

轮趣科技

M10 串口雷达上位机软件及 ROS 环境中使用教程

推荐关注我们的公众号获取更新资料



版本说明:

版本	日期	内容说明
V1.0	2021/11/23	第一次发布
V2.0	2022/11/24	第二次发布

网址: www.wheeltec.net

目录

1. 上位机软件	3
1.1 安装 cp2102 芯片驱动	3
1.2 连接使用上位机软件	4
1.3 CP2102 信息修改	6
2. ROS	8
2.1 编译 ROS_SDK 包	8
2.2 Ubuntu CP2102 驱动安装	9
2.3 使用 ROS 包驱动 M10 雷达	11

1. 上位机软件

1.1 安装 cp2102 芯片驱动

① 解压资料中的“CP2102usb 驱动.zip”文件

选中资料包中 CP2102 芯片目录下对应的驱动压缩包，进行解压使用。



图 1-1-1 解压 CP2102 驱动压缩包

② 安装 CP2102 驱动

根据系统的位数来选择相应位数的驱动进行安装。

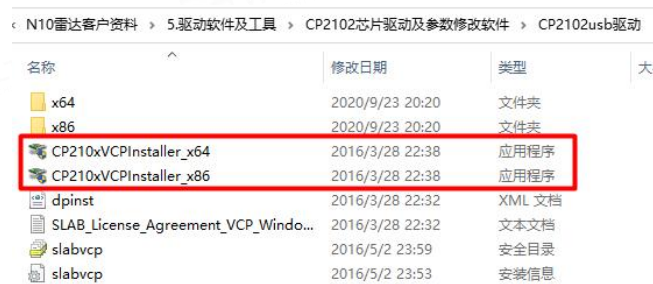


图 1-1-2 选择相应位数的驱动

双击打开安装程序后点击下一页→接受协议→下一页进行驱动安装，安装完成后会显示以下界面，这时电脑就能识别到我们的 CP2102 的 usb 了。



图 1-1-3 驱动安装完成

1.2 连接使用上位机软件

① 雷达接线

一开始到手，我们会有三个部分：雷达本体，串口转接模块，TPYE-C 线。将他们按下图方式连接。



图 1-2-1 雷达接线

之后将转接模块上的开关拨动至靠近 type-C 接口的一侧，表示雷达开启。

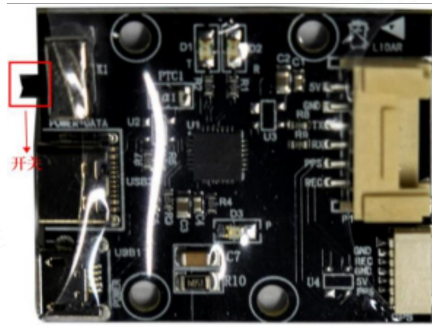


图 1-2-2 打开开关

注：转接板相关说明可以参考《M10 产品使用手册》第 11~12 页。

② 连接电脑运行上位机软件

将 USB 线接到电脑的 USB 口，在 windows 中右键此电脑→属性→设备管理器→端口，找到 CP210x USB 的端口名，其后面对应的 COM 即为雷达在 windows 中的串口号。



图 1-2-3 查看串口号

解压 windows 上位机软件中的压缩包，得到以下文件夹。

M10雷达客户资料V3.5_20221111 > M10-10KHZ串口 > 2.windows上位机软件

名称	修改日期	类型
LSLIDAR_M10_Windows上位机软件_v1.1.5_20220409	2022/11/24 14:10	文件
LSLIDAR_M10_Windows上位机软件_v1.1.5_20220409	2022/11/24 14:06	Win

图 1-2-4 解压上位机软件压缩包

进入解压后的文件夹，找到 LSLIDAR_M10_v1.1.5_20220409.exe 双击运行。

<< LSLIDAR_M10_Windows上位机软件_v1.1.5_202... > LSLIDAR_M10_v1.1.5_20220409

名称	修改日期	类型	大小
Developer	2022/4/11 10:53	文件夹	
doc	2022/4/11 10:53	文件夹	
imageformats	2022/11/24 14:10	文件夹	
platforms	2022/11/24 14:10	文件夹	
Resources	2022/4/11 10:51	文件夹	
icudt53.dll	2018/8/11 10:15	应用程序扩展	21,025 KB
icuin53.dll	2018/8/11 10:12	应用程序扩展	1,937 KB
icuuc53.dll	2018/8/11 10:14	应用程序扩展	1,324 KB
LASlib.dll	2019/3/21 8:42	应用程序扩展	14 KB
LSLIDAR_M10_v1.1.5_20220409	2022/4/11 10:50	应用程序	1,889 KB

