

TG SERIES USER MANUAL



CONTENTS

1	YDLIDAR TG SERIES LIDAR DEVELOPMENT KIT	1
1.1	Development Kit	1
2	USAGE UNDER WINDOWS	1
2.1	Device Connection	1
2.2	Driver Installation	3
2.3	Evaluation Software Usage	5
2.3.1	Start Scanning	6
2.3.2	Data Presentation	6
2.3.3	Data Storage	7
2.3.4	Scan Frequency	7
2.3.5	Angle Calibration	7
2.3.6	Smoothing	7
3	LINUX ROS OPERATION	8
3.1	Device Connection	8
3.2	Compile and Install YDLidar-SDK	8
3.3	ROS Driver Installation	9
3.4	Run the ydlidar_ros_driver	9
3.5	Modify Scan Angle	10
4	CAUTION	11
4.1	Ambient Temperature	11
4.2	Ambient Lighting	12
4.3	Power Supply	12
5	REVISE	13

1 YDLIDAR TG SERIES LIDAR DEVELOPMENT KIT

The development kit of YDLIDAR TG series lidar (including TG15, TG30, TG50 etc) is an accessory tool provided for performance evaluation and early development of the TG. Through the TG development kit, and with the evaluation software, users can observe point cloud data scanned by TG on your environment or development on the SDK.

1.1 Development Kit

The TG development kit has the following components:



TG Series Lidar



USB Type-C Cable



USB Adapter Board

FIG 1 YDLIDAR TG DEVELOPMENT KIT

CHART 1 YDLIDAR TG SERIES LIDAR DEVELOPMENT KIT DESCRIPTION

Item	Qty.	Description
TG lidar	1	Standard version of the TG series Lidar. The TG has an integrated motor drive for motor control
USB Type-C Cable	1	Use with USB adapter board to connect TG and PC. USB cable is both a power supply cable and a data cable
USB Adapter Board	1	Realize USB to UART, convenient for the rapid interconnection of TG series lidar and PC. In addition, it provides Micro USB power interface (PWR) for auxiliary power supply

Note: USB Adapter board has two USB TYPE C interface: USB_DATA、USB_PWR.

USB_DATA: Data powered interface. In most cases, this interface can be used to meet power and communication requirements.

USB_PWR: Auxiliary power supply interface. The USB interface of some development platforms has weak current drive capability. At this time, auxiliary power supply can be used.

2 USAGE UNDER WINDOWS

2.1 Device Connection

When TG is evaluated and developed under windows, TG and PC need to be interconnected. The specific process is as follows:

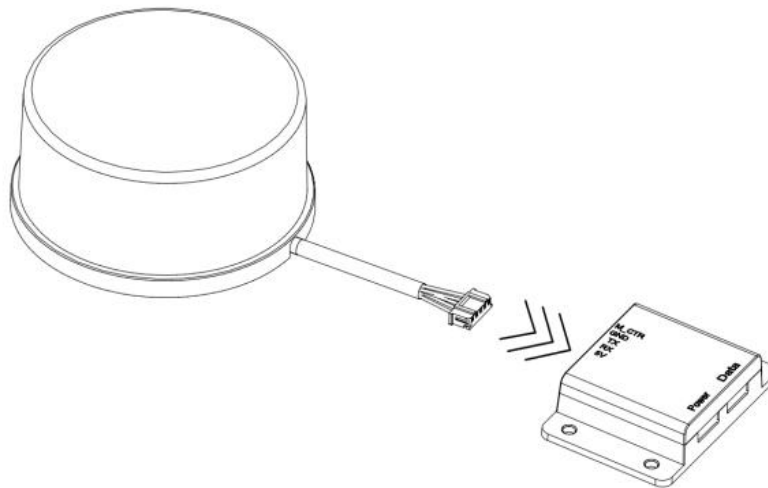


FIG 2 YDLIDAR TG DEVICE CONNECTION STEP 1

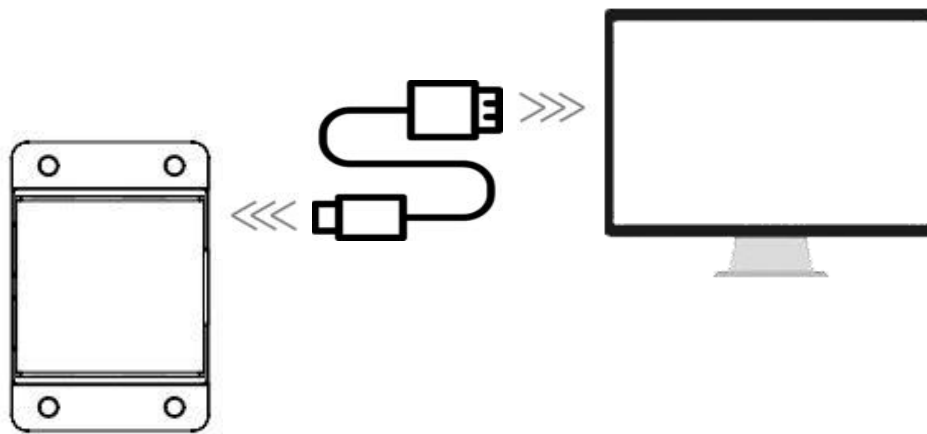


FIG 3 YDLIDAR TG DEVICE CONNECTION STEP 2

Connect the adapter board with TG first, then connect the USB cable to the USB port of the adapter board and the PC. Note that the Type-C interface of the USB cable is connected to the USB_DATA of the USB interface board, and the idle mode is used after TG is powered on. The motor does not rotate.

The drive current of USB interface of some development platforms or PC is not sufficient. TG Series need to be connected to the auxiliary power supply of +5V, otherwise the lidar will be abnormal.

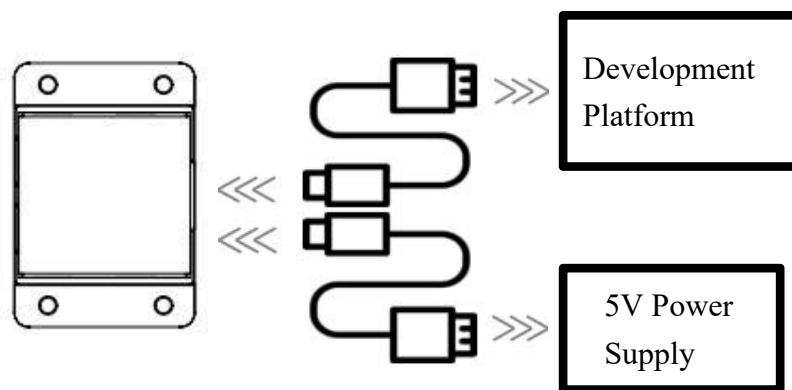


FIG 4 YDLIDAR TG AUXILIARY POWER SUPPLY

2.2 Driver Installation

To evaluate and develop the TG under Windows, you need to install the serial port driver of the USB adapter board. The USB adapter board of this kit adopts CP2102 chip to realize serial port (UART) to USB signal conversion. Its driver can be downloaded from our official website.

<https://www.ydlidar.com/download/category/lidar-sensor>

After unzip the driver package, run the CP2102's Windows driver installation file (exe file under CP210x_VCP_Windows). Please select the 32-bit version (x86) or 64-bit version (x64) installation program according to the version of the windows operating system.

x64	2013/10/25 11:39	文件夹	
x86	2013/10/25 11:39	文件夹	
CP210xVCPInstaller_x64.exe	2013/10/25 11:39	应用程序	1,026 KB
CP210xVCPInstaller_x86.exe	2013/10/25 11:39	应用程序	901 KB
dpinst.xml	2013/10/25 11:39	XML 文档	12 KB
ReleaseNotes.txt	2013/10/25 11:39	文本文档	10 KB
SLAB_License_Agreement_VCP_Windo...	2013/10/25 11:39	文本文档	9 KB
slabvcp.cat	2013/10/25 11:39	安全目录	12 KB
slabvcp.inf	2013/10/25 11:39	安装信息	5 KB

FIG 5 YDLIDAR TG DRIVER VERSION SELECTION

Double-click the exe file and follow the prompts to install it.



FIG 6 YDLIDAR TG DRIVER INSTALLING

After the installation is complete, you can right-click on [My Computer] and select [Properties]. On the open [System] screen, select [Device Manager] from the left menu to access the [Device Manager].

