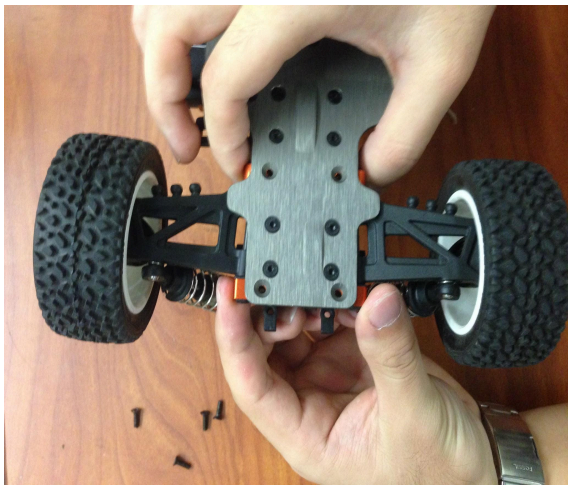
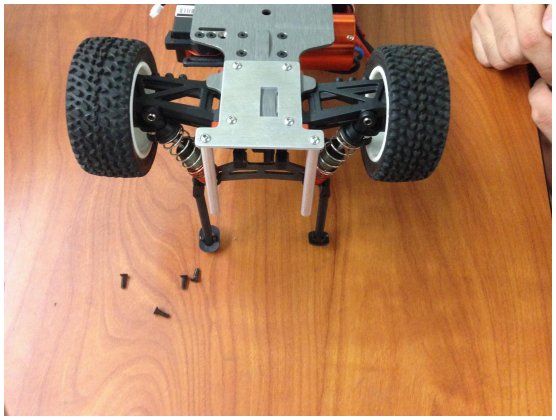
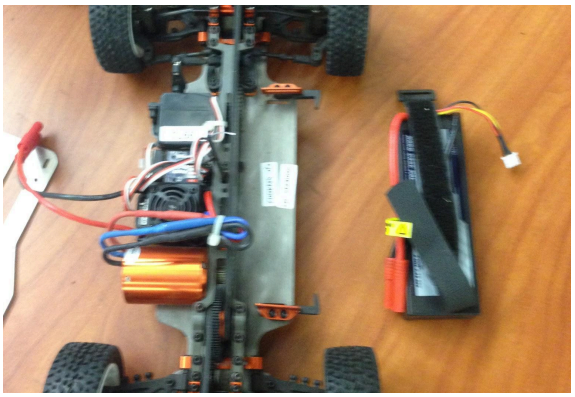
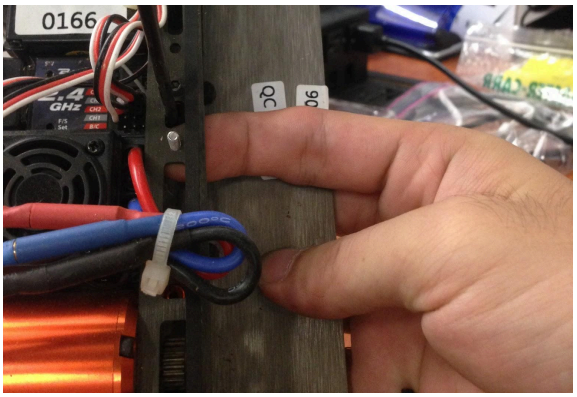
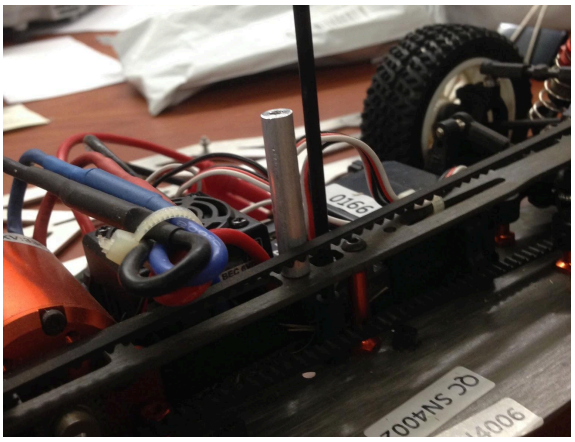
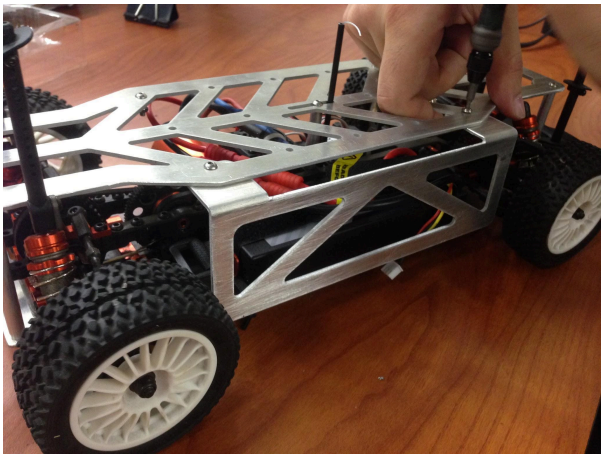
Berkeley Autonomous Race Car Project
Mechanical Assembly Guide

Assembly Instructions

Assemble the deck, standoffs.

Gather the pieces 	Remove the 4 screws (shown below) on the rear end of the chassis 
Place the standoff, and affix it to the chassis 	Apply the same procedure to the chassis front end with the thin stand off 

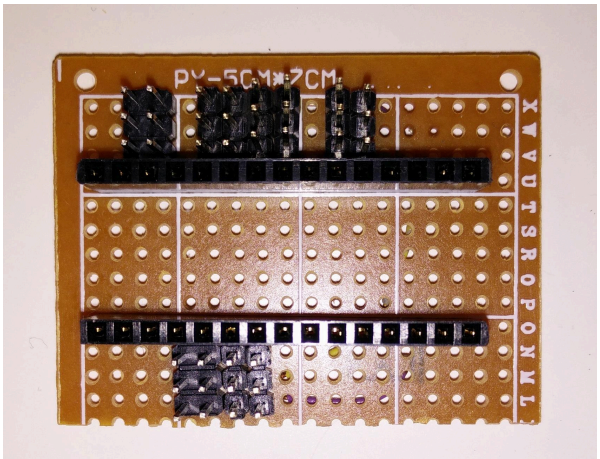
Remove the LiPo battery 	Insert the short standoff through the center vehicle slot (location shown at the tip of the index finger) 
Use a socket screw and fasten the short standoff to the RC central beam, right next to the antenna pole 	Attach micro controller to 
From the top view, install the left bracket, then reinsert the battery, lastly,	Final Result



