

SLAM Lidar ROS2

Note: This section supports the following lidar models: A1/A2/A3/S1/S2; The baud rates are different between A1/A2<115200>, A3/S1<256000> and S2<1000000>

SLAM Lidar ROS2

- 1、Remap the USB serial port
- 2、Terminal test
- 3、rviz2 Visual test
- 4、gmapping Build diagram test

ROS 2 official website: <https://docs.ros.org/>

Lidar ROS2: <https://www.slamtec.com/cn/Support#rplidar-a-series>

Lidar official website: <http://www.slamtec.com/cn/Support>

Device : PC; A1M8

Environment : Ubuntu18.04; ROS2 (eloquent)

New Workspace

```
mkdir -p my_ws/src
```

When the lidar is used, it is necessary to enter the workspace each time the command is executed

```
cd ~/my_ws
```

Decompress the [silanlidar_ros2.zip] function package and put it in the src folder of your workspace. Open the terminal in the workspace

```
colcon build          # compile
source install/setup.bash # Update the environment
```

Note: Every time a new terminal is opened, it needs to [update the environment]. Every time the code in the function pack is changed, it needs to [re-compile] and then update the environment.

1、Remap the USB serial port

To perform this step, run the installation USB port remapping command in the scripts folder of the slidar_ros2 function package

```
cd ~/my_ws/src/sllidar_ros2/scripts
sudo chmod 777 *
./create_udev_rules.sh
```

Use the following command to view the modified remapping

```
ls -l /dev | grep ttyUSB
```

```
yahboom@pc: ~74x18
yahboom@pc:~$ ls -l /dev | grep ttyUSB
lrwxrwxrwx 1 root root 7 28 10:41 rplidar -> ttyUSB0
crwxrwxrwx 1 root dialout 188, 0 28 10:41 ttyUSB0
yahboom@pc:~$
```

2、Terminal test

Step one, activate the corresponding lidar

```
ros2 launch sllidar_ros2 sllidar_launch.py # RPLIDAR A1/A2
ros2 launch sllidar_ros2 sllidar_a3_launch.py # RPLIDAR A3
ros2 launch sllidar_ros2 sllidar_s1_launch.py # RPLIDAR S1
ros2 launch sllidar_ros2 sllidar_s2_launch.py # RPLIDAR S2
```

Second, start the test application

```
roslaunch rplidar_ros rplidarNodeClient
```

