ELECENG 3EY4: Electrical System Integration Project

Lab07_Localization and Mapping with MacAEV

Junbo Wang – wangj430 – 400249823

Preet Batra - batrap5 - 400341704

Yichen Lu - luy191 - 400247938

Activity 1: Derive the homogenous transformations that relate the frame "base_link" to the frames "laser", "imu", and "camera", i.e., $T_{laser}^{base\_link}, T_{imu}^{base\_link}, T_{camera}^{base\_link}$.

Ans:

Activity 1

$$\because\ R = \begin{bmatrix} r_{11} & r_{12} & r_{13} & 0 \\ r_{21} & r_{22} & r_{23} & 0 \\ r_{31} & r_{32} & r_{33} & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \qquad D = \begin{bmatrix} 1 & 0 & 0 & X_0 \\ 0 & 1 & 0 & Y_0 \\ 0 & 0 & 1 & Z_0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$\therefore\ T = RD = \begin{bmatrix} r_{11} & r_{12} & r_{13} & X_0 \\ r_{21} & r_{22} & r_{23} & Y_0 \\ r_{31} & r_{32} & r_{33} & Z_0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$\therefore\ T_{imu}^{base-link} = R_{imu}^{base-link} D_{imu}^{base-link}$$

$$= \begin{bmatrix} 0 & -1 & 0 & 12.86 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 61.19 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$T_{laser}^{base-link} = R_{laser}^{base-link} D_{laser}^{base-link}$$

$$= \begin{bmatrix} -1 & 0 & 0 & 12.89 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & 1 & 76 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$T_{camera}^{base-link} = R_{camera}^{base-link} D_{camera}^{base-link}$$

$$= \begin{bmatrix} 0 & 0 & 1 & 292.03 \\ -1 & 0 & 0 & 0 \\ 0 & -1 & 0 & 86.00 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Activity 2: Complete the following ROS static transformation in the “experiment.launch” file that is provided to you on Avenue to Learn. The information given in Figs. 1&2 is needed for performing this task. Explain the purpose of these transformations.

```
<node pkg="tf2_ros" type="static_transform_publisher" name="base_link_to_imu"
      args="0.01286 0 0.06119 1.57 0 0 base_link imu" />

<node pkg="tf2_ros" type="static_transform_publisher" name="base_link_to_laser"
      args="0.01286 0 0.076 3.14 0 0 base_link laser" />
```

The answers are observed from the schematic of MacAEV. The purpose of these transformations is that we need to specify the transformation between the two coordinate systems in order to perform transformations in ROS. The translation and rotation vectors between the two frames are used to determine the transformation. After the transformation is defined, it can be added to the robot's startup file for use by other components.

Activity 3: Refer to the course lecture notes to briefly explain how the orientation of a frame is expressed in Quaternion vector.

Quaternions are utilized to denote three-dimensional orientations, and in ROS (Robot Operating System), they are employed for this purpose. When representing frame orientation in the form of a quaternion, it includes the rotation angle, which is denoted as w as well as the rotation axes, which are represented by the coordinates x, y, and z.

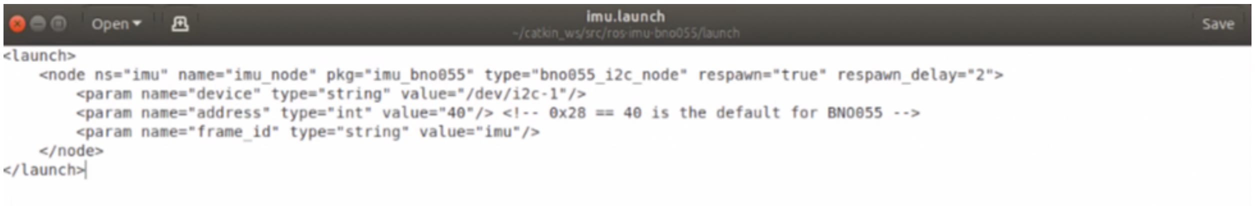
Activity 4:

For this activity, we just followed the steps in the lab manual. And the results are shown below.

```
team37@team37-desktop:~/catkin_ws/src/f1tenth_simulator/launch$ cd ~/catkin_ws/src
team37@team37-desktop:~/catkin_ws/src$ sudo git clone https://github.com/dheera/ros-imu-bno055
Cloning into 'ros-imu-bno055'...
remote: Enumerating objects: 200, done.
remote: Counting objects: 100% (41/41), done.
remote: Compressing objects: 100% (17/17), done.
remote: Total 200 (delta 30), reused 24 (delta 24), pack-reused 159
Receiving objects: 100% (200/200), 41.81 KiB | 259.00 KiB/s, done.
Resolving deltas: 100% (71/71), done.
```

```
team37@team37-desktop:~/catkin_ws/src$ cd ~/catkin_ws
team37@team37-desktop:~/catkin_ws$ catkin_make
```

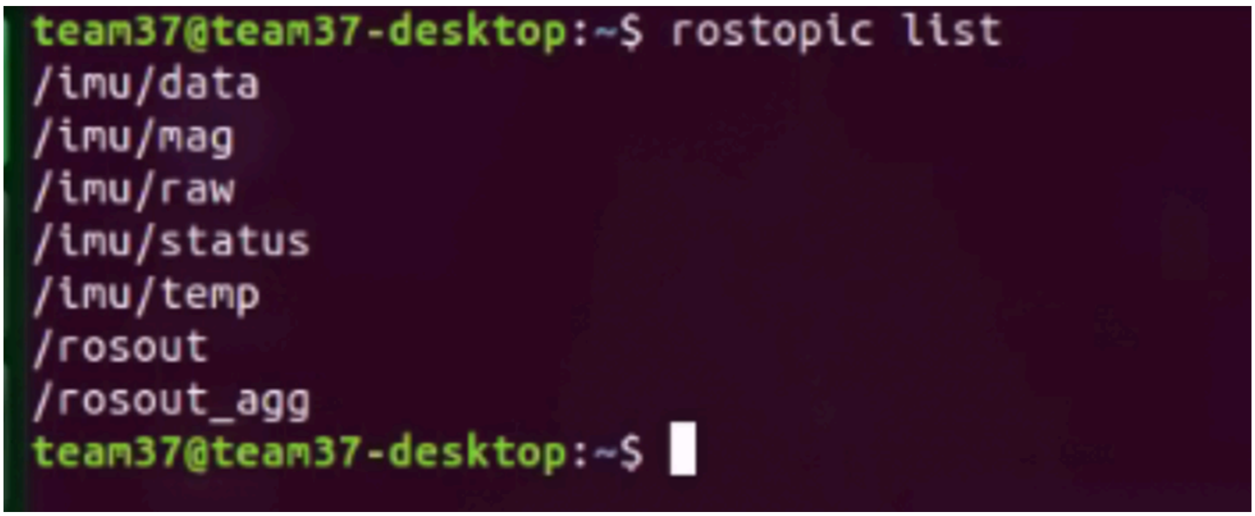
```
team37@team37-desktop:~/catkin_ws$ roscd imu_bno055/launch/
team37@team37-desktop:~/catkin_ws/src/ros-imu-bno055/launch$ sudo gedit imu.launch
```




```
<launch>
    <node ns="imu" name="imu_node" pkg="imu_bno055" type="bno055_i2c_node" respawn="true" respawn_delay="2">
        <param name="device" type="string" value="/dev/i2c-1"/>
        <param name="address" type="int" value="40"/> <!-- 0x28 == 40 is the default for BNO055 -->
        <param name="frame_id" type="string" value="imu"/>
    </node>
</launch>
```

Activity 5:Make any necessary changes to the launch file and run it from the command line. List the published ROS topics related to the IMU.

We listed all the published ROS topics related to the IMU. It is shown below.



```
team37@team37-desktop:~$ rostopic list
/imu/data
/imu/mag
/imu/raw
/imu/status
/imu/temp
/rosout
/rosout_agg
team37@team37-desktop:~$
```

Activity 6: It is usually a good idea to calibrate the IMU before utilizing its data. Follow the procedure outlined on Page 51 of the sensor data sheet in order to calibrate it. Echo the value of “Calibration status” on the terminal screen as you move the vehicle manually. What status value would indicate a fully calibrated IMU?

Calibration

Explanation:

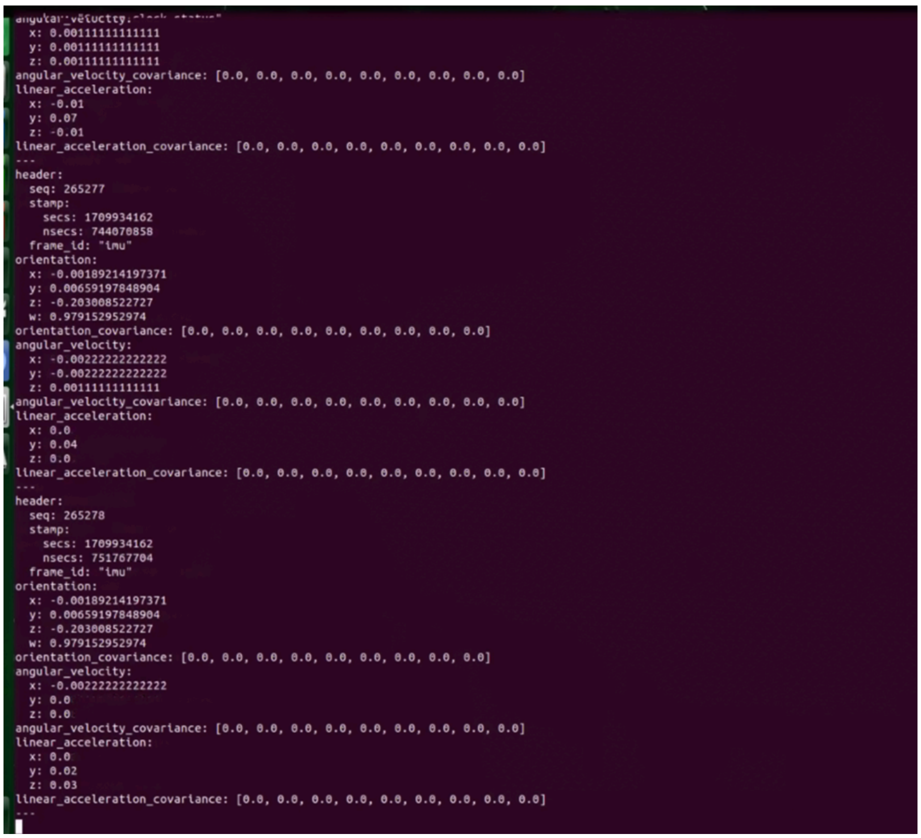
A total of eight bits are allocated for calibration purposes, with two bits assigned to each of the following components: system, gyroscope, accelerometer, and magnetometer. These bits are transformed from binary format to decimal for output purposes.

