**IMU and LRF Mapping**

[1. Roscore Execution 2](#_Toc22726)

[2. Program Compile build 3](#_Toc6701)

[3. Hokuyo LREF Sensor Execution 4](#_Toc3828)

[4. IMU Distance measurement Module execution 5](#_Toc27)

[5. LRF(hokuyo) Sensor data receiving and dynamixel execution 7](#_Toc3808)

[6. GUI view execution 8](#_Toc2515)

[7. TF 좌표 변환 9](#_Toc31083)

[8 Octree setting 10](#_Toc18112)

## 1. Roscore Execution

1. 터미널창에서 "roscore"를이용해서 ROS 프로그램실행

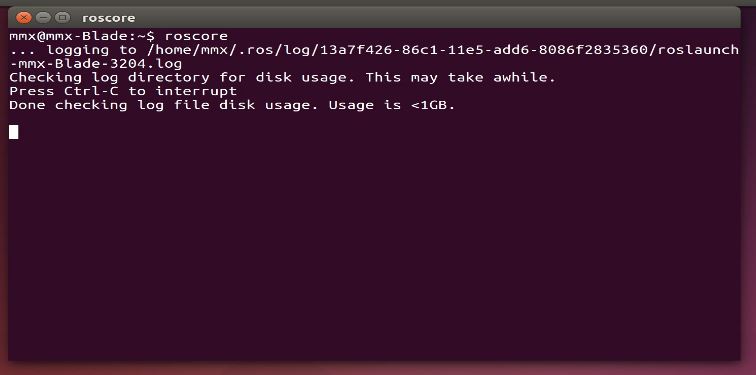


fig.1.ROS program execution

\*command: roscore

## 2. Program Compile build

1. “cd catkin\_ws/”입력하고, 폴더이동
2. “catkin\_make”입력하고, 프로그램컴파일

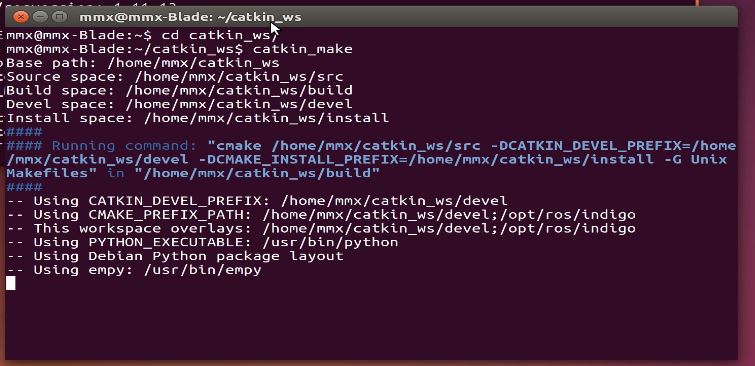


fig.2. Program build

\*command: cd catkin\_ws

\*command: catkin\_make·

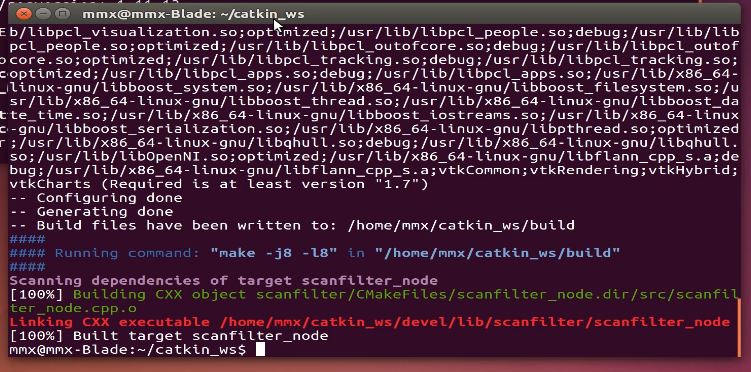


fig.3. Program build complete

## 3. Hokuyo LREF Sensor Execution

Start a new terminal and run commands.

