

Documentação do Sistema Operacional do Bareboat Necessities.

mgrouch

Version 2022-02-23, Documentação do Sistema Operacional do Bareboat Necessities

<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>

PDF version:

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

Chapter 1. Primeiros passos

1.1. Por que usar o BBN-OS?

Eletrônicos náuticos comerciais, tablets inteligentes e multiplexadores Wi-Fi não substituem um computador de bordo, componente importante capaz de monitorar, armazenar dados e controlar todos os sistemas de uma embarcação. Com o BBN-OS instalado em um computador de bordo baseado em Raspberry Pi (Pi) de baixo consumo de energia, ao contrário de computadores ou notebooks, atende suas necessidades com flexibilidade para adicionar inúmeras opções de sensores e se comunicar com todos os sistemas da embarcação tais como: Internet, Wi-Fi local, câmeras, rede NMEA e um sistema capaz de decodificar protocolos de rádio marítimos. O BBN-OS é gratuito baseado em projetos de código aberto suportados pela comunidade e comumente usados, como SignalK, PyPilot, OpenCPN e outros. A interface gráfica do usuário do BBN-OS permitirá que você crie uma estação no cockpit para todas as funcionalidades do sistema operacional, desde gráficos, painéis, informações meteorológicas, reproduutor de mídia, etc.

1.1.1. Utilizações:

- Chartplotter em monitor Touchscreen no Cockpit
- Nav Station Computer
- Computador desktop
- Laptop baseado em Pi
- Servidor com inicialização automática
- Reprodutor de mídia
- Piloto automático
- Decodificador de rádio SSB
- Servidor de dados náuticos, multiplexador
- Ponto de Acesso Wi-Fi
- Monitoramento de câmeras
- Servidor de análise de dados
- Estação meteorológica
- Painéis de instrumentos da embarcação
- Servidor de arquivos
- Autonomia

