* O N B O A R D * C O M P U T E R * S Y S T E M *

Raspberry Pi navigation system / weather station / ship computer

github.com/LASER-WOLF/OnboardComputerSystem

[GO BACK]

* THIS PAGE WAS LAST UPDATED 27. JANUARY 2019 *

(This is the new version of the Onboard Computer System, for the old version follow this link)
I will continue to update this page as the project is progressing, some planned future features:
-- A graphical interface for viewing location history/weather history and other data --

* CURRENT FEATURES *

-- 15.6" touchscreen --
-- Chartplotter --
-- GRIB weather data visualization --
-- NMEA multiplexer --
-- Weather station (barometric pressure / temperature) --
-- Sunrise & sunset times for current position --
-- Automatic position logging --
-- Automatic weather logging --
-- Automatic data synchronization with a server --
-- Automatic time synchronization with GPS --
-- PDF reader --

* PARTS LIST *

(this is just a list of the parts I used, most of these can be replaced with alternatives and a lot of them are optional)
-- Computer: Raspberry Pi 3 Model B+ --
-- Case: Raspberry Pi 3 B+ Case black --
-- SD: Sandisk microSDXC Ultra 128GB --
-- Screen: Chalkboard Electronics 15.6" Touchscreen --
-- GPS: GlobalSat MR-350 S4 --
-- GPS extension cable: 10m PS/2 Extension Cable --
-- GPS USB adapter: GlobalSat 11-BR305-USB2 --
-- AIS: dAISy AIS Receiver --
-- Temperature sensors: (2x) DS18B20 --
-- Barometric pressure sensor: BMP180 --
-- GPIO cables: 40pin GPIO Female to Female Cable --
-- USB panel mount: (2x) Square Single Port USB 2.0 Panel Flush Mount Extension Cable --
-- 12v to 5v converter: LM2596 Buck Step-down Power Converter Module --
-- Waterproof junction box: Junction Box (190x145x70 mm) --
-- 4G Wi-Fi Router: Huawei B593s --

* USAGE *

To recieve NMEA data on the local Wi-Fi from the Pi use these settings:

```
Protocol: TCP
IP: 192.168.1.99
Port: 10110
```

Location & weather history:
A new entry is added to the location database once every hour if you have moved minimum 50 meters since last entry
A new entry is added to the weather database once every hour

User interface:
Two-finger click on the touchscreen acts as a right-click

Right-click on the desktop for a comprehensive menu

The main (a more simplified) menu is at the top of the screen, with the following buttons:
OCPN (CTRL+ALT+O) - OpenCPN chartplotter
GRIB (CTRL+ALT+G) - zyGrib weather data visualization
LOG (CTRL+ALT+L) - View location history
WTHR (CTRL+ALT+W) - View weather history
HDD (CTRL+ALT+H) - PCManFM file manager
ALRM (CTRL+ALT+A) - Alarm system settings
CONF (CTRL+ALT+C) - System settings
OFF - Left click to power off, right-click to reboot

Under the main menu is the taskbar:
Single click item to open/minimize
Right-click item to close

At the bottom of the screen is the GPS info:
The first (and last) item is GPS fix [FX] means there's a 3D fix, [NO] means there's no 3D fix
TIME - Date and time
SPD - Speed (in knots)
COG - Course Above Ground (Shows course based on GPS data (only works when moving))
LOG - Total distance travelled

If there's no active windows open:
UPTIME - System uptime
SUNRISE - The time of sunrise for current GPS position
SUNSET - The time of sunset for current GPS position
BARO - Barometric pressure (in millibar)
INSIDE - Temperature inside (in celcius)
OUTSIDE - Temperature outside (in celcius)
TIME OFFSET - System time offset from UTC
NAUTICAL TIMEZONE - Current nautical timzone (from GPS coordinates)
LON - Longitude
LON - Longitude
AVG SPD - Total average speed (in knots)

Some useful commands when logging in remotely:

System settings:

```
$ ocs-config-curses
```

Alarm system settings:

```
$ ocs-alarm-curses
```

View data:

```
$ ocs-info-curses
```

View location history:

```
$ ocs-location-history-curses
```

View weather history:

```
$ ocs-weather-history-curses
```

Adjust volume:

```
$ alsamixer
```

Start cmus music player:

```
$ cmus
```

Start ranger file manager:

```
$ ranger
```

Configuration:

General configuration:

```
$ ocs-config-curses
```

Edit desktop right-click menu:

```
$ nano ~/.config/openbox/menu.xml
```

Edit main menu:

```
$ nano ~/.OnboardComputerSystem/OnboardComputerSystem
```

Change kplex settings:

```
$ nano .kplex.conf
```

PHP examples:

Showing realtime data:
https://github.com/LASER-WOLF/OnboardComputerSystem/blob/master/example_files/PHP/ocs.php

Showing location history:
(Requires a Google Maps API key)
https://github.com/LASER-WOLF/OnboardComputerSystem/blob/master/example_files/PHP/location.php

Showing weather history:
(Requires Charts.js from http://www.chartjs.org/)
https://github.com/LASER-WOLF/OnboardComputerSystem/blob/master/example_files/PHP/weather.php

* SETTING UP THE HARDWARE *

(IMPORTANT: Adjust the LM2596 voltage to 5 volt BEFORE connecting anything to the output)
12 volt wiring diagram
Raspberry Pi USB diagram
BMP180 wiring diagram
DS18B20 wiring diagram

* INSTALLATION *

Write the Raspbian image to the SD-card:
Download the "Raspbian Lite" image from https://www.raspberrypi.org/downloads/raspbian/
Write the image to your SD-card (instructions on https://www.raspberrypi.org/documentation/installation/installing-images/README.md)

Starting the Raspberry Pi for the first time:
Insert the SD-card into the Pi
Connect ethernet cable
Power on the Pi
Login with the default user "pi" and the password "raspberry"

Alright let's start off by updating everything:

(rpi-update is a Raspberry Pi firmware updater tool)

```
$ sudo apt-get update
$ sudo apt-get upgrade
$ sudo apt-get install rpi-update
$ sudo rpi-update
$ sudo reboot
```

Setup Wi-Fi (with static IP):

Screenshot

```
$ sudo apt-get install wicd-curses
```

```
$ wicd-curses
```

Select your wireless network

Press RIGHT ARROW to enter setup

Check the box that says "Use static IP"

Enter the following in the IP field:

```
192.168.1.99
```

Fill in rest of the fields, for me it's:

```
Netmask: 255.255.255.0
Gateway: 192.168.1.1
DNS domain: 192.168.1.1
DNS server 1: 192.168.1.1
```

Check the box "Automatically connect to this network"

Input your password where it says "Key:"

Press F10 to save the settings

Press Q to quit

```
$ sudo poweroff
```

Disconnect the ethernet cable

Power on the device

Now let's SSH into the Pi by using the new static IP:

```
$ ssh pi@192.168.1.99
```

Change hostname:

```
$ sudo nano /etc/hosts
```

Change the text in the last line from "raspberrypi" to "OnboardComputerSystem"

Screenshot

Press CTRL+O, then press ENTER to save

Press CTRL+X to exit

```
$ sudo nano /etc/hostname
```

Change "raspberrypi" to "OnboardComputerSystem"

Screenshot

Save the file and exit

```
$ sudo reboot
```

At the next login it should say "pi@OnboardComputerSystem"

Change username:

```
$ sudo adduser operator
```

Enter a password

You can fill in name and stuff if you want or just press enter

```
$ sudo visudo
```

Add your new user to the bottom of the file, just copy the line for the "pi" user and replace "pi" with your username, like this:

```
operator ALL=(ALL) NOPASSWD: ALL
```

Screenshot

Save the file and exit

```
$ sudo reboot
```

Add the new user to the default groups:

```
$ sudo usermod -a -G
adm,dialout,cdrom,sudo,audio,video,plugdev,games,users,input,netdev,spi,i2c,gpio,pi
operator
```

```
$ sudo visudo
```

Remove the line for the pi user

Screenshot

Save the file and exit

Delete the pi user:

```
$ sudo deluser -remove-home pi
```

Enable SSH:

```
$ sudo systemctl enable ssh
```

```
$ sudo systemctl start ssh
```

Set timezone to UTC:

```
$ sudo timedatectl set-timezone UTC
```

Configure raspi-config:

Check free harddisk space:

```
$ df -h
```

There's probably not a lot left, we need to expand the filesystem

```
$ sudo raspi-config
```

Select "Expand filesystem" press enter

Select finish and reboot

Check free space again:

```
$ df -h
```

Should be more this time around

Clone the OnboardComputerSystem git repository:

```
$ sudo apt-get install build-essential git
```

```
$ git clone https://github.com/LASER-WOLF/OnboardComputerSystem
```

```
$ cp -r OnboardComputerSystem/ocs_files/.OnboardComputerSystem/
.OnboardComputerSystem/
```

