

## 2. Handheld lidar mapping

This function needs to start the program in the slam\_gmapping function package. The source code is located in the [yahboomcar\_ws] source code. Here we use the supporting virtual machine to explain how to start the program. If you want to put it on your own motherboard, put yahboomcar\_ws in the root directory and compile it.

### 2.1. Start lidar

Terminal input, (you need to enter the same docker, see the previous section for the method)

```
ros2 launch ydlidar_ros2_driver ydlidar_launch.py
```

### 2.2. Release static odom conversion

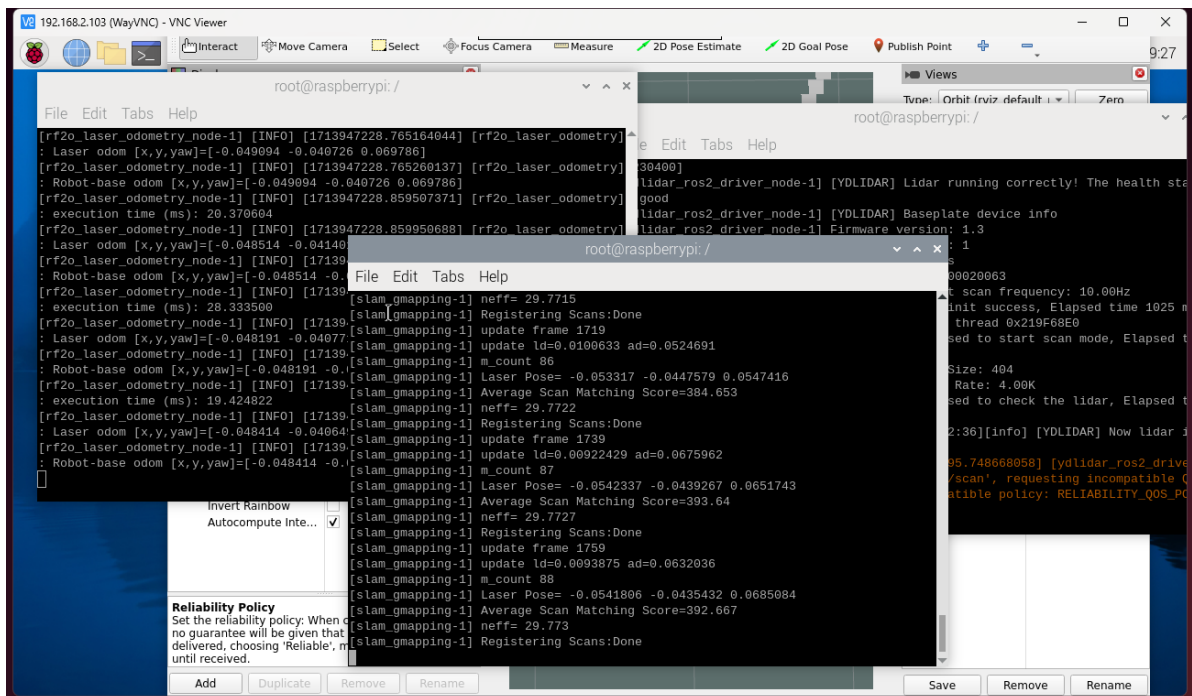
Terminal input,

```
ros2 launch rf2o_laser_odometry rf2o_laser_odometry.launch.py
```

