and be able to receive it too.

2.16. NMEA OneNet, PoE

NMEA OneNet is Ethernet based with RJ-45 or X-Coded M12 8-pin (to pass smaller openings) connectors.

PoE (Power over Ethernet) allows powering devices with same data cables. It is a nice way for reducing the number of cables in your system.

2.17. Topline (NKE)

Compared to NMEA0183, Topline can transmit significantly more data due to the compact binary data transmission and the high data rate of 38,400 baud.

NKE devices are designed and optimized for professional sports.

https://github.com/KristianJerpetjon/NkeToplineCpp

2.18. RayMarine Micronet TackTick

RayMarine protocol for wireless intruments.

https://github.com/Rodemfr/MicronetToNMEA

2.19. SAE J1939

J1939 is a CAN-BUS protocol for monitoring automotive engines, but it's also being used on boats. Usually it requires J1939 to NMEA 2000 gateways.

Vendors of such gateways:

- Maretron

- Yacht Devices

- Oceanic Systems

2.20. GPS

Several options to connect:

USB:

- Many USB "mouse" GPS should work on BBN OS after plugging-in and giving them some time to acquire satellites outside the house. Cold start can take half an hour or so with some GPS models.

Re-use GPS built-in into LTE router:

- LTE modems on LTE routers have built-in GPS. If you followed our advice on using a router with OpenWrt OS then you will be able to install KPlex (or gpsd) on it and set up a connection to it from SignalK (or/and OpenCPN, AvNav) running on your raspberry pi.

NMEA:

- Use GPS from your NMEA network. VHF have GPS and NMEA interfaces. Some NMEA multiplexers come with GPS built-in as well as most chartplotters.

PCB Boards connected to Pi:

- Example: BerryGPS. The disadvantage of those is that they use your Pi's built-in UART and you might have other devices that you want to use it instead.

2.21. SDR, SSB

RTL-SDR, SDRPlay SDR, HackRF SDR, LimeSDR, and likely others. Interfaces USB, Ethernet.

Yaesu, Icom are the popular vendors within HAM community.

SDRs consume quite some power, so it's better to connect them to pi via a powered USB Hub (in case of USB interface).

2.22. SSD

Raspberry Pi OS supports NVMe and SATA SSDs with an appropriate extension board, or a Pi case, and their vendor's driver.

NVMe apparently doesn't give much performance boost over SATA for pi4 with an exception to CM4 (Pi4 Compute Module) performing better on writes with NVMe drives. pi4 internal interface is the bottleneck for transfer rate speed.

Booting from NVMe drives got supported by linux kernel on pi4 just recently (some bugs might remain). See https://youtu.be/4Womn10v71s for instructions on booting from NVMe drives (while this feature is still in beta).

Perform a benchmark of your SSD after installation with correct drivers. Make sure it falls into expected range.

SSD is also less prone to data corruption on sudden power cut-offs than SD cards.

USB stick drives (flash drives) performance is not as high as NVMe and SATA SSDs.

2.23. NavTex

- NASA Marine PC Navtex USB with an antenna (BBN OS comes with a program called PC-NavTex for it). ~$220
- USB Devices from wetterinfobox.com (WIB2 - NAVTEX for PC) http://www.wetterinfobox.com

There are reports that PC Windows software for WIB2/3 can be run on Raspberry Pi with Box86 and Wine: https://www.cruisersforum.com/forums/f134/navtex-plugin-251561-3.html#post3533558

2.24. Moitessier HAT

Moitessier HAT has a closed source kernel driver. Contact Moitessier HAT vendor for a driver for your kernel version and with it the HAT will work.

2.25. RTC (Real Time Clock)

Raspberry pi doesn't have clock module to keep the time when it's off, so you might consider adding one.

There are a few common i2c RTC modules:

- DS3231 RTC
- DS1307 RTC
- PCF85063A RTC

Make sure there are no conflicts in addresses with your other i2c devices. Other devices usually have at least one option for switching to another address to avoid conflicts.

2.26. Sound Cards

As Raspberry Pi doesn't have an audio input, for use with SSB radio you might want to have an additional sound card which has both audio in and out. There is a good selection of options, but check reviews and the compatibility with Raspberry Pi4.

2.27. Cameras

2.27.1. IP Cameras

IP Cameras should work with MotionEye using their rtsp:// URLs.

