Falcon Kinetic LiDAR | User Manual

Preface

Product

Falcon Kinetic LiDAR

Manufacturer

INNOVUSION

Legal Disclaimer

Overview

This manual provides instructions for the installation, usage, maintenance, and diagnostic evaluation of Falcon Kinetic LiDAR (hereinafter referred to as "Falcon Kinetic" or "LiDAR"). The contents of the manual are based on the different phases of the LiDAR life cycle, including the installation, configuration, and maintenance of the LiDAR.

The intended users of this manual include project developers (R&D personnel and designers), installers, electrical professionals, safety professionals, and service personnel.

Original document

This document is the original document owned by Innovusion.

Manual description

Although this document covers instructions to handle frequent problems, it is still not guaranteed to fully resolve all problems. If you encounter other issues not covered in the manual, please contact Innovusion staff in time. This manual will be updated when new information becomes available.

Tel : (650)963-9573
E-mail: info@ innovusion.com

Precautions

This user manual provides descriptions of Falcon Kinetic introduction, installation, transferring, usage, maintenance, diagnostic evaluation, disposal, etc., and software instructions.

Considering this is a laser product (1550 nm), please thoroughly read and comprehend all information within this manual before the operation and consider the precautions to avoid danger. Please strictly follow the instructions and steps described in the manual during operation.

Safety notices

Before using the product, please read this manual carefully and strictly follow the relevant instructions.

Please contact Innovusion staff to obtain detailed specifications.

To reduce the risk of electric shock and avoid violating the warranty, please do not disassemble or modify the LiDAR without permission. This product does not contain the user's serviceable parts. Please consult Innovusion's certified service personnel for maintenance and repair.

Service procedures where the laser is energized, is only intended to be performed by Innovusion service personnel or persons trained by Innovusion to perform such service.

Caution

Use of controls, adjustments, or performance of procedures other than those specified for this product may result in hazardous radiation leakage.

Caution

- Class 1 laser product.
- Failure to use, control, adjust or operate LiDAR as specified herein may result in serious radiation hazards.
- The product incorporates a Class 4 fiber laser system which, by itself, may be hazardous. This device incorporates a protective housing and a scan failure safeguard in the machine design such that there is no exposure or human access to laser radiation generated by the fiber laser during operation or maintenance.
- UNDER NO CIRCUMSTANCE shall attempts be made to operate the laser with protective housing removed or the scan failure safeguards overridden.
- When the laser is powered up, service procedures are only intended to be operated by Innovusion service personnel or persons trained and certified by Innovusion.

Caution

This product meets the following standards:

Complies with FDA performance standards for laser products except for conformance with IEC 60825-1 Ed. 3, as described in Laser Notice No. 56, dated May 8, 2019.

- IEC 60825-1:2014
- 21 CFR 1040.10 and 1040.11, except for the deviation of Laser Notice No.50 issued on June 24, 2007.

Device maintenance

This product is made of metal, glass, plastic, and contains sensitive electronic components.

- Do not misuse the product by dropping, burning, piercing, bumping, squeezing, etc.
- Shut off the product immediately once it is hit or dropped. Please contact Innovusion staff for technical support.
- If there is any possibility that the product may have been damaged, please stop using it immediately to prevent personnel injury.
- Do not touch the LiDAR window with your hands in case of performance degradation.
- If the LiDAR window is stained, please clean the product as described in the **Cleaning** section of this manual.
- It is strictly forbidden for users to disassemble or convert the device without permission. Dismantling this product may result in product damage, loss of waterproof performance, or personal injury.

Electrical safety

- Always use the connecting cable and power adapter provided or specified by Innovusion.
- Using damaged cables or adapters in a humid environment may lead to fire, electric shock, personal injury, product damage, or other property losses.
- Voltage surges due to lightning may lead to the degeneration of electrical systems over an extended period.

Heat dissipation

- Long-time contact with the hot surface of the product may cause personal discomfort or injury.
- To avoid heat accumulation, please ensure the device is in well-ventilated environment.
- LiDAR may generate a high amount of heat after prolonged operations. It is recommended to shut off the power for a few minutes before touching it.

Operating environment

- Do not subject the product to intense vibration.
- Do not look directly at the transmitting laser through a magnifying device (such as a microscope and magnifying glass).
- Do not look directly at the transmitting laser through an electronic device.
- Do not place this product near flammable and explosive materials.
- Do not expose this product to areas with explosive air, such as areas with a high concentration of flammable chemicals or saturated vapor.
- Do not expose this product to an environment with high-density industrial chemicals, such as easily vaporized liquefied gas (e.g., helium), to avoid performance degradation.

Radio frequency interference

Before the operation, please read the product label's certification and safety information. Although the product's design, testing, and manufacturing comply with the relevant provisions of RF energy radiation, the radiation from the product may still lead to the failure of other electronic equipment.

Medical device interference

Some components and radio devices contained in this product will emit electromagnetic fields that may interfere with medical equipment, such as cochlear implants, pacemakers, and defibrillators. Consult your doctor and medical equipment manufacturer for specific information, e.g., whether you need to keep a safe distance from this product. If there's any possibility that this product is interfering with your medical equipment, please stop using it immediately.

1 Product description

1.1 Product introduction

Product overview

Falcon Kinetic LiDAR (hereinafter referred to as "Falcon K" or "LiDAR" or "FK") is an industry-leading automotive-grade LiDAR developed by Innovusion. It can detect objects as far as 500 meters and objects with 10% reflectivity up to 250 meters. Falcon can maximize point density in an adjustable region of interest (ROI) to better track objects on the road ahead. High performance LiDAR like Falcon is key to achieve L2+ safe autonomous driving.

Product features

- 500m ultra-long detection range, image-grade ultra-high resolution
- Flexible and adjustable ROI
- 1550nm laser wavelength, eye-safe
- Automotive-grade Falcon is in mass production stage

1.2 Time of flight

LiDAR performs based on the time-of-flight (ToF) methodology.

1. The LiDAR emits a light pulse of short duration and narrow divergence.
2. Upon hitting an object, the emitted light will undergo scattered reflection.
3. Some of the reflected light will return to the unit and be detected by the LiDAR's optical sensor.
4. The object's distance is calculated by measuring the time between the emission of the light pulse and the detection of the reflected light. The object's location is known since the angular direction of the emitted light pulse is known.
5. The LiDAR emits light pulses in multiple directions sequentially, thereby constructing a 3-dimensional map of the system's surroundings.

The distance is expressed as:

$$d = \frac{ct}{2}$$

d: distance   c: speed of flight   t: flight time of the laser pulse

1.3 Label description

Table 1 Label description

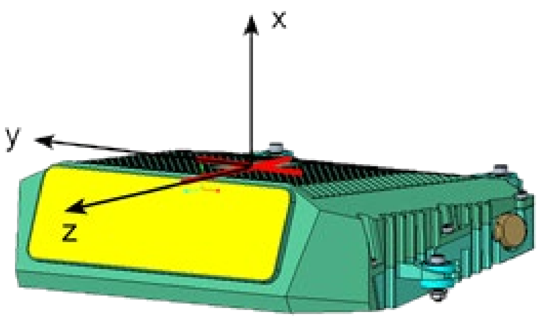
Serial No.	Name	Serial No.	Name
1	Company's LOGO	2	Product name
3	Product model	4	Rated input
5	Serial number	6	Manufacture date
7	MAC address		

1.4 LiDAR laser and coordinate system

Falcon K is a semi-solid-state LiDAR with a laser light source wavelength of 1550nm.

The three-dimensional coordinate system is defined as follows.

- X-axis is perpendicular to the ground, pointing up.
- Y-axis is parallel to the ground, pointing right.
- Z-axis is parallel to the ground, pointing forward.
- The origin in this coordinate is the optical origin and can be used for calibration reference.



1.5 Scanning pattern

Falcon K is designed with a dual-dimensional scanning pattern.

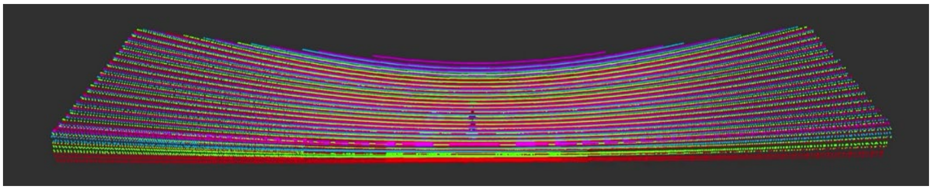
The Field of View (FOV) is the angle through which the LiDAR is sensitive to the electromagnetic radiation.

FOV (H × V): 120°×25°. Angular resolution (H × V): 0.18°×0.24°.

The Region of Interest (ROI) is the angle with high point cloud density, which is about six times higher than the non-ROI in LiDAR. FOV in ROI (H × V): 40°×4.8°. Angular resolution in ROI (H × V): 0.09°×0.08°.

The position of ROI center can be adjusted in real-time within the entire FOV through the computer commands sent to LiDAR.

An example of the LiDAR scanning pattern is shown in the figure below.