1.1.2. Barco Inteligente

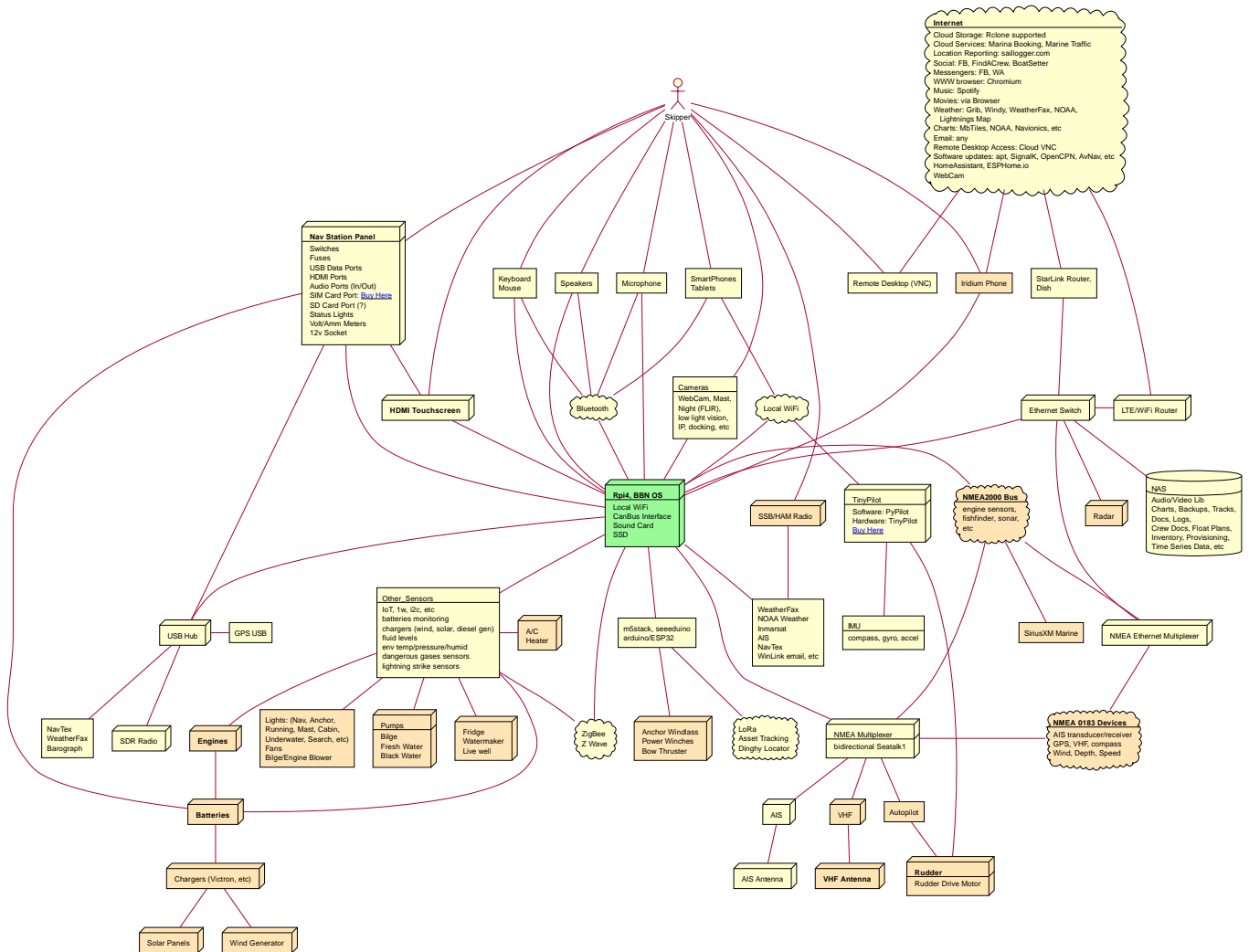


Figure 1. Smart Boat Diagram

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. Baixar imagem

Baixe a imagem do cartão SD (arquivo ~ 3Gb):

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

Endereço alternativo de download se o link anterior não funcionar.

https://github.com/bareboat-necessities/lysmarine_gen/releases/tag/v2021-12-01

Versão anterior do LTS:

<https://archive.org/details/lysmarine-bbn-linux-LTS4>

1.2.1. imagens de 32 bits vs 64 bits

- Prós e contras
- 32 bits consomem menos memória e são mais lentos
- 32 bits é mais compatível no momento (algumas câmeras não eram compatíveis com 64 bits no momento em que este artigo foi escrito)
- QtVlm, wx2img, muitos plug-ins OpenCPN são apenas de 32 bits (no momento em que este artigo foi escrito)
- Você ainda pode executar o kernel de 64 bits na imagem de 32 bits, os programas de espaço do usuário serão de 32 bits. (Adicione `arm_64bit = 1` em `/boot/config.txt`)
- 32 bits está disponível em mais hardwares
- OS de 64 bits e OpenCPN funcionam bem mas tecnicamente ainda são chamados de 'beta'
- A maioria dos problemas menores estão presentes em imagens de 32 e 64 bits

NOTA: A mudança do kernel de 32 bits para o kernel de 64 bits tem um impacto na instalação do java. Consulte (solução alternativa): https://github.com/bareboat-necessities/lysmarine_gen/issues/44

Portanto, no momento em que este artigo foi escrito, o melhor desempenho / compatibilidade é a imagem de 32 bits com o kernel alterado para 64 bits. A imagem preferida de download do BBN-OS é armhf.

- Instale a imagem armhf
- A mudança para kernel de 64 bits em `/boot/config.txt` já foi feito nas versões recentes do sistema operacional BBN
- Continue com todas as outras etapas de configuração.
- Depois de estabelecer a conectividade com a Internet, instale os hot fixes oficiais (também necessários para a configuração do touch screen):

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

- Reinicializar (uma reinicialização após a instalação é necessária para recuperar todas as configurações)

1.3. Prepare o cartão SD

Grave a imagem baixada no cartão SD. Recomenda-se um cartão SD mínimo de 32 Gb. Você pode usar o Raspberry Pi Imager para isso:

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

Se a resolução da tela for inferior a 1024x600, você precisará manualmente configurar no arquivo `/boot/config.txt`, montando a partição / boot do seu cartão SD (no Windows é feito apenas inserindo-o no slot de cartão SD e editando em um texto simples editor).

