

1.Preparation

1.1Compile function package

Unzip rplidar_ros-dev-ros2 in the [Source Code] folder to get the rplidar_ros-dev-ros2 function package. Copy rplidar_ros to the src directory of the workspace you created. Here, the workspace name is rplidar_ws as an example. The path of rplidar_ws is in the ~ directory, and then return to the workspace directory to compile.

```
cd ~/rplidar_ws
colcon build --symlink-install
```

```
---
Finished <<< rplidar_ros [14.9s]
Summary: 1 package finished [15.7s]
```

The above screen description appears and the compilation is passed.

Then enter the following command to set the environment variables.

```
echo "source ~/rplidar_ws/install/setup.bash --extend" >> ~/.bashrc
```

Open the terminal under the rplidar_ros-dev-ros2 function package, enter the following command, and copy the rplidar.rules file under the function package to /etc/udev/rules.d.

```
sudo cp rplidar.rules /etc/udev/rules.d/
```

Then re-plug the radar serial port.

Input following command:

```
ll /dev/rplidar
```

```
yahboom@VM:~/Desktop$ ll /dev/rplidar
lrwxrwxrwx 1 root root 7 11月 10 18:26 /dev/rplidar -> ttyUSB0
yahboom@VM:~/Desktop$
```

The above content indicates that the binding is successful.

The end is not necessarily 0 and changes according to the order in which the devices are inserted.

1.2 Run launch

Run lidar:

```

#A1 lidar
ros2 launch rplidar_ros rplidar_a1_launch.py
#A2M12 lidar
ros2 launch rplidar_ros rplidar_a2m12_launch.py
#A3 lidar
ros2 launch rplidar_ros rplidar_a3_launch.py
#C1 lidar
ros2 launch rplidar_ros rplidar_c1_launch.py
#S2 lidar, S2L lidar
ros2 launch rplidar_ros rplidar_s2_launch.py
#s3
ros2 launch rplidar_ros rplidar_s3_launch.py

```

Lidar node data can be viewed through the following command.

```
ros2 topic echo /scan
```

```

header:
  stamp:
    sec: 1699612558
    nanosec: 971186587
  frame_id: laser
angle_min: -3.1241390705108643
angle_max: 3.1415927410125732
angle_increment: 0.0019344649044796824
time_increment: 3.068916339543648e-05
scan_time: 0.09940220415592194
range_min: 0.15000000596046448
range_max: 30.0
ranges:
- 0.20399999618530273
- 0.20399999618530273
- 0.20399999618530273
- 0.20399999618530273
- 0.20399999618530273
- 0.2029999941587448
- 0.20200000703334808
- 0.20100000500679016
- 0.20000000298023224
- 0.19900000095367432
- 0.1979999989271164
- 0.19599999487400055
- 0.19499999284744263
- 0.1940000057220459
- 0.19200000166893005
- 0.19099999964237213
- 0.1899999976158142
- 0.1899999976158142
- 0.1889999955892563
- 0.1889999955892563
- 0.1889999955892563
- 0.1889999955892563
- 0.1889999955892563
- 0.1889999955892563
- 0.1889999955892563

```

Run the radar to display the point cloud and input it into the terminal (ctrl c is required to close the node that started the lidar before),

```

#A1 lidar
ros2 launch rplidar_ros view_rplidar_a1_launch.py
#A2M12 lidar
ros2 launch rplidar_ros view_rplidar_a2m12_launch.py
#A3 lidar
ros2 launch rplidar_ros view_rplidar_a3_launch.py
#C1 lidar
ros2 launch rplidar_ros view_rplidar_c1_launch.py
#S2 lidar, S2L lidar
ros2 launch rplidar_ros view_rplidar_s2_launch.py
#s3
ros2 launch rplidar_ros view_rplidar_s3_launch.py

```