图 1-2-5 运行上位机软件

打开上位机软件后，找到 COM 选择的下拉窗口，选择我们刚刚查看到的串口号。



图 1-2-6 选择对应串口

之后点击左上角的“雷达连接”，即可连接到我们的雷达，并在上位机软件中显示雷达点云数据。

注：上位机软件相关功能说明可以参考《M10 产品使用手册》第 16~22 页。

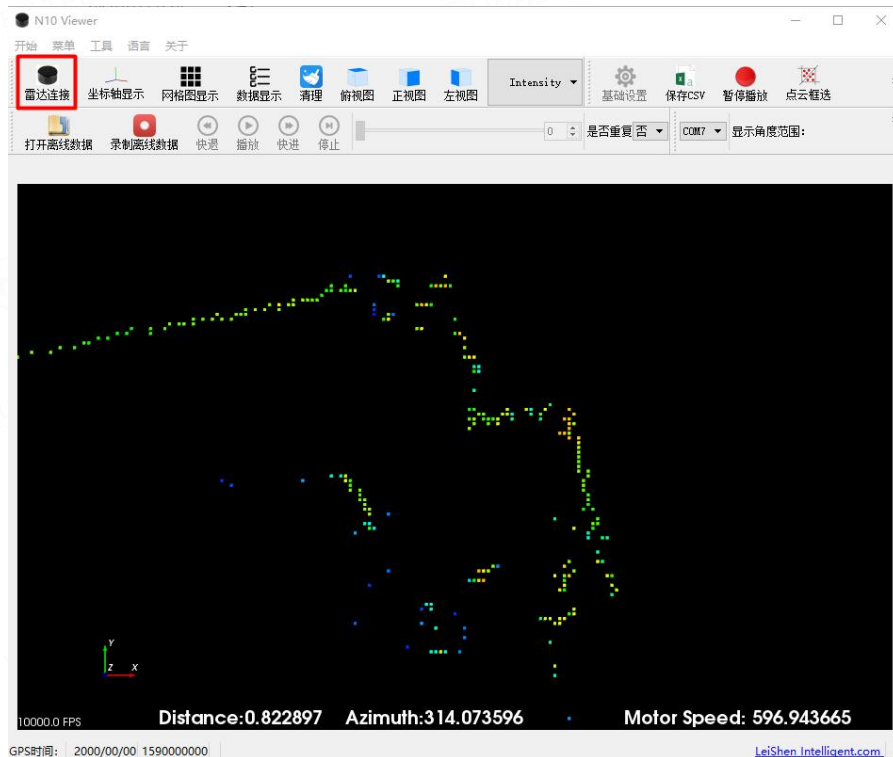


图 1-2-7 雷达连接上位机软件

1.3 CP2102 信息修改

解压 CP2102 芯片目录下的“修改端口号软件”，得到以下文件。

« M10-10KHZ串口 » 5.驱动软件及工具 » 1.Windows CP2102驱动及参数修改软件			
名称	修改日期	类型	大小
CP2102usb驱动	2022/11/24 14:07	WinRAR ZIP 压缩...	3,767 KB
修改端口号软件	2022/11/24 14:07	WinRAR ZIP 压缩...	39,438 KB

图 1-3-1 解压压缩包

打开解压后的文件夹，双击运行 CP21xxCustomizationUtility.exe。



图 1-3-2 运行 CP21xxCustomizationUtility.exe

之后若我们的电脑有连接 CP2102 芯片的设备则会显示相关参数，我们可以通过修改对应参数栏后面的 Value 值并点击“Program Device”进行应用修改。