As you can see from the screenshot, when the calibration progress was showing "255" = "11111111", it means that the calibration of all components is completed.

Activity 7: Confirm that the IMU reported orientation is consistent with this convention, e.g., by aligning the y-axis of the IMU frame with the direction of the earth north and reading the value of the orientation. Recall that you can use the ROS command “rostopic” to echo the published content of a topic on the terminal screen.

For this activity, as you can see from the screenshot, we successfully aligned the y-axis of the IMU frame with the direction of the earth north and read the value of the orientation.

North

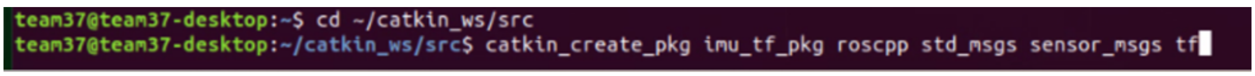


Activity 8:

For this activity, we just followed the steps in the lab manual. More specifically, we created a ROS package named **imu_tf_pkg** within our catkin workspace. Afterwards, we added the file **imu_tf_pub.cpp** as a node to your ROS package. At last, we added the imu_tf_pub.launch file to the launch folder of the package, replaced the CMakeLists.txt file in the package with the one you downloaded from avenue, and built the package.

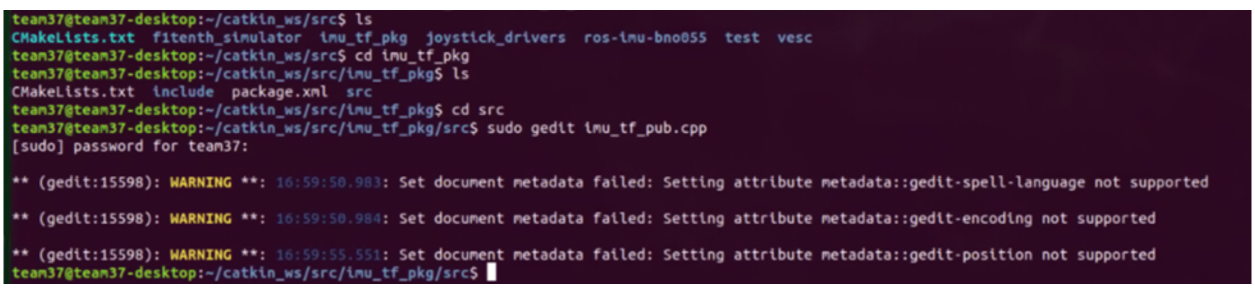
Examine this file's content and briefly explain in your report what the new node does. (the file imu_tf_pub.cpp)

This ROS node, developed in C++, subscribes to an IMU (Inertial Measurement Unit) data topic and utilizes the tf library for broadcasting transformations. Key functionalities include transforming quaternion orientation from IMU messages into ROS-compatible transforms, updating these with IMU data timestamps, and broadcasting them to maintain an updated reference of the robot's orientation in space. Parameters like frame names and IMU topics are dynamically adjustable, enhancing the node's adaptability to different robotic setups. This implementation is crucial for robotic applications requiring precise and real-time orientation tracking, such as autonomous navigation and sensor data integration.



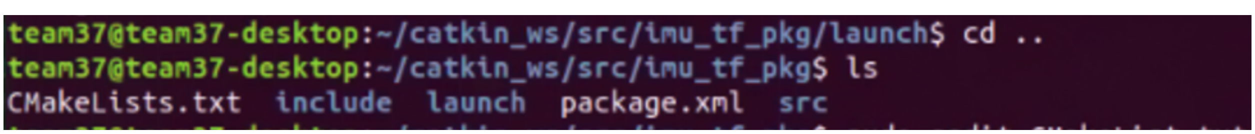
```
team37@team37-desktop:~$ cd ~/catkin_ws/src
team37@team37-desktop:~/catkin_ws/src$ catkin_create_pkg imu_tf_pkg roscpp std_msgs sensor_msgs tf
```

```
team37@team37-desktop:~$ cd ~/catkin_ws/src
team37@team37-desktop:~/catkin_ws/src$ catkin_create_pkg imu_tf_pkg roscpp std_msgs sensor_msgs tf
Created file imu_tf_pkg/package.xml
Created file imu_tf_pkg/CMakeLists.txt
Created folder imu_tf_pkg/include/imu_tf_pkg
Created folder imu_tf_pkg/src
Successfully created files in /home/team37/catkin_ws/src/imu_tf_pkg. Please adjust the values in package.xml.
```



```
team37@team37-desktop:~/catkin_ws/src$ ls
CMakeLists.txt  f1tenth_simulator  imu_tf_pkg  joystick_drivers  ros-imu-bno055  test  vesc
team37@team37-desktop:~/catkin_ws/src$ cd imu_tf_pkg
team37@team37-desktop:~/catkin_ws/src/imu_tf_pkg$ ls
CMakeLists.txt  include  package.xml  src
team37@team37-desktop:~/catkin_ws/src/imu_tf_pkg$ cd src
team37@team37-desktop:~/catkin_ws/src/imu_tf_pkg/src$ sudo gedit imu_tf_pub.cpp
[sudo] password for team37:

** (gedit:15598): WARNING **: 16:59:50.983: Set document metadata failed: Setting attribute metadata::gedit-spell-language not supported

** (gedit:15598): WARNING **: 16:59:50.984: Set document metadata failed: Setting attribute metadata::gedit-encoding not supported

** (gedit:15598): WARNING **: 16:59:55.551: Set document metadata failed: Setting attribute metadata::gedit-position not supported
team37@team37-desktop:~/catkin_ws/src/imu_tf_pkg/src$
```

```
team37@team37-desktop:~/catkin_ws/src/imu_tf_pkg/src$ cd ..
team37@team37-desktop:~/catkin_ws/src/imu_tf_pkg$ sudo mkdir launch
team37@team37-desktop:~/catkin_ws/src/imu_tf_pkg$ ls
CMakeLists.txt  include  launch  package.xml  src
team37@team37-desktop:~/catkin_ws/src/imu_tf_pkg$ cd launch
team37@team37-desktop:~/catkin_ws/src/imu_tf_pkg/launch$ sudo gedit imu_tf_pub.launch

** (gedit:15687): WARNING **: 17:03:11.680: Set document metadata failed: Setting attribute metadata::gedit-spell-language not supported

** (gedit:15687): WARNING **: 17:03:11.681: Set document metadata failed: Setting attribute metadata::gedit-encoding not supported

** (gedit:15687): WARNING **: 17:03:14.124: Set document metadata failed: Setting attribute metadata::gedit-position not supported
```



```
team37@team37-desktop:~/catkin_ws/src/imu_tf_pkg/launch$ cd ..
team37@team37-desktop:~/catkin_ws/src/imu_tf_pkg$ ls
CMakeLists.txt  include  launch  package.xml  src
```

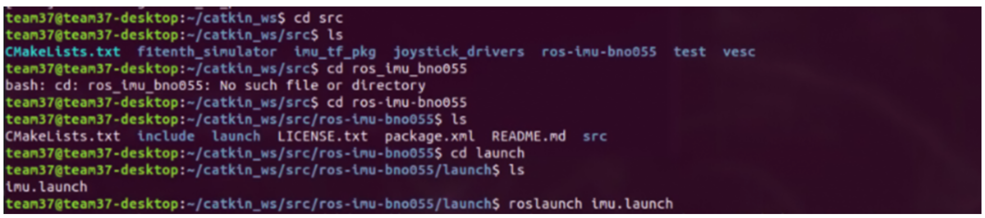
```
team37@team37-desktop:~/catkin_ws/src/imu_tf_pkg$ sudo gedit CMakeLists.txt

** (gedit:15886): WARNING **: 17:06:46.136: Set document metadata failed: Setting attribute metadata::gedit-spell-language not supported

** (gedit:15886): WARNING **: 17:06:46.137: Set document metadata failed: Setting attribute metadata::gedit-encoding not supported

** (gedit:15886): WARNING **: 17:06:47.553: Set document metadata failed: Setting attribute metadata::gedit-position not supported
```

```
team37@team37-desktop:~/catkin_ws/src/imu_tf_pkg$ cd ~/catkin_ws/src
team37@team37-desktop:~/catkin_ws/src$ catkin_make
Base path: /home/team37/catkin_ws/src
The specified source space "/home/team37/catkin_ws/src/src" does not exist
team37@team37-desktop:~/catkin_ws/src$ cd ..
team37@team37-desktop:~/catkin_ws$ catkin_make
Base path: /home/team37/catkin_ws
Source space: /home/team37/catkin_ws/src
Build space: /home/team37/catkin_ws/build
Devel space: /home/team37/catkin_ws/devel
Install space: /home/team37/catkin_ws/install
####
#### Running command: "cmake /home/team37/catkin_ws/src -DCATKIN_DEVEL_PREFIX=/home/team37/catkin_ws/devel -DCMAKE_INSTALL_PREFIX=/home/team37/catkin_ws/install -G Unix Makefiles" in "/home/team37/catkin_
ws/build"
####
-- Using CATKIN_DEVEL_PREFIX: /home/team37/catkin_ws/devel
-- Using CMAKE_PREFIX_PATH: /home/team37/catkin_ws/devel;/opt/ros/melodic
-- This workspace overlays: /home/team37/catkin_ws/devel;/opt/ros/melodic
-- Found PythonInterp: /usr/bin/python2 (found suitable version "2.7.17", minimum required is "2")
-- Using PYTHON_EXECUTABLE: /usr/bin/python2
-- Using Debian Python package layout
-- Using empy: /usr/bin/empy
-- Using CATKIN_ENABLE_TESTING: ON
-- Call enable_testing()
-- Using CATKIN_TEST_RESULTS_DIR: /home/team37/catkin_ws/build/test_results
-- Found gtest sources under '/usr/src/googletest': gtests will be built
-- Found gmock sources under '/usr/src/googletest': gmock will be built
-- Found PythonInterp: /usr/bin/python2 (found version "2.7.17")
-- Using Python nosetests: /usr/bin/nosetests-2.7
-- catkin 0.7.29
-- BUILD_SHARED_LIBS is on
-- BUILD_SHARED_LIBS is on
-- ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
-- ~~  traversing 11 packages in topological order:
-- ~~  - joystick_drivers (metapackage)
-- ~~  - vesc (metapackage)
-- ~~  - vesc_msgs
-- ~~  - imu_bno055
-- ~~  - test
-- ~~  - joy
-- ~~  - spacenav_node
-- ~~  - imu_tf_pkg
-- ~~  - f1tenth_simulator
-- ~~  - vesc_ackermann
-- ~~  - vesc_driver
-- ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
-- +++ processing catkin metapackage: 'joystick_drivers'
-- ==> add_subdirectory(joystick_drivers/joystick_drivers)
```