1. “sudo chmod a+wr /dev/ttyACM0”입력후 Sensor 이용권한허가

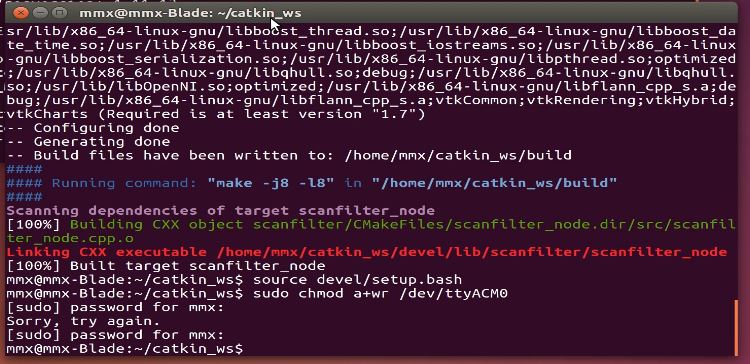


fig.4. Sensor use permission

1. -“rosrun hokuyo\_node hokuyo\_node” 입력, hokuyo\_node실행

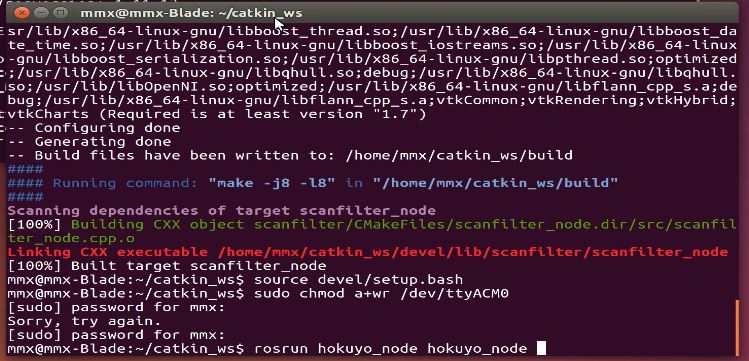


fig.5.hokuyo node execution

\*command: sudo chmod a+wr /dev/ttyACM0

\*command: rosrun hokuyo\_node hokuyo\_node

## 4. IMU Distance measurement Module execution

Start a new terminal and Enable the USB port(set the permission)

1. Run “sudo chmod a+rw /dev/ttyUSB1” , Here the “USB1” means the usb port number which can be views by command “ls –l /dev/tty\*”.