2.27.2. Connecting cameras to CSI on raspberry pi

Geekworm Raspberry Pi Hdmi-in Module, Hdmi to CSI-2 could be used to solve distance issue with CSI cameras.

https://geekworm.com/products/raspberry-pi-hdmi-to-csi-2-adapter-board-with-15-pin-ffc-cable

Another trick with Arducam CSI to HDMI Cable extension is for CSI cameras to connect to raspberry pi over longer HDMI. You can't use HDMI camera, HDMI is used merely as wire signal extension for CSI.

https://www.arducam.com/product/arducam-csi-hdmi-cable-extension-module-15pin-60mm-fpc-cable-raspberry-pi-camera-specific-pack-2-1-set/

2.27.3. Low light vision

Sionyx has some boating oriented solutions: https://www.sionyx.com/pages/boating

Image stabilization for boat movements is an important factor too.

Night vision FLIR

While FLIR cameras (from the company which bought major manufacturer Raymarine) are great, they are prohibitively expensive for most recreational boaters.

2.27.4. USB connected cameras

Supported with USB 3.0

We need your help with testing it and building some recommendations on models, etc.

2.28. ESP32 / Arduino

Using ESP32 controllers and a bit of micro python programming you can connect all kind of sensors to your system. One of product lines we like is from m5stack.com, and another is Wio Terminal by seeedstudio.com.

M5Tough: https://shop.m5stack.com/products/m5stack-tough-esp32-iot-development-board-kit

Our m5stack based boat display: https://github.com/bareboat-necessities/my-bareboat/tree/master/m5-boat-display

Here is a dinghy location tracker idea. https://www.seeedstudio.com/LoRa-GPS-Tracker-with-Wio-Terminal-p-5182.html (Needs some work on a waterproof enclosure for a transmitter, and reporting the dinghy location via AIS NMEA data stream to the mother ship's chartplotter). WIO terminal is not esp32 though.

2.29. ESP32 with SignalK

To program ESP32 for communicating with SignalK you can use SignalK SensESP library: https://github.com/SignalK/SensESP

SensESP allows connecting many sensors:

- Engine RPM
- Engine Temperature, Oil Pressure
- Liquid levels (Fuel, fresh water, black water, septic tank)
- Environment (Temperature, pressure, humidity, air quality)
- Rudder position
- Dangerous gases
- Lightning strikes detectors
- Voltage, amps
- Relays for control (lights, devices)
- Power utilization (batteries monitoring)
- Motion sensors
- Human heartbeat detection
- Anchor windlass chain counters
- Bilge pump
- Proximity, distance sensors
- LiDAR

- Fingerprint recognition

- Light sensors

- Color detection

- pH alkalinity and acidity sensor

- 433MHz RF receivers

- LoRa receivers/transmitters

- Interfaces (USB, Ethernet, CAN Bus, Rs-485, RS-232, Wi-Fi, 4G/LTE, GPS/GNSS, ModBus, etc)

- IMU (gyroscope, magnetometer, accelerometer)

- Hall effect

- Actuators, servo, motor controls

- Audio (speaker, microphone), buzzers

- Barcode/QR code readers

- External memory readers (TF card, etc)

- Laser emitters / receivers

- NCIR/IR receivers/transmitters

- Thermal cameras

- Joystick

- DAC/ADC

- Potentiometers

- Blood alcohol concentration

- Blood oxygen, oximeter

- and more

2.29.1. BBN Esp32 Projects

- MFD (multi-functional display) based on waterproof M5Stack Tough: https://github.com/bareboat-necessities/bbn-m5stack-tough

- Environmental NMEA 2000 sensors by BBN Projects: https://github.com/bareboat-necessities/bbn-nmea200-m5atom/tree/main/bbn-nmea2000-env-m5atom

- DIY N2K to USB gateway in Actisense format on M5Atom by BBN Projects: https://github.com/bareboat-necessities/bbn-nmea200-m5atom/tree/main/bbn-nmea2000-usb-gw-m5atom

2.29.2. PlatformIO IDE

SensESP requires using PlatformIO IDE, which you can install using these steps:

Install Visual Studio Code

```
sudo apt update
sudo apt install code
code
```

Install PlatformIO IDE by clicking 'Extensions' icon in Visual Studio Code and typing 'PlatformIO' in the search field.

2.30. 1-Wire

SignalK Raspberry Pi 1wire plugin supports connecting multiple DS18B20 1-wire temperature sensors.

2.31. IoT

Sonoff Switches - Supported by SignalK plugin.

Ecowitt weather sensors - Supported via SignalK plugin.