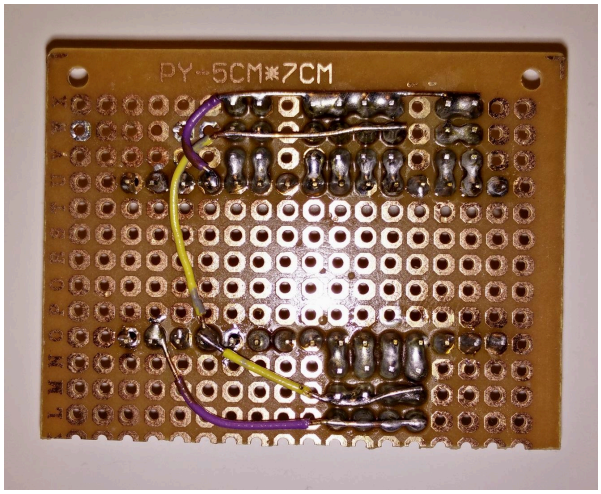
install the right bracket

Prepare the arduino breakout board

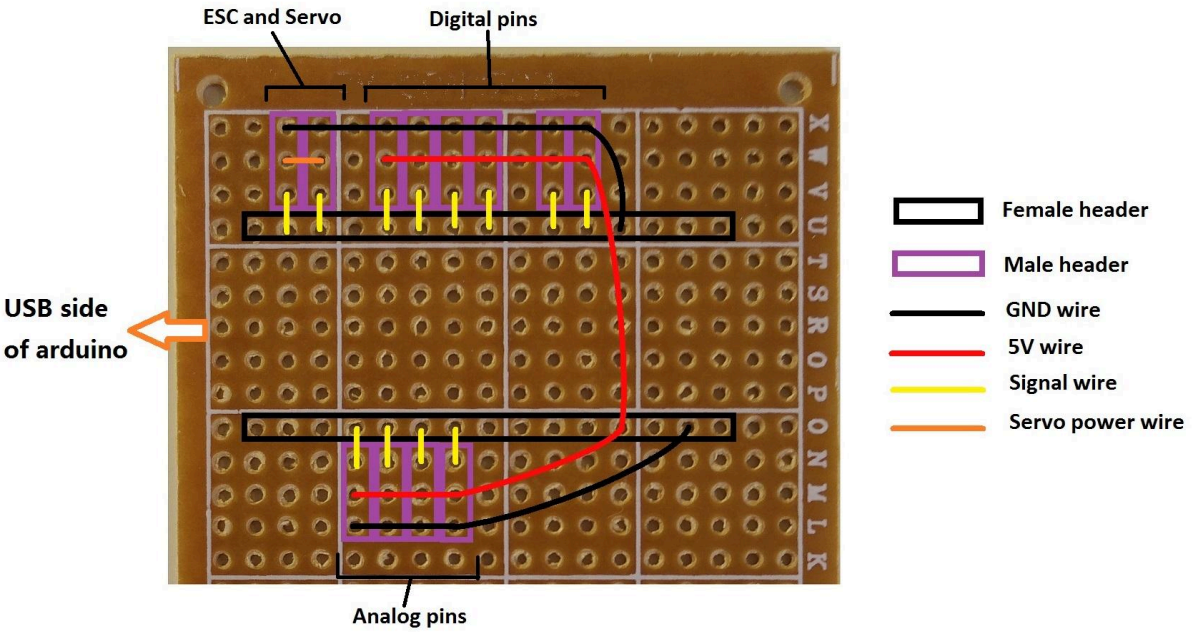
1. Layout of male and female headers. You can add more male headers if you need additional pins. Just need to make sure they align with the appropriate I/O port on the arduino.

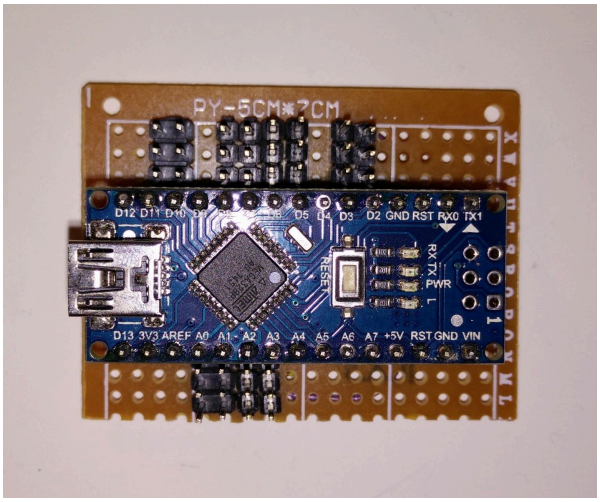


2. Soldering on the back side. Use solder and/or wires to create connections. 5V is tied to the 5V port on the arduino. Ground is tied to the GND on the arduino.

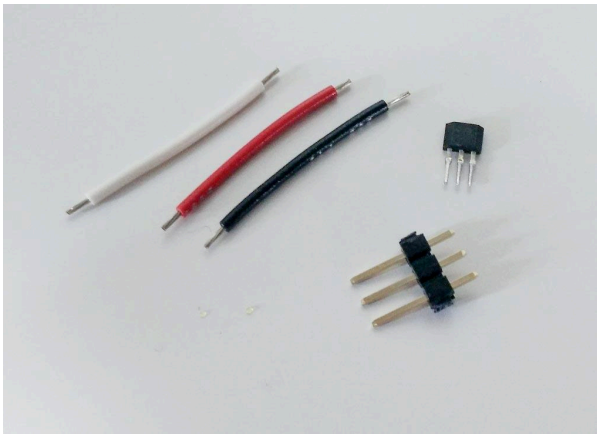
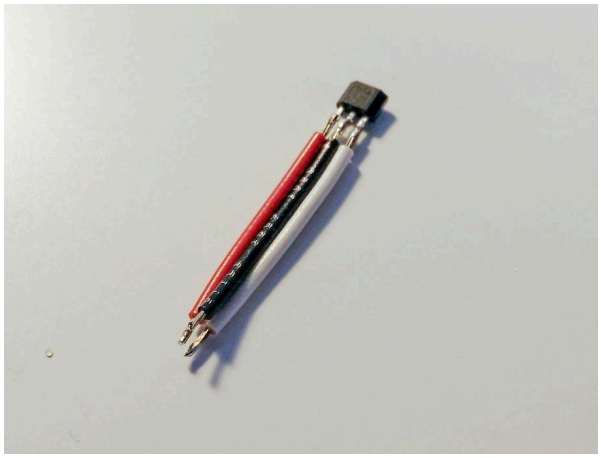


Schematics of connections. Note that **orange** and **red** are **NOT** connected to each other. This is because the ESC actually ***supplies*** power



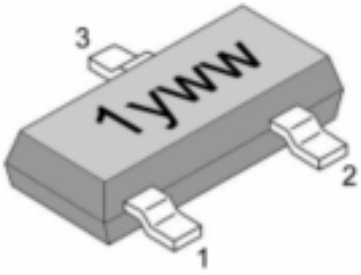
3. Attach the arduino to the board. Note the orientation of the arduino 	

Making the wheel encoders (you'll need 4 of these)

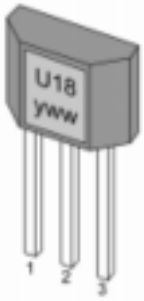
1. Cut three pieces of wire, about 1" each. Strip off about ⅛" from each end. Cut the metal pins on the hall effect sensor to ¼" long. Cut a piece of 1x3 male pin header. See diagram below. 	2. Solder the wires onto the pins of the hall effect sensor. It helps to ***tin the pins and wires*** before soldering them together. Beware of th e polarity (see data sheet below). Use red for VDD, black for GND, white for SIG/OUT. 
Hall effect sensor pinout	