A qualidade do cartão SD é essencial para o desempenho do seu sistema. BBN-OS vem com ferramenta de benchmarking de disco. Um bom cartão SD mostraria cerca de 45 MB / seg (ou mais) de velocidade de leitura e / ou de saída nele. As unidades SSD mostram resultados muito melhores: 178 MB / seg de leitura e 147 MB / seg de velocidade de E / S de gravação ou superior.

1.4. Primeira inicialização do cartão SD

Insira o cartão SD no slot de cartão SD do Raspberry Pi, conecte um mouse e um teclado e ligue-o. Aguarde até que o processo de inicialização execute a GUI (cerca de 2 minutos).

NOTA: Sua tela touch screen deve estar conectada (HDMI e USB) e ligada durante a primeira inicialização para ser devidamente reconhecida e touchscreen calibrados.

1.5. Configurando a rede

1.5.1. Ethernet com fio

Se você tiver um roteador Ethernet com fio, pode simplesmente conectar sua porta ethernet Raspberry Pi no roteador

1.5.2. Cliente Wi-Fi

Acesse o menu de configurações de rede e exclua o ponto de acesso sem fio Wi-Fi.

NOTA: Você precisa configurar o país do Wi-Fi.

Troque `/etc/wpa_supplicant/wpa_supplicant.conf` to add line for your country (exemplo):

```
country=BR
```

Troque `/etc/default/crda` to set your country (exemplo):

```
REGDOMAIN=BR
```

Quando o cartão Wi-Fi Raspberry Pi descobrir a rede do roteador Wi-Fi, clique para se conectar ao Wi-Fi e insira a senha de Wi-Fi.

Quando o gerenciador de armazenamento de senhas aparecer para inserir a senha deixe em branco para não ser solicitado novamente.

1.5.3. Ponto de acesso Wi-Fi

As conexões Wi-Fi são gerenciadas pelo Gnome NetworkManager amplamente utilizado. Procure a documentação 'nmcli' (interface de linha de comando para NetworkManager). Ou você pode descobrir em Administração / Configurações avançadas de rede menu de aplicativos. Por padrão, a imagem do sistema operacional é configurada para fornecer a você um ponto de acesso.

1.5.4. Usando o iPhone como gateway de Internet

Habilite o ponto de acesso pessoal no iPhone. Conecte-o ao Pi com BBN-OS via USB e diga ao iPhone para confiar autorizando a conexão no computador. Você terá uma conexão de Internet tethered do seu Pi via iPhone (usando usbmuxd).

Você não precisa estar conectado a um USB se o seu telefone e o computador de bordo estiverem na mesma rede Rede Wi-Fi.

1.5.5. Usando o telefone Android como gateway de Internet

Ative o tethering USB nas configurações de rede do seu telefone Android. Ligue-o ao computador do barco através de USB.

A saída do comando abaixo deve mostrar uma nova rota via interface usb0:

```
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. Power usage might be relatively high for small boats.

1.5.7. Typical Setup on a Boat

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

If you use raspberry pi WiFi it is better to disable WiFi 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
```

Follow the prompts.

1.10. Set up GPS

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

```
ls -l /dev/ttyL*SY*
```

1.11. Set up AIS

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

```
ls -l /dev/ttyL*SY*
```

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)



Due to a bug in OpenCPN https://github.com/bareboat-necessities/lysmarine_gen/issues/53 Updating plugins on a system booted with arm64 kernel doesn't work even if userspace is armhf. As a workaround: boot with armhf kernel, update all plugins and only then switch to arm64 kernel.

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

- Inicie o OpenCPN. Vá em Ferramentas / Opções / Gráficos / guia Downloader de gráficos.
- Clique em 'Adicionar Catálogo'. Para os EUA: clique em USA NOAA & Inland Charts / ENC / By Region.
- Escolha sua região, clique (ou toque) em 'OK'
- Clique em 'Atualizar' (para atualizar o catálogo)
- Clique na guia 'Baixar gráficos ...'
- Clique com o botão direito (ou toque longo) na lista de gráficos
- Clique em 'Selecionar tudo' no menu pop-up
- Pressione o botão 'Baixar gráficos selecionados' e aguarde a conclusão
- Pressione o botão 'Aplicar'
- Clique na guia 'Arquivos de gráfico'
- Pressione o botão 'Prepare all ENC Charts'
- Pressione o botão 'OK' quando terminar.