Shelly https://shelly.cloud/ - Supported via SignalK plugin.

EmpirBus NXT https://www.empirbus.com/ - Supported via SignalK plugin.

ZigBoat https://www.zigboat.com/

Keep in mind that most Wi-Fi IoT devices will not work on 5GHz Wi-Fi. They usually require 2.4GHz Wi-Fi bands.

2.32. Power, 12v → 5v Converters

2.32.1. DC Voltage Converters

This converter was used with no issues:

TOBSUN EA50-5V DC 12V 24V to DC 5V 10A 50W Converter Regulator 5V 50W Power Supply Step Down Module Transformer

```
Over-voltage, over-current, over-temperature, short-circuit auto protection
```

```
Input voltage: 12/24V, Output: 5V/10A
```

https://www.amazon.com/dp/B01M03288J

As well as this UBEC:

US Ship Hobbywing 5V/6V 3A Switch-mode UBEC, Max 5A lowest RF

https://www.amazon.com/dp/B008ZNWOYY

Avoid powering Pi4 via GPIO pins directly as it bypasses an overvoltage self-healing fuse built-in into pi.

2.32.2. Power Management, Victron, Venus OS

SignalK has good support for interfacing with Victron Energy products and their Venus OS.

SignalK comes with:

- Victron Venus OS Plugin
- ModBus Client Plugin

NodeRed comes with nodes supporting Victron too.

HomeAssistant power monitoring has good support of Victron devices as well. https://community.victronenergy.com/questions/63997/home-assitant-integration-with-victron.html

2.33. Sonars

2.33.1. EchoZilla Open Source Sonar

EchoZilla Open Source Sonar: https://echozilla.co.uk/

Source code: https://echozilla.co.uk/pages/developers

2.34. Drones

List of supported drones:

https://ardupilot.org/planner2/docs/common-rtf.html

2.35. IR (Infrared)

The Raspberry Pi can receive and send IR signals using its GPIO pins.

You configure which pins your IR diode is connected to in /boot/config.txt like this:

```
dtoverlay=gpio-ir,gpio_pin=19
dtoverlay=gpio-ir-tx,gpio_pin=26
```

LIRC (https://lirc.org/) software is a part of BBN OS. You need to configure it for your IR remote.

2.36. Actuators

Actuators discussed in this thread: https://forum.openmarine.net/showthread.php?tid=3468

2.37. Open Boat Projects

Open Boat Projects: https://open-boat-projects.org/en/

2.38. Antennas

There is a great variety of antennas of a boat. Directional vs omnidirectional, active vs passive, internal (some are just a part of a circuit on a PCB) vs external, grounded vs ungrounded, receiving-only vs receiving/transmitting, and a wide variety of shapes.

2.38.1. VHF, AIS

VHF and AIS use similar frequencies, need be in direct line of sight, so the higher they are the better range. They are limited by Earth curvature. The range is around 30 miles. They can use the same antenna with a proper splitter. Transmitter device on AIS has to be licensed and registered.

2.38.2. Wi-Fi, LTE/4G

Similar but distinct separate antennas. Can work even if inside cabin of a fiberglass boat. Range varies on antenna gain. Wi-Fi usually in hundreds of feet. LTE/4G in/under few miles. LTE/4G uses cellular network of antennas of network providers which covers most populated areas.

2.38.3. GPS

Receive-only. Doesn't need to be high. Needs a clear sky view of multiple satellites. External ones are usually active and placed inside a mushroom-looking plastic. Designed to have almost world-wide coverage (except of extreme North/South poles areas). There are several GNSS systems with different satellite constellations. GPS (US), BEIDOU (China), GALILEO (EU), GLONASS (Russia), and QZSS (Japan)

2.38.4. Satellite Data (Iridium, StarLink, etc)

Both are satellite communications. Iridium designed for a hand held phone with a small antenna. StarLink is communicating with more satellites on much lower orbits, and it is designed for a directional pizza-sized dish antenna, automatically orienting itself using a motor. Designed to have

almost world-wide coverage (except of extreme North/South poles areas).

Satellites data providers:

- Iridium
- Orbcomm
- Inmarsat
- Globalstar
- StarLink
- Swarm
- Others as well

Garmin Inreach provides reasonably priced satellite data plans via Iridium.

2.38.5. FM (Frequency Modulation)

Short piece of wire, small telescoping antenna. Receive only. 30-40 miles range. Transmitted from ground stations. VHF is FM too, but FM is loosely used term for music and local news broadcast channels.