SE Pin №	UA Pin №	Name	Type	Function
1	1	VDD	Supply	Supply Voltage pin
2	3	OUT	Output	Open Drain Output pin
3	2	GND	Ground	Ground pin

Table 2: Pin definitions and descriptions



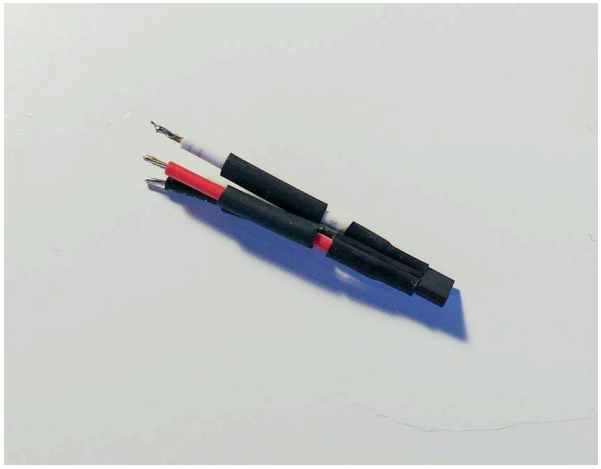
SE package



UA package

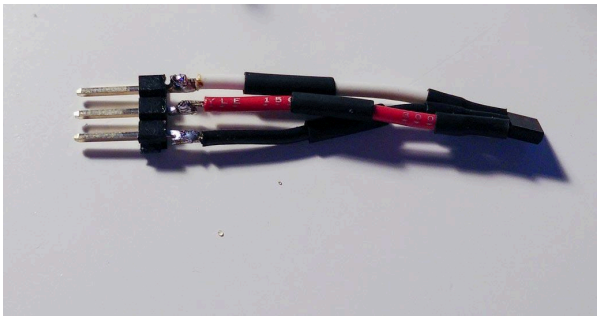
3. Insert heat shrink tubings over the solder joints and heat them with the neck of the soldering iron until they wrap snuggly over the joint. Use the thinnest heat shrink you can find (about 1mm after shrinking). No bare metal should be exposed from the joint.

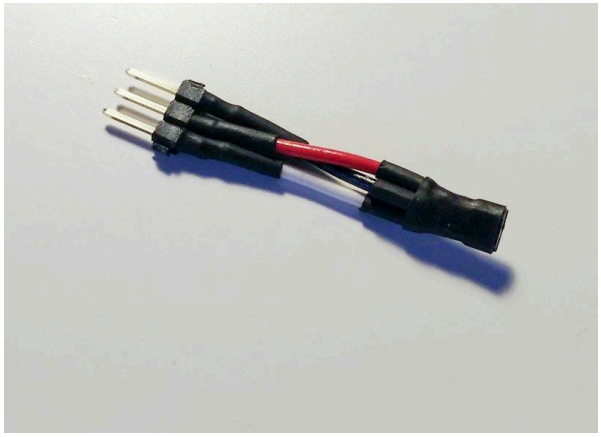
4. Insert another piece of heat shrink tubing on each wire. This is to prepare for the solder joint on the other side.



5. Solder the short side of the male pin header to the wires. Note the crossover between the red and black wire. Red should be in the middle at the pin header side. Push the heat shrink tubing over the solder joint and apply heat.

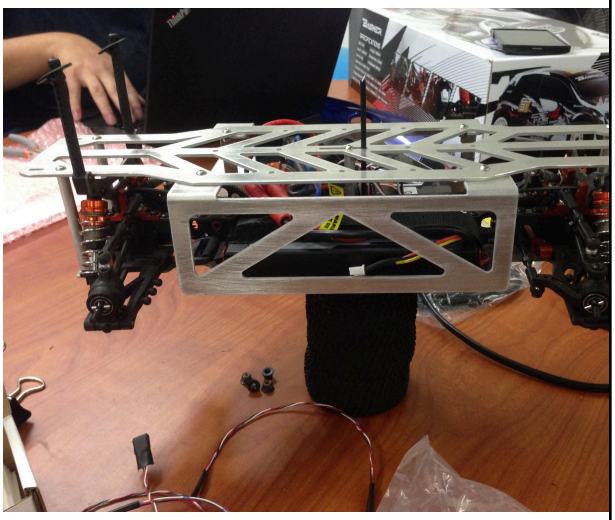
6. Use a piece of larger heat shrink tubing to wrap over the hall effect sensor and the three smaller heat shrinks (this will act as a strain-relief to protect the sensor pins).



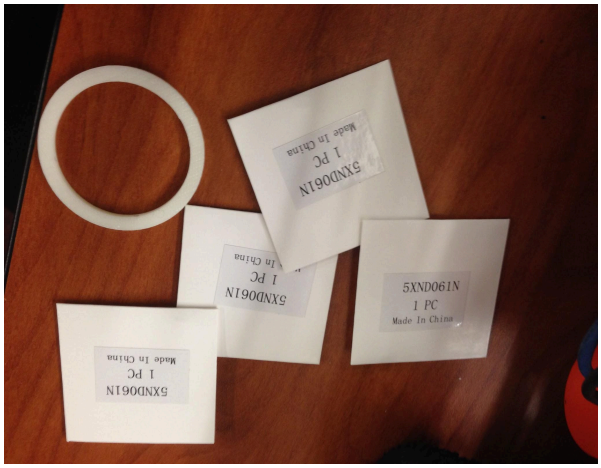


Install the wheel encoder

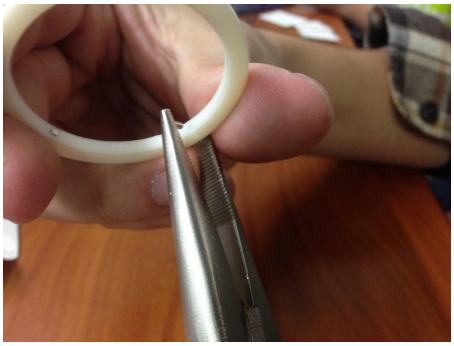
1. Remove 4 Wheels

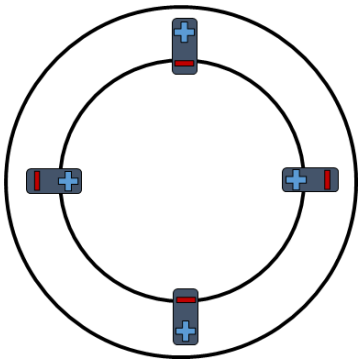


2. For each wheel there is 1 ring and 4 Magnets. Use the “stock” ring if you are using stock wheels. Use “option” if you are changing the wheels.