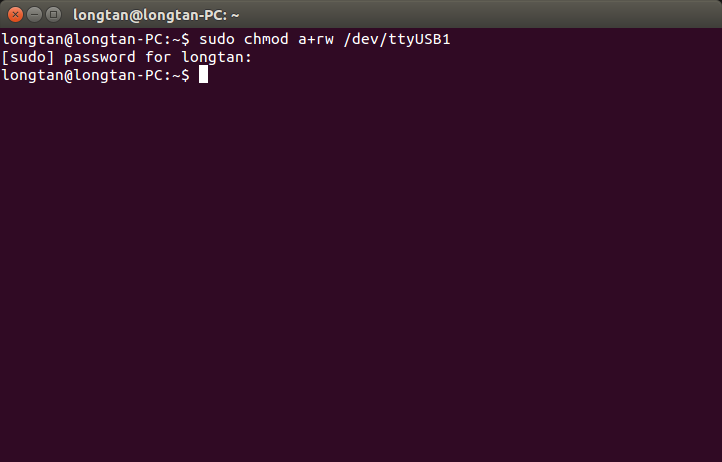


fig.6. IMU permission

Source the code

1. Run ”source catkin\_ws/devel/setup.bash”

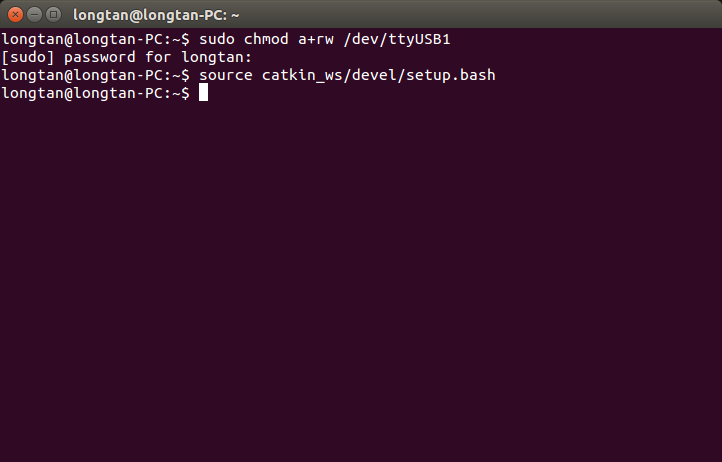


fig.7. Source the code

Run the module

1. Run ”rosrun imu imu.py /dev/ttyUSB1`” .

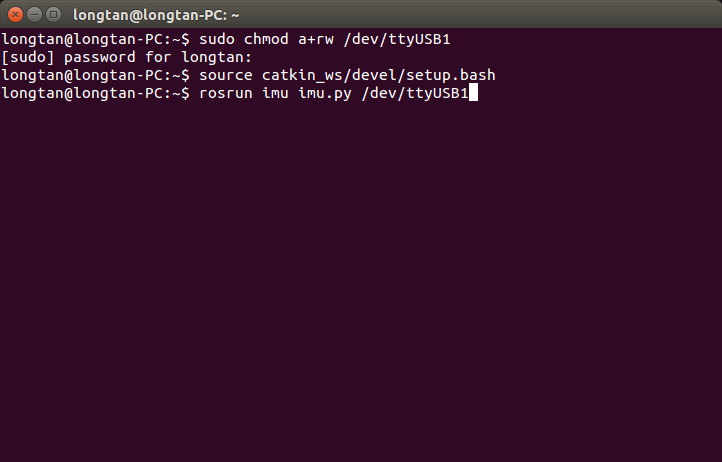


fig.8. IMU runing

command: sudo chmod a+rw /dev/ttyUSB1

command: source catkin\_ws/devel/setup.bash

command: rosrun imu imu.py /dev/ttyUSB1

## 5. LRF(hokuyo) Sensor data receiving and dynamixel execution

LRF Sensor를 설치하고, Rviz프로그램에서 서보모터 구동한 후, Octree구동해서 맵을 작성

1. -Enable the Dynamixel usb port by command “sudo chmod a+wr/ dev/ttyUSB0”

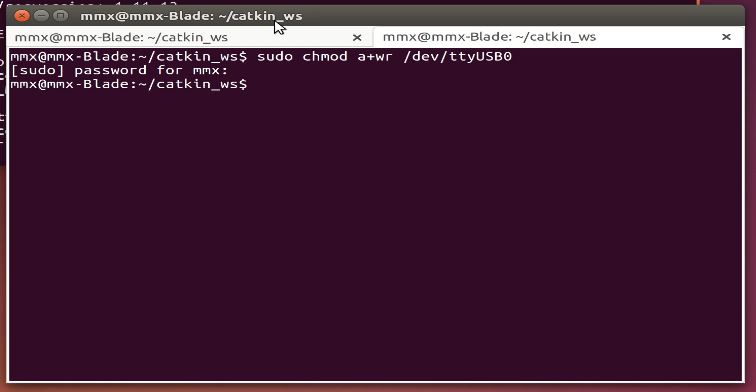


fig.11. USB Port use permission

1. - Source the code by command “source catkin\_ws/devel/setup.bash”
2. -run LRF(hokuyo) Sensor data receiving and dynamixel codes by command ”rosrun scanfilter scanfilter\_node /dev/ttyUSB0”