Install urxvt (terminal emulator):

```
$ cp OnboardComputerSystem/config_files/.bash_aliases
.bash_aliases
```

```
$ cp OnboardComputerSystem/config_files/.Xresources .Xresources
```

```
$ sudo apt-get install rxvt-unicode
```

Install Xorg (display server):

```
$ sudo apt-get install xorg
```

Install Openbox (window manager):

Copy the config files:

```
$ mkdir .config
```

```
$ cp -r OnboardComputerSystem/config_files/.config/openbox/ .config/openbox/
```

```
$ cp OnboardComputerSystem/config_files/.config/user-dirs.dirs .config/user-dirs.dirs
```

```
$ sudo apt-get install openbox
```

Install LightDM (display manager):

```
$ sudo apt-get install lightdm
```

Enable autologin:

```
$ sudo nano /etc/lightdm/lightdm.conf
```

Uncomment the following line:

```
autologin-user=
```

Change it to:

```
autologin-user=operator
```

Screenshot

Save the file and exit

Enable the LightDM service:

```
$ sudo systemctl enable lightdm.service
```

Reboot:

```
$ sudo reboot
```

After the reboot you should automatically be logged into openbox

Edit config.txt:

(See "~/OnboardComputerSystem/example_files/config.txt" for an example)

```
$ sudo nano /boot/config.txt
```

Add the following lines:

```
disable_overscan=1
hdmi_force_hotplug=1
hdmi_group=2
hdmi_mode=81
gpu_mem=128
disable_splash=1
dtoverlay=w1-gpio
dtparam=i2c_arm=on
max_usb_current=1
```

Screenshot

Save the file and exit

Reboot:

```
$ sudo reboot
```

Remove Raspberry Pi logos at boot:

(See "~/OnboardComputerSystem/example_files/cmdline.txt" for an example)

```
$ sudo nano /boot/cmdline.txt
```

Add the following at the end of the line:

```
logo.nologo
```

Screenshot

Save the file and exit

Do a reboot and check that the logos are not displayed:

```
$ sudo reboot
```

Enable two finger click on the touchpad as right-click:

(See "~/OnboardComputerSystem/xorg.conf" for an example)

Edit xorg.conf:

```
$ sudo nano /etc/X11/xorg.conf
```

Add the following lines:

```
Section "InputClass"
Identifier "calibration"
Driver "evdev"
MatchProduct "Chalkboard Electronics HID Touchscreen"
Option "EmulateThirdButton" "1"
Option "EmulateThirdButtonTimeout" "250"
Option "EmulateThirdButtonMoveThreshold" "30"
EndSection
```

Screenshot

Save the file and exit

Reboot:

```
$ sudo reboot
```

Test that right-clicking works by long clicking on the desktop with the touchpad

Install GPSd

```
$ sudo apt-get install gpsd gpsd-clients
```

(See "~/OnboardComputerSystem/example_files/gpsd" for an example)

Edit /etc/default/gpsd:

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

Modify/add the lines:

```
START_DAEMON="true"
USBAUTO="true"
DEVICES="/dev/ttyUSB0"
GPSD_OPTIONS="-n"
GPSD_SOCKET="/var/run/gpsd.sock"
```

Screenshot

Save the file and exit

Reboot:

```
$ sudo reboot
```

Test GPSd by running:

```
$ cgps
```

You should see data from the GPS

setup I2C:

```
$ sudo apt-get install i2c-tools
```

```
$ sudo nano /etc/modules
```

Add the following lines:

```
i2c-bcm2708
i2c-dev
```

Screenshot

Save the file and exit

Reboot:

```
$ sudo reboot
```

Test I2C by running:

```
$ sudo i2cdetect -y 1
```

You should see some numbers and stuff like that

Install Python:

```
$ sudo apt-get install python3 python3-dev python3-pip python3-gst-1.0
$ pip3 install gps
$ pip3 install ephem
$ pip3 install geopy
$ pip3 install playsound
```

Install SQLite3:

```
$ sudo apt-get install sqlite3
```

Install ncurses:

```
$ sudo apt-get install libncurses5-dev libncursesw5-dev
```

Setup the BMP180 barometric pressure sensor:

```
$ git clone https://github.com/adafruit/Adafruit_Python_BMP.git
$ cd Adafruit_Python_BMP
$ sudo python3 setup.py install
$ cd
$ sudo rm -rf ~/Adafruit_Python_BMP/
```

Setup the DS18B20 temperature sensors:

```
$ sudo modprobe w1-gpio
$ sudo modprobe w1-therm
$ cd /sys/bus/w1/devices
$ ls
```

You should get three results starting with "28-..."