1.6 Specifications

Table 2 Specifications

Performance	
Maximum detection range	500 m
Detection range (10% reflectivity)	250 m
Detection range (blind area)	≤ 2 m
FOV (H × V)	120° × 25°
Angular resolution (H × V)	0.18° × 0.24°
FOV in ROI (H × V)	40° × 4.8°, configurable
Angular resolution in ROI (H × V)	0.09° × 0.08°
Distance accuracy	± 5 cm for Lambertian targets
Distance precision	±2 cm (50 m@1sigma)
Vertical scanning lines	1500 lines/s, configurable
Frame rate	10 FPS, configurable
Reliability	
Safety protection	IP67 (body), IP69K (window)
Working temperature	-40°C to +85°C
Laser wavelength	1550 nm
Laser safety	Class-1 (IEC-60825)
Electrical and Data	
Input voltage	9 to 34V DC

Power consumption	about 30 W
Data transmission	1000Base-T1 Ethernet (UDP, TCP)
Data transmission cable length	5 m, configurable
Data output	Point cloud (X,Y,Z) intensity or reflectivity
Time synchronization	NTP, PTP and gPTP
Timestamp accuracy	10 µsec resolution for each data point
Mechanical	
Dimension (W × H × D)	228 ×60.4 × 149.8 mm
Weight	1.7 kg

Note

Specifications are subject to change without notice.

2 Getting started

This chapter provides the procedure to test and verify that your LiDAR is operating properly.

2.1 Delivery list

It is highly recommended to check if there are any loss or damage of the components according to the delivery list before installation.

Table 3 Standard delivery list

No.	Name	No.	Name
1	LiDAR	2	Standard cable

You can select the optional components to customize your delivery list on demand. The list of the optional components is shown below. For more information, please contact innovusion staff for support.

Table 4 Optional delivery list

No.	Name	No.	Name
3	LiDAR connector	4	Metadapter
5/6	Metadapter charger		

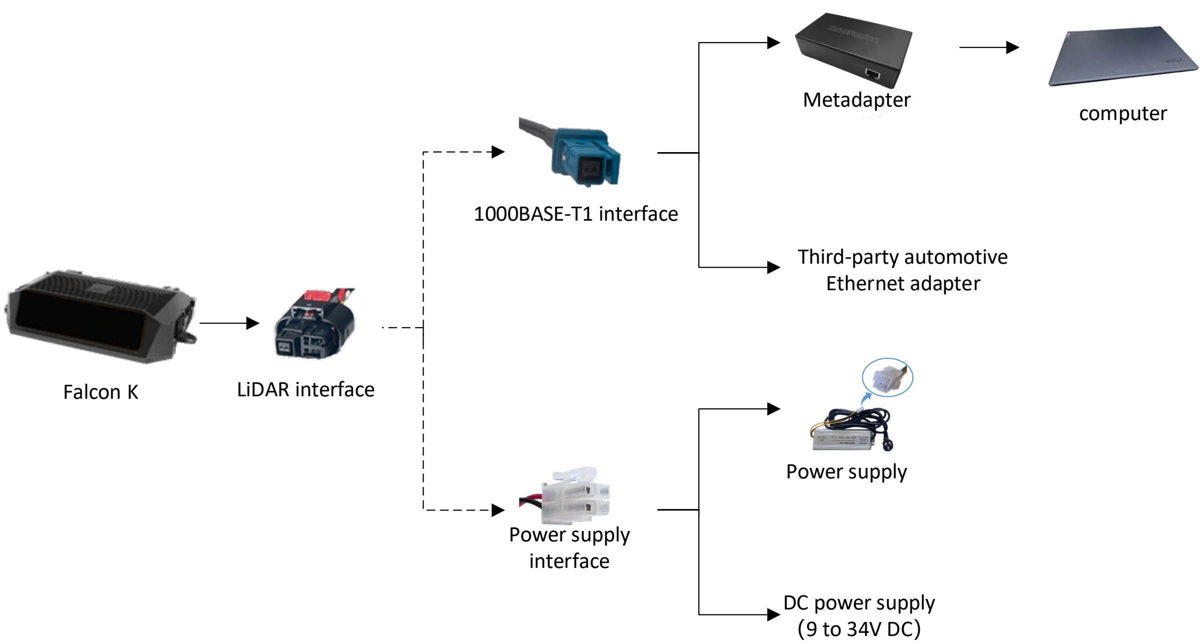
2.2 Verification

Note

It is recommended to check out a new LiDAR's functionality and start processing the LiDAR before mounting.

1. Unpack the LiDAR and the accessories on a workbench or desk. About the component list, see 2.1 Delivery list.
2. Connect the LiDAR to the power supply and startup the LiDAR. About the connection method, see 2.2.1 LiDAR connection. About the LiDAR startup, see 2.2.2 Start the LiDAR.
3. Connect the LiDAR to a test computer. Change the computer IP address to the same subnet with the LiDAR. About the connection method, see 2.2.1 LiDAR connection. About how to change the computer IP address, see 2.2.3 Change the computer IP address.
4. View the point cloud status via ILA. Follow the procedure in 2.2.4 View the point cloud status via ILA.
5. (Optional) View the point cloud status via MetaView. Follow the procedure in 2.2.5 (Optional) View the point cloud status via MetaView.

2.2.1 LiDAR connection



2.2.2 Start the LiDAR

1. Connect the power supply to start the LiDAR.
2. The LiDAR completes initialization and generates data after powering on for 11 to 18 seconds.

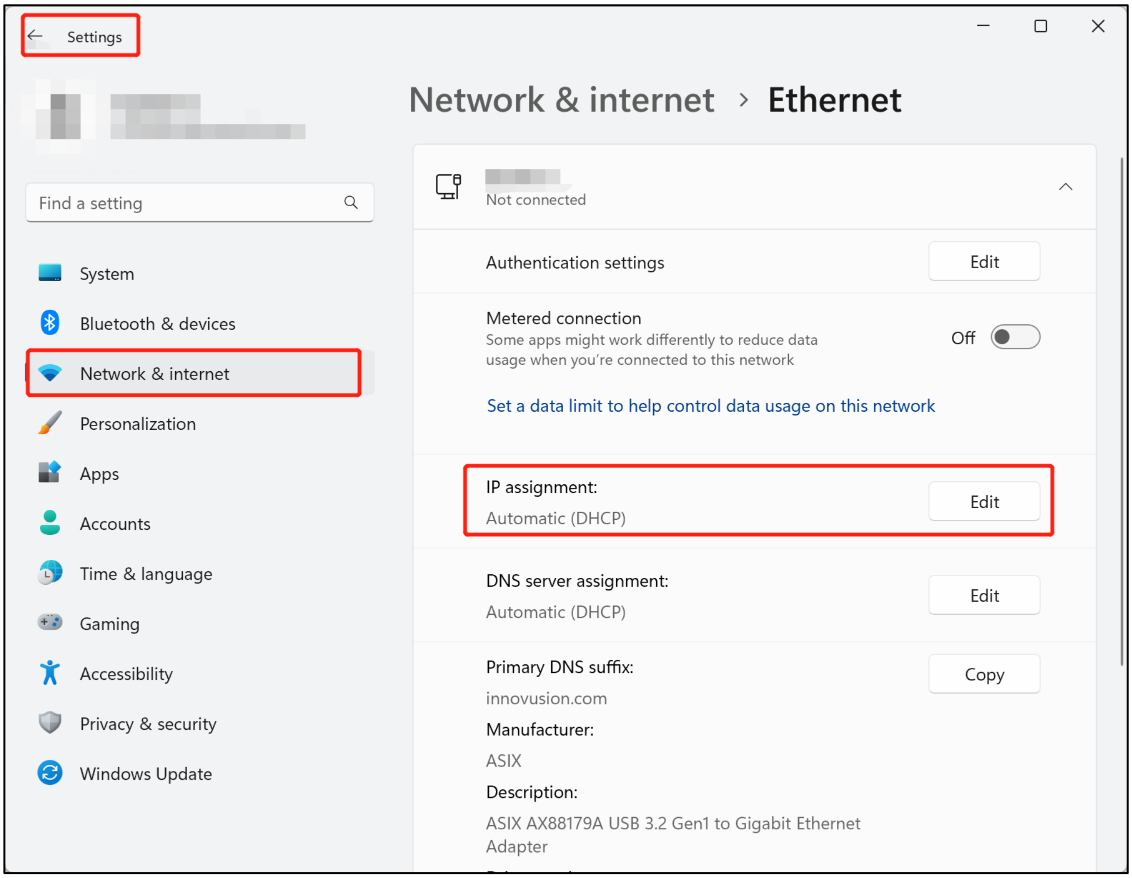
Note

The LiDAR does not have a power switch. It will become operational when power is applied.

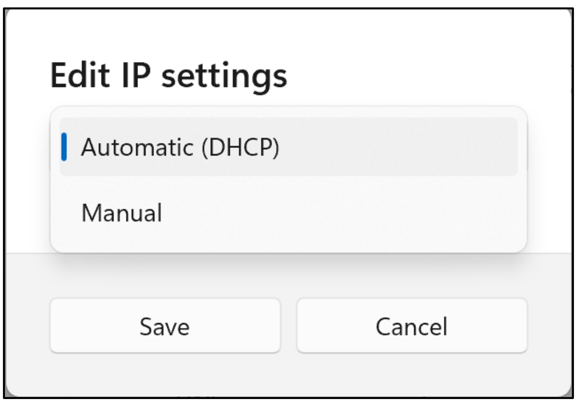
2.2.3 Change the computer IP address

The method of changing the IP address may vary with the operating system. This chapter takes operations on Windows as an example.