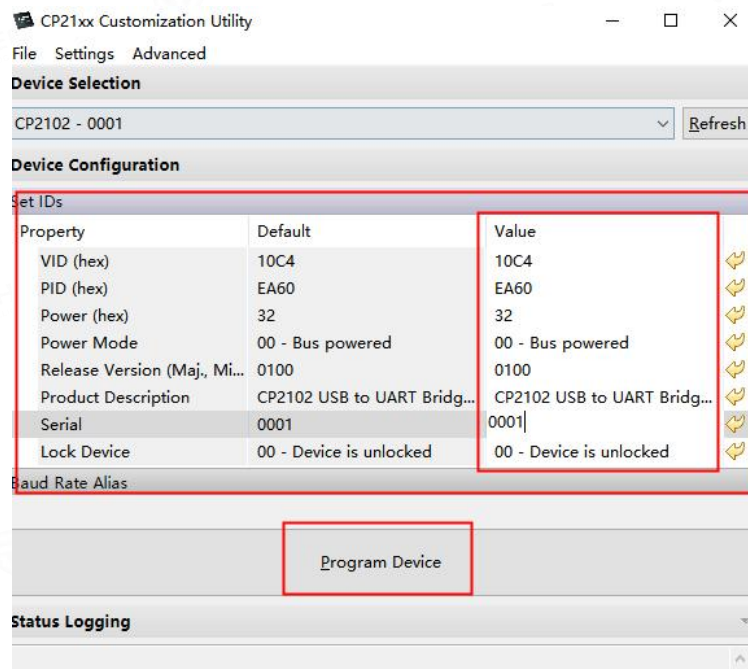


图 1-3-3 查看芯片信息及修改

2. ROS

2.1 编译 ROS_SDK 包

① ROS_SDK

进入 ROS_SDK 目录中，我们可以看到 lsx10 包。



图 2-1-1 解压 ROS_SDK 压缩包

② 将 lsx10 放入 ROS 工作空间的 src 中

将 lsx10 包复制到 ROS 工作空间的 src 目录中。



图 2-1-2 lsx10 放置到 ros 工作空间

③ 下载 libpcap 库

运行以下命令安装 libpcap-dev。（需要联网进行）

```
passoni@passoni:~$ sudo apt-get install libpcap-dev
```

图 2-1-3 sudo apt-get install libpcap-dev

④ 运行终端并编译

返回 ROS 工作空间的根目录并在该目录下打开一个终端，执行“catkin_make”命令进行编译。

```
passoni@passoni:~/catkin_ws$ catkin_make
```

图 2-1-4 编译功能包

⑤ 编译成功

执行完上一步后，系统会开始编译 lsx10 功能包并在终端显示相应的编译进

度信息。若中间无报错最后显示为 100%则编译成功。

```
0.dir/src/lslidar_serial_x10.o
[ 88%] Linking CXX shared library /home/passoni/catkin_ws/devel/lib/liblslidar_serial_x10.so
[ 88%] Built target lslidar_serial_x10
Scanning dependencies of target sensor_msgs_generate_messages_lisp
[ 88%] Built target sensor_msgs_generate_messages_lisp
Scanning dependencies of target sensor_msgs_generate_messages_eus
[ 88%] Built target sensor_msgs_generate_messages_eus
Scanning dependencies of target sensor_msgs_generate_messages_cpp
[ 88%] Built target sensor_msgs_generate_messages_cpp
Scanning dependencies of target nodelet_generate_messages_lisp
[ 88%] Built target nodelet_generate_messages_lisp
Scanning dependencies of target lslidar_x10_driver
[ 91%] Building CXX object lslidar_x10_driver/CMakeFiles/lslidar_x10_driver.dir/src/lslidar_x10_driver.cc.o
[ 94%] Linking CXX shared library /home/passoni/catkin_ws/devel/lib/liblslidar_x10_driver.so
[ 94%] Built target lslidar_x10_driver
Scanning dependencies of target lslidar_x10_driver_node
[ 97%] Building CXX object lslidar_x10_driver/CMakeFiles/lslidar_x10_driver_node.dir/src/lslidar_x10_driver_node.cc.o
[100%] Linking CXX executable /home/passoni/catkin_ws/devel/lib/lslidar_x10_driver/lslidar_x10_driver_node
[100%] Built target lslidar_x10_driver_node
```

图 2-1-4 编译成功

2.2 Ubuntu CP2102 驱动安装

一般情况下 ubuntu 系统都会自带该驱动模块，不需要手动安装，该步骤主要针对系统中没有该驱动的用户。可以通过在终端执行 `lsmod|grep cp210x` 命令确认系统是否已经有该驱动模块。

① 解压驱动压缩包

首先解压“Ubuntu CP2102 驱动”目录下的压缩包，得到以下文件。

< N10雷达客户资料 > > 5.驱动软件及工具 > Ubuntu CP2102驱动 > Linux_3.x.x_4.x.x_VCP_Driver_Source				
名称	修改日期	类型	大小	
cp210x.c	2021/1/29 19:12	C 文件	78 KB	
cp210x_gpio_example.c	2021/1/29 19:12	C 文件	2 KB	
cp210x_gpio_example_gpiolib.c	2021/1/29 19:12	C 文件	4 KB	
CP210x_VCP_Linux_4.x_Release_Notes	2021/1/29 19:16	文本文档	3 KB	
Makefile	2020/8/19 17:03	文件	1 KB	

图 2-2-1 解压驱动压缩包

② 打开解压得到的 txt 文档

打开解压得到的 txt 文档，并参照“Ubuntu:”后面四步进行驱动安装。

The bundle contains:

- * cp210x.c
- * Makefile
- * cp210x_gpio_example.c
- * cp210x_gpio_example_gpiolib.c
- * build.sh
- * CP210x_VCP_Linux_4.x_Release_Notes.txt

Build instructions:

Ubuntu:

1. make (your cp210x driver)
2. cp cp210x.ko to /lib/modules/<kernel-version>/kernel/drivers/usb/serial
3. insmod /lib/modules/<kernel-version>/kernel/drivers/usb/serial/usbserial.ko
4. insmod cp210x.ko

图 2-2-2 打开 txt 文档说明

③ 安装驱动

将我们解压得到的驱动文件夹整个复制到 ubuntu 的 home 路径下。



图 2-2-3 复制驱动文件夹到 ubuntu

打开一个终端进入到我们刚刚复制过来的驱动文件夹中，并执行 make 指令进行编译。

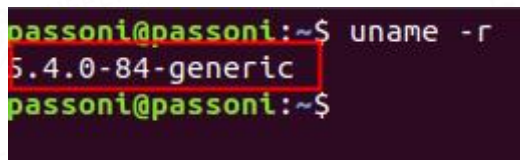
```
passoni@passoni:~$ cd Linux_3.x.x_4.x.x_VCP_Driver_Source/
passoni@passoni:~/Linux_3.x.x_4.x.x_VCP_Driver_Source$ make
make -C /lib/modules/`uname -r`/build M=/home/passoni/Linux_3.x.x_4.x.x_VCP_Driver_Source modules
make[1]: 进入目录"/usr/src/linux-headers-5.4.0-84-generic"
CC [M] /home/passoni/Linux_3.x.x_4.x.x_VCP_Driver_Source/cp210x.o
Building modules, stage 2.
MODPOST 1 modules
CC [M] /home/passoni/Linux_3.x.x_4.x.x_VCP_Driver_Source/cp210x.mod.o
LD [M] /home/passoni/Linux_3.x.x_4.x.x_VCP_Driver_Source/cp210x.ko
make[1]: 离开目录"/usr/src/linux-headers-5.4.0-84-generic"
passoni@passoni:~/Linux_3.x.x_4.x.x_VCP_Driver_Source$
```

图 2-2-4 执行 make 编译驱动包

编译完成之后，文件夹内会生成几个新的文件。我们继续在这个终端执行一个复制文件的命令

```
sudo cp cp210x.ko /lib/modules/<kernel-version>/kernel/drivers/usb/serial
```

其中<kernel-version>是系统发行编号，可以通过 `uname -r` 指令查看。



```
passoni@passoni:~$ uname -r
5.4.0-84-generic
passoni@passoni:~$
```

图 2-2-5 查看系统发行编号

在得到系统发行编号之后，我们就可以执行上面那条复制指令了。

```
passoni@passoni:~/Linux_3.x.x_4.x.x_VCP_Driver_Source$ sudo cp cp210x.ko /lib/modules/5.4.0-84-generic/kernel/drivers/usb/serial
passoni@passoni:~/Linux_3.x.x_4.x.x_VCP_Driver_Source$
```

图 2-2-6 复制 cp210x.ko 到指定目录

接下来继续在终端执行以下命令进行模块的载入：

```
sudo insmod /lib/modules/<kernel-version>/kernel/drivers/usb/serial/usbserial.ko
```

```
passoni@passoni:~/Linux_3.x.x_4.x.x_VCP_Driver_Source$ sudo insmod /lib/modules/5.4.0-84-generic/kernel/drivers/usb/serial/usbserial.ko
passoni@passoni:~/Linux_3.x.x_4.x.x_VCP_Driver_Source$
```

图 2-2-7 insmod 载入 usbserial.ko 模块

最后再执行 cp210x.ko 模块的载入命令

sudo insmod cp210x.ko

```
passoni@passoni:~/Linux_3.x.x_4.x.x_VCP_Driver_Source$ sudo insmod cp210x.ko
passoni@passoni:~/Linux_3.x.x_4.x.x_VCP_Driver_Source$
```

图 2-2-8 insmod 载入 cp210x.ko 模块

通过 lsmod|grep cp210x 命令查看已安装驱动，发现 cp210x 说明我们的驱动安装成功。

```
passoni@passoni:~$ lsmod|grep cp210x
cp210x                40960      0
usbserial              49152      1 cp210x
```