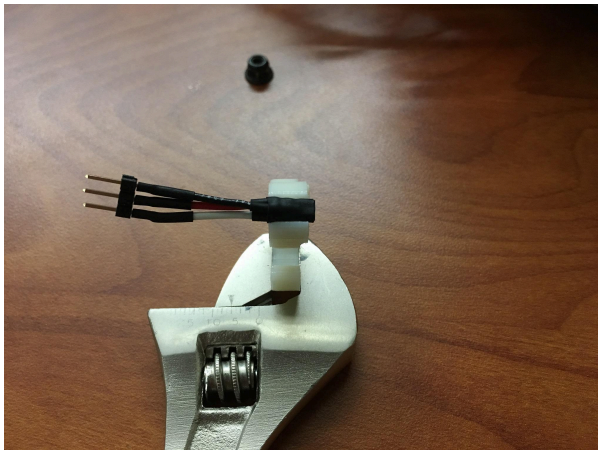


3. Place the magnets in the ring, alternating the polarity, as shown below. If you install a magnet in the wrong direction, you can tug it out with an exacto knife





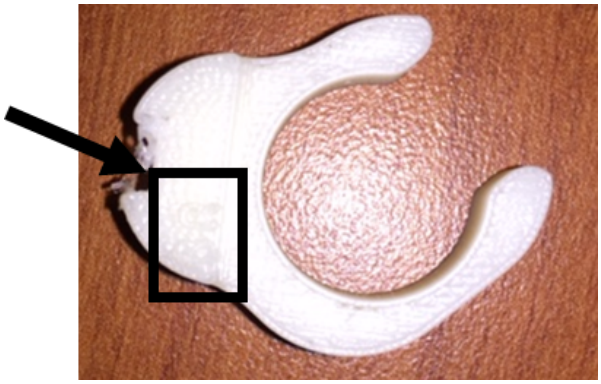
4. Glue the hall effect sensor into the C fixture as according to the picture, with the **chamfered side facing up**.



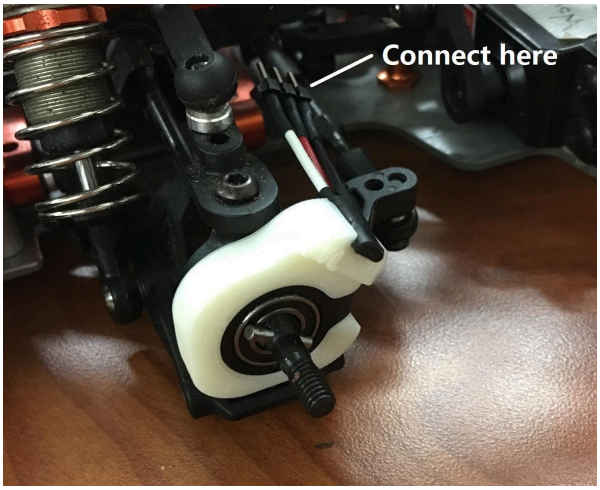
5. If all 4 rings have right polarity, they should adhere to each other like this. The thicker stock rings will not stick because of the protruding letters on one side (the letters say “INSIDE” to indicate which side face towards the center/inside of the car).

6. The four C-fixture and sensors glued together should look like this (see below).

Observe, in the example image below, there is inscribed text that reads BR, for “back right.” It is difficult to see here, but each mount has such a marking, for correct placement on the vehicle
{B = back, F = front}, {L= left, R= right}



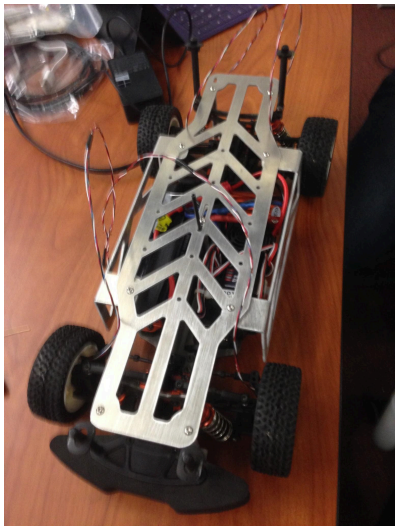
7. Push the fixture onto the wheel hubs, with the pins facing towards the car. The sensor wiring should connect at the labelled connector. (Have the sensor positioned so the pins are easily accessible.)



8. Same as Step 7, this time the rear wheels. See picture below.

9. Press the magnet rings into the wheels, one in each. If you are using stock rings and wheels, the text marking “INSIDE” on the ring should face toward the chassis. “Option” rings are symmetric so you don’t have to worry about orientation.

10. Insert the wheels back onto the tires. The futaba male to female wires are to connect the digital pins on the arduino to the sensors pin header, (make sure VDD, GND, and the signal wires are in correct pins on breakout board, see page 9.)



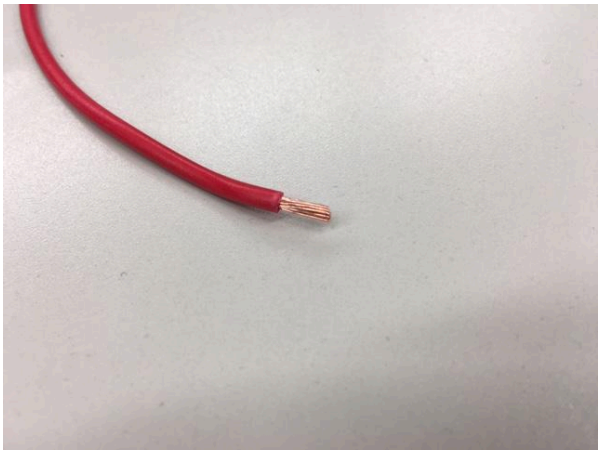
.

Install buck-converter (to power the Odroid from car battery)

(Steps 1-4 illustrate how to install the Anderson powerpole connector in general.)

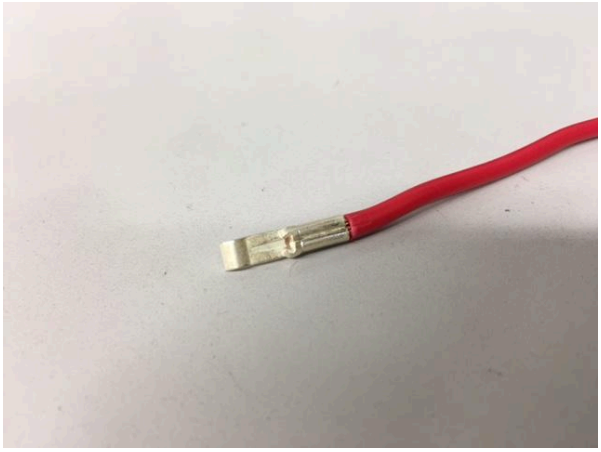
1. How to install an Anderson powerpole connector:

Strip off about 8-10mm of insulation from a piece of 14-18 AWG wire.

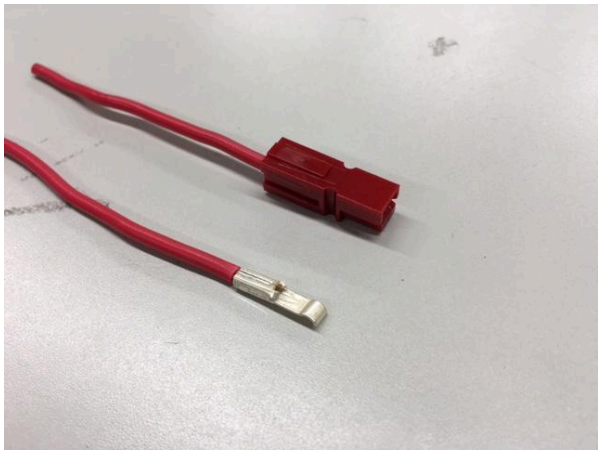


2. Insert the silver insert into the Anderson powerpole crimper, then the stripped wire into the insert piece. Squeeze the handles, which will ratchet-lock until proper pressure has been applied, at which point they will release and spring open.