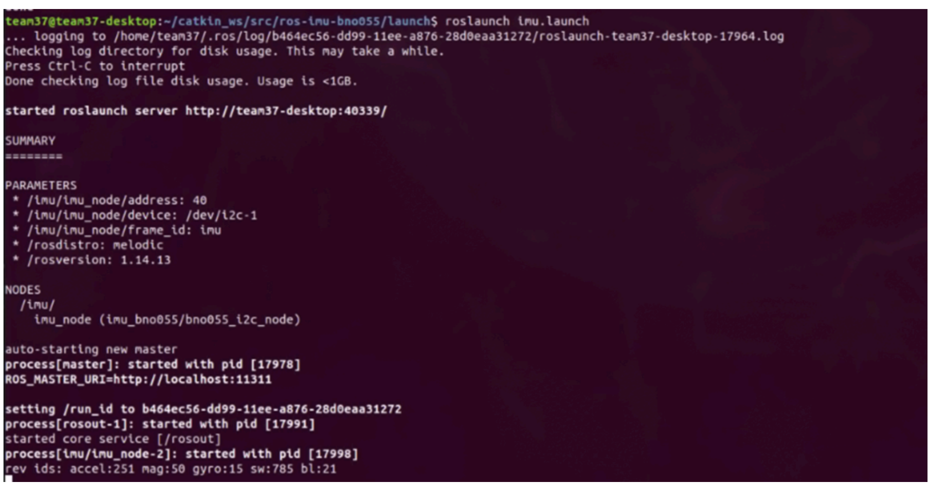
```
[ 23%] Built target tf_generate_messages_eus
Scanning dependencies of target actionlib_generate_messages_lisp
[ 23%] Built target actionlib_generate_messages_eus
Scanning dependencies of target tf_generate_messages_py
Scanning dependencies of target actionlib_msgs_generate_messages_nodejs
[ 23%] Built target actionlib_generate_messages_lisp
[ 23%] Built target tf_generate_messages_py
Scanning dependencies of target actionlib_generate_messages_nodejs
[ 23%] Built target actionlib_msgs_generate_messages_nodejs
Scanning dependencies of target actionlib_generate_messages_py
[ 23%] Built target actionlib_generate_messages_nodejs
Scanning dependencies of target actionlib_msgs_generate_messages_eus
[ 23%] Built target actionlib_generate_messages_py
Scanning dependencies of target actionlib_msgs_generate_messages_lisp
Scanning dependencies of target tf2_msgs_generate_messages_cpp
[ 23%] Built target actionlib_msgs_generate_messages_eus
[ 23%] Built target actionlib_msgs_generate_messages_lisp
[ 23%] Built target tf2_msgs_generate_messages_cpp
[ 23%] Built target ackermann_msgs_generate_messages_eus
[ 23%] Built target ackermann_msgs_generate_messages_lisp
[ 23%] Built target ackermann_msgs_generate_messages_nodejs
[ 23%] Built target ackermann_msgs_generate_messages_py
[ 23%] Built target ackermann_msgs_generate_messages_cpp
[ 23%] Built target bond_generate_messages_cpp
[ 23%] Built target nodelet_generate_messages_nodejs
[ 23%] Built target nodelet_generate_messages_cpp
[ 23%] Built target bond_generate_messages_eus
[ 23%] Built target nodelet_generate_messages_eus
[ 23%] Built target bond_generate_messages_lisp
[ 23%] Built target nodelet_generate_messages_py
[ 23%] Built target bond_generate_messages_nodejs
[ 23%] Built target bond_generate_messages_py
[ 26%] Built target vesc_msgs_generate_messages_nodejs
[ 31%] Built target vesc_msgs_generate_messages_py
[ 34%] Built target vesc_msgs_generate_messages_cpp
[ 39%] Built target vesc_msgs_generate_messages_eus
[ 42%] Built target vesc_msgs_generate_messages_lisp
[ 42%] Built target f1tenth_simulator_generate_messages_cpp
[ 42%] Built target f1tenth_simulator_generate_messages_nodejs
[ 43%] Built target f1tenth_simulator_generate_messages_eus
[ 43%] Built target f1tenth_simulator_generate_messages_lisp
[ 43%] Built target f1tenth_simulator_generate_messages_py
[ 46%] Built target random_walk
[ 50%] Built target experiment
[ 54%] Built target vesc_to_odom_node
[ 62%] Built target vesc_ackermann_nodelet
[ 67%] Built target ackermann_to_vesc_node
[ 76%] Built target vesc_driver_node
[ 85%] Built target vesc_driver_nodelet
[ 85%] Built target vesc_msgs_generate_messages
[ 89%] Built target mux
[ 92%] Built target keyboard
[ 95%] Built target behavior_controller
[ 95%] Built target f1tenth_simulator_generate_messages
[ 98%] Built target simulator
[100%] Linking CXX executable /home/team37/catkin_ws/devel/lib/imu_tf_pkg/imu_tf_pub
[100%] Built target imu_tf_pub
team37@team37-desktop:~/catkin_ws$
```

Activity 9:

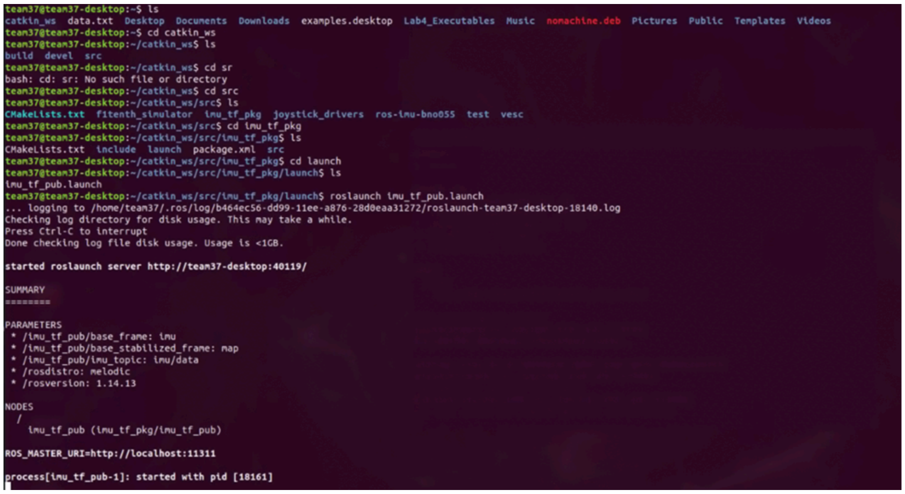
For this activity, we launched the new node and confirmed that a transformation between the fixed frame “map” and the IMU frame “imu” is published over the topic “/tf”. Afterwards, we launched the ROS visualization tool “rviz” and added TF displays from the left windowpane. And then we setted the fixed frame to “map”. At last, we manually moved the vehicle and noted the relative alignment of the two frames “map” and “imu”. All the results and steps are shown below.



```
team37@team37-desktop:~/catkin_ws$ cd src
team37@team37-desktop:~/catkin_ws/src$ ls
CMakeLists.txt  f1tenth_simulator  imu_tf_pkg  joystick_drivers  ros-imu-bno055  test  vesc
team37@team37-desktop:~/catkin_ws/src$ cd ros_imu_bno055
bash: cd: ros_imu_bno055: No such file or directory
team37@team37-desktop:~/catkin_ws/src$ cd ros-imu-bno055
team37@team37-desktop:~/catkin_ws/src/ros-imu-bno055$ ls
CMakeLists.txt  include  launch  LICENSE.txt  package.xml  README.md  src
team37@team37-desktop:~/catkin_ws/src/ros-imu-bno055$ cd launch
team37@team37-desktop:~/catkin_ws/src/ros-imu-bno055/launch$ ls
imu.launch
team37@team37-desktop:~/catkin_ws/src/ros-imu-bno055/launch$ roslaunch imu.launch
```