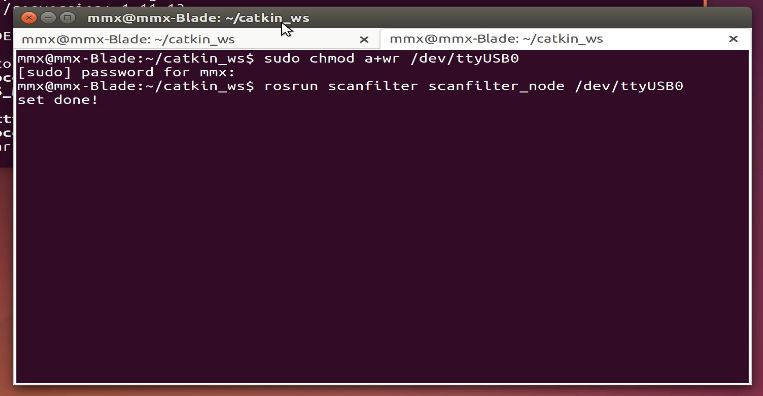


fig.12. USB Port setting and dynamical

\*command: sudo chmod a+wr/ dev/ttyUSB0

\*command: source catkin\_ws/devel/setup.bash

\*command: rosrun scanfilter scanfilter\_node /dev/ttyUSB0

## 6. GUI view execution

1. start a new terminal and “rosrun rviz rviz” command and rviz program execution

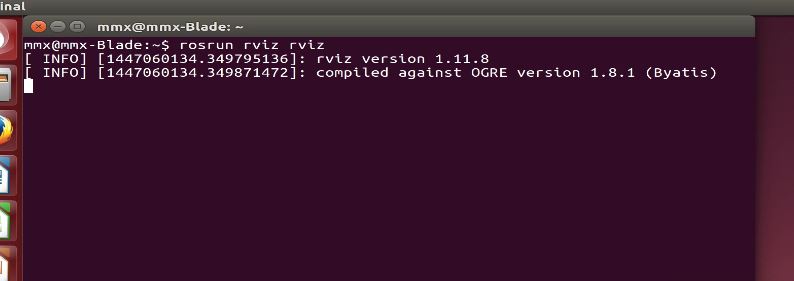


fig.13.rviz program execution

command: rosrun rviz rviz

## 7. TF 좌표 변환

* 좌표 변환

좌표 변환을 설정한 후, Octree 설치 하고 실행.

1. "rosrun tf static\_transform\_publisher 0 0 0 0 0 0 map dynamixel 30", command and password