Screenshot

Take note of the names

```
$ cd
$ nano .OnboardComputerSystem/OnboardComputerSystem.py
```

Find the lines:

```
temp1name='28-0115905a2fff'
temp2name='28-0415901b95ff'
```

And replace the names with the ones you found

Screenshot

Save the file and exit

Testing the buzzer:

```
$ .OnboardComputerSystem/buzzer-alarm.py
```

Install tint2 (taskbar):

```
$ cp -r OnboardComputerSystem/config_files/.config/tint2/ .config/tint2/
$ sudo apt-get install tint2
```

Install lemonbar (statusbar):

(lemonbar doesn't have xft font support so we'll be using a fork with xft support)

```
$ sudo apt-get install libxcb-xinerama0-dev libxcb-randr0-dev libxft-dev libx11-xcb-dev
$ git clone https://github.com/krypt-n/bar
$ cd bar
$ make
$ sudo make install
$ cd
$ sudo rm -rf ~/bar/
```

Install Conky (system monitor):

```
$ sudo apt-get install conky
$ cp OnboardComputerSystem/config_files/.conkyrc .conkyrc
$ touch .datatext
```

Install font, theme and icons:

```
$ cp -r OnboardComputerSystem/theme_files/.fonts/ .fonts/
$ cp -r OnboardComputerSystem/theme_files/.themes/ .themes/
$ cp -r OnboardComputerSystem/theme_files/.icons/ .icons/
$ cd .icons
$ tar xvzf Faenza.tar.gz
$ rm Faenza.tar.gz
$ cd
$ cp OnboardComputerSystem/config_files/.gtkrc-2.0 .gtkrc-2.0
$ cp -r OnboardComputerSystem/config_files/.config/gtk-3.0/ .config/gtk-3.0/
```

Reboot:

```
$ sudo reboot
```

Setup NTP to syncronize with GPSd:

```
$ sudo apt-get install ntp
$ sudo nano /etc/init.d/ntp
```

Find the line:

```
RUNASUSER=ntp
```

Change it to:

```
RUNASUSER=root
```

Screenshot

Save and exit

Edit ntp.conf:(See "~/OnboardComputerSystem/example_files/ntp.conf" for an example)

```
$ sudo nano /etc/ntp.conf
```

Find the lines:

```
server 0.debian.pool.ntp.org iburst
server 1.debian.pool.ntp.org iburst
server 2.debian.pool.ntp.org iburst
server 3.debian.pool.ntp.org iburst
```

Add the following lines under the server lines:

```
server 127.127.28.0
fudge 127.127.28.0 refid GPS
server 127.127.28.1 prefer
fudge 127.127.28.1 refid PPS
```

Screenshot

Save the file and exit

```
$ sudo poweroff
```

Test that the synchronization works by powering off your wifi router and the Raspberry Pi, leave them both off for a couple of minutes, then power on only the Raspberry Pi, check the clock, it should run a couple of minutes late, but after some ti
it will syncronize with the GPS and be correct, if it works correctly power on the wifi router again

Setup audio:

Set audio output to 3.5mm jack:

```
$ amixer cset numid=3 1
```

Change volume:

```
$ alsamixer
```

Screenshot

Use UP/DOWN ARROWS to change volume, press ESC when done

Copy apps:

```
$ cp -r OnboardComputerSystem/ocs_files/Apps/ Apps/
```

The "Apps" folder contains two apps:
"JTides" for viewing tidal data
"PiSNES" a SNES emulator

Copy scripts:

```
$ cp -r OnboardComputerSystem/ocs_files/Scripts/ Scripts/
```

Setup documents:

```
$ cp -r OnboardComputerSystem/ocs_files/Documents/ Documents/
```

Copy all your documents to the ~/Documents folder
Check out https://www.reddit.com/r/sailing/comments/3hwagp/a_collection_of_free_useful_sailing_maritime/ for a collection of documents
Download link for my document collection: http://perpetual.voyage/ocs/OnboardComputerSystemDocuments.zip

Install Kplex (NMEA multiplexer):

```
$ cp OnboardComputerSystem/config_files/.kplex.conf .kplex.conf
$ git clone https://github.com/stripydog/kplex
$ cd kplex
$ make
$ sudo make install
$ cd
$ rm -rf kplex/
```