O-Charts podem ser registrados usando plug-ins OpenCPN pré-instalados (no espaço do usuário arm32, certifique-se de atualizar os plug-ins on-line do catálogo OpenCPN) com chave ou dongle USB O-Charts.

O-Charts

O-Charts são gráficos comerciais compatíveis com OpenCPN. Veja: <https://www.o-charts.org/>

Fazendo seus próprios gráficos MbTiles

Fazendo seus próprios gráficos MbTiles com SASPlanet: http://svocelot.com/Cruise_Info/Equipment/mbTiles.htm

1.13.2. AvNav

Quando você está online, os blocos de MB raster NOAA devem funcionar imediatamente. O-Charts podem ser registrados usando o plugin AvNav com chave ou dongle USB O-Charts.

1.13.3. SignalK, FreeBoard-SK, TukTuk

Siga a documentação do SignalK para instalar gráficos offline para eles.

1.14. Configure seus parâmetros da embarcação

Configure os parâmetros do seu navio nas aplicações SignalK e Vessel. Eles podem ser úteis.

No plugin OpenCPN Polar, escolha um arquivo polar para a sua embarcação em ~/Polars, /usr/share/opencpn/plugins/weather_routing_pi/data/polars/ ou construa a partir de suas gravações de dados de viagens anteriores. Será necessário para a rota do tempo, etc.

Os polares são importantes para os plug-ins Weather Routing e Dashboard Tactics.

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://signalk-dev.slack.com/>

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.

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 login (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.

Off-shore 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.

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. 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.

1.18.4. 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.

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 WiFi).

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 WiFi. 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.

1.21.2. NMEA 2000

Check SignalK plugin settings and SignalK documentation.

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.
- 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.
- 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

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

- 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.

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.



You need to change the authentication method or the default unix user password before connecting the system to the Internet.

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 AppStore 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 WiFi. 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).

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

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

1.26. 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.

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 pre-loaded 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.

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. 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

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 file in /var/www/bbn-launcher.

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.



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. Wi-Fi DNS

If you use your devices on LAN or WiFi 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.

1.36. 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, KPlax

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.

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.

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. Media Library

You can install JellyFin via add-ons:

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

1.44. HomeAssistant

You can install HomeAssistant (for many BBN users their boat is their home) via add-ons:

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

1.45. 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 require 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.

1.46. 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 (Model: SL07W, Brand Sihovision, Capacitive Touch Screen 7-inch, (1000 nits), IP65, 1024x600, Cost under \$300): <https://www.sihovision.com/industrial-touch-monitor/7-inch-industrial-wide-temperaturer-lcd-monitor-with-remote-control->

- 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

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

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

1.47. 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 to choose a printer.

1.48. Social Networks and Messaging

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

1.49. Location Reporting

SignalK comes with preinstalled plugin for saillogger.com

1.50. 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 WiFi hotspot and LTE gateway

1.51. IoT

1.51.1. 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.

1.51.2. 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.