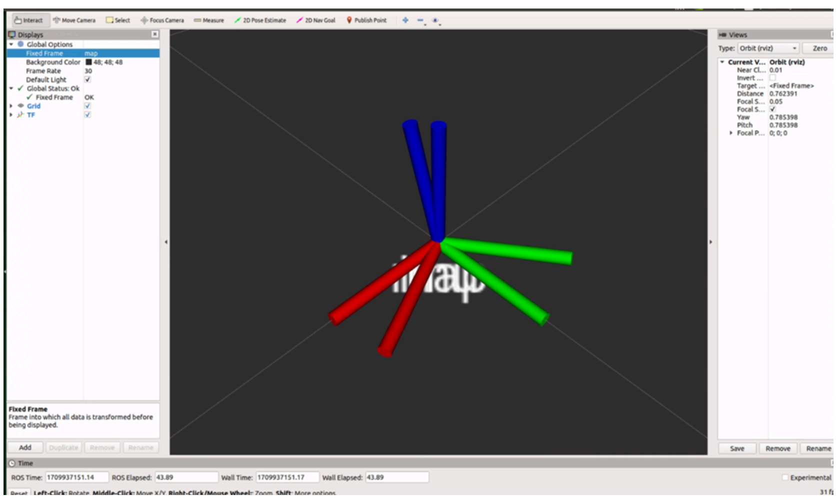


```
team37@team37-desktop:~/catkin_ws/src/ros-imu-bno055/launch$ roslaunch imu.launch
... logging to /home/team37/.ros/log/b464ec56-dd99-11ee-a876-28d0eaa31272/roslaunch-team37-desktop-17964.log
Checking log directory for disk usage. This may take a while.
Press Ctrl-C to interrupt
Done checking log file disk usage. Usage is <1GB.

started roslaunch server http://team37-desktop:40339/

SUMMARY
========

PARAMETERS
 * /imu/imu_node/address: 40
 * /imu/imu_node/device: /dev/i2c-1
 * /imu/imu_node/frame_id: imu
 * /rosdistro: melodic
 * /rosversion: 1.14.13

NODES
  /imu/
    imu_node (imu_bno055/bno055_i2c_node)

auto-starting new master
process[master]: started with pid [17978]
ROS_MASTER_URI=http://localhost:11311

setting /run_id to b464ec56-dd99-11ee-a876-28d0eaa31272
process[rosout-1]: started with pid [17991]
started core service [/rosout]
process[imu/imu_node-2]: started with pid [17998]
rev ids: accel:251 mag:50 gyro:15 sw:785 bl:21
```



```
team37@team37-desktop:~$ ls
catkin_ws  data.txt  Desktop  Documents  Downloads  examples.desktop  Lab4_Executables  Music  nomachine.deb  Pictures  Public  Templates  Videos
team37@team37-desktop:~$ cd catkin_ws
team37@team37-desktop:~/catkin_ws$ ls
build  devel  src
team37@team37-desktop:~/catkin_ws$ cd sr
bash: cd: sr: No such file or directory
team37@team37-desktop:~/catkin_ws$ cd src
team37@team37-desktop:~/catkin_ws/src$ ls
CMakeLists.txt  f1tenth_simulator  imu_tf_pkg  joystick_drivers  ros-imu-bno055  test  vesc
team37@team37-desktop:~/catkin_ws/src$ cd imu_tf_pkg
team37@team37-desktop:~/catkin_ws/src/imu_tf_pkg$ ls
CMakeLists.txt  include  launch  package.xml  src
team37@team37-desktop:~/catkin_ws/src/imu_tf_pkg$ cd launch
team37@team37-desktop:~/catkin_ws/src/imu_tf_pkg/launch$ ls
imu_tf_pub.launch
team37@team37-desktop:~/catkin_ws/src/imu_tf_pkg/launch$ roslaunch imu_tf_pub.launch
... logging to /home/team37/.ros/log/b464ec56-dd99-11ee-a876-28d0eaa31272/roslaunch-team37-desktop-18140.log
Checking log directory for disk usage. This may take a while.
Press Ctrl-C to interrupt
Done checking log file disk usage. Usage is <1GB.

started roslaunch server http://team37-desktop:40119/

SUMMARY
========

PARAMETERS
 * /imu_tf_pub/base_frame: imu
 * /imu_tf_pub/base_stabilized_frame: map
 * /imu_tf_pub/imu_topic: imu/data
 * /rosdistro: melodic
 * /rosversion: 1.14.13

NODES
  /
    imu_tf_pub (imu_tf_pkg/imu_tf_pub)

ROS_MASTER_URI=http://localhost:11311

process[imu_tf_pub-1]: started with pid [18161]
```

```
team37@team37-desktop:~$ rosrun rviz rviz
[ INFO] [1709937105.773419406]: rviz version 1.13.30
[ INFO] [1709937105.773553307]: compiled against Qt version 5.9.5
[ INFO] [1709937105.773596014]: compiled against OGRE version 1.9.0 (Ghadamon)
[ INFO] [1709937105.789080734]: Forcing OpenGl version 0.
[ INFO] [1709937106.731756882]: Stereo is NOT SUPPORTED
[ INFO] [1709937106.731978646]: OpenGL device: NVIDIA Tegra X1 (nvgpu)/integrated
[ INFO] [1709937106.732085465]: OpenGl version: 4.6 (GLSL 4.6).
```



Activity 10:

```
        double current_steering_angle, current_angular_velocity;
        current_steering_angle =
            ( last_servo_state.data - steering_angle_to_servo_offset) / steering_angle_to_servo_gain;

        // following lines must be completed
        current_angular_velocity = current_speed * tan(current_steering_angle) / wheelbase;


        // calc elapsed time
        ros::Duration dt = state.header.stamp - last_vesc_state.header.stamp;

       // save state for next time
        last_vesc_state = state;


// following lines must be completed

        // odometry information
        double x_dot = current_speed * cos(yaw);
        double y_dot = current_speed * sin(yaw);
        x_odom+= x_dot * dt.toSec();
        y_odom+= y_dot * dt.toSec();
        yaw+= current_angular_velocity * dt.toSec();


        // Publish odometry message
        odom.header.stamp = state.header.stamp;

        // following lines must be completed
        // Position
        odom.pose.pose.position.x = x_odom;
        odom.pose.pose.position.y = y_odom;
        odom.pose.pose.orientation.x = 0.0;
        odom.pose.pose.orientation.y = 0.0;
        odom.pose.pose.orientation.z = sin(yaw/2.0);
        odom.pose.pose.orientation.w = cos(yaw/2.0);

        // Velocity
        odom.twist.twist.linear.x = current_speed;
        odom.twist.twist.linear.y = 0.0;
        odom.twist.twist.linear.z = 0;
        odom.twist.twist.angular.z = current_angular_velocity;

        odom_pub.publish(odom);

        // Publish odometry tranformation
        odom_transform.transform.translation.x= x_odom;
        odom_transform.transform.translation.y= y_odom;
        odom_transform.transform.translation.z= 0;
        geometry_msgs::Quaternion odom_quat = tf::createQuaternionMsgFromYaw(yaw);
        odom_transform.transform.rotation = odom_quat;

        odom_transform.header.stamp=state.header.stamp;
```

We complete the empty line by using three first differential equations for vehicle motions, wheel odometry localization integrals, and the Euler’s integration method.

first differential equations

$$
\begin{aligned}
\dot{x} &= v_s \cos\theta \\
\dot{y} &= v_s \sin\theta \\
\dot{\theta} &= \frac{v_s}{l}\tan\delta
\end{aligned}
$$

wheel odometry localization integrals

$$x(t) = \int_0^t v_s(\tau)\cos\big(\theta(\tau)\big)\,d\tau$$

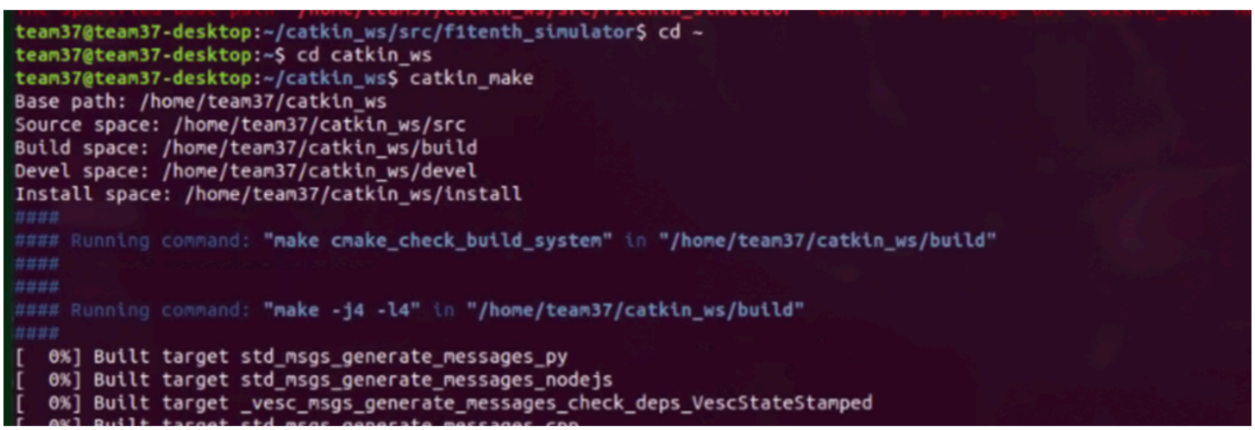
$$y(t) = \int_0^t v_s(\tau)\sin\big(\theta(\tau)\big)d\tau$$

$$\theta(t) = \int_0^t \frac{v_s(\tau)}{l}\tan\big(\delta(\tau)\big)\,d\tau$$

Euler's integration method

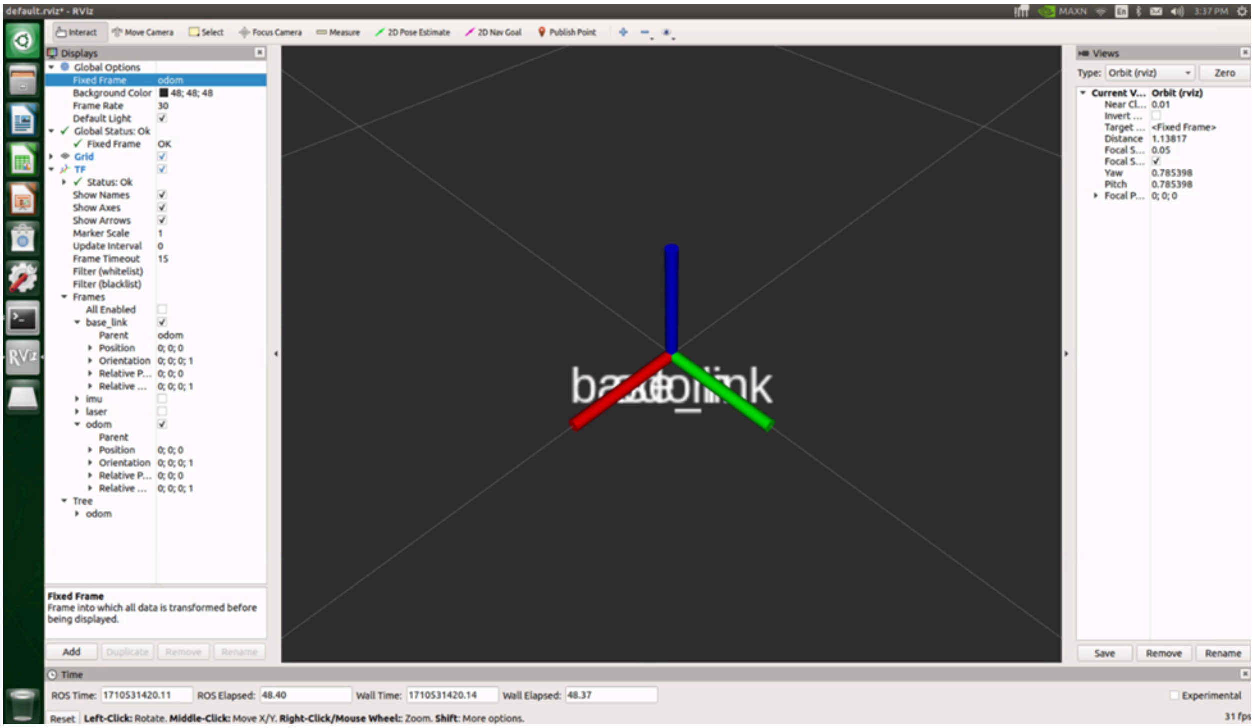
$$x_{k+1} = x_k + \delta t_k v_{s_k} cos\ (\theta_k)$$
$$y_{k+1} = y_k + \delta t_k v_{s_k}\, sin(\theta_k)$$
$$\theta_{k+1} = \theta_k + \delta t_k \frac{v_{s_k}}{l} tan\ (\delta_k)$$
$$x_0 = 0, \qquad y_0 = 0, \qquad \theta_0 = 0$$