2.38.6. Bluetooth, 433MHz FM RF

Bluetooth antennas are small and often just a part of a PCB circuit. Bluetooth range is really short, and it is about 10 meters.

433MHz FM RF range is about 300 meters. Often used for remote controls. Wireless masthead wind transducers, etc.

2.38.7. Zigbee, Z Wave

Zigbee operates on short distances 10-100m in about same spectrum range as Wi-Fi with lower bandwidth but higher reliability. There are some products for boats such as from https://www.zigboat.com/ Zigbee is for LAN (comparing with Bluetooth being point-to-point).

Z-Wave is similar to Zigbee but uses different RF spectrum which doesn't clash with Wi-Fi.

2.38.8. NFC

NFC (Near field communication) is for wireless communication for two devices over distance up to 4 cm.

2.38.9. RFID

Radio frequency identification. Passive and active tags to carry an identifier information which can be

read over some distance avd not necessarily withing light on sight (not like barcodes).

2.38.10. NavTex, WeatherFax

Receive only. Transmitted from ground stations with predefined intervals of time. WeatherFax frequency depends on the region. You need to know a schedule of WeatherFax transmitting stations for different areas.

NavTex frequency is 518 kHz in the medium frequency band. 490 kHz in addition in some countries. Both NavTex and WeatherFax designed for almost world-wide coverage and offshore sailing.

PA0RDT active mini-whip antenna is good for NavTex and WeatherFax frequencies: http://dl1dbc.net/SAQ/miniwhip.html

2.38.11. SSB, HAM

Transmitting on SSB (Single Side Band) radio requires a license for an operator except of in an emergency. Designed for long range communications at sea and for offshore vessels. An insulated backstay of a mast can serve as SSB antenna. Propagation of SSB radio waves work on reflected or refracted back toward Earth from the ionosphere. Performance ranges in excess of 4000 miles. Some parts of SSB frequency spectrum work better at night, others during daytime.

HAM (aka amateur) radio also requires a license for an operator to transmit.

2.38.12. LoRa

LoRa (Long Range) radio is license-free for both transmitting and receiving. Range 2-5 km in open. Used as a low power (and low traffic bandwidth) wireless platform for Internet of Things (IoT). LoRa antennas are small and usually attached to its devices.

Chapter 3. How you can help

1. Add a star on our GitHub project page https://github.com/bareboat-necessities/lysmarine_gen

2. Report a bug (or contribute to resolving an existing issue) https://github.com/bareboat-necessities/lysmarine_gen/issues

3. Spread the word. There are many sailing forums on Internet and Facebook. Let other people know what BBN OS could do for you.

4. Respond to a question in our discussion forums: https://github.com/bareboat-necessities/lysmarine_gen/discussions

5. Post a video on YouTube explaining how you use BBN OS on your boat

6. Submit a GitHub pull request with code changes to improve BBN OS

7. Post a blog article or contribute to HOWTOs section of this document.

8. Of course, the BBN OS is completely open-source and free to use.

Chapter 4. HOWTOs

Please send us your HowTo, and we can add it here for everyone to find. Thanks

4.1. BerryGPS-IMU V3

- Install the BerryGPS-IMU V3 hat

- Follow steps to enable i2c and disable serial port https://ozzmaker.com/berrygps-setup-guide-raspberry-pi/

- Remove serial console mentioned in /boot/cmdline.txt (argument with serial0 and baud rate)

- Reboot

- sudo i2cdetect -y 1 (should show you addresses)

- Do not create pypilot connection to signalK as it creates a data loop (see https://bareboat-necessities.github.io/my-bareboat/bareboat-os.html#_imu)

- Set up barometer feed from SignalK (https://bareboat-necessities.github.io/my-bareboat/bareboat-os.html#_barometer_temperature_humidity)

- IMU data heading, etc. should come from pypilot NMEA

- check it with telnet localhost 20220

Brief explanation:

- i2c should be enabled, serial console disabled in config.txt

- i2c driver should be loaded at boot (that's what raspi-config step does)

- At this point you should have data readable from GPS (via /dev/serial0) and IMU / barometer (via i2c)

- Now you set up routing of this data into dashboards, chartplotters

- IMU is read by pypilot, which feeds it via 20220 tcp port using NMEA 0183 format

- Do not create authorized READ/WRITE pypilot connection via SignalK web socket to port 3000 of signalK in signalK using their token exchange procedure. (Causes a data loop)

- GPS is read by signalK by creating NMEA connection to /dev/serial0 port (set correct baud rate), or add it as a device for GPSd into /etc/default/gpsd

- Barometric/temp data is read by SignalK using signalK BME280 plugin. Make sure setting polling interval below 30 seconds (because OpenCPN expires non-navigational data every 30 seconds)

- Calibrate level IMU on water. Calibrate your IMU compass, using pyPilot calibration utility. Shutdown SignalK during calibration.

