Installation

1. BIOS/UEFI -> On power loss: Power on
2. Install Ubunu 16.04 LTS ([Link](https://www.ubuntu.com/download/desktop))
   1. PC name: Dynamo-PC
   2. User: dynamo
   3. Password: dynamo
3. Update Ubuntu and other software using Software Updater

sudo apt-get update

sudo apt-get upgrade

* 1. (Optional) Uninstall unneeded software using Ubuntu Software or apt-get uninstall
     1. Amazon, Activity Log Manager, AisleRiot Solitaire, ImageMagick, Mahjong, Mines, Thunderbird, Sudoku, Shotwell…
  2. (Optional) If running 7th generation Intel processor or newer, the kernel might need to be updated (e.g. if Wi-Fi doesn’t work)

sudo apt-get install linux-headers-4.10.2-041002

sudo apt-get install linux-headers-4.10.2-041002-generic

sudo apt-get install linux-image-4.10.2-041002-generic

1. Install ROS Kinetic Kame ([Instructions](http://wiki.ros.org/kinetic/Installation/Ubuntu))

sudo sh -c 'echo "deb http://packages.ros.org/ros/ubuntu $(lsb\_release -sc) main" > /etc/apt/sources.list.d/ros-latest.list'

sudo apt-key adv --keyserver hkp://ha.pool.sks-keyservers.net:80 --recv-key 421C365BD9FF1F717815A3895523BAEEB01FA116

sudo apt-get update

sudo apt-get install ros-kinetic-desktop-full

sudo rosdep init

rosdep update

echo "source /opt/ros/kinetic/setup.bash" >> ~/.bashrc

source ~/.bashrc

sudo apt-get install python-rosinstall python-rosinstall-generator python-wstool build-essential

1. Install Nice-To-Have utilities
   1. GitKraken
   2. Chromium
   3. TeamViewer
   4. CLion
   5. RoboWare Studio
   6. Set all .sh scripts in ~/Scripts as executable
   7. Set .desktop files in ~/Desktop as executable
2. Use GitKraken or git to clone Autonomous Ecocar repository

git clone https://DTUDynamo@bitbucket.org/dtucar/autonomous-ecocar.git

Login: [dynamo@dtucar.com](mailto:dynamo@dtucar.com)

Password : OekobilGit2017

After cloning, move the repository root to be located in /home/dynamo (e.g. /home/dynamo/Source exists). If you have non-standard folders in /home/dynamo, ensure that they are added to .gitignore before committing.

1. Install dependencies[[1]](#footnote-1)
   1. autoreconf (required for autogen in other libraries)

sudo apt-get install autoconf

* 1. (Optional) serial (required for lightware\_sf40c)[[2]](#footnote-2)

cd ~/Source/Linux/serial

make

sudo make install

* 1. libserialport ([Link](https://sigrok.org/wiki/Libserialport), requires autoreconf)

cd ~/Source/Linux/libserialport

./autogen.sh

./configure

make

sudo make install

* 1. SerialTools (update serial numbers and rerun this when adding new devices)

sudo cp ~/Source/Linux/SerialTools/serial\_tools.h /usr/local/include

* 1. libusb (required for libphidget21)

sudo apt-get install libusb-1.0-0-dev

* 1. libphidget21 ([Link](https://www.phidgets.com/docs/OS_-_Linux), tested 2.1.8)

cd ~/Source/Linux/libphidget

autoreconf -f -i (only needed if newer version than 1.14 of aclocal is installed)

./configure

make

sudo make install

* 1. JsonCpp - Already installed in Ubuntu.
  2. MyQueue

cd ~/Source/Linux/MyQueue

cmake ./

make

sudo make install

* 1. libsbp (SwiftNav Binary Protocol, [Link](https://github.com/swift-nav/libsbp))

git clone git://github.com/swift-nav/libsbp.git ~/SwiftNav/libsbp

cd ~/SwiftNav

sudo apt-get install build-essential pkg-config cmake

cd libsbp/c/

mkdir build

cd build

cmake ../

make

sudo make install

* 1. Velodyne driver

cd ~/catkin\_ws/src/velodyne/

rosdep install --from-paths ./ --ignore-src --rosdistro kinetic -y

**To establish connection:**

Link: http://wiki.ros.org/velodyne/Tutorials/Getting%20Started%20with%20the%20Velodyne%20VLP16