3. The crimped connection should look like this. If you don't have a crimper, you can also solder the wire into the insert.

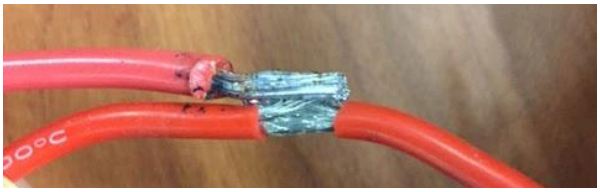
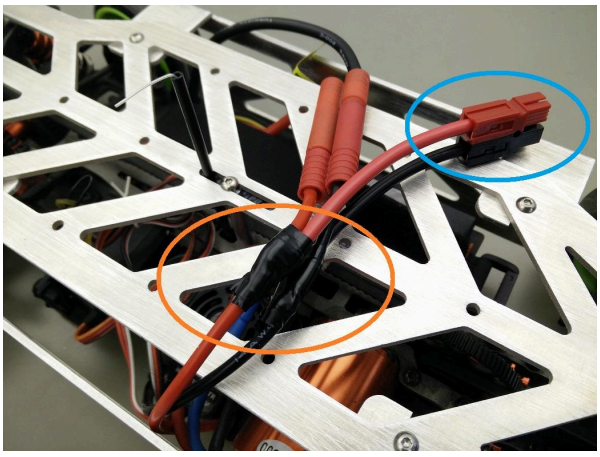
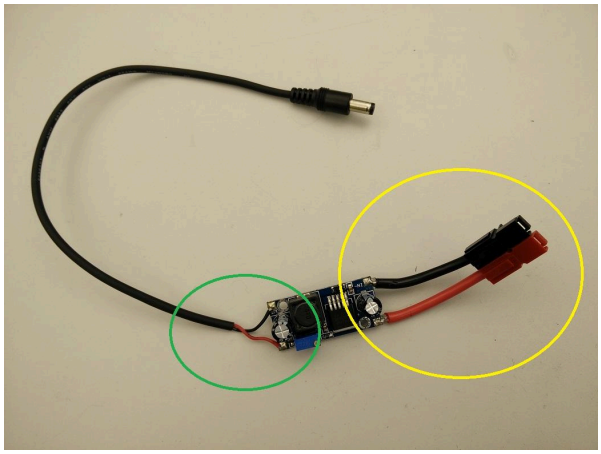


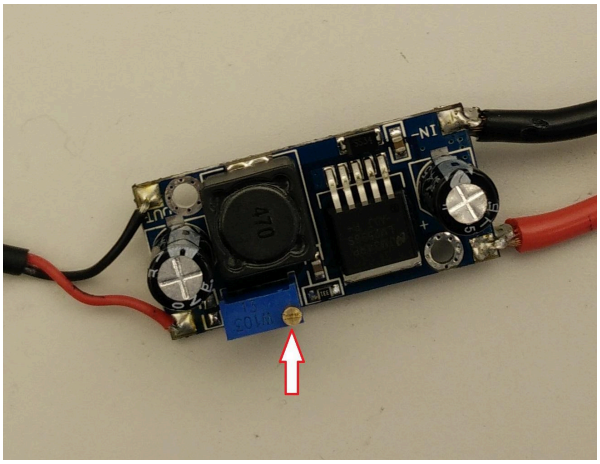
4. Push the insert into the back of the housing, until you hear a click. A small screwdriver can help you apply force on the insert rather than the wire. After the click, you should not be able to pull the wire out of the housing with bare hands.



5. Splice the power cables on the ESC in the place shown (orange). To splice the wire, cut off a piece of insulation on the

6. Install Anderson connectors on the INPUT side of the buck converter (yellow). Solder the "barrel-jack to open

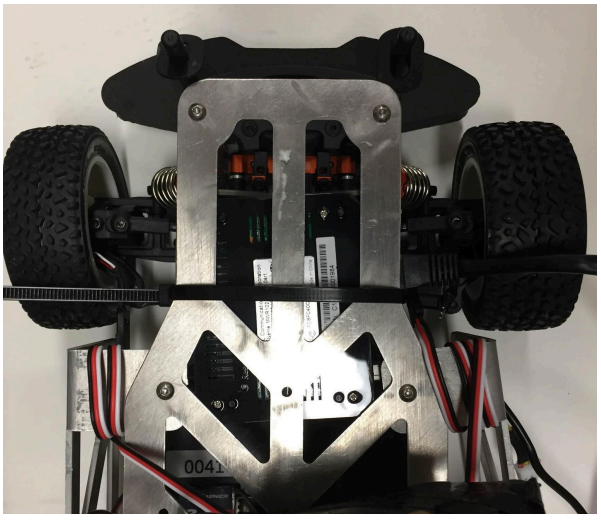
ESC power wires, tin the wire with solder, and solder another wire to it. DO NOT cut the wire, you are simply removing a section of the wire covering. After the splice, wrap that area with electrical tape. The spliced leads will be the input to the buck converter. You can attach Anderson connectors to them using the procedure shown above. Note that you can bind individual Anderson housings together using the tab and slots on the side of the connectors by sliding them into each other (blue).  	leads" cable to the OUTPUT side of the converter (green). Beware of polarity. 
7. Adjust the screw on the potentiometer to trim to output voltage to 5.15V (as close as you can, but do not exceed 5.20V). You **MUST** do this before powering on the odroid for the first time. It is advisable to check the output every time you power the odroid.	8. Insulate the board with electrical tape or other materials of your preference.




5.156
DDM645
DIGITAL MULTIMETER
ON
OFF
HOLD
COM
10A

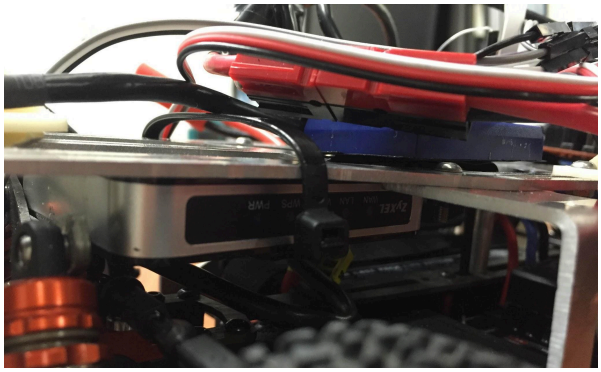
Mount Router underneath the deck

1. On a sheet of paper, Record the login credentials for the router, listed as Wireless SSID and Wireless Password. Place the router on the underside of the front part of the deck as shown below. The router should be oriented such that (a) the router is pushed as far forward as it can, and (b) the ethernet cable extends out to the right. Use a zip-tie to hold the router in place. Insert a ethernet cable into the LAN port of the router, and insert the mini-usb power cable. Rotate the zip tie loop so that the excess end points to the side or downward (this is to minimize contact with any rough surfaces after cutting the zip-tie end). Next, clip the excess ends of the zip-tie