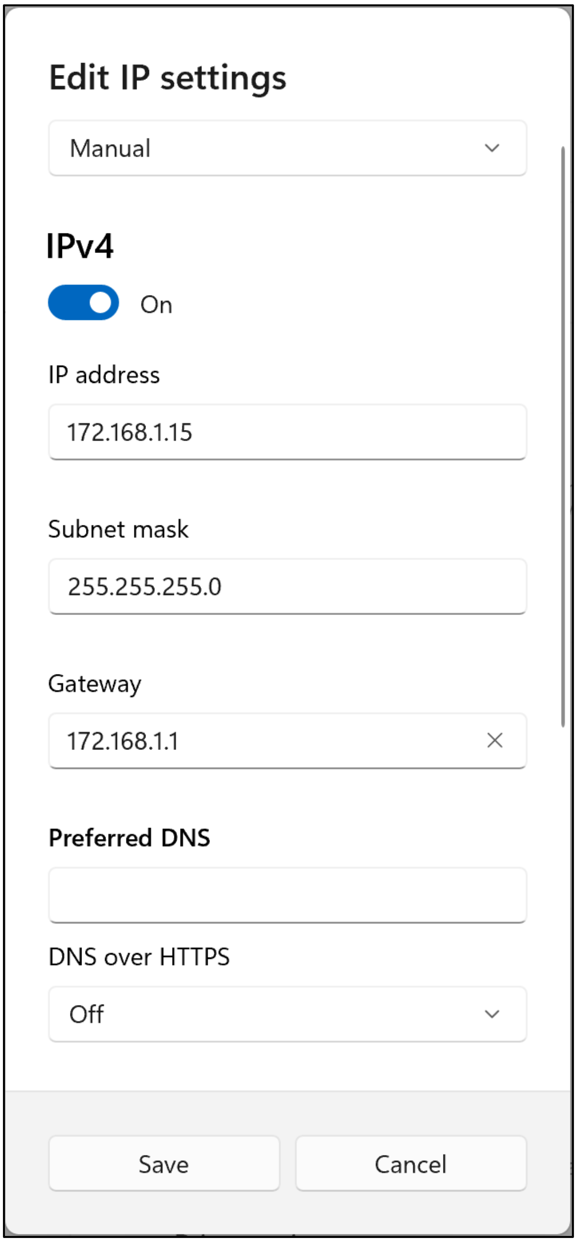
1. Connect the computer to LiDAR and ensure the Ethernet connection.
2. Select **Settings** > **Network & internet** > **Ethernet**.
3. Click **Edit** on the **IP assignment** bar.



4. Select **Manual** in the **Edit IP settings** dialog box.



5. Toggle on **IPv4**. Enter the **IP address**, **Subnet mask** and **Gateway** to make the computer with the same subnet to the LiDAR. See below for example of the settings for a default factory LiDAR. Click **Save** to save the settings.



6. (Optional) Verify the connection.

   It is recommended to check the access to the LiDAR IP address by using the ping command. The return value is shown in the figure below. The default factory LiDAR will usually have an IP address of 172.168.1.10.

```
demo@demo-OMEN-by-HP-Laptop-16-b0xxx:~$ ping 172.168.1.10
PING 172.168.1.10 (172.168.1.10) 56(84) bytes of data.
64 bytes from 172.168.1.10: icmp_seq=70 ttl=64 time=0.448 ms
64 bytes from 172.168.1.10: icmp_seq=71 ttl=64 time=0.222 ms
64 bytes from 172.168.1.10: icmp_seq=72 ttl=64 time=0.200 ms
64 bytes from 172.168.1.10: icmp_seq=73 ttl=64 time=0.208 ms
64 bytes from 172.168.1.10: icmp_seq=74 ttl=64 time=0.200 ms
64 bytes from 172.168.1.10: icmp_seq=75 ttl=64 time=0.219 ms
64 bytes from 172.168.1.10: icmp_seq=76 ttl=64 time=0.255 ms
64 bytes from 172.168.1.10: icmp_seq=77 ttl=64 time=0.212 ms
64 bytes from 172.168.1.10: icmp_seq=78 ttl=64 time=0.206 ms
64 bytes from 172.168.1.10: icmp_seq=79 ttl=64 time=0.170 ms
64 bytes from 172.168.1.10: icmp_seq=80 ttl=64 time=0.207 ms
64 bytes from 172.168.1.10: icmp_seq=81 ttl=64 time=0.207 ms
64 bytes from 172.168.1.10: icmp_seq=82 ttl=64 time=0.145 ms
64 bytes from 172.168.1.10: icmp_seq=83 ttl=64 time=0.168 ms
64 bytes from 172.168.1.10: icmp_seq=84 ttl=64 time=0.316 ms
64 bytes from 172.168.1.10: icmp_seq=85 ttl=64 time=0.192 ms
64 bytes from 172.168.1.10: icmp_seq=86 ttl=64 time=0.309 ms
64 bytes from 172.168.1.10: icmp_seq=87 ttl=64 time=0.295 ms
^C
--- 172.168.1.10 ping statistics ---
87 packets transmitted, 18 received, 79% packet loss, time 88040ms
rtt min/avg/max/mdev = 0.145/0.232/0.448/0.069 ms
demo@demo-OMEN-by-HP-Laptop-16-b0xxx:~$
```

2.2.4 View the point cloud status via ILA

1. Open the browser. Enter the LiDAR IP address and port number in the address bar <IP Address>:<PORT> to access the ILA.

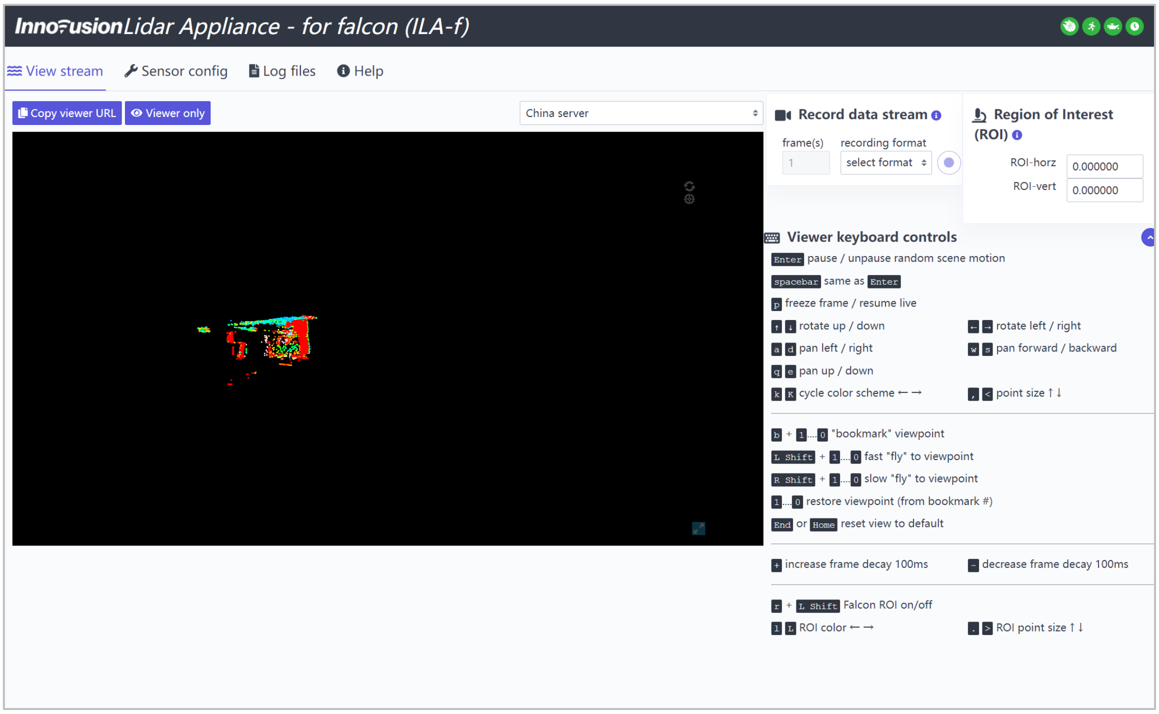
Note

- The default LiDAR IP address is 172.168.1.10. By default, the ILA port number is 8675. The default ILA login address is 172.168.1.10:8675.
- It is recommended to use the Google Chrome browser to log in to the ILA.

2. You can directly view the status of the LiDAR point cloud in real-time on the **View Stream**.
3. (Optional) Change the point of view and distance in which the point cloud data is displayed using the keyboard shortcuts and mouse.

   The available keyboard shortcuts are listed on the **View Stream**.
4. (Optional) Click **Viewer Only** to launch the point cloud viewer and view the live point cloud in a new tab.

   From the new tab, it is possible to maximize the display into full-screen mode.