图 2-2-9 lsmod 查看已安装驱动

2.3 使用 ROS 包驱动 M10 雷达

① 将雷达连接至 Ubuntu 系统

将雷达连接至 ubuntu 系统，通过 lsusb 命令查看已连接的 usb 设备，发现 CP210x 的设备说明连接成功。

```
passoni@passoni:~$ lsusb
Bus 004 Device 001: ID 1d6b:0003 Linux Foundation 3.0 root hub
Bus 003 Device 002: ID 10c4:ea60 Cygnal Integrated Products, Inc. CP210x UART Br
idge / myAVR mySmartUSB light
Bus 003 Device 001: ID 1d6b:0002 Linux Foundation 2.0 root hub
Bus 001 Device 001: ID 1d6b:0002 Linux Foundation 2.0 root hub
Bus 002 Device 002: ID 0e0f:0002 VMware, Inc. Virtual USB Hub
Bus 002 Device 003: ID 0e0f:0003 VMware, Inc. Virtual Mouse
Bus 002 Device 001: ID 1d6b:0001 Linux Foundation 1.1 root hub
passoni@passoni:~$
```

图 2-3-1 lsusb 查看已连接设备

② 配置 M10 ROS 驱动包

这里我用 sublime 打开 M10 的 ROS 包，方便查看文件目录结构和文件内容。我们打开 lsldar_x10_driver/launch 下的 lsldar_x10_serial.launch 进行查看，这个 launch 文件也是我们之后要运行的启动雷达的文件。

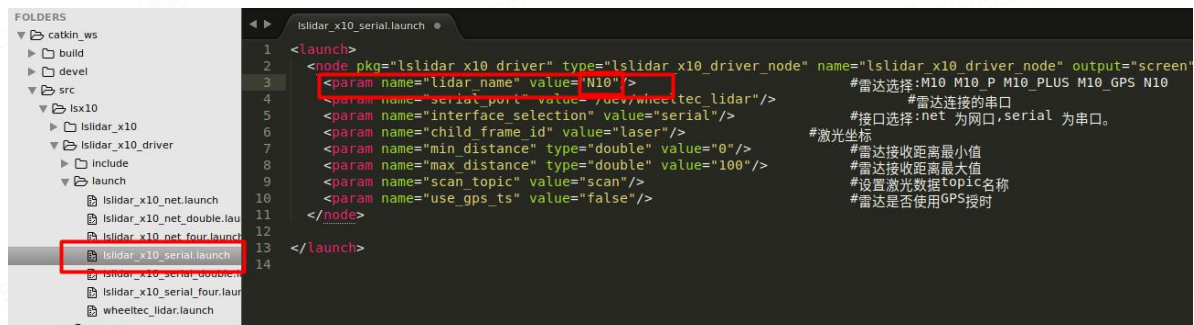


图 2-3-2 查看 lsldar_x10_serial.launch

我们先将参数 lidar_name 修改成我们目前雷达的型号, 根据实际修改成 M10 或者 M10_P。

用 lsldar_x10_serial.launch 开启 M10 雷达前需要先确认雷达在 ubuntu 中的串口。我们可以通过 `ll /dev/ | grep ttyUSB` 命令查看。一般情况下串口名为 ttyUSB0。这里我通过创建串口别名将其串口名重映射为 wheeltec_lidar (这一步不是必须的)。

```
passoni@passoni:~$ ll /dev/ | grep ttyUSB
crwxrwxrwx 1 root dialout 188, 0 Nov 24 09:46 ttyUSB0
lrwxrwxrwx 1 root root      7 Nov 24 09:46 wheeltec_lidar -> ttyUSB0
passoni@passoni:~$
```

图 2-3-3 查看串口名

注: 串口别名的创建可以参考雷达资料包根目录中的“wheeltec_udev.sh”脚本。本质是通过设备属性来定位设备并为其创建命名规则。

接下来将 lsldar_x10_serial.launch 中对应的 serial_port 的值改为雷达的串口名, 串口名即为我们上图查看到的, 这个参数一般不需要修改, 一般默认的串口名就是“/dev/ttyUSB0”。这里我进行了串口别名, 所以也可以将其改为 wheeltec_lidar。

```
<launch>
<node pkg="lsldar_x10_driver" type="lsldar_x10_driver_node" name="lsldar_x10_driver_node" output="screen">
  <param name="lidar_name" value="M10"/> #雷达选择:M10 M10_P M10_PLUS M10_GPS N10
  <param name="serial_port" value="/dev/wheeltec_lidar"/> #雷达连接的串口
  <param name="interface_selection" value="serial"/> #接口选择:net 为网口,serial 为串口。
  <param name="child_frame_id" value="laser"/> #激光坐标
  <param name="min_distance" type="double" value="0"/> #雷达接收距离最小值
  <param name="max_distance" type="double" value="100"/> #雷达接收距离最大值
  <param name="scan_topic" value="scan"/> #设置激光数据Topic名称
  <param name="use_gps_ts" value="false"/> #雷达是否使用GPS授时
</node>
```