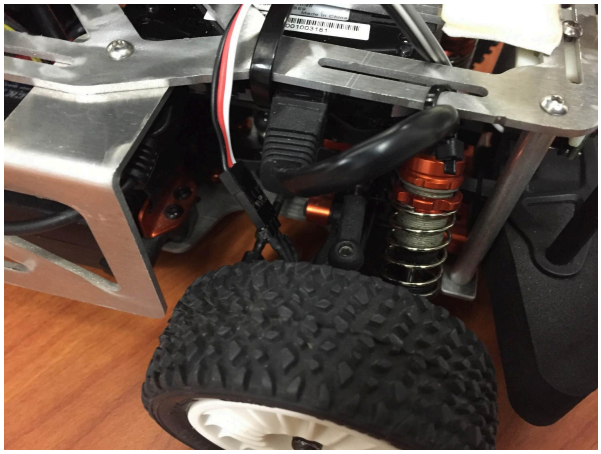


2.It is recommend that you route the ethernet cable underneath the vehicle to prevent cuttering cables on the surface of the deck. Below is one possible way to route the cable underneath the vehicle. The other end of the ethernet cable will connect to the odroid.

(View from left front, looking inward)



(view from right front, looking inward)



On the side of the encoder, make sure the ‘router’ mode is selected. (Between router and

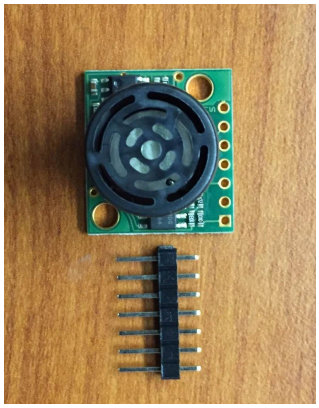
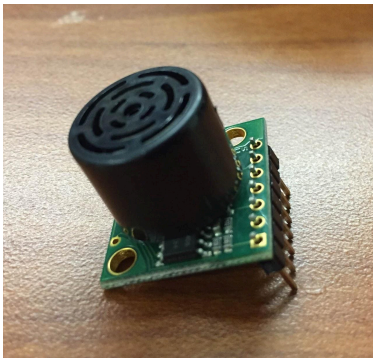
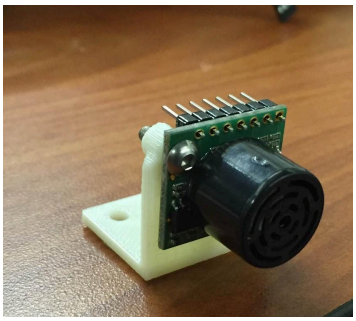
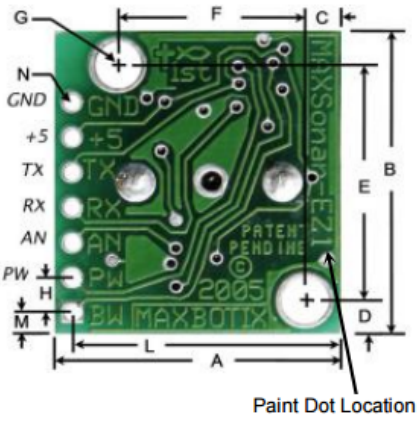
Mount the USB Expansion Hub on the right side bracket

1. Next, mount the usb expansion hub on the right side back. Orient it such that the usb ports face upward and the on/off buttons face inward toward the center of the vehicle. Use zip ties to fasten the hub in place. 	2. After tightening securely, clip the excess ends off the zip tie.

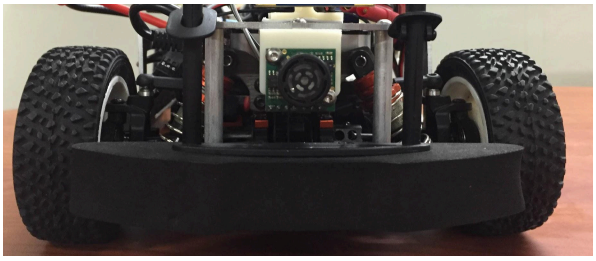
Mount the the camera

1. Gather the camera, the camera base-fixture, and the camera top-fixture. Use M3 button sockets to fix the camera to the top-fixture; note that the holes are slightly smaller than the M3 button socket, so as you turn the socket, you will create threads in the fixture.	2. Correctly orient the base-fixture and top-fixture at the pivot.

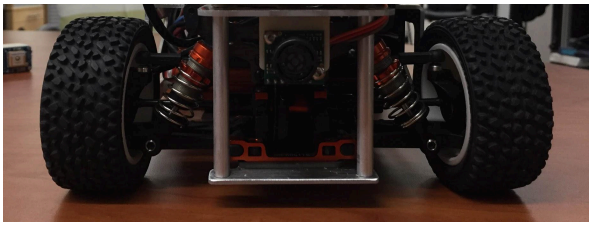
Mount Ultrasound Sensor

1. Gather the ultrasound sensors male pin header connectors 	2. Solder the pins to the sensor in the orientation shown below 
3. Mount the ultrasound sensor to the 3d printed mount. Use M3 nuts and and bolts to fix the sensor. Mount the sensors along the four sides of the car. Make sure it is mounted in a place without any obstructions in front of the sensor. 	4. Connect three ribbon cables to the male header pins (GND, +5, PW). The connection to the arduino are on the discussed in the next section 
5. Install the four mounts along the four sides of the vehicle. One possible configuration would have the front and rear ultrasound sensors on the underside of the deck, and the left and right ones on the top side	

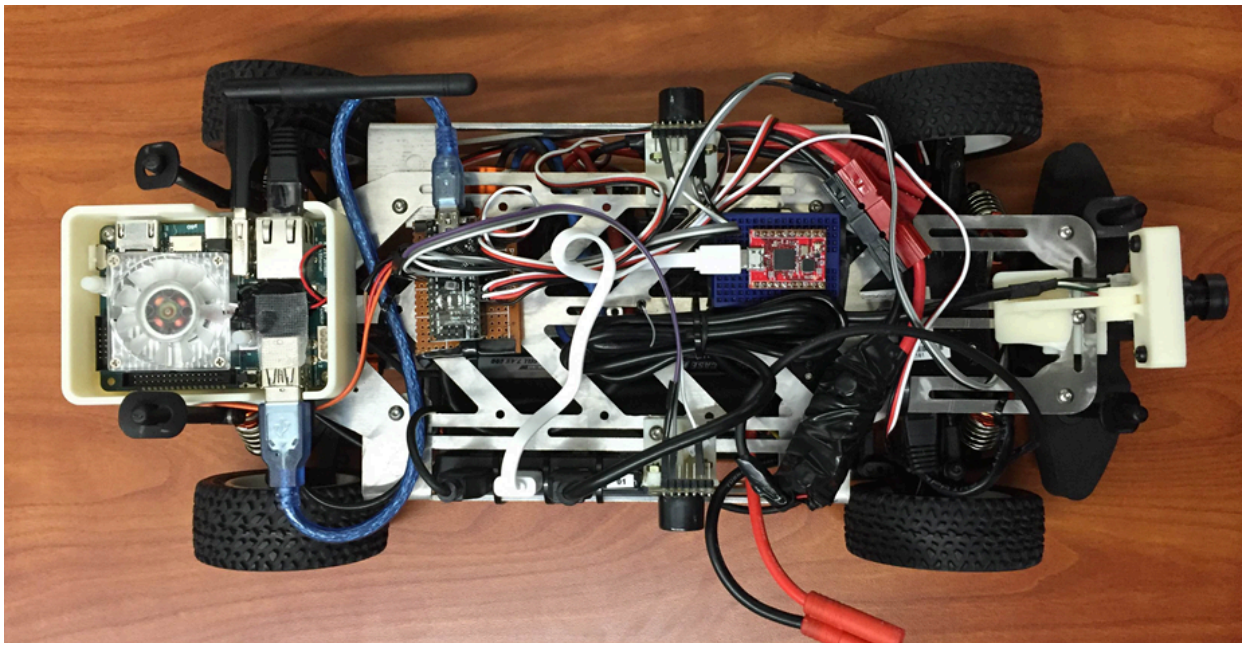
(front view)