Reboot:

```
$ sudo reboot
```

Test Kplex by running OpenCPN on another machine (not the Pi), go to connections, and add a new connection with:

```
Protocol: TCP
Address: 192.168.1.99
Port: 10110
```

You should get GPS data from the Pi

Screenshot

Install OpenCPN:

```
$ mkdir Charts
```

Copy all your charts to the ~/Charts directory

```
$ cp -r OnboardComputerSystem/config_files/.opencpn/ .opencpn/
$ sudo apt-get install build-essential cmake gettext git-core gpsd gpsd-clients libgps-dev wx-common libwxgtk3.0-dev libglu1-mesa-dev libgtk2.0-dev wx3.0-headers libbz2-dev libtinyxml-dev libportaudio2 portaudio19-dev libcurl4-openssl-dev libexpat1-dev libcairo2-dev libsqlite3-dev
$ git clone git://github.com/OpenCPN/OpenCPN.git
$ cd OpenCPN
$ mkdir build
$ cd build
$ cmake ../
$ make
```

(This is gonna take a while)

```
$ sudo make install
$ cd
$ rm -rf ~/OpenCPN/
```

Press the OCPN button on the top menu to start OpenCPN
Go to settings and add your Charts folder

Install Watchdog (OpenCPN plugin):

```
$ git clone git://github.com/seandepagnier/watchdog_pi.git
$ cd watchdog_pi
$ mkdir build
$ cd build
$ cmake ../
$ make
$ sudo make install
$ cd
$ rm -rf ~/watchdog_pi/
```

Install Climatology (OpenCPN plugin):

```
$ git clone git://github.com/seandepagnier/climatology_pi.git
$ cd climatology_pi
$ mkdir build
$ cd build
$ cmake ../
$ make
```

```
$ sudo make install
$ cd
$ rm -rf ~/climatology_pi/
$ sudo tar -C /usr/local/share/opencpn/plugins/climatology_pi
-xvf OnboardComputerSystem/other_files/CL-DATA-1.0.tar.xz
```

Install AIS Radar View (OpenCPN plugin):

```
$ git clone git://github.com/Verezano/radar_pi.git
$ cd radar_pi
$ mkdir build
$ cd build
$ cmake ../
$ make
$ sudo make install
$ cd
$ rm -rf ~/radar_pi/
```

Install zyGrib (weather data visualization)

```
$ mkdir GRIB
$ sudo apt-get install build-essential g++ make libqt4-dev
libbz2-dev zlib1g-dev libproj-dev libnova-dev nettle-dev
$ tar xvzf OnboardComputerSystem/other_files/zyGrib-7.0.0.tgz
$ cd zyGrib-7.0.0
$ make
$ sudo make install
$ cd
$ rm -rf ~/zyGrib-7.0.0/
$ mkdir -p ~/.zygrib/config/
$ cp OnboardComputerSystem/config_files/zygrib.ini
.zygrib/config/zygrib.ini
```

Press the GRIB button on the top menu to test zyGrib

Install FoxtrotGPS (non-maritime maps):

```
$ mkdir -p ~/.gconf/apps/foxtrotgps
$ sudo apt-get install foxtrotgps
$ cp OnboardComputerSystem/config_files/.gconf/apps/foxtrotgps
/%gconf.xml .gconf/apps/foxtrotgps/%gconf.xml
```

Install unclutter (hides the mouse when idle):

```
$ sudo apt-get install unclutter
```

Install scrot (take screenshots):

```
$ sudo apt-get install scrot
```

Install ranger (console based file manager):

```
$ sudo apt-get install ranger
```

Install PCManFM (file manager)

```
$ mkdir -p ~/.config/pcmanfm/default/
$ cp OnboardComputerSystem/config_files/.config/pcmanfm/default
/pcmanfm.conf .config/pcmanfm/default/pcmanfm.conf
$ cp -r OnboardComputerSystem/config_files/.config/libfm/
.config/libfm/
$ sudo apt-get install pcmanfm
```

Install slurm (network load monitor):

```
$ sudo apt-get install slurm
```

Install htop (process monitor):

```
$ sudo apt-get install htop
```

Install Chromium (web browser):

```
$ sudo apt-get install chromium-browser
```

Install x11vnc (vnc server):

```
$ sudo apt-get install x11vnc
```

use the vnc script to start the server

Install feh (image viewer):

```
$ sudo apt-get install feh
```

Install Okular (pdf reader):

```
$ mkdir -p ~/.kde/share/config
$ cp OnboardComputerSystem/config_files/.kde/share/config
/okularrc .kde/share/config/okularrc
$ cp OnboardComputerSystem/config_files/.kde/share/config
/okularpartrc .kde/share/config/okularpartrc
$ sudo apt-get install okular
```