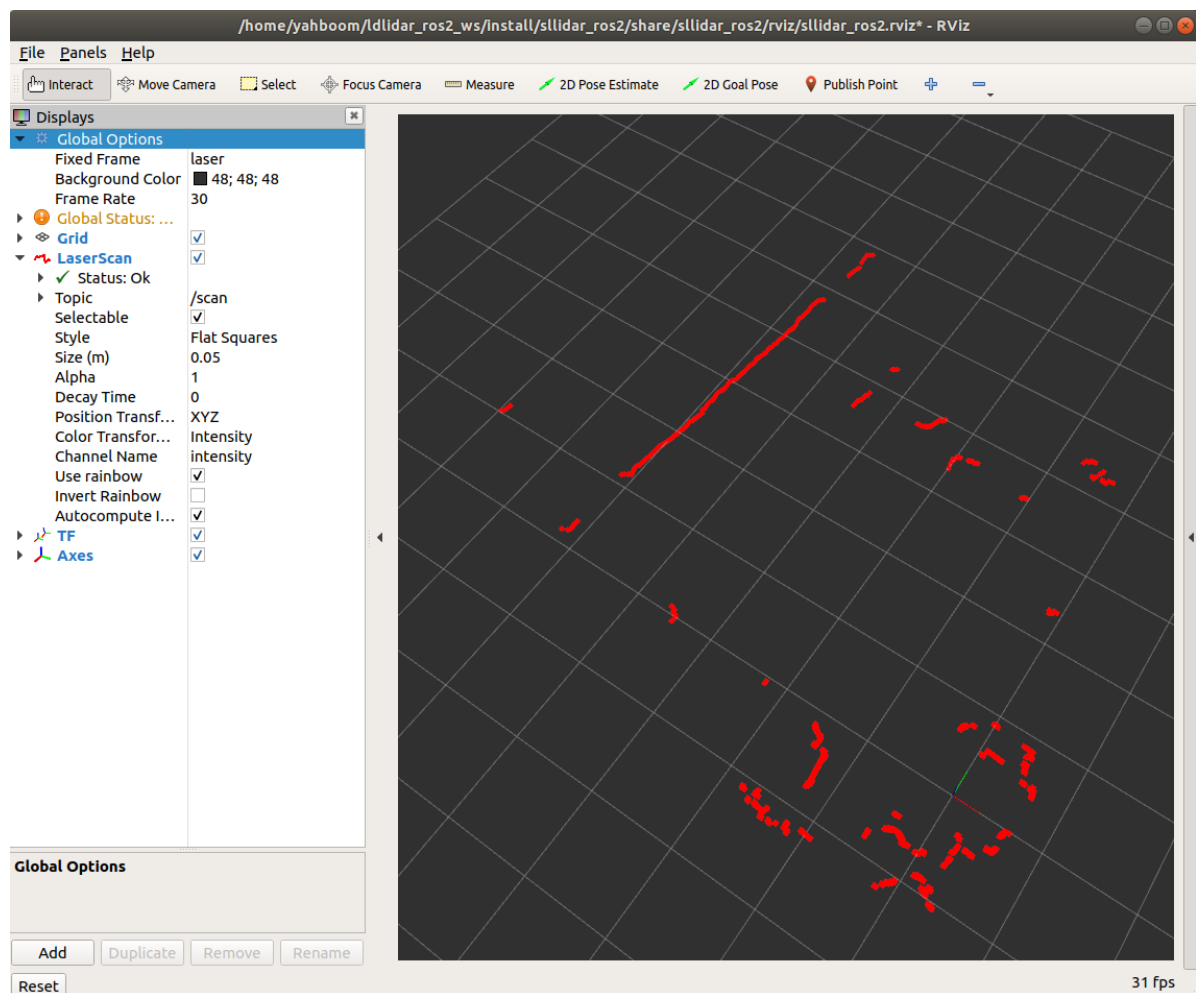
You should see the rPLidar scan results in the console

```
yahboom@pc: ~89x45
[SLLIDAR INFO]: angle-distance : [14.307703, 0.133000]
[SLLIDAR INFO]: angle-distance : [14.640415, inf]
[SLLIDAR INFO]: angle-distance : [14.973127, inf]
[SLLIDAR INFO]: angle-distance : [15.305840, inf]
[SLLIDAR INFO]: angle-distance : [15.638565, inf]
[SLLIDAR INFO]: angle-distance : [15.971277, inf]
[SLLIDAR INFO]: angle-distance : [16.303989, inf]
[SLLIDAR INFO]: angle-distance : [16.636702, inf]
[SLLIDAR INFO]: angle-distance : [16.969414, inf]
[SLLIDAR INFO]: angle-distance : [17.302139, inf]
[SLLIDAR INFO]: angle-distance : [17.634851, inf]
[SLLIDAR INFO]: angle-distance : [17.967564, inf]
[SLLIDAR INFO]: angle-distance : [18.300276, inf]
[SLLIDAR INFO]: angle-distance : [18.633001, inf]
[SLLIDAR INFO]: angle-distance : [18.965714, inf]
[SLLIDAR INFO]: angle-distance : [19.298426, inf]
[SLLIDAR INFO]: angle-distance : [19.631138, inf]
[SLLIDAR INFO]: angle-distance : [19.963863, inf]
[SLLIDAR INFO]: angle-distance : [20.296576, inf]
[SLLIDAR INFO]: angle-distance : [20.629288, inf]
[SLLIDAR INFO]: angle-distance : [20.962000, inf]
[SLLIDAR INFO]: angle-distance : [21.294727, inf]
[SLLIDAR INFO]: angle-distance : [21.627439, inf]
[SLLIDAR INFO]: angle-distance : [21.960152, inf]
[SLLIDAR INFO]: angle-distance : [22.292864, inf]
[SLLIDAR INFO]: angle-distance : [22.625589, inf]
[SLLIDAR INFO]: angle-distance : [22.958302, inf]
```

3、rviz2 Visual test

```
ros2 launch sllidar_ros2 view_sllidar_launch.py # RPLIDAR A1/A2
ros2 launch sllidar_ros2 view_sllidar_a3_launch.py # RPLIDAR A3
ros2 launch sllidar_ros2 view_sllidar_s1_launch.py # RPLIDAR S1
ros2 launch sllidar_ros2 view_sllidar_s2_launch.py # RPLIDAR S2
```

You should see the lidar scan results in rviz2



Note: the [Fixed Frame] coordinate system and [LaserScan] topics should be consistent with those published.