1.52. Data Analytics

1.52.1. Grafana

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

1. Edit the `/etc/grafana/grafana.ini` file and change line `;'http_port = 3000'` to `'http_port = 3080'` (to avoid conflict with SignalK)

```
sudo nano /etc/grafana/grafana.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'.

1.52.2. 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 Kapasitor also come pre-installed.

1.53. 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.

1.54. Upgrading your System

1.54.1. Verified Updates

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

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

1.54.2. 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 SignalK App Store.

OpenCPN charts can be updated using OpenCPN chart downloader plugin.



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

1.55. Data Flow

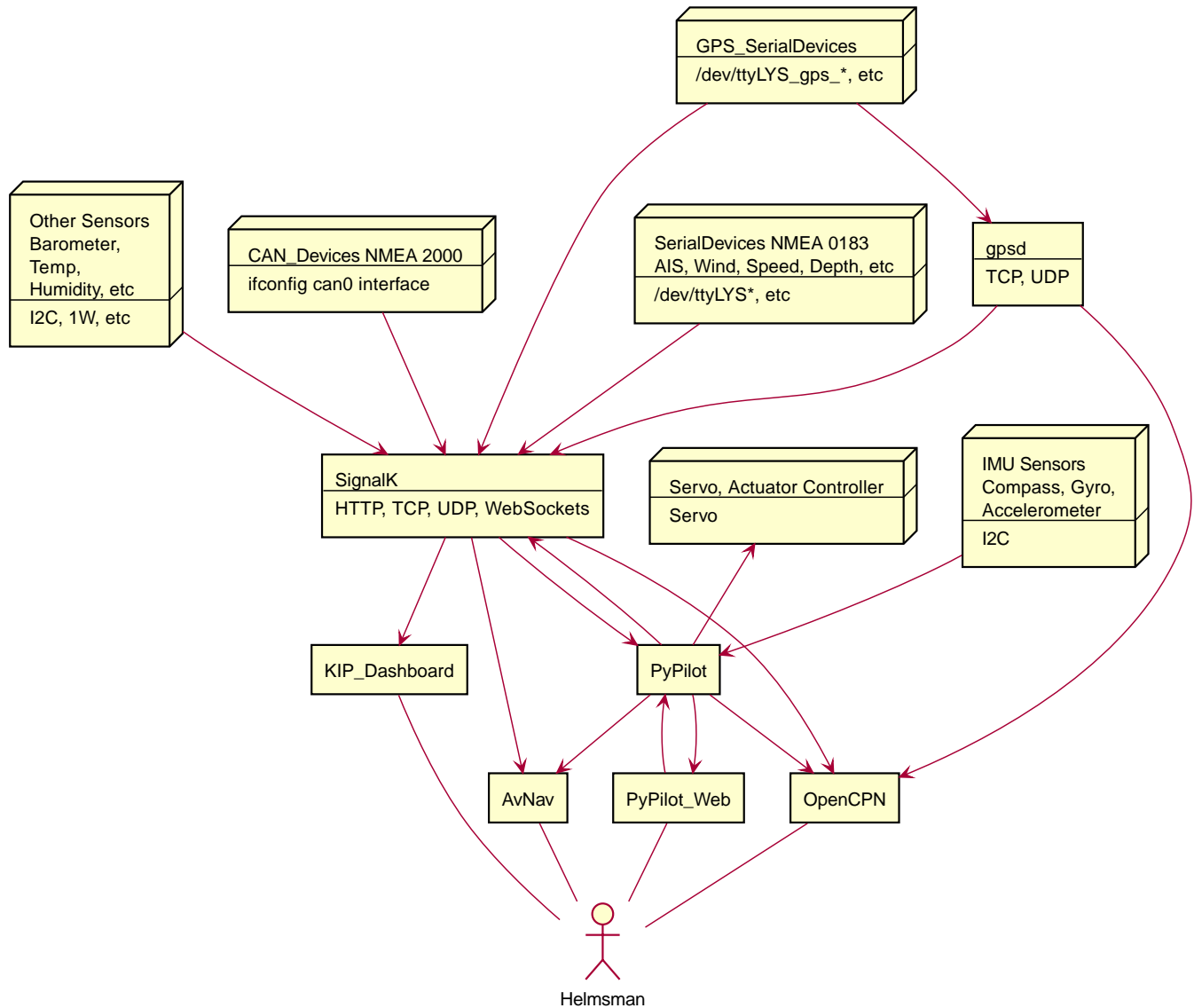


Figure 2. Data Flow

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, 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>

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 multi-touch 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.

2.2. 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 wifi 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.

Another option is the board and a case from McuZone. The advantage 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/>

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



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.

2.4. 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 mpu9255.

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

2.5. Environmental Sensors

2.5.1. 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.

2.6. 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>

2.7. 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.

2.8. 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
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

2.9. 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.

2.10. SD Cards

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

2.11. 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 bi-directional 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 bi-directional

multiplexer to be able to control autopilot from your pi via NMEA. There is a dedicated Autohelm autopilot plugin in SignalK.

- Buy a dedicated Seataalk1 to NMEA converter.
- For read/only Seataalk1 connection you can use this solution: [https://github.com/SignalK/signalk-server/blob/master/Seataalk\(GPIO\).md](https://github.com/SignalK/signalk-server/blob/master/Seataalk(GPIO).md)

SeaTalk1 had been reverse engineered and a converter into NMEA is available at: <https://github.com/MatsA/seataalk1-to-NMEA0183>

<https://github.com/Thomas-GeDaD/Seataalk1-Raspi-reader>

2.12. 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)



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 WiFi 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/>

2.13. 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

TCP/IP gateways:

- Bi-directional NMEA 2000 Multiplexer WiFi (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

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.

2.14. 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.

2.15. 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 of 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.

2.16. 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).

2.17. 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).

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.

2.18. 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>

2.19. 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

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.

2.20. 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.

2.21. Cameras

2.21.1. IP Cameras

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

2.21.2. 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-ffc-cable-raspberry-pi-camera-specific-pack-2-1-set/>

2.21.3. 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.

2.21.4. USB connected cameras

Supported with USB 3.0

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