- I use external GPS antenna with it. (It needs to be an active antenna). There is a jumper/or switch on the Berry board to choose external antenna

Cellular, GPS, Wi-Fi Antenna (3 in one), Model AFCJF3, Price $80 Designed for marine and recreational vehicles (RV), this multi-band Cellular, GPS, Wi-Fi Antenna (696 MHz - 5900 MHz / 5.9 GHz) is a 3-way omnidirectional, IP67 waterproof antenna for harsh environment communication applications. https://www.signalbooster.com/products/cellular-gps-wifi-antenna

Check with commands:

```
sudo i2cdetect -y 1
```

should show addresses on the bus, then

```
systemctl status pypilot@pypilot
```

should show that pypilot service as enabled, running and has no errors.

```
telnet localhost 20220
```

should show stream of heading NMEA data from pypilot.

```
telnet localhost 10110
```

should show both GPS and heading NMEA sentences from SignalK.

Also, check this FAQ for common berryGPS issues:

https://ozzmaker.com/berrygps_imu-faq/

disabling echo on serial port might be required, as well as enabling ZDA NMEA time sentence.

u-blox GPS receivers have extensive configuration. Check: https://gpsd.io/ubxtool-examples.html

4.2. Other Wi-Fi drivers

There is a script to install various additional Wi-Fi cards drivers

```
cd /home/user/add-ons
./wifi-drivers-install.sh
```

4.3. Cloud Storage

Using rclone you can store and sync your files on many cloud vendors storage. More: https://github.com/rclone/rclone

4.4. Backup

There are few programs (installed) you can use for backups:

- Restic https://github.com/restic/restic
- Timeshift https://github.com/teejee2008/timeshift
- rclone https://github.com/rclone/rclone
- rsync https://rsync.samba.org/
- rpi-clone https://github.com/billw2/rpi-clone
- and other scriptable solutions

4.5. Using your smartphone as NMEA GPS receiver

Install GPS2IP application on your smartphone. Start it and enable GPS2IP in it. Your phone should be on the same local network as your boat computer. In SignalK add gps2ip_on_smartphone NMEA TCP connection to the IP address shown in GPS2IP app and the port specified there (usually 11123). Your phone ip address should be a static address. So make sure it's mapped to a static address on your Wi-Fi networks with a MAC address of your phone.

4.6. How to navigate with your smartphone as GPS receiver and have an Internet access on a boat within cellular reception

If you have a smartphone on board then you can use it as a source of NMEA GPS data and Internet provider for all your boat devices connected to your boat computer.

- Follow steps Using iPhone as Internet gateway (or Using Android phone as Internet gateway) to connect to your phone and use it as an Internet hotspot.
- Follow steps Using your smartphone as NMEA GPS receiver to use your phone GPS as NMEA 0183 source of GPS data.

4.7. Dual monitor setup

Raspberry Pi comes with two HDMI outputs. However, its software is still not able to handle two heterogeneous displays properly. The hardest case is when one is a touchscreen and another is not,

and they have different screen resolutions, and a mouse attached which needs to flow from one screen workspace to another workspace of a secondary screen. Another challenge is at boot time. What if one screen is powered and another is not, but a user decides to turn power on the secondary one later. Raspberry Pi OS seems unable to cover all these scenarios (nor other linux distributions at the current state).

4.8. Autostart programs on desktop startup

You can set up which programs you want to autostart when desktop comes up in Administration→Budgie Desktop Settings. Many users would probably want OpenCPN and KIP Dashboard right away.

4.9. Running with Higher Resolution Monitors

With higher resolution monitors you might want to adjust font size and UI elements to be bigger on your screen.

- Go into Administration/Settings/Universal Access

- Turn on 'Large Text', set 'Cursor Size' to 'Medium' (Or go into Budgie Desktop Preferences and increase font scaling there. Decreasing scaling below 1 is also useful on monitors with lower resolutions)

- In OpenCPN go to Tools/Options/User Interface

- Set 'User Interface Scale Factor' slider (and all other sliders) to 1

- Press 'OK' to apply settings

- Icon sizes of desktop toolbar are controlled by toolbar size in Budgie Desktop Preferences

4.10. Setting up multiple Wi-Fi access points

For seamless switching for Wi-Fi client from one access point to another you can set up all access points with the same SSID and credentials on different radio frequency channels.