4. gmapping Build diagram test

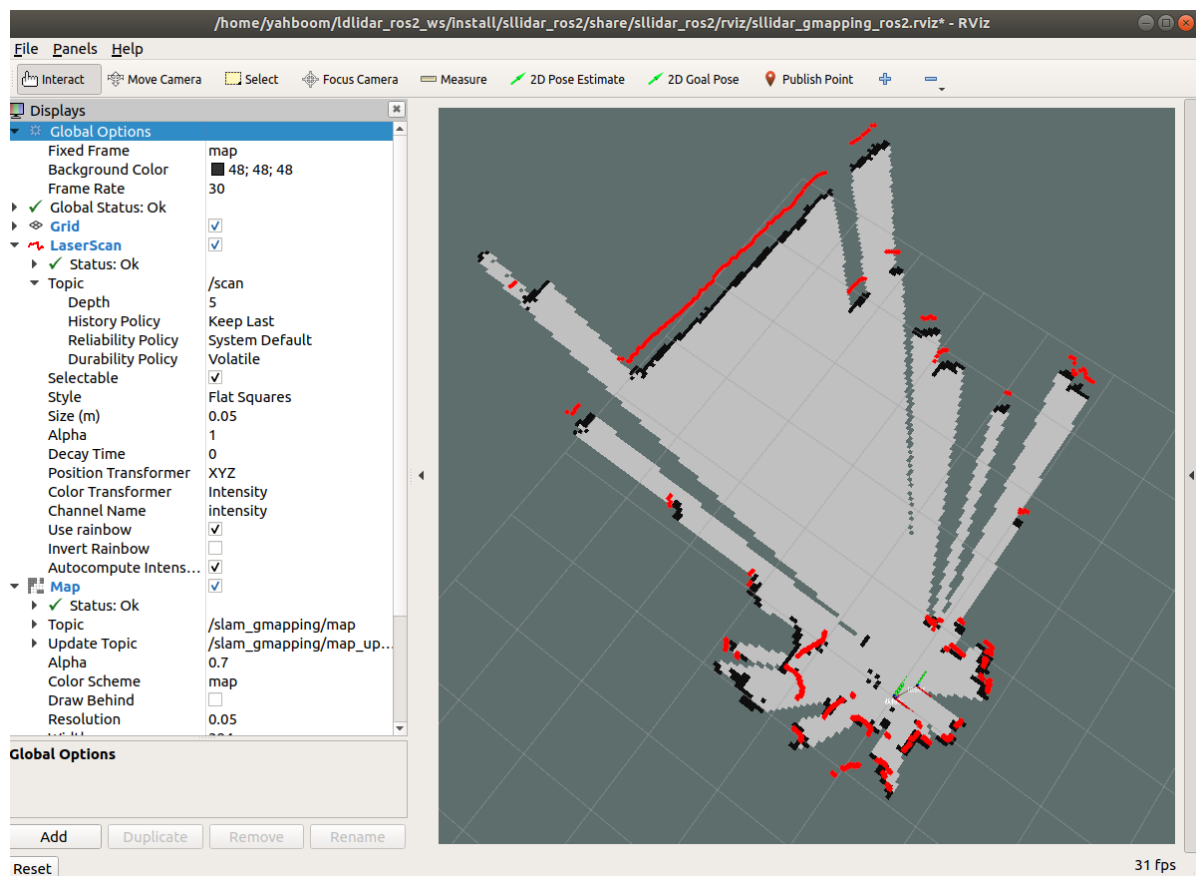
Step one, activate the corresponding lidar

```
ros2 launch sllidar_ros2 sllidar_launch.py # RPLIDAR A1/A2
ros2 launch sllidar_ros2 sllidar_a3_launch.py # RPLIDAR A3
ros2 launch sllidar_ros2 sllidar_s1_launch.py # RPLIDAR S1
ros2 launch sllidar_ros2 sllidar_s2_launch.py # RPLIDAR S2
```

Second, start gmapping

```
ros2 launch sllidar_ros2 gmapping_launch.py # other
ros2 launch sllidar_ros2 gmapping_s2_launch.py # S2
```

You should see the result in rviz2



Note: [Fixed Frame] coordinate system, [LaserScan] topic, [map] topic should be the same as published.