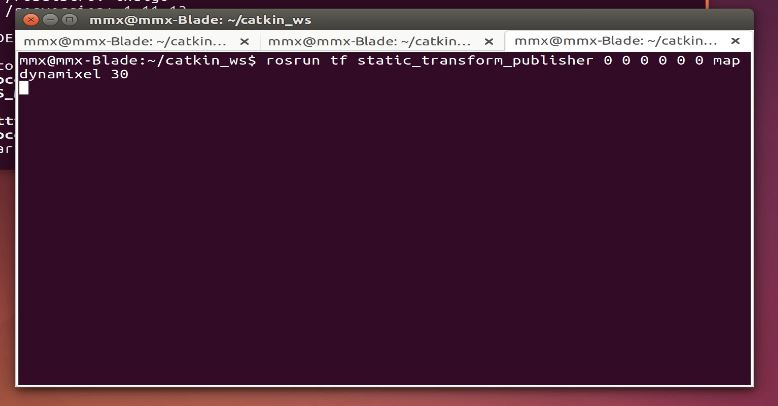


fig.14. Command complete

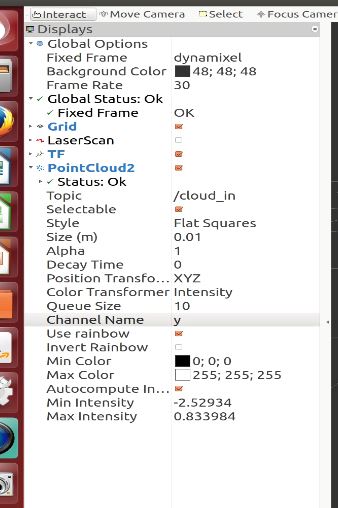


Fig.15.Check the information frame rate

\*command: rosrun tf static\_transform\_publisher 0 0 0 0 0 0 map dynamixel 30

## 8 Octree setting

1. Open a new terminal and run “rosrun octomap\_server octomap\_server\_node”
2. Set the gui Point cloud topic to “octomap\_point\_cloud\_centers”

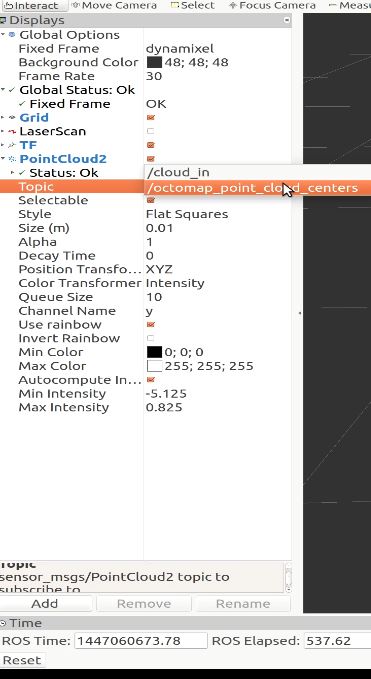


fig.16. Rviz topic change octomap point cloud centers

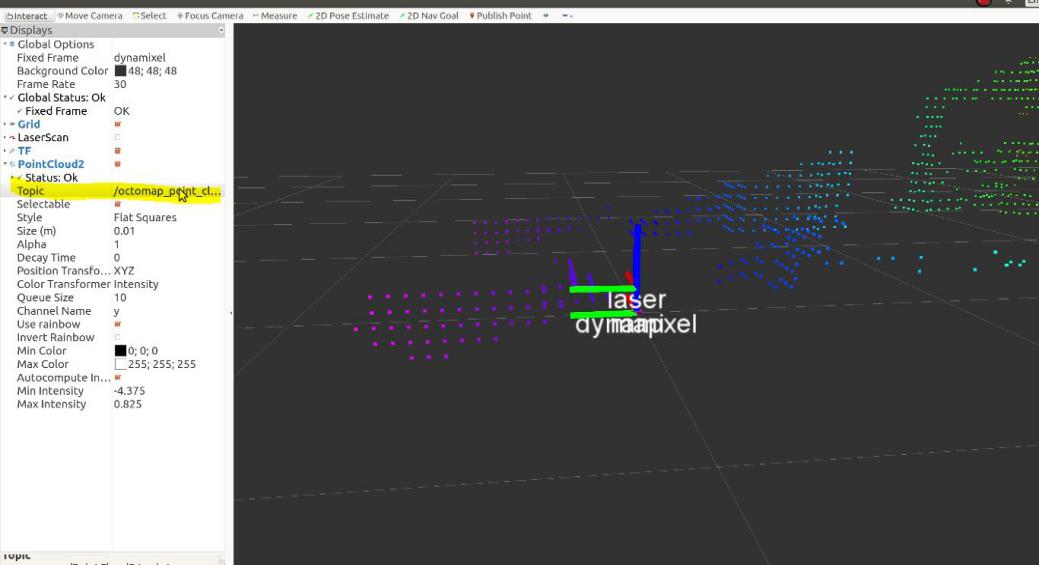


fig.17.Rviz octree servis

\*command: rosrun octomap\_server octomap\_server\_node