Link: https://www.youtube.com/watch?v=Pa-q5elS\_nE

**Instructions:**

When Velodyne is connected to computer via ethernet

Give the ethernet manual IP: 192.168.1.77

Give the ethernet manual netask: 255.255.255.0

If ethernet connection is named "enp0s25" then (see name using ifconfig):

---- Terminal Instructions: ----

sudo route add 192.168.1.77 enp0s25

1. Prepare the workspace:

cd ~/catkin\_ws/

catkin\_make

(Optional) If this build fails with weird errors, try to rename the repository catkin\_ws to catkin\_ws\_temp, then recreate the repository and copy the src folder from the renamed repository to the recreated on and choose merge, skip (do not replace):

mkdir -p ~/catkin\_ws/src

cd ~/catkin\_ws/

catkin\_make

1. Set you catkin\_ws environment automatically:

echo "source ~/catkin\_ws/devel/setup.bash" >> ~/.bashrc

source ~/.bashrc

See more notes in AutonomousEcocar\Desktop\Notes.txt as well as USB.txt. Optionally, add ~/Scripts/ autostart.sh to autostart (see Notes) for startup melody.

1. Disable CUPS (prevents shutdowns that take 1min30s)

sudo systemctl disable cups-browsed.service

1. Set power button to shutdown

gsettings set org.gnome.settings-daemon.plugins.power button-power suspend

Running

roscore

* Teensy

rosrun teensy teensy\_node

* Gyroscope

rosrun gyro gyro\_node

* Phidget (when running with Steering and Brake motors, Brake can be uncommented in code)

rosun phidget phidget\_node

* + When running “Simply autonomy” you can use the arguments:

rosun phidget phidget\_node record

rosun phidget phidget\_node playback

Record mode will record all steering positions with regards to Teensy reported distance driven into a file called ~/Scripts/steering\_program.txt. Playback will read this file and try to recreate the steering output.

* Remote

Make sure ~/Scripts/register\_ip.sh is set to executable. Start this server before opening the Dynamo Remote Android app. (Only works when Dynamo-PC has static (non-NAT / non-firewall blocked) IP).

rosrun remote remote\_node

* Piksi (GPS)

By default, this will set up the RTK sync server and wait for connection. Connect the base station to an Android phone through a USB OTG cable to the RTK port on the Piksi Multi board and open the Piksi RTK Sync app (only works when Dynamo-PC has static IP). Disable in code if using radio RTK synchronization (set USE\_APP\_REMOTE\_BASE\_SYNC to 0) and rebuild.

rosrun piksi piksi\_node

* Lightware SF40C (only when using sf40)

rosrun lightware\_sf40\_ros sf40\_node

* Navigation Node (incomplete, currently only pose estimation)

rosrun navigation estimate\_pose

* Dynamo Helper (Currently not used, intended for keep-alive and audio feedback)

rosrun dynamo\_helper dynamo\_helper\_node

* Live Webcam

To live stream webcam feed, go to <https://dtucar.com/a/stream.html> on Dynamo-PC and <https://dtucar.com/a> on any client (works well in Chrome/Chromium). This uses minimal latency WebRTC.

* Camera node (Captures images at specified rate, saving images in ~/catkin\_ms/camera\_images)

rosrun camera\_package image\_capturing

* Velodyne driver

rosrun velodyne\_driver velodyne\_node \_model:=VLP16

* LiDAR node

rosrun lidar\_package vlp16\_obst\_detection

For simulation

rosrun lidar\_package vlp16\_sim\_obst\_detection

1. When running executables (e.g. ./configure) from downloaded files, you will need to set the executable permission on that file (Right Click -> Permissions -> Allow executing file as program). [↑](#footnote-ref-1)
2. If using downloaded version, change line 3 to CMAKE\_FLAGS := -DCMAKE\_INSTALL\_PREFIX=/usr/local [↑](#footnote-ref-2)