图 2-3-4 修改串口参数

③ 屏蔽雷达角度

根据自己实际需要选择屏蔽雷达角度。(此步非必须)

将下列代码复制粘贴在 lsldar_x10_serial.launch 中。

```
<!-- 镭神 M10 N10 雷达参数 -->
<param name="lsldar_x10_driver_node/truncated_mode" value="0"/>
<!--0:不屏蔽角度 1:屏蔽角度-->
<rosparam param="lsldar_x10_driver_node/disable_min">[120]</rosparam>
<!--角度左值-->
<rosparam param="lsldar_x10_driver_node/disable_max">[240]</rosparam>
<!--角度右值-->
<!-- 镭神 M10 N10 雷达参数 -->
```

```

<launch>
<node pkg="lslidar_x10_driver" type="lslidar_x10_driver_node" name="lslidar_x10_driver_node" output="screen">
  <param name="lidar name" value="M10"/> #雷达选择:M10 M10 P M10_PLUS M10_GPS N10
  <param name="serial port" value="/dev/wheeltec lidar"/> #雷达连接的串口
  <param name="interface_selection" value="serial"/> #接口选择:net 为网口,serial 为串口。
  <param name="child_frame_id" value="laser"/> #激光坐标
  <param name="min_distance" type="double" value="0"/> #雷达接收距离最小值
  <param name="max_distance" type="double" value="100"/> #雷达接收距离最大值
  <param name="scan_topic" value="scan"/> #设置激光数据topic名称
  <param name="use_gps_ts" value="false"/> #雷达是否使用GPS授时
</node>

<!-- 镭神M10 N10雷达参数 -->
<param name="lslidar_x10_driver_node/truncated_mode" value="0"/> <!-- 0:不屏蔽角度 1:屏蔽角度 -->
<rosparam param="lslidar_x10_driver_node/disable_min">[120]</rosparam> <!-- 角度左值 -->
<rosparam param="lslidar_x10_driver_node/disable_max">[240]</rosparam> <!-- 角度右值 -->
<!-- 镭神M10 N10雷达参数 -->
</launch>

```

图 2-3-5 ROS 中 M10 角度屏蔽

在 lslidar_x10_serial.launch 中，雷达多角度屏蔽共由 3 个参数组成，truncated_mode 及 disable_min 和 disable_max。其中 truncated_mode 为一个开关，truncated_mode 为 0 时默认不屏蔽角度，当它为 1 时开启雷达的多角度屏蔽，disable_min 和 disable_max 中可以填入一组或者多组角度，假如这里填入三组数据：disable_min=[40,90,320] disable_max=[50,120,360] 表示雷达扫描时屏蔽 40° ~50°，90° ~120°，320° ~360° 的区间。

多角度屏蔽时，以雷达正前方为 0 度角方向，扫描角度顺时针增加，如图所示。

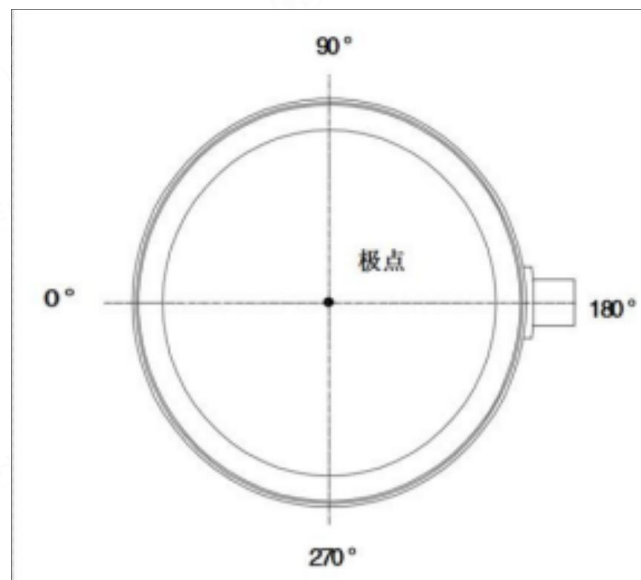


图 2-3-6 ROS 中 M10 角度及坐标系

④ 启动雷达

在配置完 launch 文件后，我们打开一个终端，执行以下命令：

```
roslaunch lslidar_x10_driver lslidar_x10_serial.launch
```