Expand [Port] to see the serial port name corresponding to the identified USB adapter, that is, the driver installation is successful. The following figure shows COM3. (Note that the port must be checked in case of TG and PC interconnection).

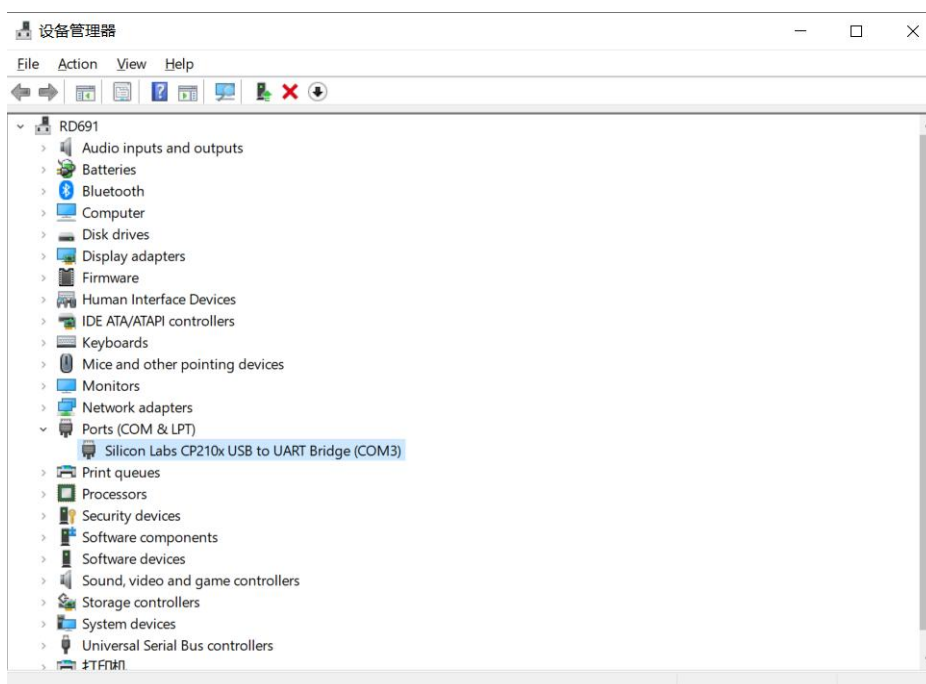


FIG 7 YDLIDAR TG DRIVER INSTALLATION CHECK

2.3 Evaluation Software Usage

YDLIDAR provides EaiLidarTest, a point cloud data LidarClient software for TG real-time scanning. Users can intuitively observe the TG scanning effect chart. Real-time point cloud data and real-time scanning frequency are provided on YDLIDAR. At the same time, the version information of TG can be read. LidarClient software download link:

<https://www.ydlidar.com/download/category/lidar-sensor>

Before using the YDLIDAR software, make sure that the TG USB adapter board serial port driver is installed successfully, and interconnect the TG with the USB port of the PC. Run the evaluation software: EaiLidarTest.exe, select the corresponding serial port number and model number. Meanwhile, users could choose language on the top right corner.

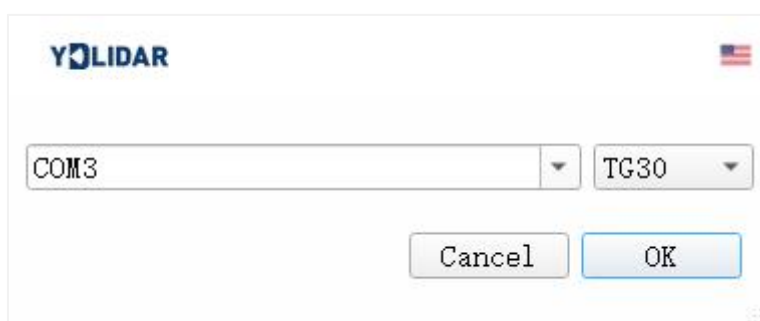


FIG 8 YDLIDAR TG EVALUATION SOFTWARE

If the connection is correct, you will see the following screen:

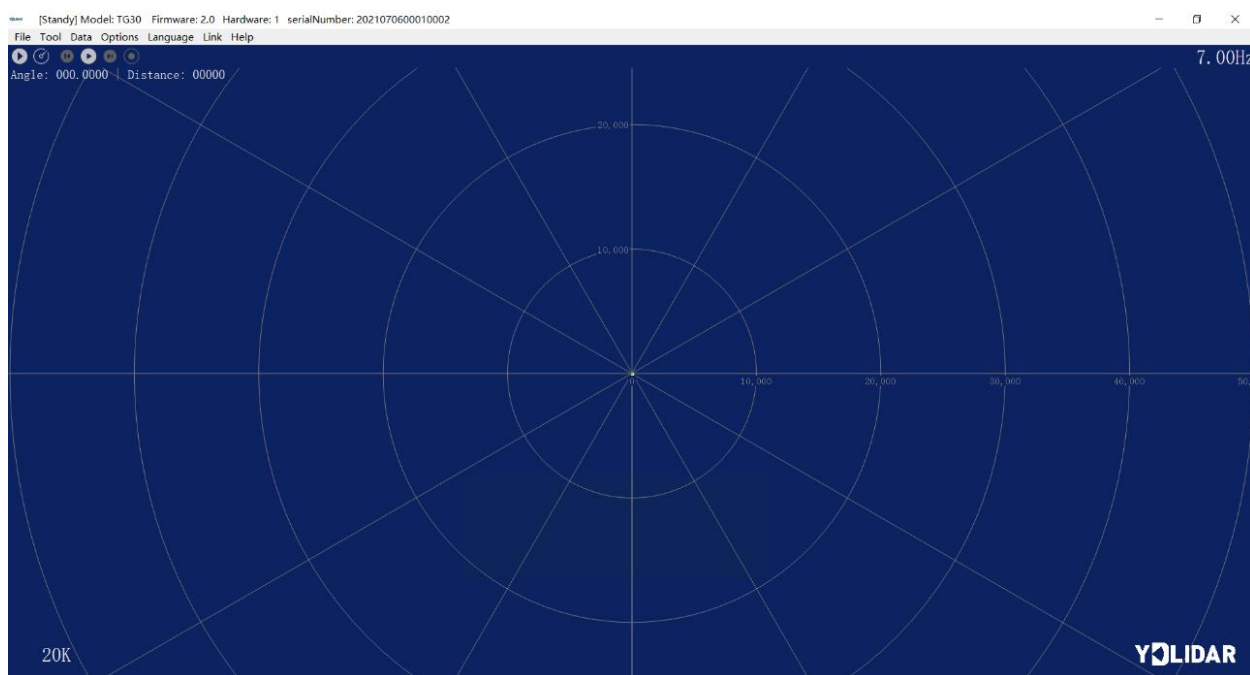


FIG 9 POINTCLOUD VIEWER INTERFACE

2.3.1 Start Scanning

Click  to start scanning and display the environment point cloud. Click  to stop it. as shown below:

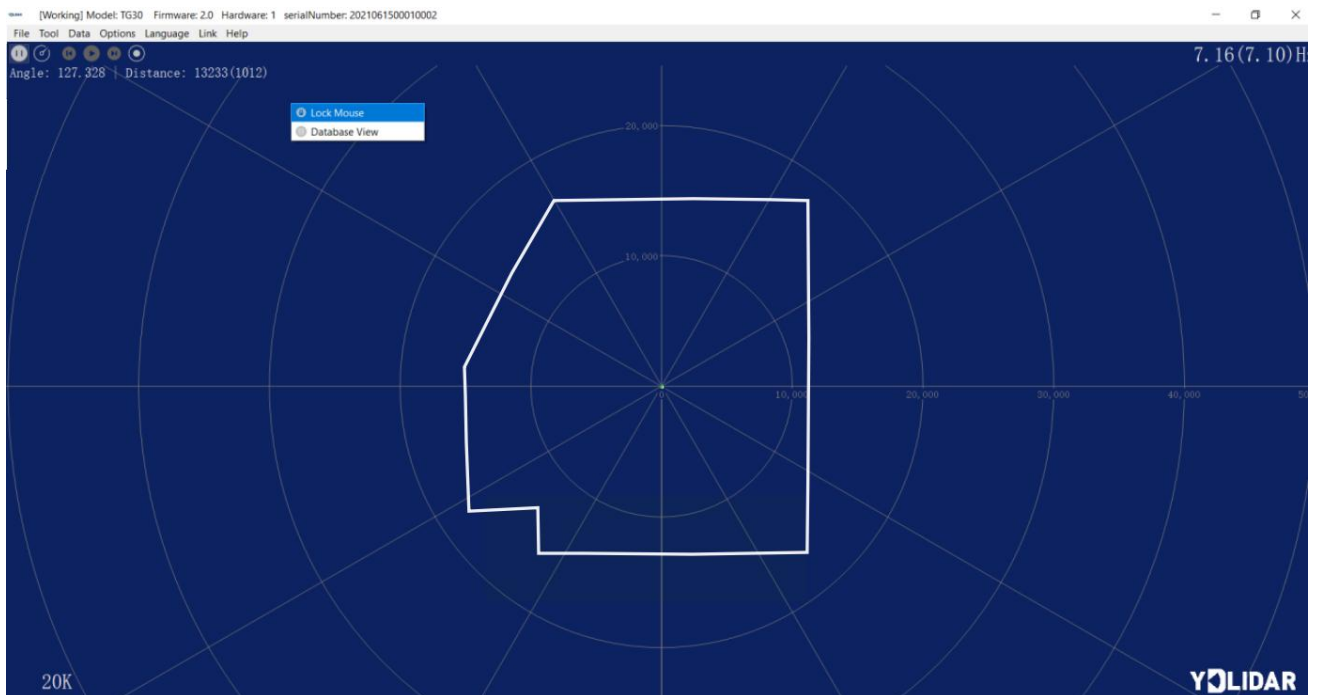


FIG 10 LIDAR SCANNING POINT CLOUD DISPLAY

2.3.2 Data Presentation

F(unit: mm)