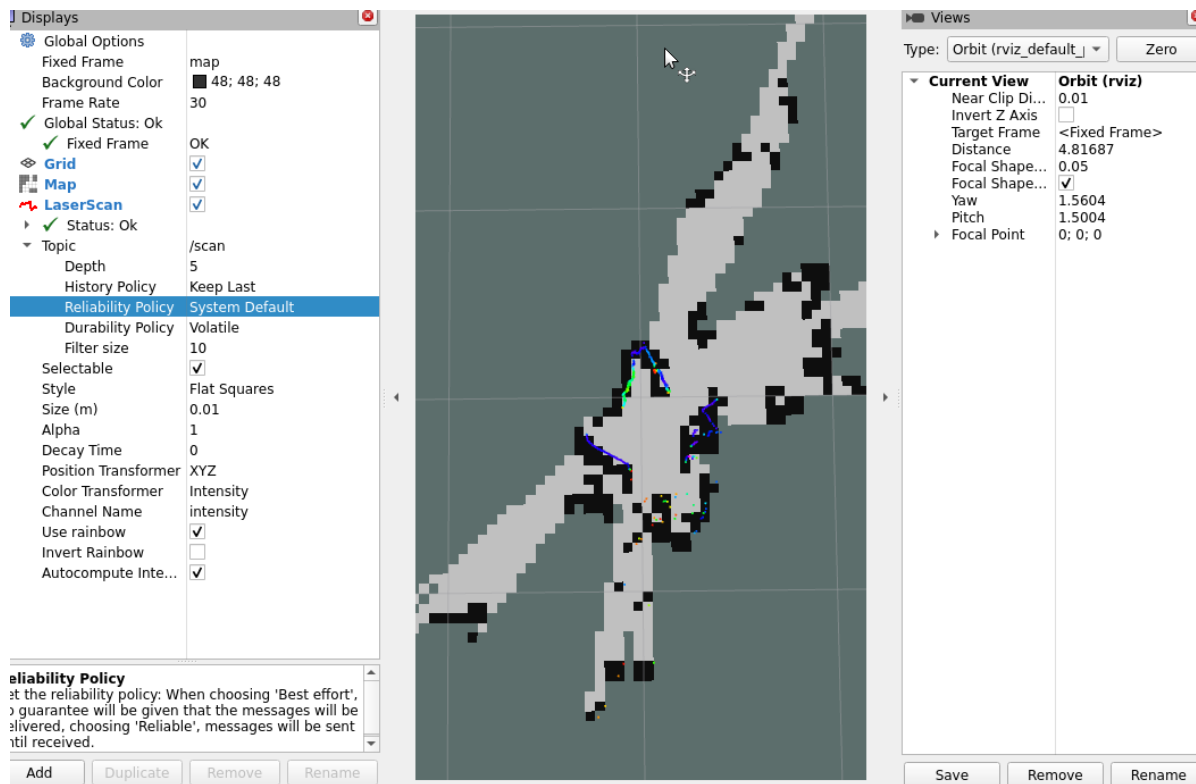
### 2.3. Start gmapping mapping

Terminal input,

```
ros2 launch slam_gmapping slam_gmapping.launch.py
```



rviz displays as follows,



## 2.4. View TF tree

Terminal input,

```
ros2 run tf2_tools view_frames
```

```
pi@raspberrypi:~$ docker exec -it b43d /bin/bash
root@raspberrypi:/# ros2 run tf2_tools view_frames
[INFO] [1713947299.218645649] [view_frames]: Listening to tf data for 5.0 second
S...
[INFO] [1713947304.234233411] [view_frames]: Generating graph in frames.pdf file
...
[INFO] [1713947304.242069623] [view_frames]: Result:tf2_msgs.srv.FrameGraph_Response(frame_yaml="laser_frame: \n parent: 'base_link'\n broadcaster: 'default_a
uthority'\n rate: 10000.000\n most_recent_transform: 0.000000\n oldest_transf
orm: 0.000000\n buffer_length: 0.000\nbase_link: \n parent: 'base_footprint'\n
broadcaster: 'default_authority'\n rate: 10000.000\n most_recent_transform:
0.000000\n oldest_transform: 0.000000\n buffer_length: 0.000\nodom: \n parent
: 'map'\n broadcaster: 'default_authority'\n rate: 20.012\n most_recent_trans
form: 1713947304.231238\n oldest_transform: 1713947301.382888\n buffer_length:
2.848\nbase_footprint: \n parent: 'odom'\n broadcaster: 'default_authority'\n
rate: 10.370\n most_recent_transform: 1713947303.990728\n oldest_transform:
1713947301.290644\n buffer_length: 2.700\n")
```

A frames.pdf file will be generated in the directory where the command terminal is started. This is the generated TF tree.

```
root@raspberrypi:/# ls
bin      etc      home     mnt      root     srv      usr
boot    frames_2024-04-24_16.28.24.gv  lib      opt      run      sys      var
dev     frames_2024-04-24_16.28.24.pdf media    proc     sbin     tmp
```