2.2.5 (Optional) View the point cloud status via MetaView

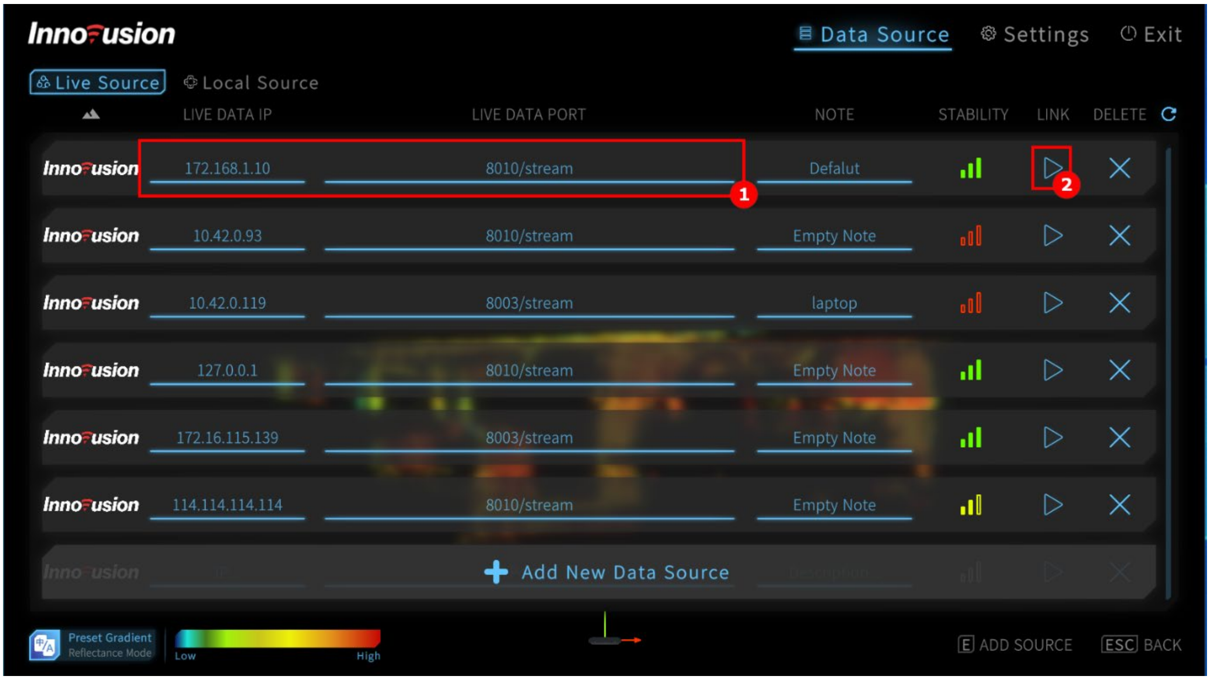
1. Double-click to open MetaView.exe.

Note

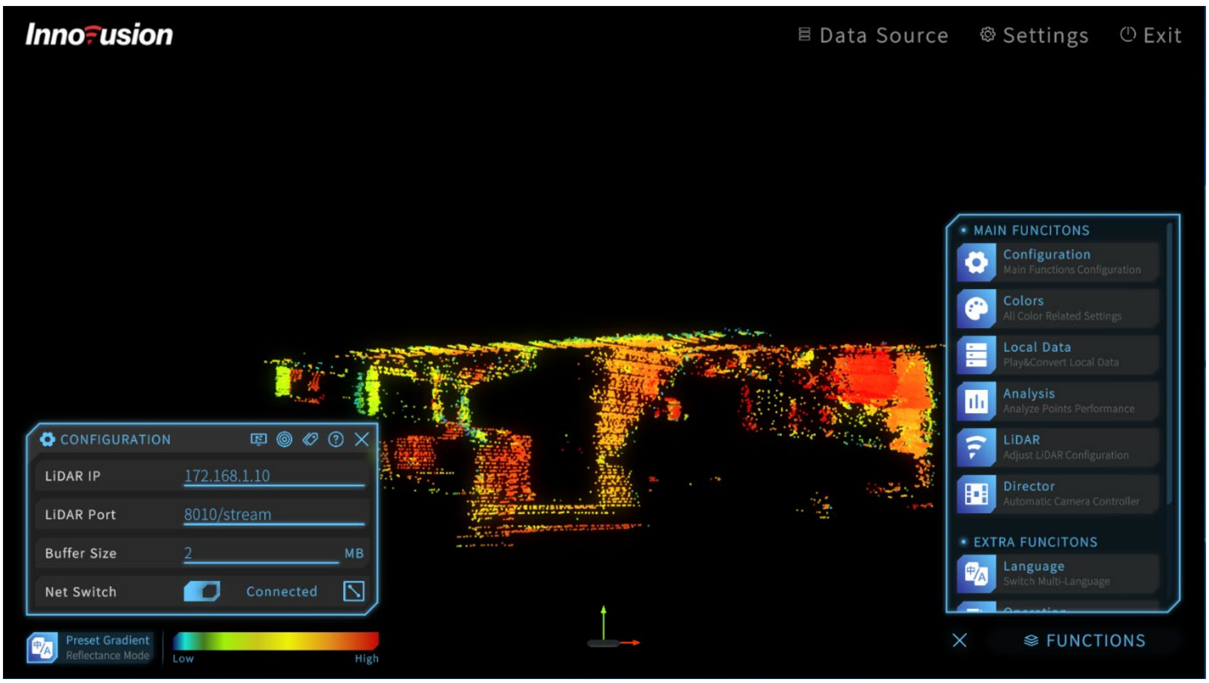
You can download the latest MetaView at the Innovusion Resource Hub at https://hub.innovusion.com/.

2. Go to **Data Source** > **Live Source**. Click **Add New Data Source**.

3. Enter LiDAR IP address and port number. Click ▷ to connect the LiDAR. The default LiDAR IP address is 172.168.1.10 and the default port number is 8010.



4. You could check the point cloud status after adding the LiDAR.



3 Installation

3.1 Precautions

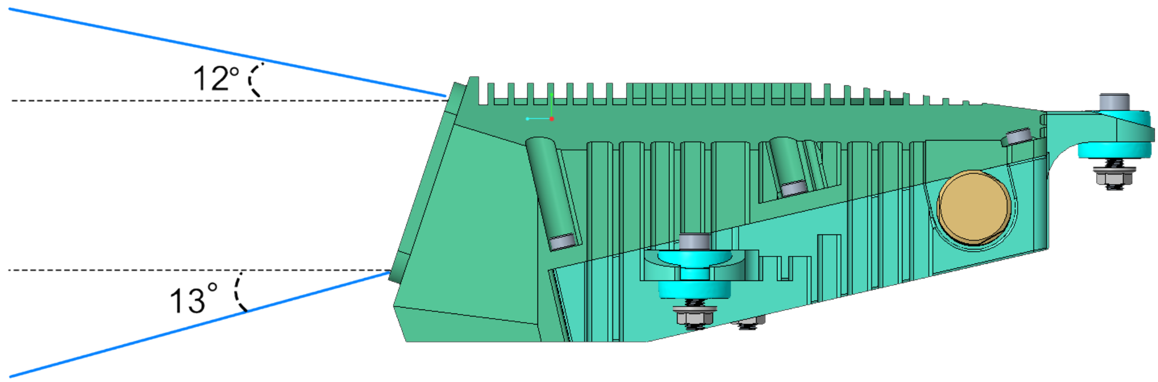
Falcon K is generally intended to be mounted on vehicles. Please follow the instructions during the installation process.

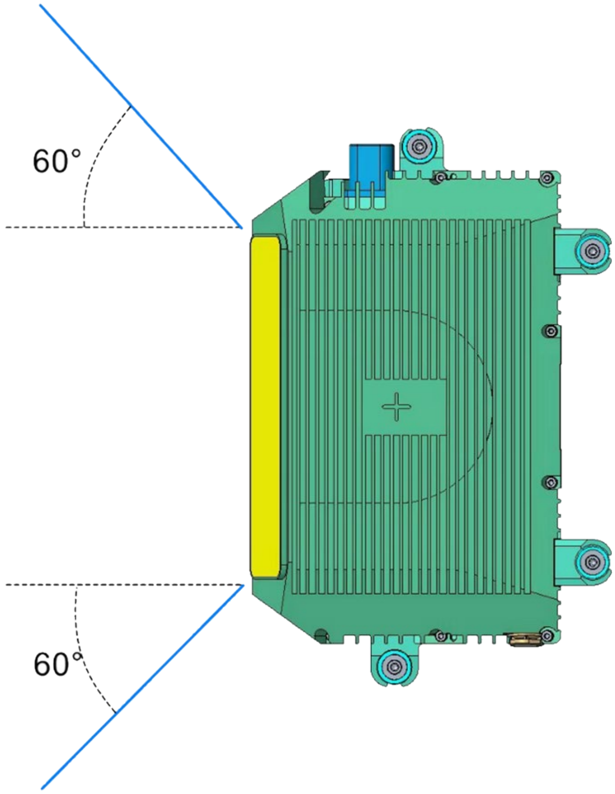
Personnel Requirements

The device is only intended to be installed by Innovusion service personnel or persons trained by Innovusion.

Installation

- If components are damaged or lost before installation, please contact Innovusion staff for support.
- It is recommended to connect the ground to the vehicle chassis to improve the Falcon's immunity to emissions.
- It is recommended to install the device on the center of the vehicle top.
- Make sure the LiDAR installation is flat and the tilt angle should be less than 3.5°. If there are specific tilt angle requirements, please contact Innovusion staff for support.
- It is recommended that the mounting surface should be made of aluminum alloy to facilitate the heat dissipation of the LiDAR.
- Make sure that the LiDAR cable keeps a certain degree of slackness.
- A space of 8 cm should be reserved at the outlet of the LiDAR to facilitate cabling.
- Before installation, please ensure there is no obstruction in the FOV (field of view) of the LiDAR. The vertical FOV of the LiDAR is between -13 ° to +12°, and the horizontal FOV of the LiDAR is between -60 ° to +60°. The specific FOV is shown in the figures below.