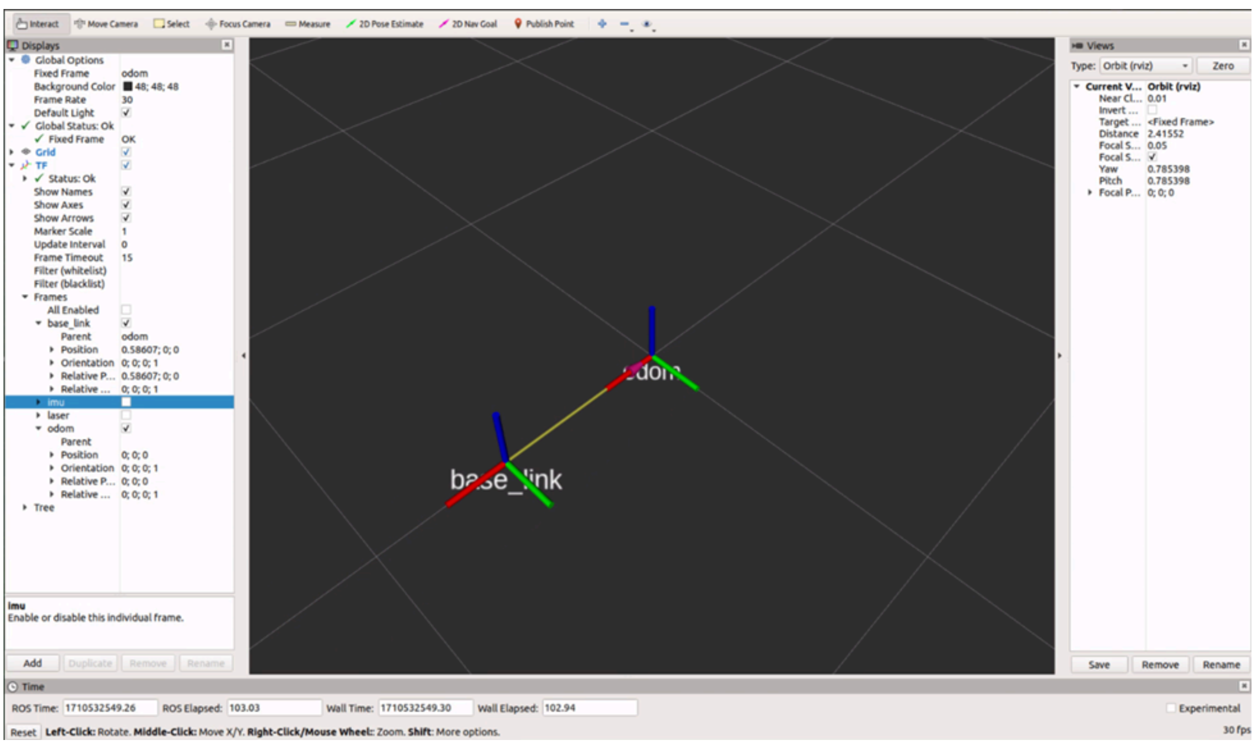
Activity 11:



```
team37@team37-desktop:~/catkin_ws/src/f1tenth_simulator$ cd ~
team37@team37-desktop:~$ cd catkin_ws
team37@team37-desktop:~/catkin_ws$ catkin_make
Base path: /home/team37/catkin_ws
Source space: /home/team37/catkin_ws/src
Build space: /home/team37/catkin_ws/build
Devel space: /home/team37/catkin_ws/devel
Install space: /home/team37/catkin_ws/install
####
#### Running command: "make cmake_check_build_system" in "/home/team37/catkin_ws/build"
####
####
#### Running command: "make -j4 -l4" in "/home/team37/catkin_ws/build"
####
[  0%] Built target std_msgs_generate_messages_py
[  0%] Built target std_msgs_generate_messages_nodejs
[  0%] Built target _vesc_msgs_generate_messages_check_deps_VescStateStamped
```

Activity 12:


default.rviz* - RViz
Interact
Move Camera
Select
Focus Camera
Measure
2D Pose Estimate
2D Nav Goal
Publish Point
Displays
Global Options
Fixed Frame
odom
Background Color
48; 48; 48
Frame Rate
30
Default Light
Global Status: Ok
Fixed Frame
OK
Grid
TF
Status: Ok
Show Names
Show Axes
Show Arrows
Marker Scale
1
Update Interval
0
Frame Timeout
15
Filter (whitelist)
Filter (blacklist)
Frames
All Enabled
base_link
Parent
odom
Position
0; 0; 0
Orientation
0; 0; 0; 1
imu
laser
odom
Tree
Fixed Frame
Frame into which all data is transformed before being displayed.
Add
Duplicate
Remove
Rename
Views
Type: Orbit (rviz)
Zero
Current V... Orbit (rviz)
Near Cl... 0.01
Target ... <Fixed Frame>
Distance 1.13817
Focal S... 0.05
Yaw 0.785398
Pitch 0.785398
Focal P... 0; 0; 0
Save
Remove
Rename
Time
ROS Time: 1710531420.11
ROS Elapsed: 48.40
Wall Time: 1710531420.14
Wall Elapsed: 48.37
Experimental
Reset
Left-Click: Rotate. Middle-Click: Move X/Y. Right-Click/Mouse Wheel: Zoom. Shift: More options.
31 fps

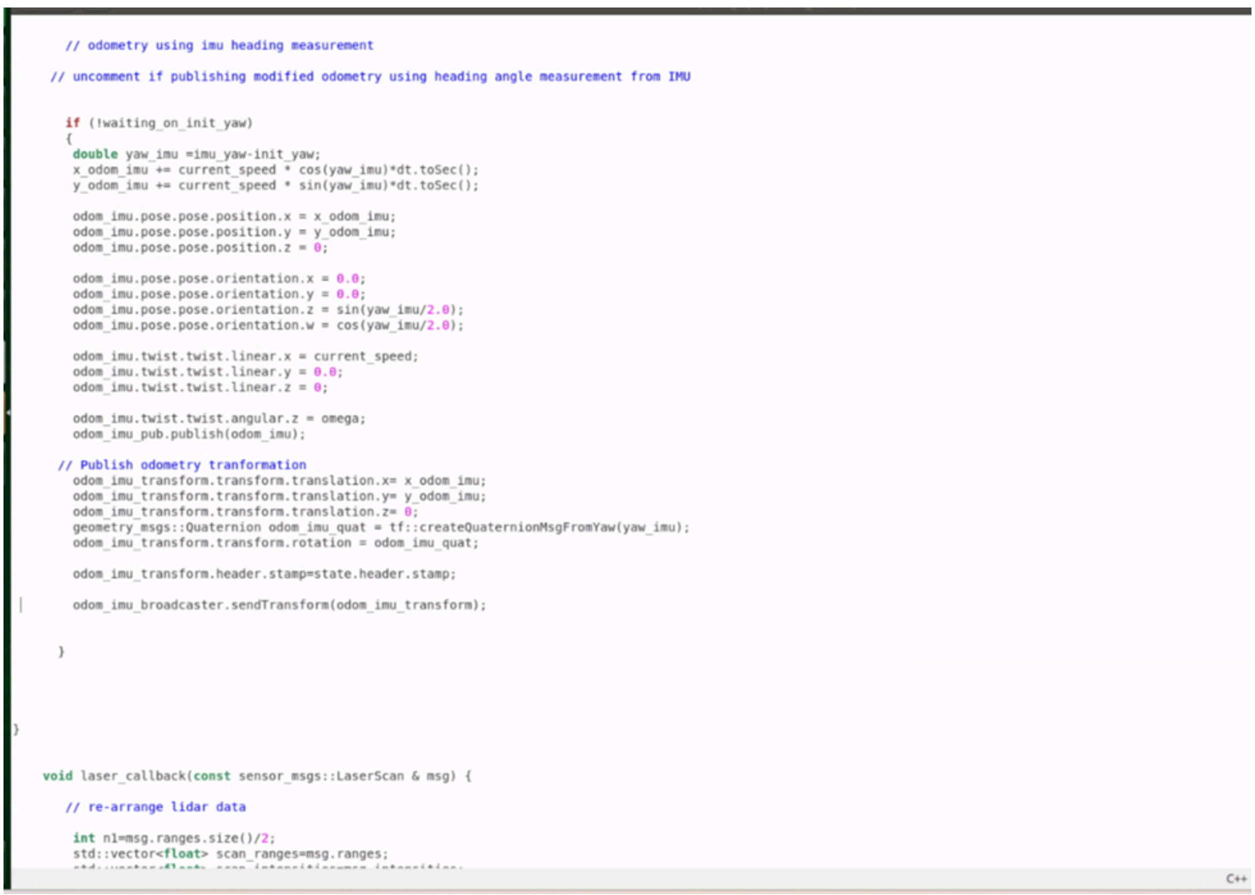

Interact
Move Camera
Select
Focus Camera
Measure
2D Pose Estimate
2D Nav Goal
Publish Point
Displays
Global Options
Fixed Frame
odom
Background Color
48; 48; 48
Frame Rate
30
Default Light
Global Status: Ok
Fixed Frame
OK
Grid
TF
Status: Ok
Show Names
Show Axes
Show Arrows
Marker Scale
1
Update Interval
0
Frame Timeout
15
Filter (whitelist)
Filter (blacklist)
Frames
All Enabled
base_link
Parent
odom
Position
0.58607; 0; 0
Orientation
0; 0; 0; 1
imu
laser
odom
Tree
odom
base_link
imu
Enable or disable this individual frame.
Add
Duplicate
Remove
Rename
Views
Type: Orbit (rviz)
Zero
Current V... Orbit (rviz)
Near Cl... 0.01
Target ... <Fixed Frame>
Distance 2.41552
Focal S... 0.05
Yaw 0.785398
Pitch 0.785398
Focal P... 0; 0; 0
Save
Remove
Rename
Time
ROS Time: 1710532549.26
ROS Elapsed: 103.03
Wall Time: 1710532549.30
Wall Elapsed: 102.94
Experimental
Reset
Left-Click: Rotate. Middle-Click: Move X/Y. Right-Click/Mouse Wheel: Zoom. Shift: More options.
30 fps

Activity 13: Do you observe any errors in the wheel odometry-based localization in your experiment? Explain. Discuss potential sources of errors in wheel-based odometry computations that may cause a drift in the computed position and orientation of the vehicle.

According to the image in activity 12, the vehicle is initially aligned to the Odom frame. When we moved the car a distance of 3 floor tiles and then drove it back to the starting point, it caused a deviation at rviz.

When wheels slip due to lack of traction, uneven floor, or sudden changes in motion, encoder readings can become inaccurate, causing deviations in travel distance. Also, wheel behavior can deviate from linearity due to factors such as springiness or hysteresis, resulting in non-linear encoder readings and further distorting distance estimates. Moreover, movement on uneven surfaces may cause the distance traveled by each wheel to vary, resulting in inaccurate estimates of vehicle position and orientation.