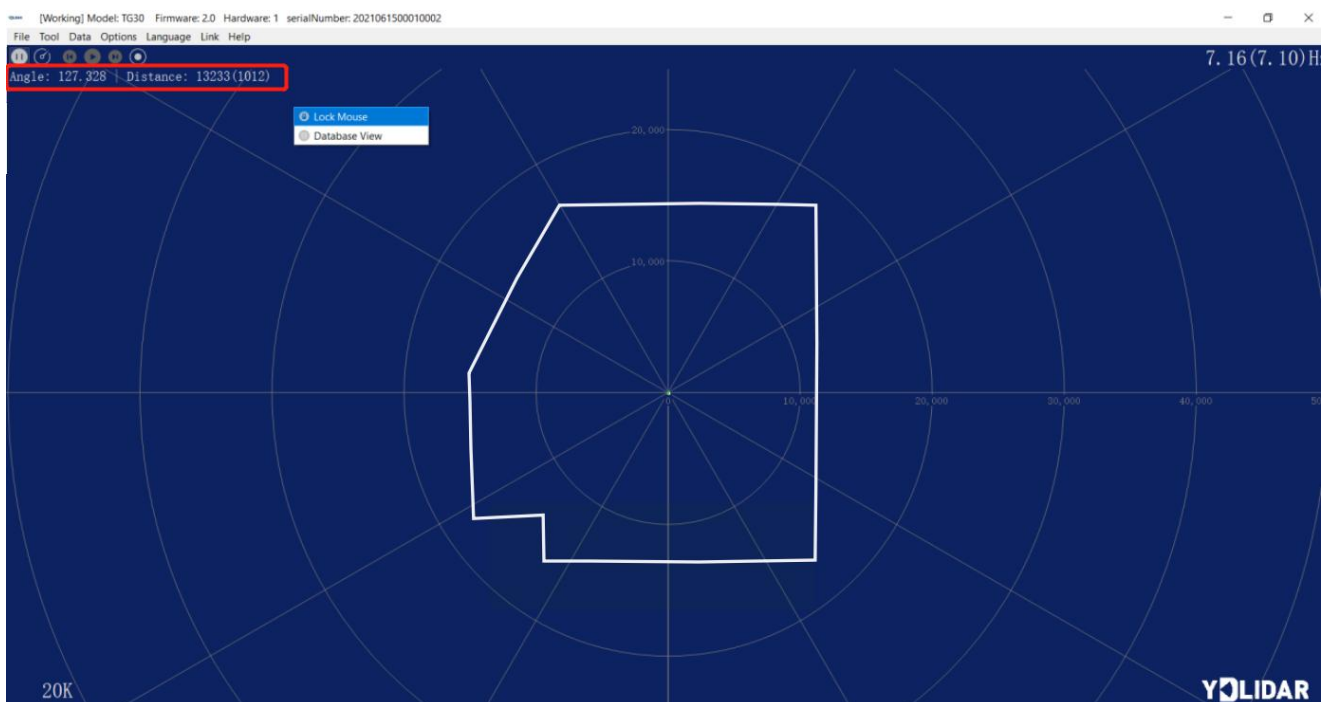


FIG 11 OBTAIN LIDAR ANGLE & RANGE DATA

2.3.3 Data Storage

During lidar scanning, click [File] in the main menu, select [Export to Excel], and save point cloud data according to the prompts. Then the system will save the point cloud information scanned in a circle in Excel format.



FIG 12 SAVE DATA

2.3.4 Scan Frequency

⌚ is used to adjust the scanning frequency (motor speed) of the lidar. Click and drag to adjust according to the demand. When the lidar is in the scanning state, you need to click the scan control again after adjusting the scan frequency.



FIG 13 SCAN FREQUENCY ADJUSTMENT

2.3.5 Angle Calibration

In the process of mechanical assembly of the lidar, the user may have deviation of zero Angle. At this time, the Angle calibration function of the client can be used to calibrate according to the actual demand. The specific operation is as follows:

- 1) Turn on the calibration function

During lidar scanning, click [Tools] in the main menu and select [Zero Adjustment].

These controls ⌚ ⌚ will be displayed.

- 2) Adjust angle

Click control ⌚ to adjust the Angle to the appropriate position.

- 3) Save configuration

After adjustment, click control ⌚, the system will automatically save the calibration parameters, and the calibration will take effect after saving.

2.3.6 Smoothing

Click [Tools] in the main menu, then select [Filtering] to add the lidar data filtering algorithm.

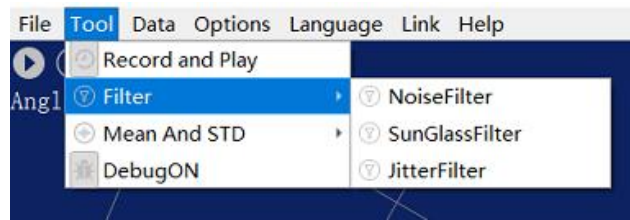


FIG 14 YDLIDARTG SERIES RUN EVALUATION SOFTWARE

Note: please click [Help] and select [More Information] to learn more about how to use the EaiLidarTest. (e.g. display mean and standard deviation, playback and recording, debugging, etc.)

3 LINUX ROS OPERATION

There are many Linux versions, this article only uses Ubuntu 18.04, Melodic version ROS as an example.

SDK driver address:

<https://github.com/YDLIDAR/YDLidar-SDK>

ROS driver address:

https://github.com/YDLIDAR/ydlidar_ros_driver

3.1 Device Connection

Under Linux, the TG and PC interconnect processes are consistent with those under Windows. See Device Connection under Window.

3.2 Compile and Install YDLidar-SDK

ydlidar_ros_driver depends on the YDLidar-SDK library. If you have never installed the YDLidar-SDK library, or it has expired, you must first install the YDLidar-SDK library. If you have the latest version of YDLidar-SDK installed, please skip this step, then go to the next step.

```
$ git clone https://github.com/YDLIDAR/YDLidar-SDK.git
$ cd YDLidar-SDK/build
$ cmake ..
$ make
$ sudo make install
```

3.3 ROS Driver Installation

- 1) Cloning GitHub's ydlidar_ros_driver Package:

```
$ git clone https://github.com/YDLIDAR/ydlidar_ros_driver.git  
ydlidar_ws/src/ydlidar_ros_driver
```

- 2) Build the ydlidar_ros_driver software package:

```
$ cd ydlidar_ws  
  
$ catkin_make
```

- 3) Package environment Settings:

```
$ source ./devel/setup.sh
```

Note: Add a permanent workspace environment variable. It will be very convenient if ROS environment variables are automatically added to your bash session every time you start a new shell:

```
$ echo "source ~/ydlidar_ws/devel/setup.bash" >> ~/.bashrc  
$ source ~/.bashrc
```

- 4) Verify that your package path is set, echo the ROS_PACKAGE_PATH variable.

```
$ echo $ROS_PACKAGE_PATH
```

You should see something like this: /home/tony/ydlidar_ws/src:/opt/ros/melodic/share

- 5) Create Serial Port Alias [Optional]

```
$ chmod 0777 src/ydlidar_ros_driver/startup/*  
  
$ sudo sh src/ydlidar_ros_driver/startup/initenv.sh
```

Note: After completing the previous operation, re-insert the LiDAR again.