Storage

- The storage environment should be dry and well-ventilated. The recommended storage temperature is between -40°C and +85°C. The recommended storage humidity is lower than 85%.
- Without the official consent of Innovusion, do not continuously immerse the product in water.
- Avoid exposing the product to an environment exceeding the ingress protection rating.

Transportation

- The equipment should be packed in a packing box filled with cushioning materials to avoid product damage during transportation.
- Please handle the device with care. Impact to the device can damage or decrease performance of optical components in the device.
- Consider the space and location during transportation to minimize the handling distance. Utilize auxiliary tools or assistants if needed.
- Please do not place the device in an unstable position or handle it in an incorrect posture in case of device damage and personal injury.

Disposal of packaging materials

➤ Packaging materials are recyclable. Please dispose of them correctly when discarding.

➤ Packaging bags, cartons, or plastic films should be kept out of reach of infants and children to avoid injury or suffocation.

3.2 LiDAR installation

3.2.1 Power description

The operating voltage range of Falcon K is 9 to 34 VDC. It is recommended to use standard 12 VDC or 24 VDC for the power supply. The power consumption of LiDAR is about 30W during regular operating. The peak power when the LiDAR is activated is about 55W and lasts less than 1 second. The peak power should be considered when choosing the power supply.

3.2.2 Cooling solutions

3.2.2.1 Cooling method

To prolong the lifetime and improve stability, the recommended working temperature of the LiDAR is between -40°C to +85°C.

A cooling method should be considered when designing the installation solution of LiDAR to make sure the LiDAR works in the thermal equilibrium and the working temperature is under 85°C. (The working temperature should be measured at 75mm from the LiDAR)

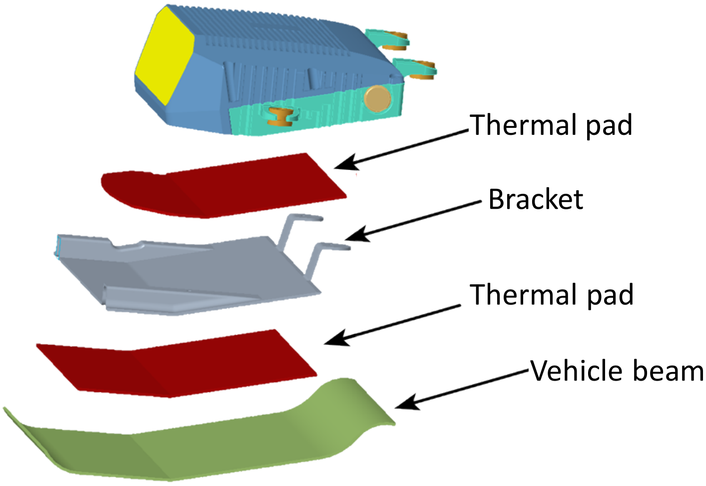
Two recommended cooling methods are as follows. You can choose one or multiple methods to integrate into your installation solution.

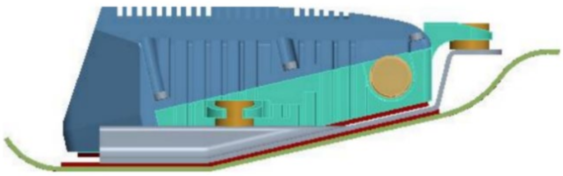
Method 1: Heat conduction

Methodologies

Due to heat accumulation of the LiDAR component, two thermal pads need to be installed at the bottom of LiDAR for heat dissipation.

One thermal pad is between the LiDAR bottom and the bracket, and the other is between the bracket and the beam. The installation positions are shown in the figure below.





1. The heat is conducted from the LiDAR bottom to the bracket through a thermal pad.
2. The heat is conducted from the bracket to the vehicle beam through a thermal pad.
3. The beam dissipates heat through the vehicle.

Mechanical requirements

The bracket and the vehicle beam should meet the following requirements for better heat dispassion.

- To increase the contact area with the LiDAR bottom, the shape of the bracket's upper surface should fit the LiDAR bottom as much as possible with uniform clearance.
- To increase the contact area with the vehicle beam, the shape of the bracket's lower surface should fit the top of the beam as much as possible.

The thermal pad for heat dissipation should meet the following requirements.

Table 5 Requirements for thermal pad

Item	Requirement
Material	Thermal pad
Thickness	1 mm
Thermal conductivity	3.0
Compression ratio	20%
Contact area	Full coverage of contact surface

Method 2: Heat convection

Methodologies

- ❖ Induce outer air or AC wind to form air convection for cooling.

Mechanical requirements

- ➢ The LiDAR installation environment should be designed with openings to facilitate air convection. The wind speed for air convection should be at least 2 m/s.
- ➢ It is recommended to induce the AC wind to or under the beam via pipelines or other structures to accelerate LiDAR cooling.

3.2.2.2 Overheat protection

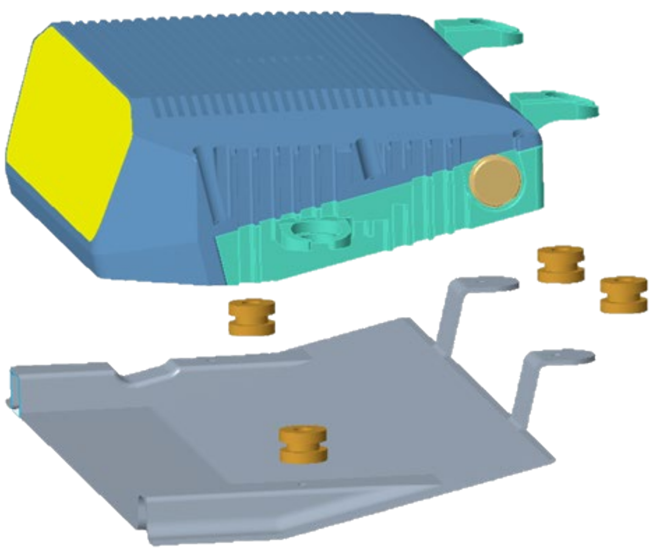
The LiDAR automatically switches to the protection mode when its working temperature exceeds 100°C. In the protection mode, the laser is turned off. The LiDAR automatically switches back to the normal mode after the working temperature drops to 95°C.

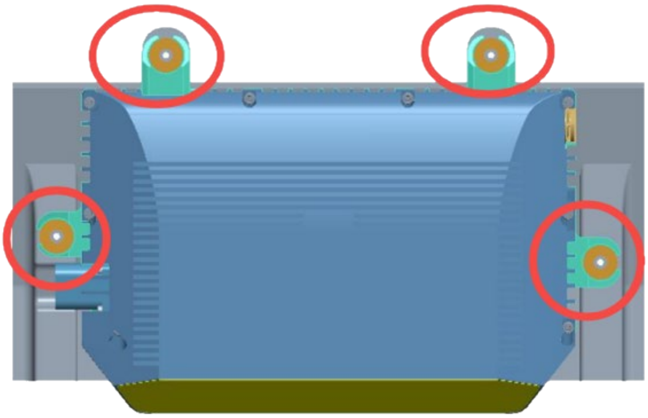
You can switch to the standby mode by commands. In the standby mode, the laser, polygon and Galvo are turned off, Only the IC components functions. You can utilize the standby mode to reduce the overheating risk and prolong the lifetime. i.e., when the vehicle stops or travels at a low speed, you can switch the LiDAR to the standby mode until it accelerates again. Since the vehicle speed exceeds 20 km/h, the cooling performance will be greatly improved.

3.2.3 Damping Solutions

The vibration frequencies of the LiDAR's equilibrium points are mainly distributed in 80/400/800/1200 Hz when the motor speed is constant.

The LiDAR is installed on the vehicle through 4 lugs, each of which is equipped with a secondary bushing to prevent vibration. The dynamic stiffness of the bushings' mounting point should be more than 2000 N/mm. The installation of the bushings is shown in the figure below.

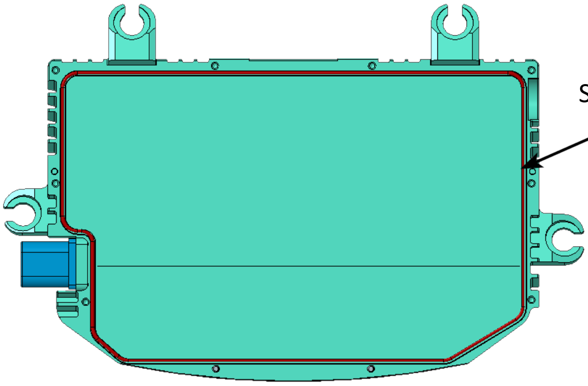




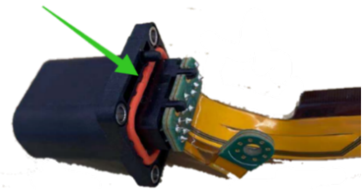
3.2.4 Sealing solutions

LiDAR IP rating: IP67 (body), IP69K (window).

