**ROS Launch Files Guide**

**weeblebot\_remote\_controller\_with\_slider.launch**

Launches following nodes:

* imu\_filter\_node
* imu\_mpu6050\_node.py
* imu\_madgwick\_filter.py
* sensing\_and\_wheel\_control\_node\_v02.py
* sensors\_processing\_v01.py
* manual\_tuning\_sliders.py
* (data\_plot.py) (need to fix errors)
* remote\_control\_driver.py

Install ROS-Mobile app on Android, as <https://github.com/ROS-Mobile/ROS-Mobile-Android>

To start, enter superuser mode:

sudo su

Source the bash location:

source weeblebot/devel\_isolated/setup.bash

Launch the program:

roslaunch mobrob weeblebot\_remote\_controller\_with\_slider.launch

Wait until it completes calibration procedure, as shown below.

Text

Description automatically generated

After installing ROS-Mobile app, making sure Raspberry PI and phone are both connected to the same WIFI, UWNet or any other WIFI. Open the app, input IP address into Master URL section, and 11311 as its Master port. Raspberry PI IP address can be found using:

ifconfig

Text

Description automatically generated

In the app, if not yet done, add a joystick widget as below. Back to Master page, click CONNECT, and use generated widget under VIZ page to send X and Z command to the remote driver:

(Edit on 1/27/2022 by Allison Willhite: Make sure to use ifconfig before launching the program, the program must be launched for the app to connect.)

Graphical user interface, application

Description automatically generatedGraphical user interface, text, application

Description automatically generated

A screenshot of a computer screen

Description automatically generated with medium confidenceA screenshot of a computer

Description automatically generated with medium confidence

Open a new tab in Terminal, open rqt\_plot:

rqt\_plot

Graphical user interface

Description automatically generated

Add topics to plot them interactively,

Graphical user interface, chart

Description automatically generated

If necessary, open another rqt\_plot,

rqt\_plot

Graphical user interface, application

Description automatically generated

The following manual tuning slider will show as the calibration completes, while looking at the interactive plots, this slider can be used to tune the robot to its optimal condition:

Graphical user interface, application

Description automatically generated

**weeblebot\_remote\_controller.launch**

Launches following nodes:

* imu\_filter\_node
* imu\_mpu6050\_node.py
* imu\_madgwick\_filter.py
* sensing\_and\_wheel\_control\_node\_v02.py
* sensors\_processing\_v01.py
* remote\_control\_driver.py

Install ROS-Mobile app on Android, as <https://github.com/ROS-Mobile/ROS-Mobile-Android>

This launch file is the same version as weeblebot\_remote\_controller\_with\_slider.launch, except it doesn’t launch sliders and plotter. These two were specifically programmed for PID tuning purpose. Once the optimal PID parameters are found, it can be directly applied and launch this program for user control.

**weeblebot\_angle\_speed\_loop.launch**

Launches following nodes:

* imu\_filter\_node
* imu\_mpu6050\_node.py
* imu\_madgwick\_filter.py
* sensing\_and\_wheel\_control\_node\_v02.py
* sensors\_processing\_v01.py
* set\_desired\_wheel\_speeds\_v01.py

To start, enter superuser mode:

sudo su

Source the bash location:

source weeblebot/devel\_isolated/setup.bash

Launch the program:

roslaunch mobrob weeblebot\_angle\_speed\_loop.launch

This launch program should control the robot to do specific motion. There’s also a feature to plot the data points out if needed.

**mpu6050.launch**

Launches following nodes:

* imu\_filter\_node
* imu\_mpu6050\_node.py
* imu\_madgwick\_filter.py

To start, enter superuser mode:

sudo su

Source the bash location:

source weeblebot/devel\_isolated/setup.bash

Launch the program:

roslaunch mobrob mpu6050.launch

This launch program solely runs IMU features on Weeblebot and was used mainly for IMU troubleshooting.

**CLDRoperate.launch**

Launches following nodes:

* imu\_filter\_node
* imu\_mpu6050\_node.py
* imu\_madgwick\_filter.py
* sensing\_and\_wheel\_control\_node\_v02.py
* sensors\_processing\_v01.py
* dead\_reckoning\_v01.py
* mobile\_robot\_animator\_v00.py
* closed\_loop\_path\_follower.py
* set\_path\_to\_follow.py

To start, enter superuser mode:

sudo su

Source the bash location:

source weeblebot/devel\_isolated/setup.bash

Launch the program:

roslaunch mobrob CLDRoperate.launch

This launch program controls Weeblebot to move a set path under the closed loop path controller.

mobile\_robot\_animator\_v00.py will plot the robot’s movement out and it allows the user to see Weeblebot location in xy coordinate even if he/she is not looking at Weeblebot