SLAM Simulation

First we go to the turtlebot3 site "TurtleBot3 (robotis.com)", change the visibility to noetic, and then follow the steps. We only need option 3 "Quick Start Guide and 6 "Simulation.

3.0 quick start guide: Install ROS on Remote PC

$ sudo apt update $ sudo apt upgrade $ wget
https://raw.githubusercontent.com/ROBOTIS
GIT/robotis_tools/master/install_ros_noetic.sh

 $ chmod 755 ./install_ros_noetic.sh $ bash ./install_ros_noetic.sh

Install Dependent ROS Packages

$ sudo apt-get install ros-noetic-joy ros-noetic-teleop-twist-joy

ros-noetic-teleop-twist-keyboard ros-noetic-laser-proc

ros-noetic-rgbd-launch ros-noetic-rosserial-arduino

ros-noetic-rosserial-python ros-noetic-rosserial-client

ros-noetic-rosserial-msgs ros-noetic-amcl ros-noetic-map-server

ros-noetic-move-base ros-noetic-urdf ros-noetic-xacro

ros-noetic-compressed-image-transport ros-noetic-rqt* ros-noetic-rviz

ros-noetic-gmapping ros-noetic-navigation ros-noetic-interactive-markers

Install TurtleBot3 Packages

$ sudo apt install ros-noetic-dynamixel-sdk $ sudo apt install ros-noetic-turtlebot3-msgs $ sudo apt install ros-noetic-turtlebot3

6.0 simulation

6.1 Gazebo simulation

Install Simulation Package

The TurtleBot3 Simulation Package requires turtlebot3 and turtlebot3_msgs packages as prerequisite. Without these prerequisite packages, the Simulation cannot be launched. Please follow the PC Setup instructions if you did not install required packages and dependent packages.

```
$ cd ~/catkin_ws/src/ $ git clone -b noetic-devel
https://github.com/ROBOTIS-GIT/turtlebot3_simulations.git $ cd ~/catkin_ws
&& catkin_make
```

Launch Simulation World

Three simulation environments are prepared for TurtleBot3. Please select one of these environments to launch Gazebo.

(burger , waffle , house)

I picked waffle.

```
$ export TURTLEBOT3_MODEL=waffle $ roslaunch turtlebot3_gazebo
turtlebot3_world.launch
```

Operate TurtleBot3 In order to teleoperate the TurtleBot3 with the keyboard, launch the teleoperation node with below command in a new terminal window.

```
roslaunch turtlebot3_teleop turtlebot3_teleop_key.launch
```

6.2 SLAM simulation

The following instructions require prerequisites from the previous sections

Launch Simulation World

```
$ export TURTLEBOT3_MODEL=waffle $ roslaunch turtlebot3_gazebo
turtlebot3_world.launch
```

Run SLAM Node

```
$ export TURTLEBOT3_MODEL=waffle $ roslaunch turtlebot3_slam
turtlebot3_slam.launch slam_methods:=gmapping
```

Run Teleoperation Node

```
$ export TURTLEBOT3_MODEL=waffle $ roslaunch turtlebot3_teleop
turtlebot3_teleop_key.launch
```

Control Your TurtleBot3!

Moving around: w a s d x

w/x : increase/decrease linear velocity a/d : increase/decrease angular velocity space key, s : force stop

CTRL-C to quit

Here we can move the little car by pressing the move keys (a,w,d,x and s for stop).