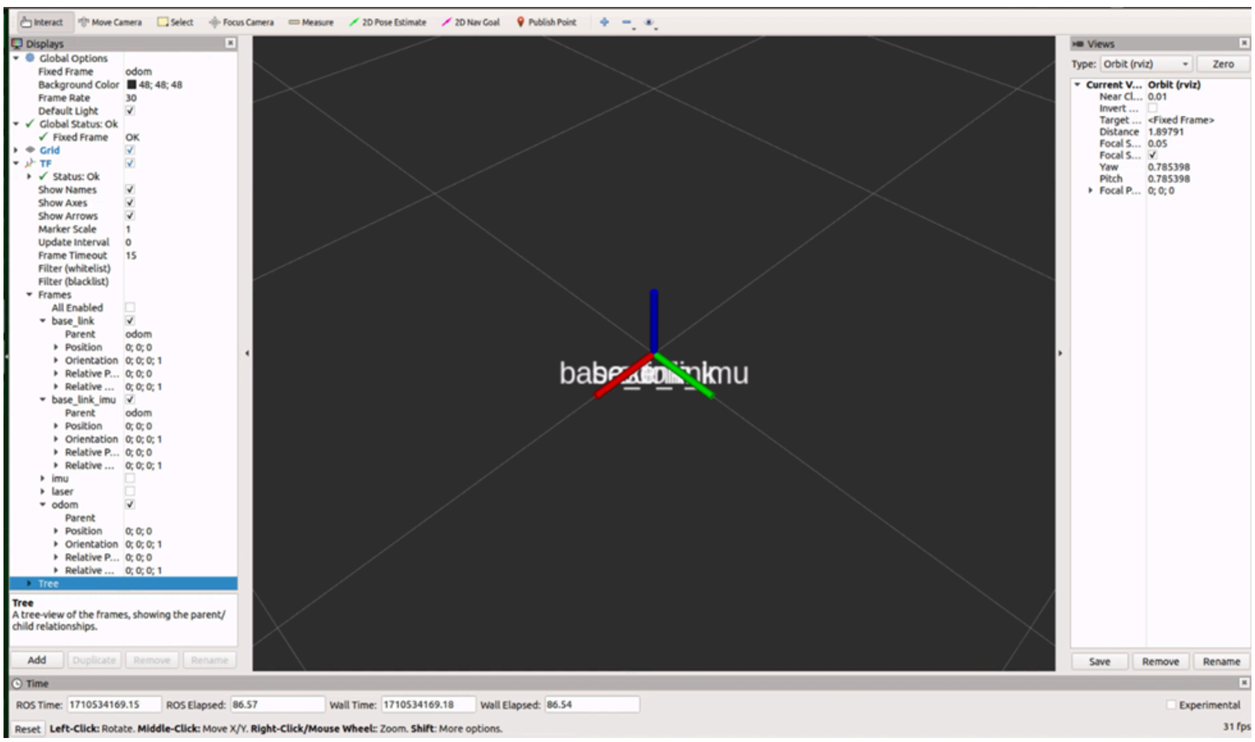
Activity 14 & 15:

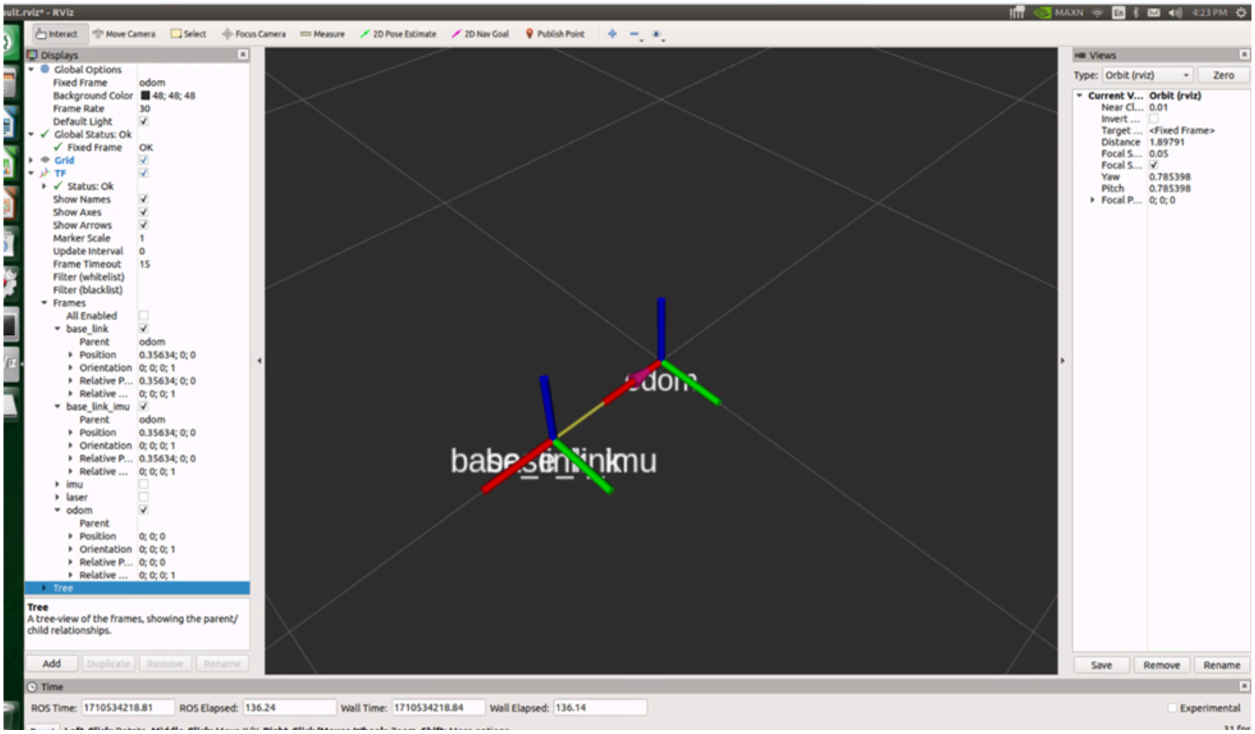


```cpp
    // odometry using imu heading measurement

  // uncomment if publishing modified odometry using heading angle measurement from IMU

    if (!waiting_on_init_yaw)
    {
     double yaw_imu =imu_yaw-init_yaw;
     x_odom_imu += current_speed * cos(yaw_imu)*dt.toSec();
     y_odom_imu += current_speed * sin(yaw_imu)*dt.toSec();

     odom_imu.pose.pose.position.x = x_odom_imu;
     odom_imu.pose.pose.position.y = y_odom_imu;
     odom_imu.pose.pose.position.z = 0;

     odom_imu.pose.pose.orientation.x = 0.0;
     odom_imu.pose.pose.orientation.y = 0.0;
     odom_imu.pose.pose.orientation.z = sin(yaw_imu/2.0);
     odom_imu.pose.pose.orientation.w = cos(yaw_imu/2.0);

     odom_imu.twist.twist.linear.x = current_speed;
     odom_imu.twist.twist.linear.y = 0.0;
     odom_imu.twist.twist.linear.z = 0;

     odom_imu.twist.twist.angular.z = omega;
     odom_imu_pub.publish(odom_imu);

   // Publish odometry tranformation
     odom_imu_transform.transform.translation.x= x_odom_imu;
     odom_imu_transform.transform.translation.y= y_odom_imu;
     odom_imu_transform.transform.translation.z= 0;
     geometry_msgs::Quaternion odom_imu_quat = tf::createQuaternionMsgFromYaw(yaw_imu);
     odom_imu_transform.transform.rotation = odom_imu_quat;

     odom_imu_transform.header.stamp=state.header.stamp;

     odom_imu_broadcaster.sendTransform(odom_imu_transform);


   }



}


  void laser_callback(const sensor_msgs::LaserScan & msg) {

    // re-arrange lidar data

     int n1=msg.ranges.size()/2;
     std::vector<float> scan_ranges=msg.ranges;
```

We uncomment this code by using the IMU yaw angle measurement.

Activity 16: Compare the results of vehicle localization with and without using the IMU yaw angle measurement. This can be observed by comparing the frames “base_link” and “base_link_imu” in rviz visualization tool, while you are driving the vehicle around. You should set the rviz fixed frame to “odom”. As in the previous case, you should drive back the vehicle to its position/orientation and observe any accumulated error in localization. Explain how the code in “experiment.cpp” handles the difference in the frame of reference for measuring the yaw angle between IMU (E-N-U frame) and the wheel odometry “odom”. Do you still expect to see localization error accumulation even after using the IMU data? Why?





In the “experiment.cpp” file, the code handles the difference by converting the IMU yaw angle measurement. The IMU's yaw angle is converted using the function "tf::createQuaternionMsgFromYaw(yaw_imu)", which correctly aligns the IMU data with the vehicle Odom frame by adjusting for different starting directions. However, according to our images, there is still localization error accumulation after using IMU angle measurements. Since the IMU drifts over time, the orientation estimate may contain inaccuracies. At the same time, the IMU may be sensitive to external factors such as vibration, which will affect accurate direction estimation.

Activity 17:

We uncomment this part in the file “experiment.launch”:

```
<!-- launch laser scan matcher -->


  <node pkg="laser_scan_matcher" type="laser_scan_matcher_node"
    name="laser_scan_matcher_node" output="screen">
    <param name="fixed_frame" value = "odom"/>
    <param name="max_iterations" value="10"/>
    <param name="base_frame" value="base_link"/>
    <param name="use_imu" value="true"/>
    <param name="use_odom" value="true"/>
    <param name="kf_dist_linear" value="0.1"/>
    <param name="kf_dist_angular" value="0.175"/>
  </node>
```

Activity 18: Explain the roles of the parameters “use_odom”, “use_imu”, “kf_dist_linear” and “kf_dist_angular” in the Scan Matcher algorithm. You can consult the course lecture materials and the documentation and resources for the algorithm.

“use_odom”: This parameter determines whether odometry data is used as part of the positioning process. When set to "true", the algorithm integrates odometry data to refine the vehicle's position and orientation estimates, improving accuracy by adding motion data.

“use_imu”: It determines whether the algorithm should use data from an inertial measurement unit (IMU). If set to "true", the algorithm will utilize IMU data to enhance attitude correction.

“kf_dist_linear”: It sets a distance threshold that determines how far the vehicle moves linearly before the algorithm updates its position estimation.

“kf_dist_angular”: This parameter sets the angle threshold in radians, specifying the angular rotation required before the direction, frequency, and accuracy are updated.