4.11. Watchdog

Raspberry Pi has built in hardware which allows automatic restarts in case it freezes.

To enable this functionality you would need to add:

```
RuntimeWatchdogSec=10
ShutdownWatchdogSec=5min
```

into /etc/systemd/system.conf, reboot, then check watchdog:

```
systemctl show | grep -i watchdog
dmesg | grep watchdog
```

```
RuntimeWatchdogUSec=10s
ShutdownWatchdogUSec=5min
ServiceWatchdogs=yes
[  3.125165] bcm2835-wdt bcm2835-wdt: Broadcom BCM2835 watchdog timer
[  5.200029] systemd[1]: Hardware watchdog 'Broadcom BCM2835 Watchdog timer', version 0
[  5.206438] systemd[1]: Set hardware watchdog to 10s.
```

To test:

```
sudo su
echo 1 > /proc/sys/kernel/sysrq
echo "c" > /proc/sysrq-trigger
```

More details here: https://pysselilivet.blogspot.com/2021/10/raspberry-pi-watchdog-made-simple.html

4.12. Troubleshooting /dev/ttyXXXX device or resource busy

If you get this message likely there is another process which opened /dev/ttyXXXX. To find out the source of the conflict you can use:

```
sudo lsof /dev/ttyXXXX
```

Once you disable a process from using a serial port you are troubleshooting, you should be able to just

```
cat /dev/ttyXXXX
```

and see the data flow.

If it shows some unexpected binary data for a text protocol you can (and should) experiment with baud rate.

```
stty -F /dev/ttyXXXX 4800
cat /dev/ttyXXXX
```

or

```
stty -F /dev/ttyXXXX 38400
cat /dev/ttyXXXX
```

4.13. Display what serial devices you have

```
sudo python3 -m serial.tools.list_ports -v -s
```

4.14. Display what USB devices you have

```
sudo usb-devices
```

4.15. Good IDEs for programmers

- IntelliJ Idea (Community Edition) for Java https://www.jetbrains.com/idea/download/#section=linux
- PyCharm (Community Edition) for Python https://www.jetbrains.com/pycharm/download/#section=linux
- WxFormBuilder for UI App Design for Python https://github.com/wxFormBuilder/wxFormBuilder
- Visual Studio Code for C/C++
- Arduino IDE for micro-controllers C/C++

4.16. Video Editing

```
sudo apt install openshot
```

or another program

```
sudo apt install shotcut
```

4.17. How to mount IMU

IMU PCBs are marked with X-Y-Z labels. There are two conventions:

- ENU (X-East, Y-North, Z-Up)
- NED (X-North, Y-East, Z-Down)

Check what your IMU shows according to the diagram on its board. Mount your IMU in a waterproof box outside your Pi away from magnetic fields, and close to midline of the vessel.

4.18. WaveShare 10 dof install

This is what I had to do to set up WaveShare 10 dof v2 © IMU and baro/temp sensors.

After connecting 4 wires to Pi GPIO pins (as per vendor instructions) for IMU to work I just needed to restart pypilot.

For baro/temp sensors (which are part of Waveshare 10 dof) I had to activate BMP280/BME280 plugin in SignalK. I put 'outside' into SignalK path. And set polling period to 20 seconds. After that I restarted SignalK and got the values in SignalK data feed.

4.19. OpenCPN upgrade

Making a backup copy of your disk is recommended before doing big changes to your system.

```
sudo apt update
sudo apt install opencpn
```

Start OpenCPN and update catalog in Plugin Download Manager. Then update plugins one by one.

4.20. Tripwire

Open Source Tripwire® is a security and data integrity tool for monitoring and alerting on file & directory changes. See: https://github.com/Tripwire/tripwire-open-source

Install:

```
cd ~/add-ons
./tripwire-install.sh
```

4.21. Setting up your system as NTP (time) server

Your NTP server is chrony. Edit /etc/chrony/chrony.conf and add a line with 'allow' directive

```
allow
```

(or better allow followed by your subnet address). Save the file and restart chronyd

```
sudo systemctl restart chrony
```

4.22. Renaming host

```
sudo /usr/local/sbin/bbn-rename-host <my-new-host-name>
```

4.23. Profiling OpenCPN

```
sudo apt install linux-perf-5.10
opencpn
perf top -p $(pidof opencpn)
```

4.24. List dependencies of a package

```
apt-cache depends --recurse --no-recommends --no-suggests --no-conflicts --no-breaks --no
-replaces --no-enhances <your-package-here> | grep "^\w" | sort -u
```

4.25. How to use VNC without monitor attached to pi

You need to set up fixed resolution using

```
sudo raspi-config
```

Without it VNC would show message that it couldn't show desktop.