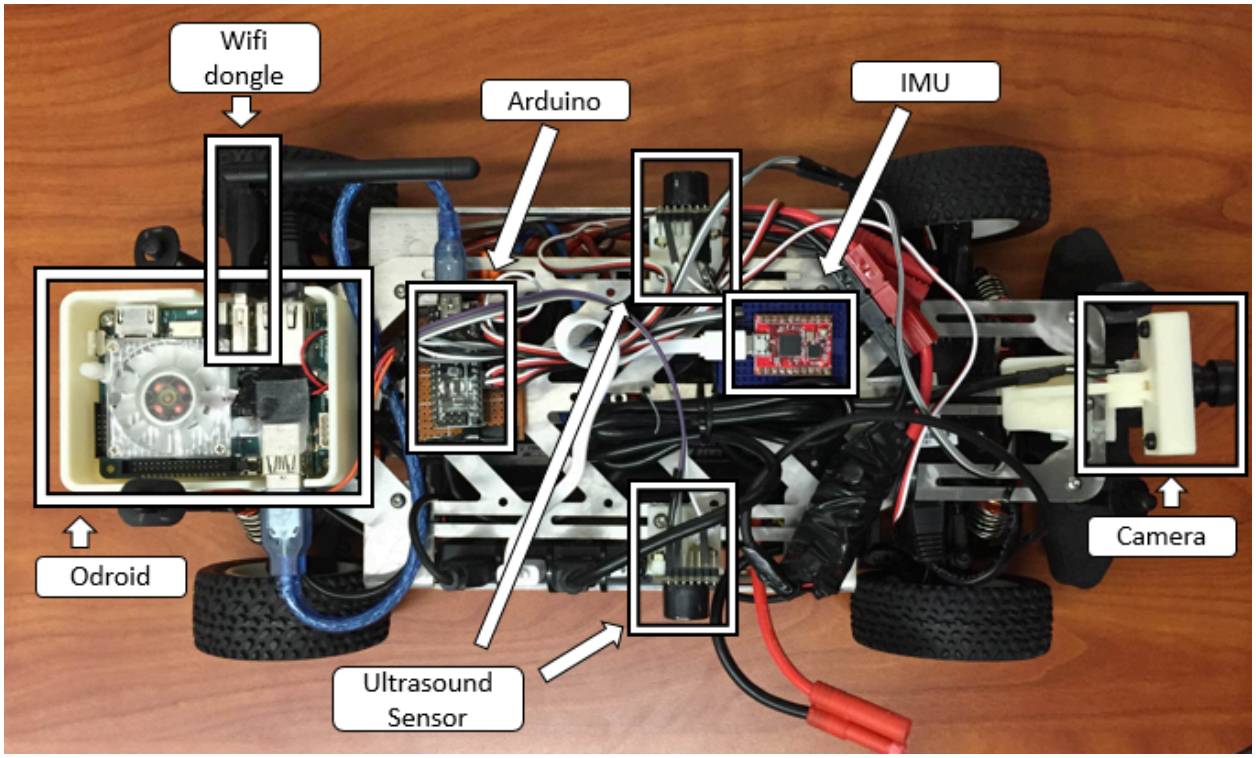


(rear view)



Install Arduino, Odroid, and IMU. Place them in this locations and connect everything. (top view picture)





In this picture, both breakout boards are taped down onto the deck using 3M 110 double-sided tape. The odroid case is fastened to the deck using the 1.5”x1.5” spacing screw holes on the deck. Using different positions for these components will change the physical characteristics of the car, and thus requires changing the code modeling the dynamics. The wifi dongle and the power connector on the odroid should face the rear of the car.

At this point the ESC and Servo wires that are connected to the radio receiver need to be unplugged and instead connected to the odroid via the Arduino Nano on the breakout board. Using the white wire as the signal wire, the red as VDD, and black as GND for each ESC and Servo, connect to the breakout board, using the schematic on page 9. The servo power wire will be powered while the car is on.

Refer to the git page for the most up-to-date pin layout to connect the arduino to the sensors and actuators, here
Next, you will flash the arduino with the code inside the barc repo. Open a terminal, and enter:
arduino
A simple-blue text editor window should open up. Go to File -> Open. In the file selection window, navigate to odroid/barc/arduino/arduino_node, select the file within the folder. Then verify and upload the code by clicking the check mark and right arrow icons, respectively. If everything is successful, the console at the bottom of the text editor should should say “done uploading”

Use tape, zip-ties, or fasteners as needed to make all wiring as neat as possible.

4- Testing

Run the following program to test if everything is working properly. Make sure to go to ~/barc and do a “git pull” to make sure you have the latest code.

To enable cloud services on your vehicle, you’ll need to do the following
First edit the file ~/team_name.sh with your team name (note the ~ means that it is in your home directory)

Then, open a terminal and launch the following commands

- source ~/barc/scripts/reset_database.sh

This will clean your database by removing all old data (if any) from the odroid image