Activity 19:

```
team37@team37-desktop:~/catkin_ws$ cd src
team37@team37-desktop:~/catkin_ws/src$ ls
CMakeLists.txt  f1tenth_simulator  imu_tf_pkg  joystick_drivers  ros-imu-bno055  scan_tools  test  vesc
team37@team37-desktop:~/catkin_ws/src$ cd scan_tools
team37@team37-desktop:~/catkin_ws/src/scan_tools$ ls
laser_ortho_projector  laser_scan_matcher  laser_scan_sparsifier  laser_scan_splitter  ncd_parser  polar_scan_matcher  README.md  scan_to_cloud_converter  scan_tools
team37@team37-desktop:~/catkin_ws/src/scan_tools$ cd laser_scan_matcher
team37@team37-desktop:~/catkin_ws/src/scan_tools/laser_scan_matcher$ ls
CHANGELOG.rst  CMakeLists.txt  demo  include  laser_scan_matcher_nodelet.xml  package.xml  src  test
team37@team37-desktop:~/catkin_ws/src/scan_tools/laser_scan_matcher$ ls
CHANGELOG.rst  CMakeLists.txt  demo  include  laser_scan_matcher_nodelet.xml  package.xml  src  test
team37@team37-desktop:~/catkin_ws/src/scan_tools/laser_scan_matcher$
team37@team37-desktop:~/catkin_ws/src/scan_tools/laser_scan_matcher$ cd src
team37@team37-desktop:~/catkin_ws/src/scan_tools/laser_scan_matcher/src$ ls
laser_scan_matcher.cpp  laser_scan_matcher_node.cpp  laser_scan_matcher_nodelet.cpp
team37@team37-desktop:~/catkin_ws/src/scan_tools/laser_scan_matcher/src$ sudo gedit laser_scan_matcher.cpp
```

```
// the correction of the base's position, in the base frame
corr_ch = base_to_laser_ * corr_ch_l * laser_to_base_;

// update the pose in the world frame
f2b_ = f2b_kf_ * corr_ch;

// **** publish
```

Activity 20:

```
// **** estimated change since last scan

double dt = (time - last_icp_time_).toSec();
double pr_ch_x, pr_ch_y, pr_ch_a;
getPrediction(pr_ch_x, pr_ch_y, pr_ch_a, dt);

// the predicted change of the laser's position, in the fixed frame

tf::Transform pr_ch;
createTfFromXYTheta(pr_ch_x, pr_ch_y, pr_ch_a, pr_ch);

// account for the change since the last kf, in the fixed frame

pr_ch =  (f2b_ * pr_ch *f2b_kf_.inverse());

// the predicted change of the laser's position, in the laser frame

tf::Transform pr_ch_l;
pr_ch_l = laser_to_base_ * pr_ch * base_to_laser_ ;

input_.first_guess[0] = pr_ch_l.getOrigin().getX();
input_.first_guess[1] = pr_ch_l.getOrigin().getY();
input_.first_guess[2] = tf::getYaw(pr_ch_l.getRotation());
```

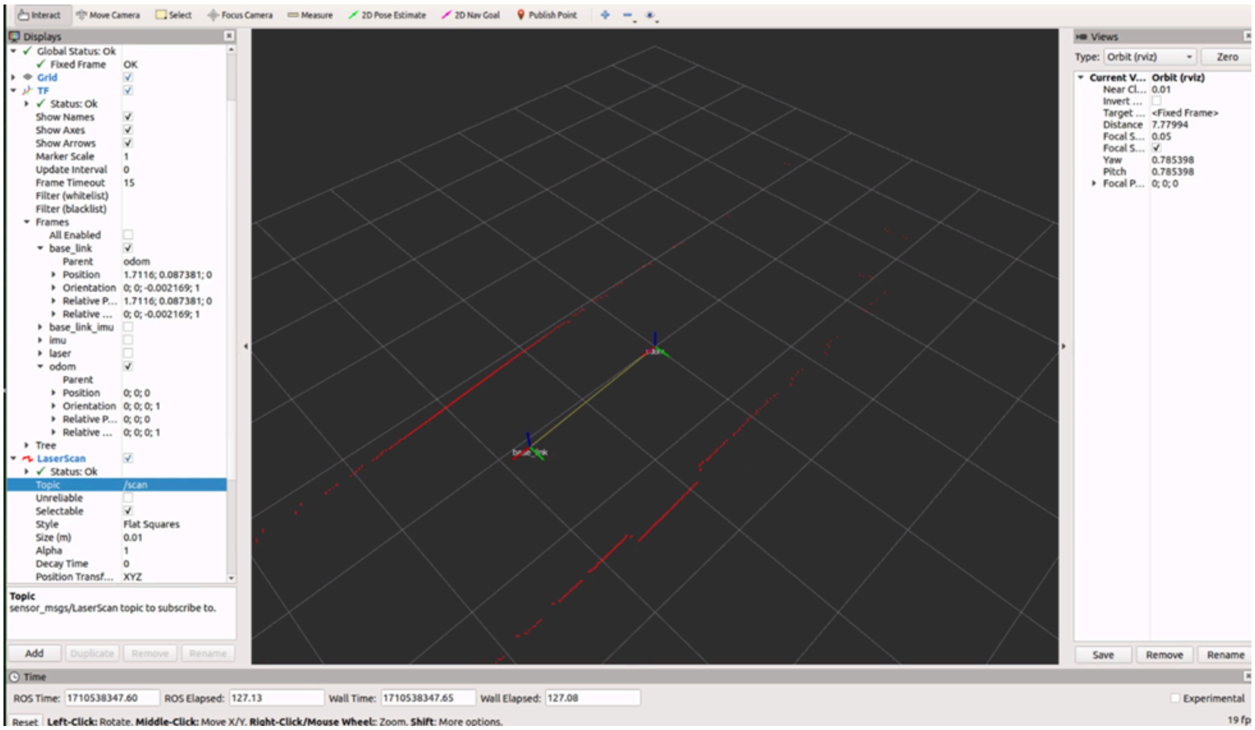
Code Correction: pr_ch = (f2b_ * pr_ch * f2b_kf_.inverse()) ;

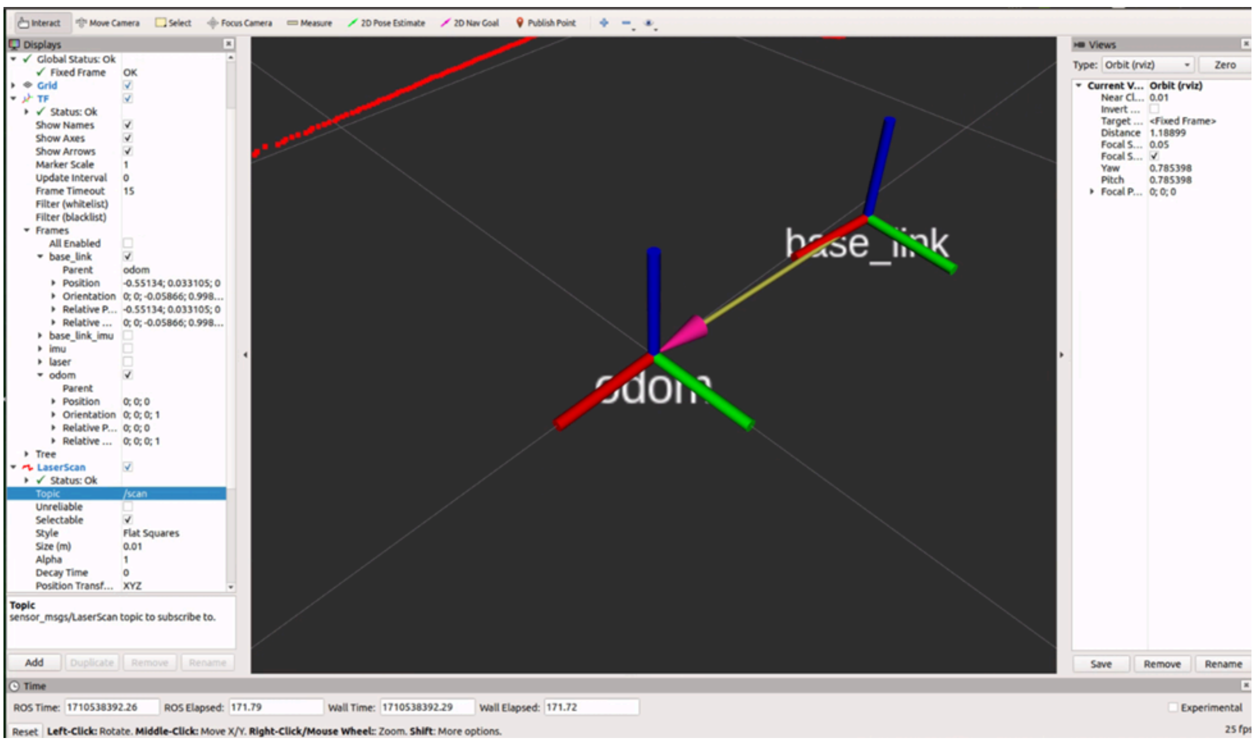
pr_ch_l = laser_to_base_ * pr_ch * base_to_laser_ ;

Activity 21: Launch the file “experiment.launch”. Use ROS visualization tool, rviz, to explore the performance of the localization algorithm. Set the fixed frame to “odom” and add a TF type display from the left panel. Also add a LaserScan type display with the topic “scan” to show the laser scan data. Observe how the frame “base_link” moves with the respect to the frame “odom” as you drive the vehicle around. Pay attention to the segments of the drive where the vehicle surrounding environment may have uniform features, e.g., when driving along two parallel walls. Would Scan Matcher by itself without wheel odometry and IMU would be effective in such case? Compare the performance of the Scan Matcher algorithm in localization with and without this data. Record the result for use in the report.

When a vehicle is driving around surroundings that may have uniform features, such as along two parallel walls, the Scan Matcher will not work without the wheel odometry and IMU because the localization algorithm cannot distinguish the environmental features. In this case, it is difficult for the positioning algorithm to accurately estimate the position and direction of the vehicle, resulting in errors and drift in the path.

However, it will improve the performance in localization when the Scan Matcher algorithm adds wheel odometry and IMU data. The odometry sensor monitors the vehicle’s movement based on wheel rotation and the IMU sensor detects its acceleration and angular velocity. When driving in an environment of uniform features, such as along two parallel walls, adding odometry and IMU data can help the algorithm correct for errors and drift that may occur when only relying on laser scan data, resulting in more accurate positioning.