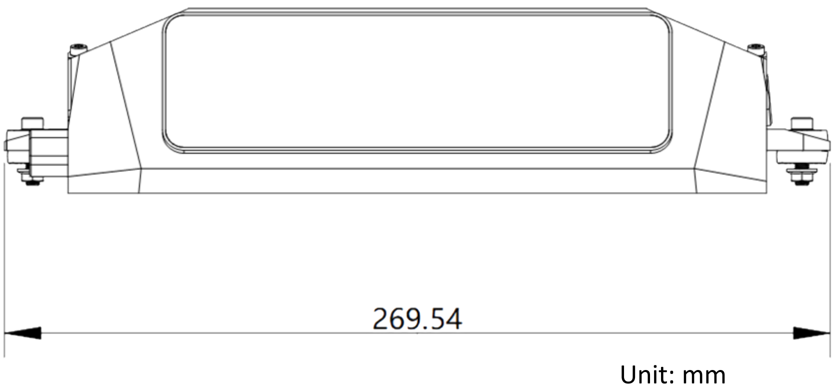
- The sealing rings on the LiDAR's upper and lower assembly surface can seal the device.



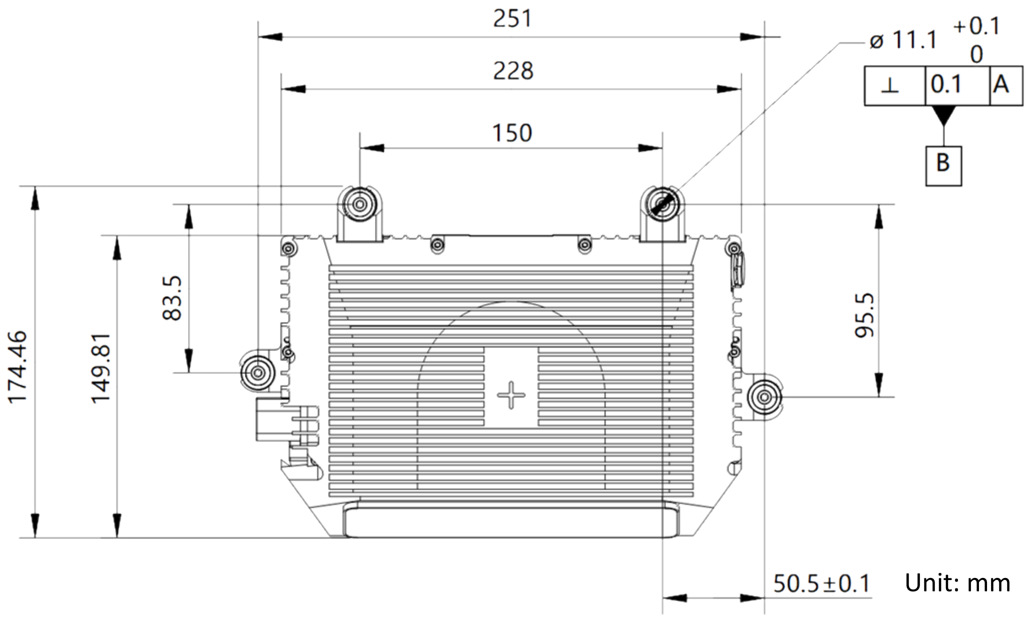
- The sealing gasket can seal LiDAR's external connectors after connection.
- The LiDAR window will be exposed after the installation. Therefore, the vehicle should perform a sealing design for the LiDAR window. You can take the solution of gluing the sealing ring as a reference.



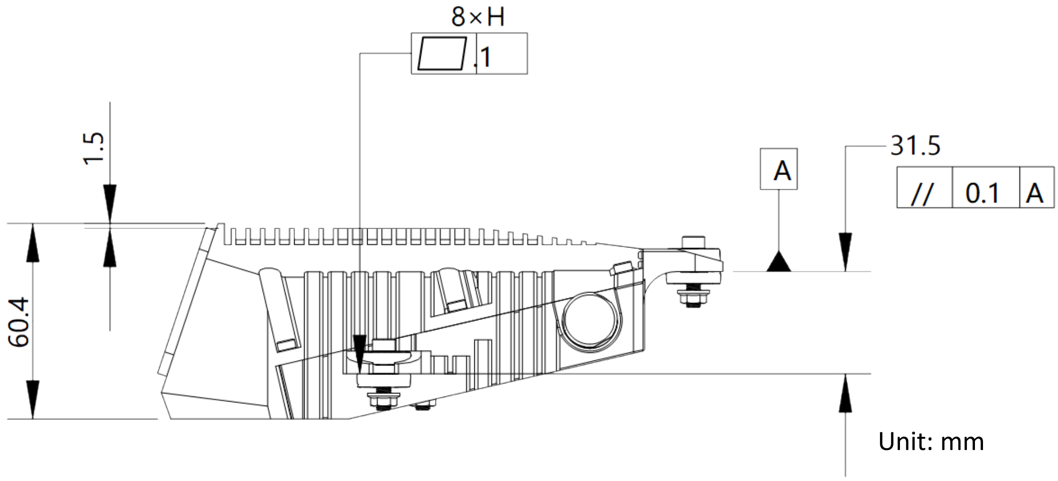
3.2.5 Dimensions



Front View



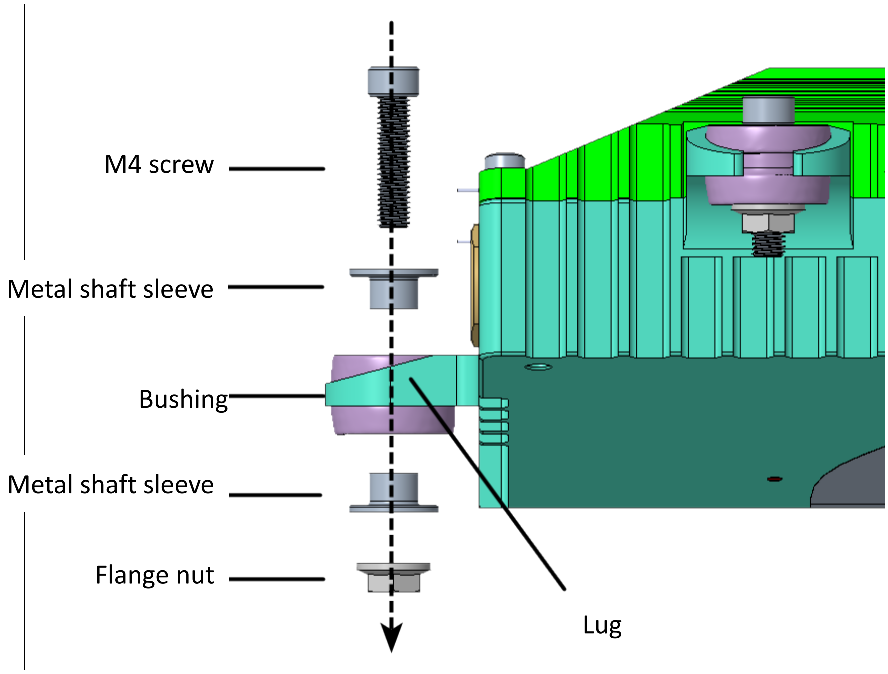
Top view



Left view

3.2.6 Installation method

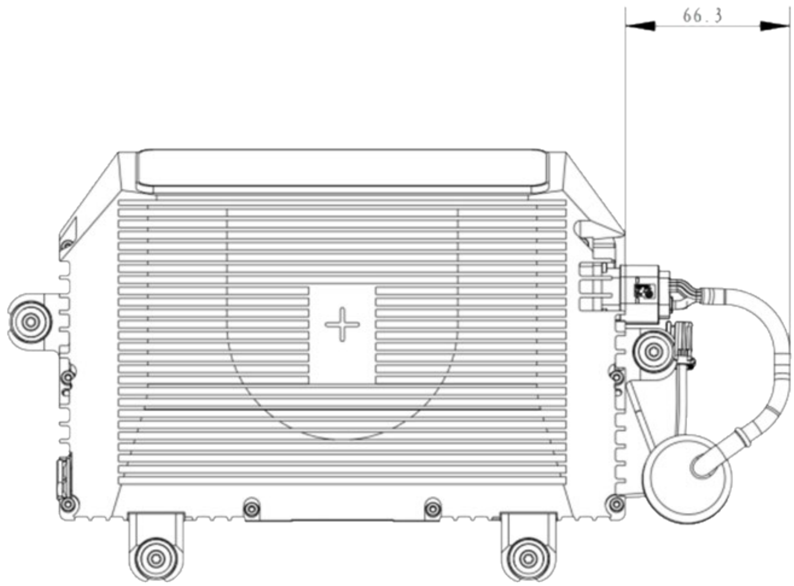
The LiDAR is installed on the vehicle through 4 lugs, each equipped with a secondary bushing for damping. Before installation, it is essential to reserve sufficient space according to the vehicle condition and installation craft. You can refer the lugs' size and relative positions to the LiDAR in 3.2.5 Dimensions. The installation method is shown below.



3.3 Cable description

3.3.1 Cable bending

Reservation of sufficient space according to the bend radius of the LiDAR cable is required before Installation. Refer to the following diagram for the required space. The unit is mm.



3.3.2 Cable definition

This section describes two types of cable respectively, which are both provided by Falcon Kinetic. You can choose the cable type to be included.

Please notice that if the Falcon must connect to industrial Ethernet with an RJ45 connector, it is necessary to adopt a media converter for the switch from the 1000Base-T1 interface to an RJ45 interface.