4.26. Remote Desktop with NoMachine

NoMachine might ba a better alternative to VNC.

Download and install NoMachine for Raspberry arm64 DEB (Bullseye) from
https://download.nomachine.com/download/8.1/Raspberry/nomachine_8.1.2_1_arm64.deb

Install command:

```
sudo dpkg -i nomachine_<version>_arm64.deb
```

Install NoMachine on the connecting client. For cloud version configuration see:

4.27. Free Cloudflare Tunnel To HomeAssistant

https://peyanski.com/connecting-cloudflare-tunnel-to-home-assistant/

4.28. Install AirMail on Raspberry Pi

AirMail is a radio mail program for sending and receiving messages via Pactor over HF radio, either via the ham radio system or participating marine and commercial services. It can communicate with PACTOR modem and other HF radio devices.

To install AirMail on Raspberry Pi with Bullseye follow this guide: https://bareboat-necessities.github.io/my-bareboat/airmail-install.html

To install AirMail on Raspberry Pi with Bookworm follow this guide: https://bareboat-necessities.github.io/my-bareboat/airmail-install-bookworm.html

4.29. Install SAS.Planet on Raspberry Pi

SAS.Planet is a popular in offshore sailing community free and opensource program to prepare free off-line raster charts for OpenCPN from GoogleEarth and others satellite images. More on the subject:

http://svocelot.com/Cruise_Info/Equipment/mbTiles.htm

And some prepared charts: http://svocelot.com/Cruise_Info/Equipment/KAP_Downloads.htm

To install SAS.Planet on Raspberry Pi with Bullseye follow this guide: https://bareboat-necessities.github.io/my-bareboat/sasplanet-install.html

To install SAS.Planet on Raspberry Pi with Bookworm follow this guide: https://bareboat-necessities.github.io/my-bareboat/sasplanet-install-bookworm.html

4.30. Install QGIS GIS

Another program used for creating DIY charts and mbtiles is QGIS. To install:

```
sudo apt-get -y install qgis
reboot
```

4.31. Protecting Electronics from Environment

• Search for "Conformal Coating".

- Waterproof IP67 boxes and cable glands

- Search for "Drip loop"

- Waterproof hot clue

- Waterproof heat shrink tubing

- Epoxy potting for electronics

- Dielectric grease for electrical connectors

4.32. Screen Recording

https://github.com/vkohaupt/vokoscreenNG

```
sudo apt-get -y install vokoscreen
```

4.33. Calculating wave height/period from a moving boat using IMU accelerometer

Also, the link provides a method to estimate true wind from course over ground and actual true heading based on apparent wind measurements.

https://bareboat-necessities.github.io/my-bareboat/bareboat-math.html

4.34. Setting up monitor

/boot/config.txt contains most important settings for your monitor. You might need to adjust them manually.

See:

https://www.raspberrypi.com/documentation/computers/config_txt.html#video-options

https://www.raspberrypi.com/documentation/computers/config_txt.html#common-display-options

Here are few tip to resolve most common issues. (For touch related issues see different parts of this document)

Common issues:

- Borders on monitor. Check overscan related settings.

- Icons on desktop do not appear square. Not optimal resolution is set in config.txt resulting in wrong aspect ratio.

- I need sound to come out of monitor via HDMI. Set hdmi_drive=2 in /boot/config.txt. (Some

individual programs might also require changes in their own settings)

- Maximized windows appears cut from the screen. Check aspect ratio and HDMI mode related settings for your monitor in /boot/config.txt (read about hdmi_cvt options)

- I have high resolution monitor and all icons and fonts are too small. See Running with Higher Resolution Monitors

- Common parameters to look for are: hdmi_cvt, hdmi_drive, hdmi_group, hdmi_mode, disable_fw_kms_setup, disable_overscan, overscan_*