3.4 Run the ydlidar_ros_driver

- 1) Run ydlidar_ros_driver with startup file, as shown below:

```
# TG15, TG30, TG50  
  
$ roslaunch ydlidar_ros_driver TG.launch
```

- 2) RVIZ scan result checking

```
$ roslaunch ydlidar_ros_driver lidar_view.launch

### The scan results are visible in RVIZ
```

Note: Take G4 as an example by default. If you use other lidars, change the lidar.launch file in lidar_view.launch file to the corresponding **.launch. (If using TG series lidar, change to TG.Launch)



```
<launch>
<include file="$ (find ydlidar_ros_driver)/launch/lidar.launch" />
<node name="rviz" pkg="rviz" type="rviz" args="-d $ (find ydlidar_ros_driver)/launch/lidar.rviz" />
</launch>
```

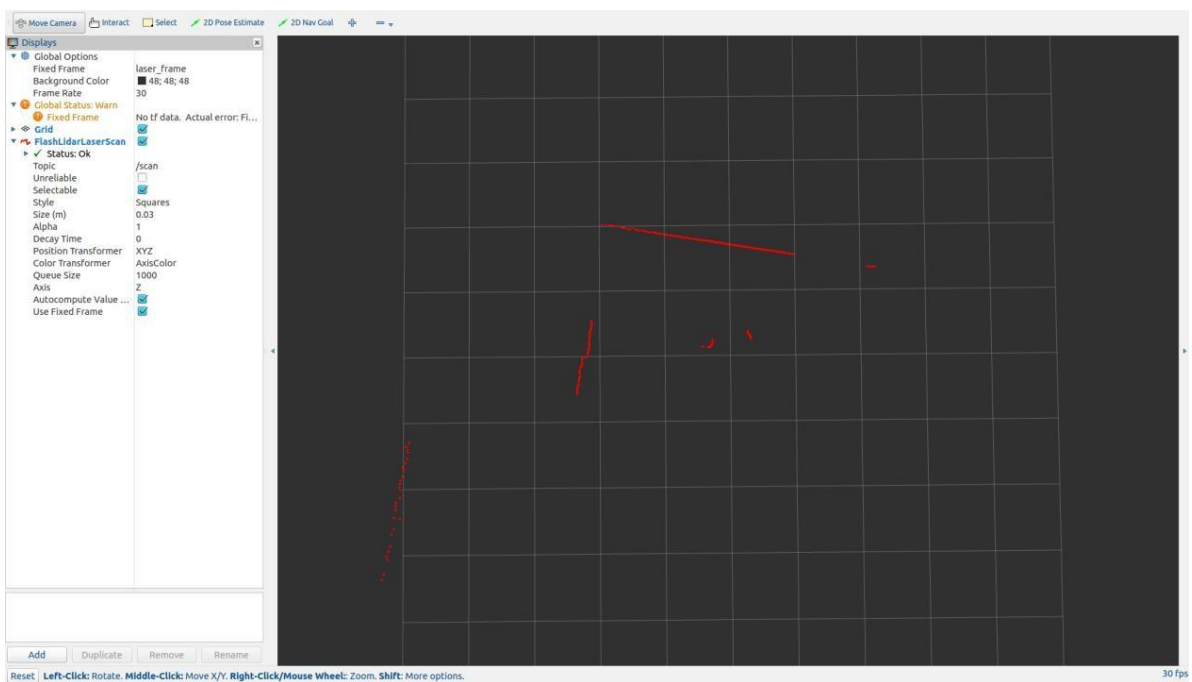


FIG 15 YDLIDAR TG RVIZ

3.5 Modify Scan Angle

The scanning data seen by running the launch file is displayed by default with 360-degree data. To modify the display range, you need to modify the configuration parameters in the launch file. The specific operation is as follows:

- 1) Go to TG.launch directory and use vim to edit lidar.launch, the contents are as shown in the figure:

```
$ vim TG.launch
```

```
<launch>
  <node name="ydlidar_lidar_publisher" pkg="ydlidar_ros_driver" type="ydlidar_ros_driver_node" output="screen" respawn="false" >
    <!-- string property -->
    <param name="port" type="string" value="/dev/ydlidar"/>
    <param name="frame_id" type="string" value="laser_frame"/>
    <param name="ignore_array" type="string" value=""/>

    <!-- int property -->
    <param name="baudrate" type="int" value="512000"/>
    <!-- 0:TYPE_TOF, 1:TYPE_TRIANGLE, 2:TYPE_TOF_NET -->
    <param name="lidar_type" type="int" value="0"/>
    <!-- 0:YDLIDAR_TYPE_SERIAL, 1:YDLIDAR_TYPE_TCP -->
    <param name="device_type" type="int" value="0"/>
    <param name="sample_rate" type="int" value="20"/>
    <param name="abnormal_check_count" type="int" value="4"/>

    <!-- bool property -->
    <param name="resolution_fixed" type="bool" value="true"/>
    <param name="auto_reconnect" type="bool" value="true"/>
    <param name="reversion" type="bool" value="true"/>
    <param name="inverted" type="bool" value="true"/>
    <param name="isSingleChannel" type="bool" value="false"/>
    <param name="intensity" type="bool" value="false"/>
    <param name="support_motor_dtr" type="bool" value="false"/>
    <param name="invalid_range_is_inf" type="bool" value="false"/>

    <!-- float property -->
    <param name="angle_min" type="double" value="-180" />
    <param name="angle_max" type="double" value="180" />
    <param name="range_min" type="double" value="0.01" />
    <param name="range_max" type="double" value="50.0" />
    <param name="frequency" type="double" value="10.0" />
  </node>
  <node pkg="tf" type="static_transform_publisher" name="base_link_to_laser4"
    args="0.0 0.0 0.2 0.0 0.0 0.0 /base_footprint /laser_frame 40" />
</launch>
```

FIG 16 TG.LAUNCH FILE

Note: For more information about the file contents, please refer to

https://github.com/YDLIDAR/ydlidar_ros_driver#configure-ydlidar_ros_driver-internal-parameter

- 2) The TG lidar coordinates follow the right-hand rule within ROS, with an angle range of [-180, 180]. "angle_min" is the start angle, and "angle_max" is the endangle. The specific scope needs to be modified according to actual use.

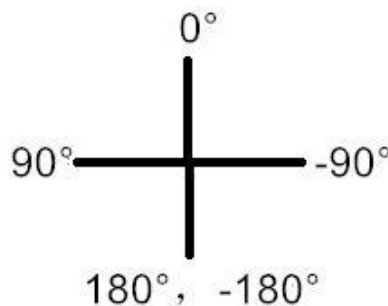


FIG 17 YDLIDAR TG COORDINATES DEFINITION

4 CAUTION

4.1 Ambient Temperature

When the working environment temperature of TG is too high or too low, it will affect the accuracy of the distance measuring system. It may also damage the structure of the

scanning system and reduce the life of the TG lidar. Avoid use in high temperature (>50 degrees Celsius) and low temperature (<0 degrees Celsius) conditions.

4.2 Ambient Lighting

TG use 905nm narrow pulse laser source, optimized optical and electric design enable strong light interference resistance, thus it could meet indoor and outdoor application requirements.

4.3 Power Supply

During the development process, since the drive current of the USB interface of each platform or the USB interface of the computer may be too low to drive the TG, the external power supply of the +5V to the TG needs to be provided through the USB_PWR interface of the USB interface board. It is not recommended to use mobile phone power bank because the voltage ripple of power bank is too large.

5 REVISE

Date	Version	Content
2019-05-28	1.0	Composing a first draft
2020-01-15	1.1	Client description and information modification
2020-06-09	1.2	Client function description and ROS driver modification
2021-08-02	1.3	Adapt EaiLidarTest client, update SDK and ROS tutorial
2025-06-21	1.4	Revised Lidarclient and UART driver address