- Cable 1

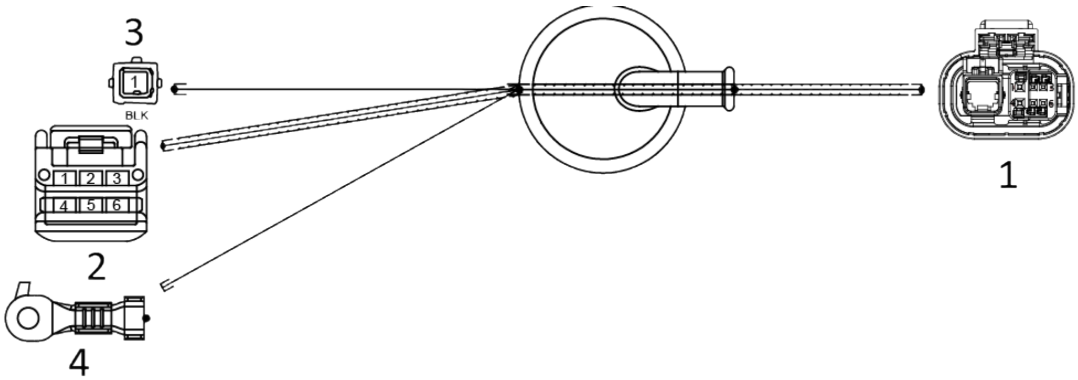
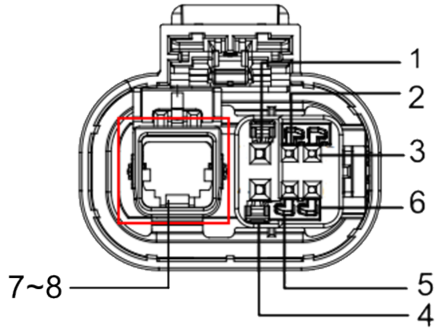
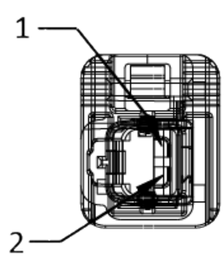


Table 6 Interface description of cable 1

NO.	Name	Description
1	LiDAR interface	8-pin interface. Pins are defined as follows.

		 [1]: Grounding [2]: Empty pin [3]: Shield grounding [4]: Power supply [5]: CAN high [6]: CAN low [7] and [8]: Gigabit Ethernet
2	Power supply and CAN bus	6-pin interface. Pins are defined as follows.  [1]: Power supply [2]: Empty pin [3]: Grounding [4]: Empty pin [5]: CAN high [6]: CAN low
3	1000Base-T1 automotive Ethernet	1000Base-T1 automotive Ethernet (support TCP and UDP protocol). Pins are defined as follows.  [1]: Ethernet D+ [2]: Ethernet D-
4	Grounding	Shield grounding

❖ Cable 2

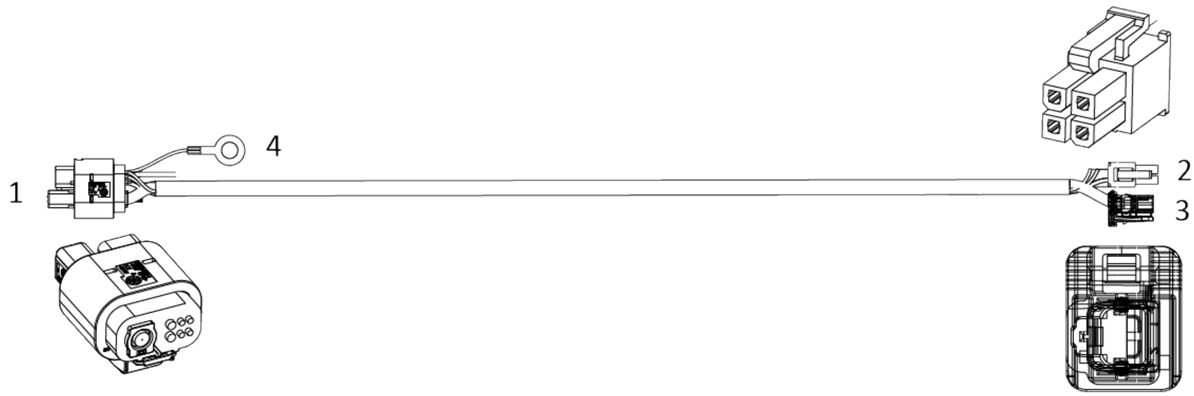
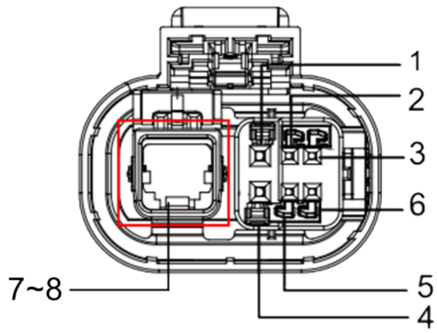
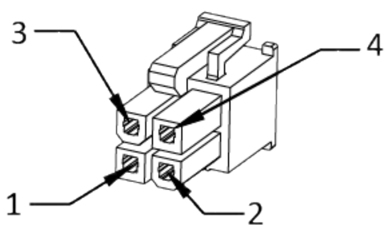
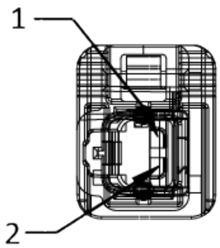


Table 7 Interface description of cable 2

NO.	Name	Description
1	LiDAR interface	8-pin interface. Pins are defined as follows.  [1]: Grounding [2]: Empty pin [3]: Shield grounding [4]: Power supply [5]: CAN high [6]: CAN low [7] and [8]: Gigabit Ethernet
2	Power supply	4-pin interface. Pins are defined as follows.  [1]: Power supply- [2]: Empty pin

		[3]: Power supply+ [4]: Empty pin
3	1000Base-T1 automotive Ethernet	1000Base-T1 automotive Ethernet. Pins are defined as follows.  [1]: Ethernet D+ [2]: Ethernet D-
4	Grounding	Shield grounding

3.4 Cleaning

For optimal performance of LiDAR, please keep the front window of the sensor clean and free of dirt, bugs, and other debris. The steps to clean the optical window are as follows.

1. Prepare a clean microfiber cloth, soak it in the alcohol, and wring it out.
2. Loose the debris from the LiDAR window with the clean, dust-free wiper for 1 minute. Do not wipe dirt directly off the LiDAR window glass without loosening it sufficiently.

LiDAR window

3. Please wait 1 minute, gently wipe the window with the clean microfiber cloth and dry it.
4. Wipe the window with a high-quality paper towel or mirror paper. Do not apply excessive force to avoid damaging the optical coating.

Note

- Please wash your hands or wear PVC powder-free clean gloves before touching the product.
- Do not use solvents since they may damage the window coating.
- Please use a new dust-free wiper to wipe the LiDAR window.
- The LiDAR window is made of special plastic material. Please pay attention to the following items when cleaning: Avoid direct skin contact with the optical window. Do not use corrosive cleaners and solvents. Do not use paper towels to clean the window to avoid scratches.

4 Software operation

You can operate the LiDAR on the following platforms. For how to operate LiDAR using SDK tools, please refer to 5 SDK configuration. If you want to get more information, please contact Innovusion staff to obtain related manuals.

Table 8 platform instructions

NO.	Software Interface	Description
1	ILA (Innovusion LiDAR Utility)	ILA is the quickest way to view live point clouds and record data. The ILA GUI can be accessed through a web browser (preferably Chrome) and does not require any SW installation. Running on Linux, Windows, or Mac OS, and does not require any SW installation. ILA does not support the replaying of recorded data.
2	ROS	ROS (ROS1) drivers are available for Kinetic, Melodic and Noetic. Please notice that drivers for ROS Kinetic will be discontinued soon. ROS2 drivers are also available for Foxy, Fitzroy, and Humble Hawksbill.
3	MetaView	The MetaView can be run from any Linux or Windows OS computer. It allows viewing of the live point cloud, data recording, and replaying recorded data. It is recommended to use MetaView as the application for evaluation of the Falcon point cloud.
4	Docker	Docker drivers are available for users to view and replay data. It also provides a convenient way for calibration and troubleshooting.

4.1 Operate on ILA

The ILA operations in this chapter are based on the 3221 version of the firmware.

4.1.1 Start the LiDAR

1. Connect the power supply to start the LiDAR.
2. The LiDAR completes initialization and generates data after powering on for 11 to 18 seconds.

Note

The LiDAR does not have a power switch. It will become operational when power is applied.

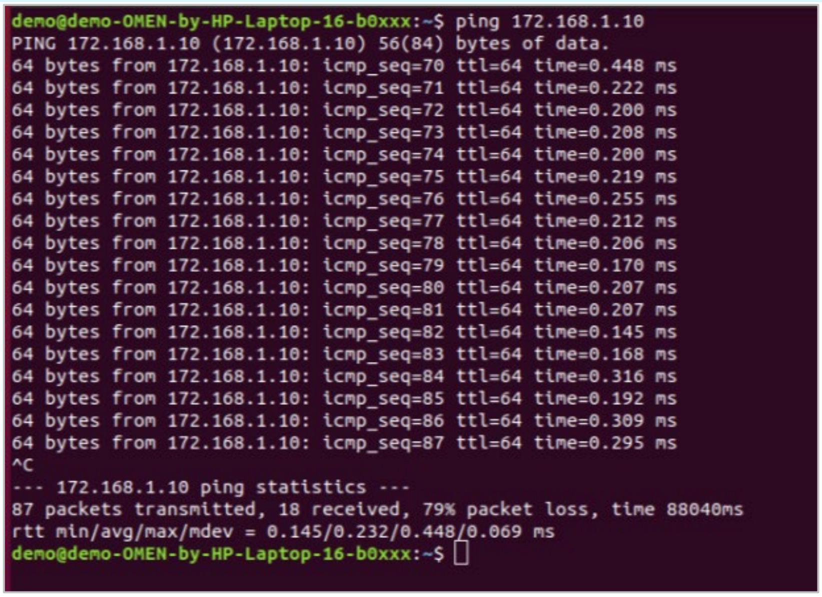
4.1.2 Login

1. Connect the computer to LiDAR and ensure the Ethernet connection.
2. Change the computer IP address to the same subnet with the LiDAR.