成功启动后终端会出现以下字样。

```
SUMMARY
=====

PARAMETERS
* /lslidar_x10_driver_node/child_frame_id: laser
* /lslidar_x10_driver_node/disable_max: [240]
* /lslidar_x10_driver_node/disable_min: [120]
* /lslidar_x10_driver_node/interface_selection: serial
* /lslidar_x10_driver_node/lidar_name: M10
* /lslidar_x10_driver_node/max_distance: 100.0
* /lslidar_x10_driver_node/min_distance: 0.0
* /lslidar_x10_driver_node/scan_topic: scan
* /lslidar_x10_driver_node/serial_port: /dev/wheeltec_lidar
* /lslidar_x10_driver_node/truncated_mode: 0
* /lslidar_x10_driver_node/use_gps_ts: False
* /roscpp: melodic
* /rosversion: 1.14.10

NODES
/
  lslidar_x10_driver_node (lslidar_x10_driver/lslidar_x10_driver_node)

auto-starting new master
process[master]: started with pid [9711]
ROS_MASTER_URI=http://192.168.0.136:11311

setting /run_id to 04d7a944-6b9f-11ed-971e-000c291e1698
process[rosout-1]: started with pid [9722]
started core service [/rosout]
process[lslidar_x10_driver_node-2]: started with pid [9725]
Lidar is M10
port = /dev/wheeltec_lidar, baud_rate = 460800
open_port /dev/wheeltec_lidar OK !
[ INFO] [1669256625.261681965]: Initialised lslidar x10 without error
```

图 2-3-7 成功启动雷达

⑤ 雷达数据查看

在 ROS 中雷达数据的展示形式可以是文本形式，也可以是点云图像形式。

当运行 `lslidar_x10_serial.launch` 启动雷达后，`/lslidar_x10_driver_node` 节点便会在 `ros` 中发布一个 `/scan` 话题。

```
passoni@passoni:~$ rostopic info /scan
Type: sensor_msgs/LaserScan

Publishers:
* /lslidar_x10_driver_node (http://192.168.0.136:45653/)

Subscribers: None

passoni@passoni:~$
```

图 2-3-8 查看 `/scan` 话题发布者

我们可以通过执行 `rostopic echo /scan` 命令直接进行雷达数据的查看。`/scan` 话题中的消息类型是 `LaserScan`，该消息类型的具体描述可以参考以下链接：
http://docs.ros.org/en/api/sensor_msgs/html/msg/LaserScan.html

```

passoni@passoni: ~
文件(F) 编辑(E) 查看(V) 搜索(S) 终端(T) 帮助(H)
passoni@passoni:~$ rostopic echo /scan

header:
  seq: 1611
  stamp:
    secs: 1649655850
    nsecs: 406630537
  frame_id: "laser"
angle_min: 0.0
angle_max: 0.28318548203
angle_increment: 0.0140249673277
time_increment: 0.000221969748964
scan_time: 0.0992204770446
range_min: 0.0
range_max: 30.0
ranges: [0.6830000281333923, 0.6830000281333923, 0.7139999866485596, 0.713999986
6485596, 0.7599999904632568, 0.7749999761581421, 0.7910000085830688, 0.822000026
7028809, 0.8529999852180481, 0.8690000176429749, inf, 0.8989999890327454, 0.9300
00071525574, 0.9610000252723694, 0.9769999980926514, 1.0080000162124634, 1.0379
999876022339, 1.8910000324249268, 1.8760000467300415, 1.8760000467300415, 1.8760
00467300415, 0.3790000081062317, inf, 0.34700000286102295, 0.4269999861717224,
inf, inf, 0.2980000078678131, 0.3149999976158142, 0.34700000286102295, 0.3790000
081062317, 0.39500001072883606, 0.4269999861717224, 0.45899999141693115, 0.49099
9966621399, 0.5249999761581421, 0.5400000214576721, 0.5720000267028809, 0.60500

```

图 2-3-9 查看雷达话题数据

除了直接通过话题查看文本形式的雷达数据，我们也可以通过 rviz 来查看雷达的点云图像。直接在终端输入 rviz 并执行。打开 rviz 后，先将 Fixed Frame 后面对应的值修改为 lslidar_x10_serial.launch 中 frame_id 对应的 value 值。