2.22. 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>

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 chartplotter).

2.23. 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)
- Dangerous gases
- Lightning strikes
- Voltage, amps
- Relays for control (lights, devices)
- Power utilization (batteries monitoring)
- Motion sensors
- Human heartbeat detection
- Anchor windlass chain counters
- Bilge pump
- Proximity
- Fingerprint recognition
- Light
- Color detection
- 433MHz RF receivers
- LoRa receivers/transmitters

- Interfaces (USB, Ethernet, CAN Bus, Rs-485, RS-232, 4G/LTE, GPS/GNSS, ModBus, etc)
- IMU (gyroscope, magnetometer, accelerometer)
- Hall effect
- Actuators, servo, motor controls
- Audio (speaker, microphone), buzzers
- Barcode readers
- External memory readers (TF card, etc)
- Laser emitters / receivers
- NCIR/IR receivers/transmitters
- Thermal cameras
- Joystick
- DAC/ADC
- Potentiometers
- Rudder position
- Blood alcohol concentration
- Blood oxygen, oximeter
- and more

2.24. 1-Wire

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

2.25. IoT

Sonoff Switches - Supported by SignalK plugin.

Ecowitt weather sensors - Supported via SignalK plugin.

Selly <https://shelly.cloud/> - Supported via SignalK plugin.

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

2.26. Power, 12v → 5v Converters

2.26.1. 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/B008ZNW0YY>

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

2.26.2. 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

2.27. Open Boat Projects

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

2.28. 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.

2.28.1. 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.

2.28.2. 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.

2.28.3. 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 of 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)

2.28.4. 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

2.28.5. 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.

2.28.6. 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.

2.28.7. 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>

2.28.8. 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.

2.28.9. 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)
- Create pypilot connection to signalK (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
- pypilot also needs to have connection via SignalK web socket to port 3000 of signalK
- That connection needs to be authorized made READ/WRITE in signalK using their token exchange procedure
- GPS is read by signalK by creating NMEA connection to /dev/serial0 port (set correct baud rate)
- 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.
- 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, WiFi Antenna (3 in one), Model AFCJF3, Price \$80 Designed for marine and recreational vehicles (RV), this multi-band Cellular, GPS, WiFi Antenna (696 MHz - 5900 MHz / 5.9 GHz) is a 3-way omni-directional, 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.

ublox GPS receivers have extensive configuration. Check: <https://gpsd.io/ubxtool-examples.html>

4.2. RTL8812AU drivers

Following these steps it compiled fine from the first attempt <https://github.com/aircrack-ng/rtl8812au>

There is also a script to install various additional WiFi 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. 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 WiFi networks with a MAC address of your phone.

4.5. 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 [Usando o iPhone como gateway de Internet](#) (or [Usando o telefone Android como gateway de Internet](#)) 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.

4.6. 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).

4.7. 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.

4.8. Running with Higher Resolution Monitors

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

- Go into Administration/Settings/Universal Access
- Turn on 'Large Text', set 'Cursor Size' to 'Medium'
- 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

4.9. 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>

4.10. 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
```