> ℹ️ Versions with KMS driver such as bookworm (instead of fkms in older ones as buster) ignore screen resolution settings in /boot/config.txt. KMS driver takes video parameters from /boot/cmdline.txt. Like this:

```
vc4.force_hotplug=1 video=HDMI-A-1:1280x720@60
```

HDMI-A-1 - name of your screen

1280x720 - its resolution

60 - refresh frequency in Hz

/boot/config.txt needs setting:

```
disable_fw_kms_setup=0
```

For XServer resolution add into your GUI start-up script (such as openbox autostart, etc)

```
# Example
xrandr --output HDMI-1 --mode 640x480 --rate 60
```

4.35. Rotate Screen

There are reports that putting

```
lcd_rotate=2
```

into /boot/config.txt works. However, it might be specific to a monitor model.

4.36. TouchThink Marine HDMI Touchscreen

https://www.touchtecs.com/ produces reasonably priced touchscreen, which can be used in a cockpit.

The product is coded as TouchThink TPC-CD-M101, but you have to spec out your order precisely:

- 10.1 inch, 1280*800 16:10

- 10 points capacitive touch screen monitor, optically bonded

- 1000nit high brightness, -20 to 70 degree wide use temperature.

- industrial front IP67 waterproof dust-proof vehicle use aluminum alloy bezel style.

- 7H hardness tempered glass grade A

- with 12-36V DC.

- waterproof aviation ports: HDMI+touch+DC

- build-in HDMI speakers (on sides under bezel)

- front light sensor, photo sensitive head. change brightness with ambient light

- VESA mountable

- 5 waterproof control buttons on front bezel

- front status light indicator

- with Aviation port to standard port cables: HDMI cable, with touch cable, DC pigtail power cable, with AC/DC power adapter. The EXW price is ~us$222/unit for 1-50 units.

- Ask for a gasket with a panel flush mount kit too.

udevadm info reports this device name as "WingCool Inc. TouchScreen"

You can also request an embedded version with EDATech CM4 module built-in:

https://www.edatec.cn/en/Product/Camera_Modules/2019/0826/76.html

and customize interfaces. For example, you can have Ethernet, USB, I2C on the back.

Chapter 5. How it is Made

For those who are comfortable writing software the scripts to create this image stored on the image itself (for the reference) in /install_scripts. The full source code to create the image is available at: https://github.com/bareboat-necessities/lysmarine_gen

The code should be much easier to understand than openplotter. The code is really just a script of commands you run to get things installed.

5.1. Building without CircleCi

Install Ubuntu 16.04.7 64-bit on PC. Run from shell:

```
curl -L \
 https://raw.githubusercontent.com/bareboat-necessities/my-bareboat/master/bbn-os-
build/bbn-os-build-on-ubuntu.sh \
 | sudo bash
```

Chapter 6. Comparison vs OpenPlotter

See: https://bareboat-necessities.github.io/my-bareboat/bareboat-necessities-vs-openplotter.html

Chapter 7. Cost of Open Source System vs Commercial Marine Electronics

How much* does it cost to build my system? Does it cost less than commercial marine products? The answer is really depends on what you are trying to achieve.

- Open source like BBN OS have features simply not available in traditional marine electronics
- Your cost-effective solution for your boat is going to be a mix of commercial and open products
- If you just need a chart plotter in a cockpit then a commercial product will do
- If you want a smart boat with a lot of sensors then you need to get smart about open source options
- If you live aboard and need a board computer then look at BBN OS and lower power consuming Pi
- With commercial products you end up with lots of black boxes with expensive proprietary connectors and plugs
- With open source you have modular system in which components are easier replaceable and way cheaper
- Day cruiser within cellphone receptions will be fine with a smart tablet or phone
- Live aboard person or someone who cruises for weeks and offshore sailors can benefit greatly from BBN OS and other open source systems
- If you want a boat computer below deck (non-waterproof screen and computer) even if you are just a day sailor then you can build much cheaper system with open source like BBN OS combined with sensors you have onboard
- Consider other features costs like onboard entertainment (music, videos watching) and cameras
- Internet connectivity on your boat via Wi-Fi at marinas or via LTE and a local boat Wi-Fi hotspot is an important feature to have
- For offshore sailors with SSB radio experience consider what you can have with open source solutions on Pi
- DIY solutions for autopilot can be quite cost-effective (You don't need whole BBN OS for it)
- Knowledge costs money. With commercial electronics your knowledge is dialing service support line. With open source you will learn how things work
- Commercial sailing software is just as expensive as the hardware. Ex: https://www.expeditionmarine.com/