Save Map

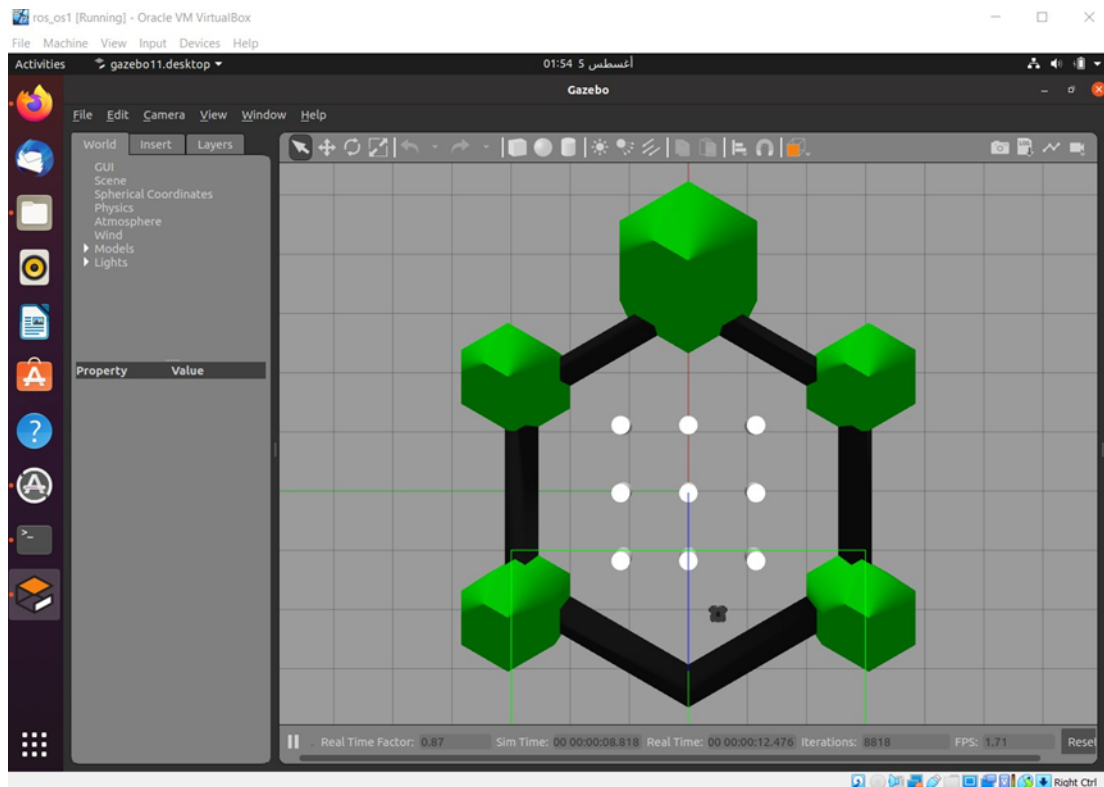When the map is created successfully, we can save it.

```
$ rosrun map_server map_saver -f ~/map
```

Finally, I screen shot the result and put it in pdf file.

After writing this step

$ export TURTLEBOT3_MODEL=Haneen

$ roslaunch turtlebot3_gazebo turtlebot3_world.launch
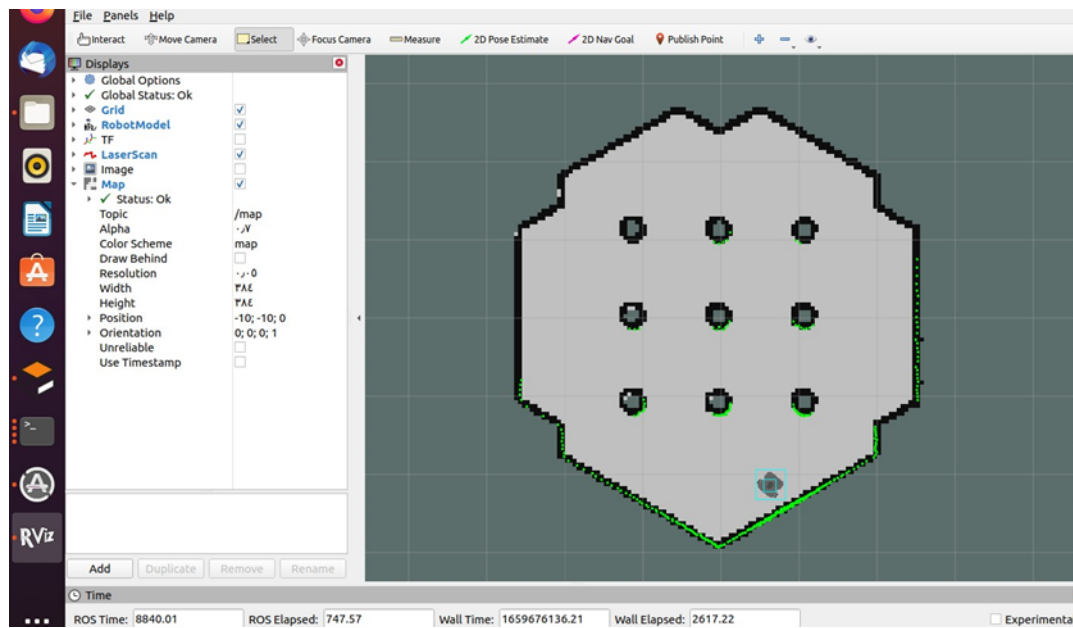

And then write this step

$ export TURTLEBOT3_MODEL=Haneen

$ roslaunch turtlebot3_slam turtlebot3_slam.launch slam_methods:=gmapping

To We need to move the little car to create the map ,so we need this step :

$export TURTLEBOT3_MODEL= Haneen

$roslaunch turtlebot3_teleop turtlebot3_teleop_key.launch

after moving the car the map will be created

Finally Saving the map

$rosrun map_server map_saver -f ~/map