Robot Control Steps

1. Connect battery power and turn on main switch, also ensure that the contactor is on using the wireless remote.

(Note: If you go through all steps and the machine does not run, first try making sure the contactor is on and the controllers have green lights, then reboot the pi using the command: sudo reboot , sometimes there is an issue where the pi does not recognize the controllers unless things are powered in the correct order.)

1. Connect to Pi via WIFI using VNC Viewer. The login information is as follows:
   1. SSID = SD\_pi
   2. PSK = seniordesign

(Note: The SSID and PSK can be changed, I setup the pi as an access point following this page: https://elinux.org/RPI-Wireless-Hotspot)

* 1. IP = 192.168.42.1

(Note: You can SSH into it but the main app will not work due to the gui needing to start an X11 window.)

1. Start a Terminal session. Change directories using the following command:

cd ./app/app

(Note: There are a lot of test scripts in this directory and scattered throughout the file system. But this directory has the latest tested scripts.)

1. You are now ready to start the program using the following command:

sudo python3 APP\_GPS.py

(Note: The motors start at 0, when you input a number between -1000,1000 the motors will not respond until you hit the enter key. The inputted number is 0.1\*MAX\_RPM, i.e. 100 = 10% power.)

(Note: Linux will do odd things sometimes, meaning every once in awhile the serial ports are swapped. This causes the Right motor to actually be the left and vice versa, this has been experience a few times and should be mentioned.)

(Note: The hydraulic functions have a software sleep function that lets it incrementally move. As of 12/21/2017 this function is a blocking function meaning all other processes stop, i.e. the motors.)

1. When ready to power down the machine it is recommended that you run the following command so that you do not run the risk of corrupting the disk image:

sudo shutdown now

1. Other noteworthy things:
   1. The contactor acts as an emergency stop to cut power to the motor controllers.
   2. As of 12/21/2017 the ultrasonic sensors are outputting 0’s and should be troubleshooted.
   3. There is no acceleration profile implemented but one can be using the roboteq serial commands that reside in the roboteq.py script.