Open two more terminals, and launch the following commands

- roscore
- rosrun data_service service.py

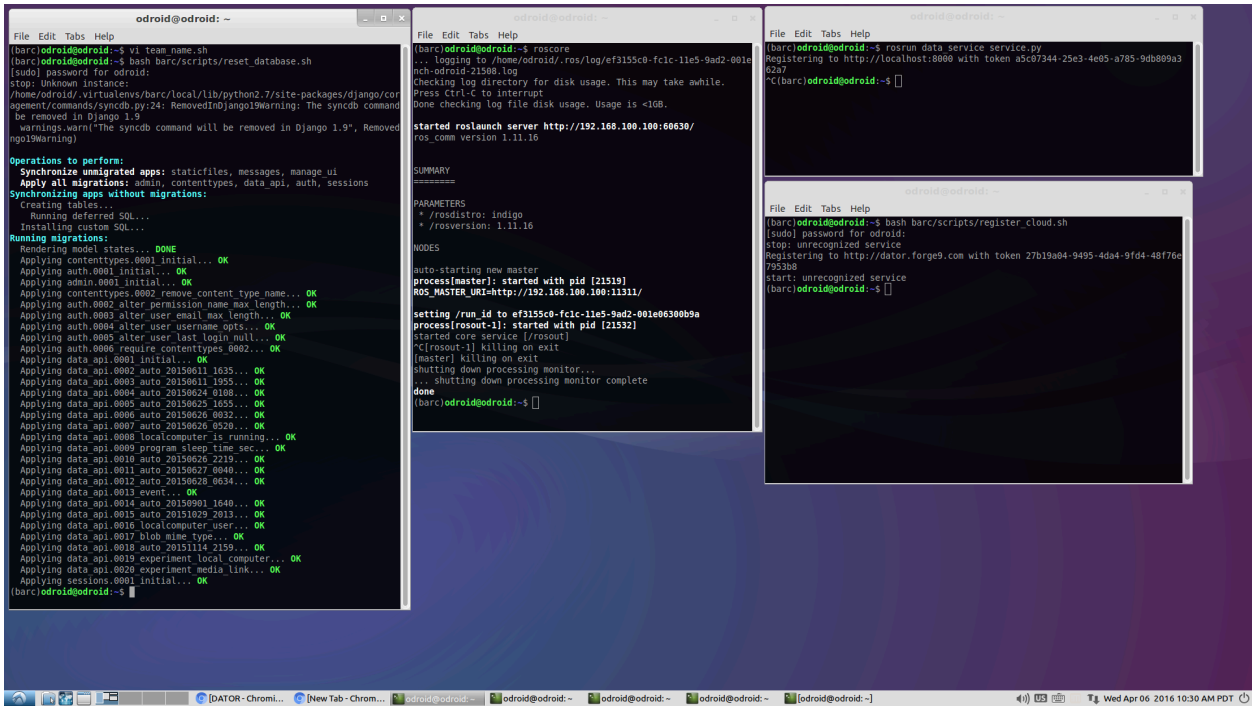
The console should report that it registered with the local database with a unique key after a second or two. After outputting this, close kill the ROS processes (Ctrl-C) and exit the terminals.

Install the following python package using the pip utility
pip install boto3

Make sure you are now connected to the internet
Then, open another terminal, and run the following

- bash ~/barc/scripts/register_cloud.sh

The following screen shows all the above steps



Now you have successfully registered to the cloud

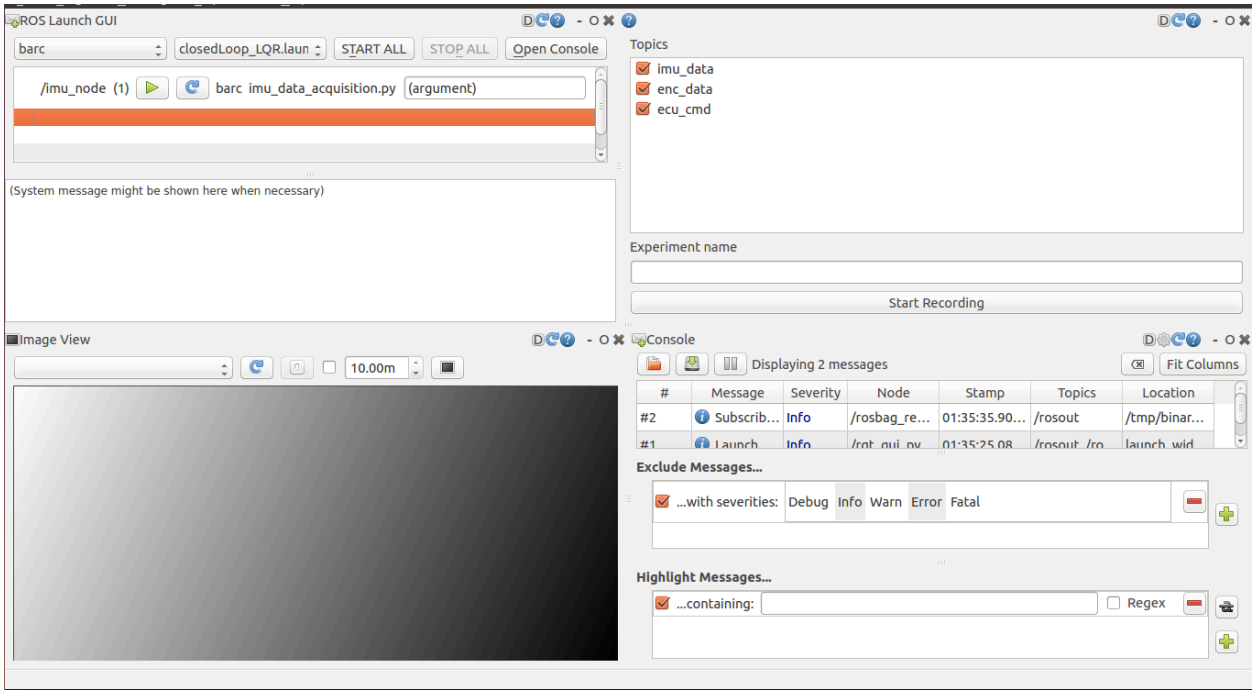
To build your workspace, open a terminal, and enter the following commands

- source ~/barc/scripts/rebuild_system.sh

Make sure to have roscore running, then execute the following in order:
rosrun data_service service.py
rqt -p ~/barc/perspectives/experiment.perspective

These should be run in two separate terminals.

Then you should be able to see the following GUI



If you see a blank screen, you can go to the “Plugins” and you can add the plugins. Add the “ROS Launch GUI” from the dropdown. Add BARC GUI. You can also add the “Image View” and “Console” from the “Plugins” dropdown menu. These plugins can be dragged and re-arranged by clicking the upper-left icon.

In the ROS Launch GUI section, the top left quadrant, you can start the launch files in ~/barc/workspace/src/barc/launch by selecting the file and pressing the “START ALL” button. This will launch all the nodes defined in the launch file and the car should start.

In the pane to the right is a GUI that corresponds on how to record your data, you can see the code at ~/barc/workspace/src/barc_gui/src/barc_gui/gui.py

The checkboxes correspond to the topics that the data will be collected on, you can add more topics to be collected by editing the gui.py file. You will need to name your experiment and press “Start Recording” button before running your experiment via the launch file UI.

Press the stop recording button once you are done and the data will be collected. You can then view the data by going to the url “localhost:8000” in your car. The data will be available under the experiment name that you entered.

If you connect your car to wifi, it will automatically push your data to the Amazon cloud server, you can see the data that we have collected so far at dator.forge9.com

NOTE: There is a current bug with the rqt_launch where it will ignore the parameters set the launch file. We have fixed the bug but it is not pushed to the original upstream repository on Github, so you should uninstall any rqt_common_plugins previously installed with the command “sudo apt-get remove ros-indigo-rqt-common-plugins”. The current BARC repository has rqt_common_plugins that includes the bug fix to rqt_launch. Make sure to use this one and not the one you get through apt-get.

If you have the camera setup, you can launch the camera.launch file. Then click the refresh button in the Image View section, then click the topic in the dropdown to see the live feed from the camera.

You have completed all the steps, enjoy your car!!

Contributors

Steven van Leeuwen, Chenliu Lu, Tong-Hua Yu, Cheng Hao Yuan, Jon Gonzales, Kiet Lam, Francesco Borrelli