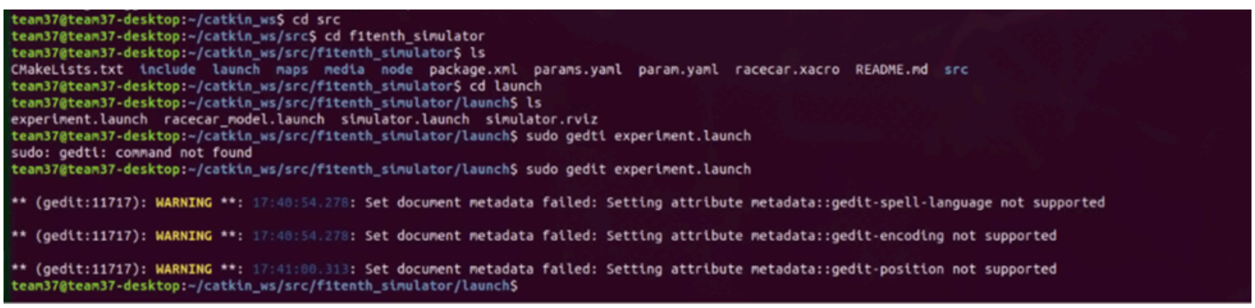



base_link
odom

Activity 22: Add a “LaserScan” display type and set its topic to “scan”. This should allow you to observe the laser scan data while driving the vehicle. Observe and compare the laser scan data in the two cases: (i) the rviz display (fixed) frame is set to “base_link” (ii) the rviz display (fixed) frame is set to “odom”. Explain what is happening in case (ii). How would you transform the LiDAR scan data to the fixed frame?

When the rviz displayed frame is set to “base_link”, the laser scan data will be displayed relative to the coordinate system based on the vehicle, which means the data will move as the vehicle moves. On the other hand, when the rviz displayed frame is set to “odom”, the laser scan data is presented relative to a fixed odometry coordinate. This means that the robot is not affected by its current position, which can cause the laser scan data to be offset or distorted relative to the robot's actual position. In the file “experiment.launch”, we are using tf packages which correctly align the LiDAR scan data with the vehicle Odom frame.

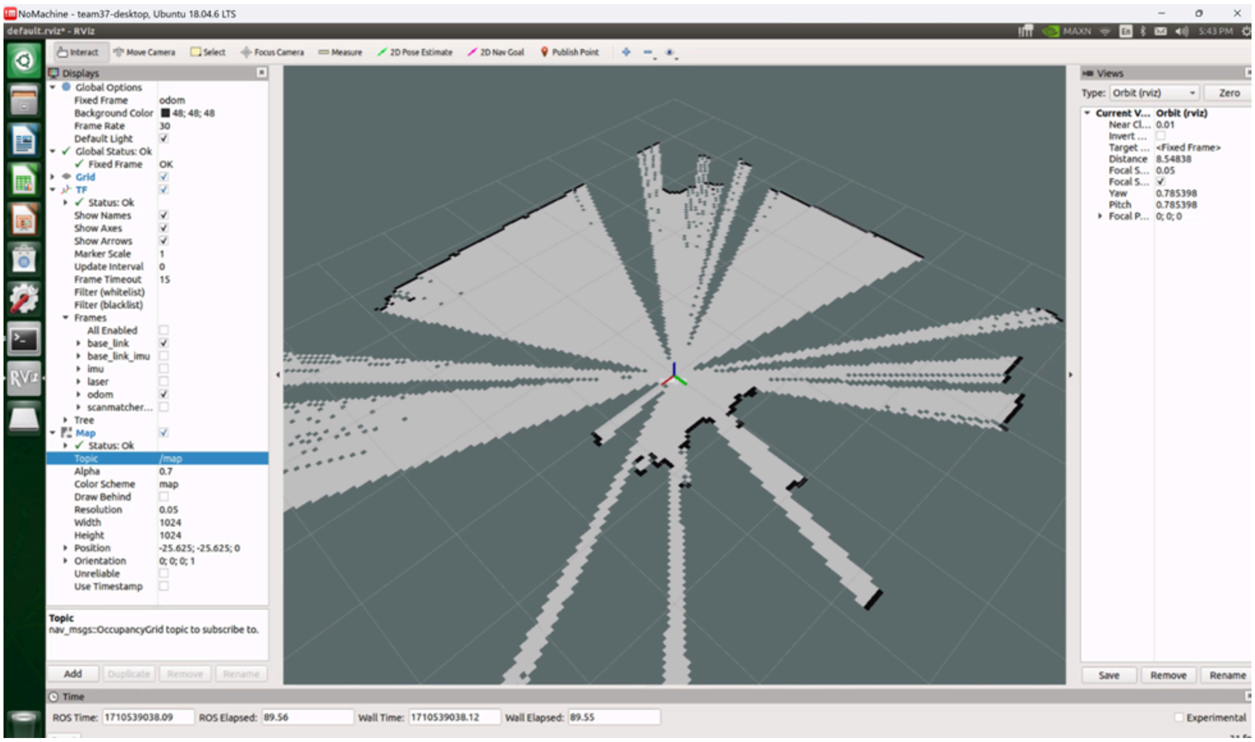
Activity 23:



We use the above command to install the hector SLAM on our ROS system.

Activity 24: Ensure that the vehicle surrounding is clear of objects and people as you will be driving the vehicle using the Joystick. Launch the modified “experiment.launch” file to start the SLAM algorithm. Also launch the ROS visualization tool rviz to display the results of the SLAM algorithm. Set the rviz fixed frame to “map” and add displays of TF and Map types from the left panel. Deselect all frames except for map (fixed frame) and base_link (body-attached frame) under TF. To display the map data, add “map” under the topic option of Map display. Drive the vehicle around slowly and observe

how the map of environment is constructed while the vehicle is simultaneously localized in this map. Comment on what you observe.

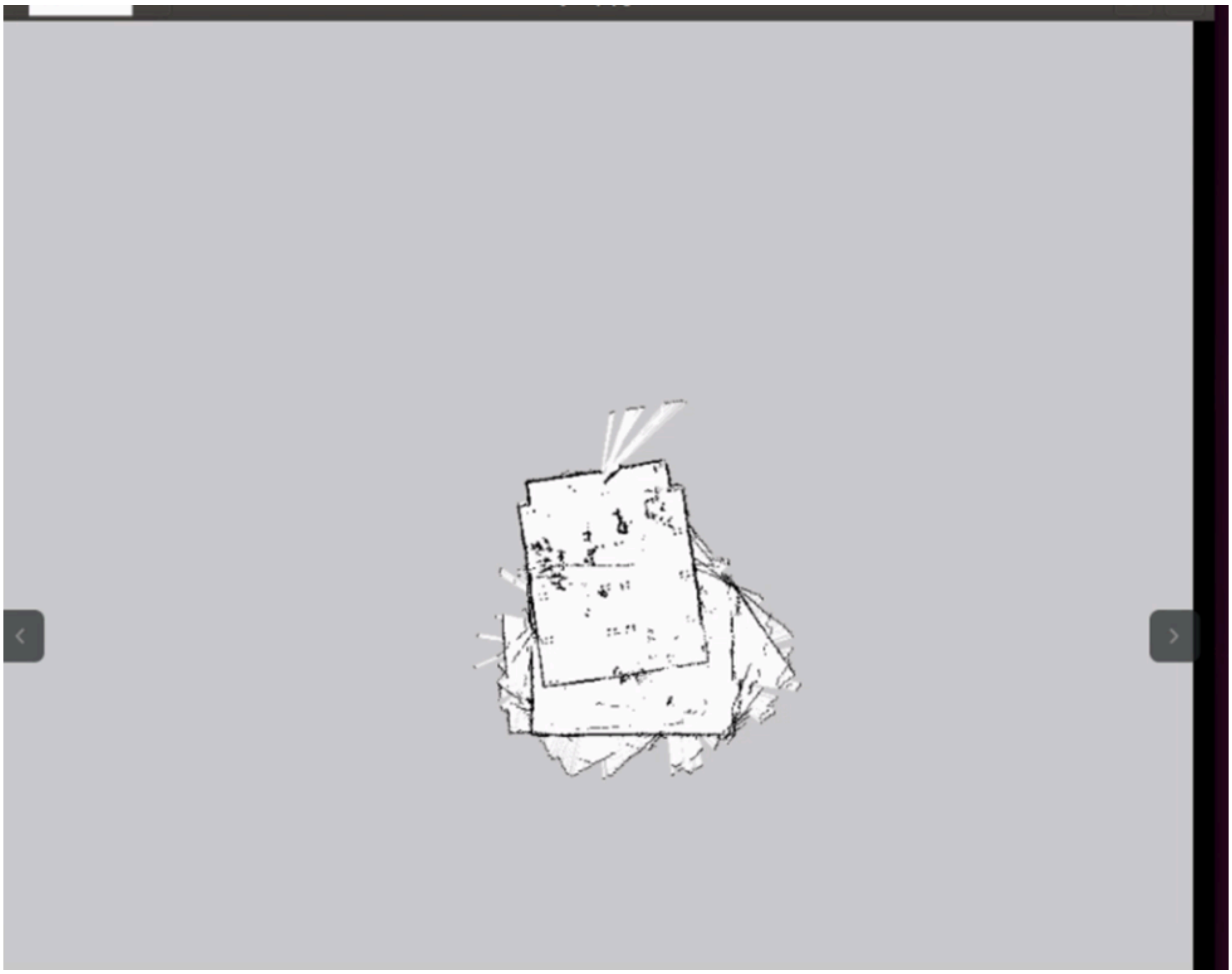


We drove the vehicle from side to side in the lab room which required 2 right angle turns. In order to display the map data, we add “map” under the Map topic. As sensors on the vehicle scan more of the environment, the map will become more complete and detailed. However, inconsistencies may arise if the algorithm encounters ambiguous features. Moreover, the accuracy of positioning estimates may change based on environmental conditions and vehicle driving errors or drift over time. Due to limited time and the complex driving environment, our picture doesn’t fully represent the complete environment during vehicle driving. Therefore, the SLAM algorithm can build a real-time map of the environment and determine the position of the vehicle through driving. At the same time, we can understand the surrounding environment by observing the real-time construction and updating of the environment map.

Activity 25: The hector SLAM lacks a loop closure mechanism. This simply means that if you revisit a part of the environment that you have seen before, the algorithm would not be able to recognize this and it could produce erroneous results. You should try to observe this potential issue by driving the vehicle around and then returning to areas that you have explored before. Save of a copy of the construct

map for your report. You can use the “map_saver” for this purpose. See here for information about how to use the command.

Map_saver picture



As seen above there are various scans overlapping at the edge of the main scan, that is because of the algorithm not recognizing that this area has been observed already and thereby adding it to the scan.