Installation Documentation

September 16, 2020

First, copy ubuntu_sim_ros_melodic.sh to your home directory.
Then, open your terminal and call the command

```
source ubuntu_sim_ros_melodic.sh
```

This command may take a few minutes, depending on internet connection. It downloads and installs ros, mavros, mavlink, Gazebo, and builds necessary things.
Then, call these commands in your home directory.

```
git clone https://github.com/METUrone/Firmware.git
```

Open a new terminal, and write

```
gedit .bashrc
```

To this file's end, paste the following text.

```
source Firmware/Tools/setup_gazebo.bash $(pwd)/Firmware $(pwd)/Firmware/build/px4_sitl_default
export ROS_PACKAGE_PATH=$ROS_PACKAGE_PATH:$(pwd)/Firmware
export ROS_PACKAGE_PATH=$ROS_PACKAGE_PATH:$(pwd)/Firmware/Tools/sitl_gazebo
```

In the end, you should save and close the file.
After that point, all installations are done.

To run the simulation, you must execute the following commands every time.

Now open a new terminal, and call the command

```
roscore
```

Then, in a new terminal, call

```
cd Firmware
no_sim=1 make px4_sitl_default gazebo
```

This command will take at least 5 minutes in first call.
After the installation, the terminal will say "Waiting for simulator to accept connection on TCP port 4560".
That means installation is done, after we inserted a drone into simulator, pixhawk will automatically connect to drone.

Just for the first time, please do the following, again in another terminal.

```
gedit ./.ignition/fuel/config.yaml
```

In that file, there is a line that writes

```
url: https://api.ignitionfuel.org
```

Change it as the following

```
url: https://api.ignitionrobotics.org
```

Then open a new terminal and call

```
roslaunch mavros px4.launch fcu_url:="udp://:14540@127.0.0.1:14557"
```

This command will execute much faster, in the end the last line will be similar to this: "MAVROS started. MY ID 1.240, TARGET ID 1.1".
Again, in a new terminal, call

```
roslaunch gazebo_ros empty_world.launch world_name:=$(pwd)/Tools/sitl_gazebo_worlds_iris.world
```

This command will open Gazebo after a few moments.
From left, click insert, and from the left menu click 3DR Iris and place it on the simulation.
Then, look for the terminals. You should see some changes.
To try if it works, open the second terminal(the one that we call "no_sim=1 make px4_sitl_default gazebo") and write the command

```
commander takeoff
```

This should takeoff the drone on the simulator.
After that, you can give your commands using your ros nodes.
To try that, download the example package in the repository, build the ros package, and run it while all the terminals and simulation working.
The drone should takeoff and stay at 2.5m.