Install cmus (music player):

```
$ mkdir Music
```

Copy all your music to the "~/Music" folder

```
$ cp -r OnboardComputerSystem/config_files/.cmus/ .cmus/
$ cp -r OnboardComputerSystem/ocs_files/Radio/ Radio/
$ sudo apt-get install cmus
```

(optional) Install cmusfm (last.fm scrobbler):

```
$ sudo apt-get install autoconf libssl-dev
$ git clone https://github.com/Arkq/cmusfm
$ cd cmusfm
$ autoreconf --install
$ mkdir build && cd build
$ ../configure
$ make
$ sudo make install
$ cd
$ rm -rf ~/cmusfm/
$ cmusfm init
$ cmus
```

Press 5 navigate to the files/folders you want to add and press A to add them to the library
I'm gonna add all the radio files so I'll navigate into the ~/Radio folder and press A on all the files
Then press 1 to go back to library view

Install bluetooth:

```
$ sudo apt-get install bluetooth bluez blueman pulseaudio-module-bluetooth
```

Setup JTides:

```
$ sudo apt-get install oracle-java8-jdk
```

Setup PiSNES (SNES emulator):

```
$ sudo apt-get install libsdl1.2debian
```

Copy roms to the ~/Apps/pisnes/roms/ folder
Edit ~/Apps/pisnes/snes9x.cfg to setup button mappings

Install ScummVM (emulator):

```
$ sudo apt-get install scummvm
$ mkdir ScummVM
```

Copy ScummVM games to the ~/ScummVM folder

Install Nethack (video game):

```
$ sudo apt-get install nethack-console
$ cp OnboardComputerSystem/config_files/.nethackrc .nethackrc
```

Install Cataclysm DDA (video game):

```
$ sudo apt-get install libglib2.0-dev ccache clang liblua5.2-0
$ git clone https://github.com/CleverRaven/Cataclysm-DDA.git
$ cd Cataclysm-DDA
$ make clean
$ make CLANG=1 CCACHE=1 RELEASE=1 ASTYLE=0
$ cd
$ mv Cataclysm-DDA/ Apps/Cataclysm-DDA/
```

(optional) Setup synchronization of the OCS databases with a server:

```
$ nano Scripts/update-server.sh
```

Change the script to match your server settings
Screenshot
Test the script:

```
$ Scripts/update-server.sh
```

Generate SSH keys (for passwordless login):
(This is how I do it on my host, might be different on another host)

```
$ ssh-keygen -t rsa
```

Press ENTER, ENTER and ENTER
Then add the SSH key to the server:

```
$ ssh-copy-id -i ~/.ssh/id_rsa.pub username@website.com
```

Run the update-server.sh script again (you shouldn't need a password this time):

```
$ Scripts/update-server.sh
```

Setup update-server.sh to run automatically once every hour:(See "~/OnboardComputerSystem/example_files/crontab" for an example)

```
$ crontab -e
```

Add the line:

```
0 * * * * /home/operator/Scripts/update-server.sh
```

Screenshot
Save the file and exit

Remove the OnboardComputerSystem folder (we have copied everything we need from it):

```
$ rm -rf ~/OnboardComputerSystem/
```

Reboot:

```
$ sudo reboot
```

Done and done!
Have a beer or something

* THANKS TO *

---

http://octant.no

http://www.catb.org/gpsd/gpsd-time-service-howto.html

http://www.danmandle.com/blog/getting-gpsd-to-work-with-python/

https://learn.adafruit.com/adafruits-raspberry-pi-lesson-4-gpio-setup/configuring-i2c

https://learn.adafruit.com/using-the-bmp085-with-raspberry-pi

https://learn.adafruit.com/adafruits-raspberry-pi-lesson-11-ds18b20-temperature-sensing/

http://opencpn.org/ocpn/compiling_source_linux

http://www.stripydog.com/kplex/examples/nmeaserver.html

https://github.com/krypt-n/bar

https://sourceforge.net/projects/pisnes/

http://code.google.com/p/tintwizard/

http://arachnoid.com/JTides/

http://www.proggyfonts.net/

http://box-look.org/content/show.php/Bygone?content=83149

http://debian-art.org/content/show.php/BSM+Simple?content=121685

http://tiheum.deviantart.com/art/Faenza-Icons-173323228