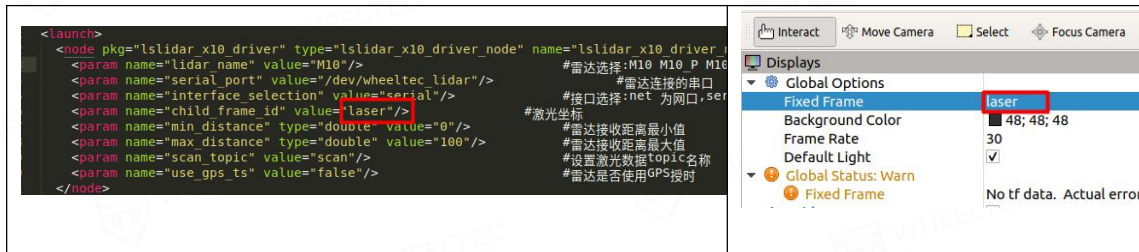


图 2-3-10 修改 Fixed Frame

之后点击 Rviz 左下角的 Add 按钮，在弹出的窗口中点击 By topic 选中 /scan 话题下的 LaserScan 并点击 OK。

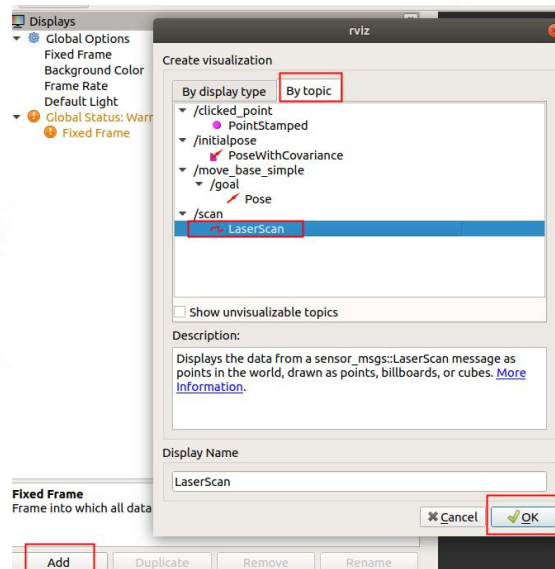


图 2-3-11 添加 LaserScan

成功添加 LaserScan 后我们便可以在 Rviz 中看到这样的雷达点云图像。

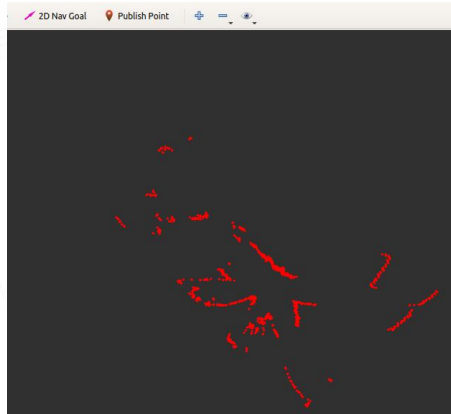


图 2-3-12 Rviz 显示雷达点云

⑥ 雷达数据在 ros 的应用

当/lslidar_x10_driver_node 节点通过话题发布 M10 雷达的数据之后，我们就可以通过订阅它发出的/scan 话题在不同的功能中获取使用雷达的数据。比如建图、导航避障、雷达跟随等等。下图就是一个在 python 中订阅/scan 话题的示例。

```
def __init__(self):
    self.lastScan=None
    self.winSize = rospy.get_param('~winSize')
    self.deltaDist = rospy.get_param('~deltaDist')
    self.scanSubscriber = rospy.Subscriber('scan', LaserScan, self.registerScan)
    self.positionPublisher = rospy.Publisher('object_tracker/current_position', Pose, queue_size=10)
    self.infoPublisher = rospy.Publisher('object_tracker/info', StringMsg, queue_size=10)
```

图 2-3-13 订阅/scan 话题

话题名默认情况下为/scan，但我们也可以通过手动修改 lslidar_x10_serial.launch 中的 scan_topic 的值来修改启动后发布的雷达话题名。

```
<launch>
  <node pkg="lsllidar_x10_driver" type="lsllidar_x10_driver_node" name="lsllidar_x10_driver_node" output="screen">
    <param name="lidar_name" value="M10"/> #雷达选择:M10 M10_P M10_PLUS M10_GPS N10
    <param name="serial_port" value="/dev/wheeltec lidar"/> #雷达连接的串口
    <param name="interface_selection" value="serial"/> #接口选择:net 为网口,serial 为串口。
    <param name="child_frame_id" value="laser"/> #激光坐标
    <param name="min_distance" type="double" value="0"/> #雷达接收距离最小值
    <param name="max_distance" type="double" value="100"/> #雷达接收距离最大值
    <param name="scan_topic" value="scan"/> #设置激光数据topic名称
    <param name="use_gps_ts" value="false"/> #雷达是否使用GPS授时
  </node>
```

图 2-3-14 修改雷达话题名