3. Open Chrome. Enter the LiDAR IP address and port number in the address bar `<IP Address>: <PORT>` to access the ILA.

Note

- The default LiDAR IP address is 172.168.1.10. By default, the ILA port number is 8675. The default ILA login address is 172.168.1.10:8675.
- It is recommended to check the access to the LiDAR IP address by using the ping command. The return value is shown in the figure below.
- It is recommended to use the Google Chrome browser to log in to the ILA.



4.1.3 View the point cloud status of the LiDAR

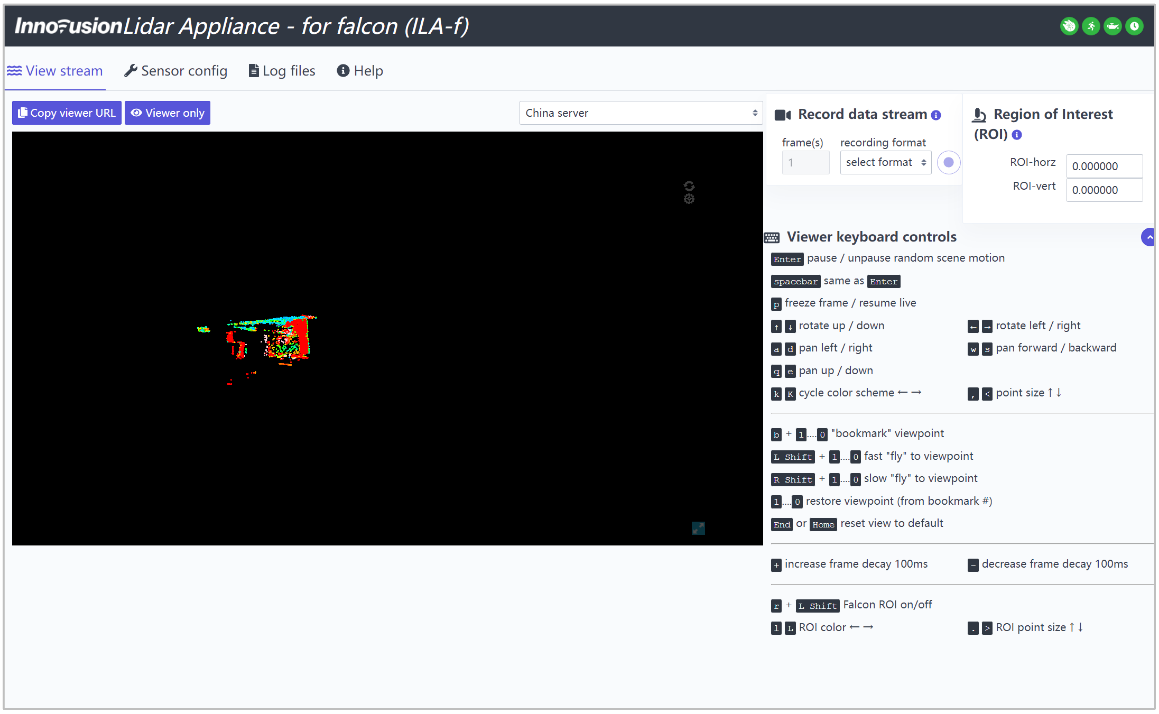
Note

- Ensure the internal point cloud service (PCS) runs on the **Sensor config**. If PCS is running, the green Run button should be greyed out.
- Before viewing the point cloud status, please ensure the computer can access to the Internet. While the latest ILA has a built-in WebGL viewer, a more feature-rich viewer is available with an Internet connection.

4. You can directly view the status of the LiDAR point cloud in real-time on the **View Stream**.
5. (Optional) Change the point of view and distance in which the point cloud data is displayed using the keyboard shortcuts and mouse.

   The available keyboard shortcuts are listed on the **View Stream**.
6. (Optional) Click **Viewer Only** to launch the point cloud viewer and view the live point cloud in a new tab.

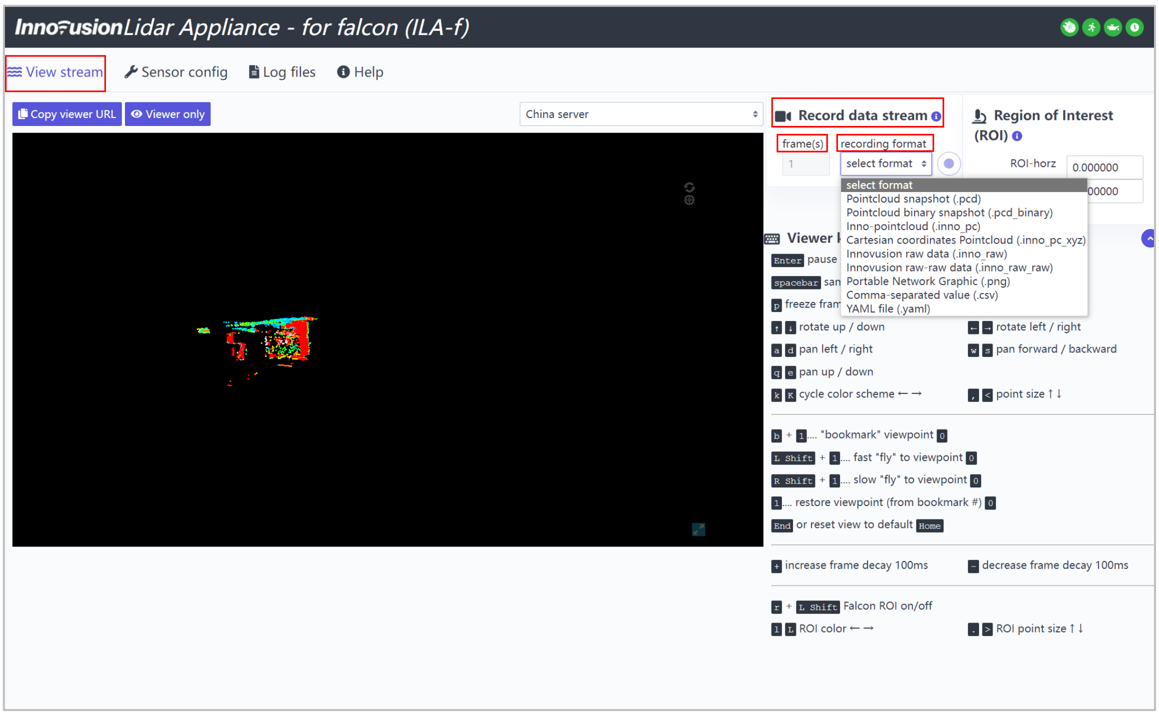
   From the new tab, it is possible to maximize the display into full-screen mode.



4.1.4 Record LiDAR point cloud data

You can record LiDAR point cloud data in different formats.

1. Go to **View stream** > **Record data stream**.



2. Select the file format and size of the data to be recorded.

- Record a file in pcd format.
  Select **Pointcloud snapshot (.pcd)** in **recording format**. Enter the number of frames to be recorded in **frame (s)**. The range is from 0 to 20.
- Record a file in inno_pc format.
  Select **Inno-pointcloud (.inno_pc)** in **recording format**. Enter the number of frames to be recorded in **frame (s)**.
  inno_pc is a proprietary format of Innovusion point cloud files and cannot be read by 3rd party software. The points in inno_pc files are in spherical coordinates. inno_pc is in the optimum compression.
- Record a file in inno_pc_xyz format.
  Select Cartesian **coordinates Pointcloud (.inno_pc_xyz)** in **recording format**. Enter the number of frames to be recorded in **frame(s)**. The range is from 0 to 10.
  inno_pc_xyz is a proprietary format of Innovusion point cloud files and cannot be read by 3rd party software. The points in inno_pc_xyz files are in Cartesian coordinates. inno_pc_xyz is in the less optimum compression compared to inno_pc.
- Record a file in inno_raw format.
  Select **Innovusion raw data (.inno_raw)** in **recording format**. Enter the size of the file in **[MiB]**.
  The inno_raw is a proprietary format of Innovusion point cloud files.
- Record a file in inno_raw_raw format.
  Select **Innovusion raw-raw data (.inno raw raw)** in **recording format**. Select channel number in **channel**. The range is from 0 to 3.
- Record a file in png format.
  Select **Portable Network Graphic (.png)** in **recording format**.
- Record a file in csv format.
  Select **Comma-separated value (.csv)** in **recording format**. Enter the number of frames to be recorded in **frame (s)**. The range is from 0 to 20.
- Record a file in yaml format.
  Select **YAML file (.yaml)** in **recording format**.

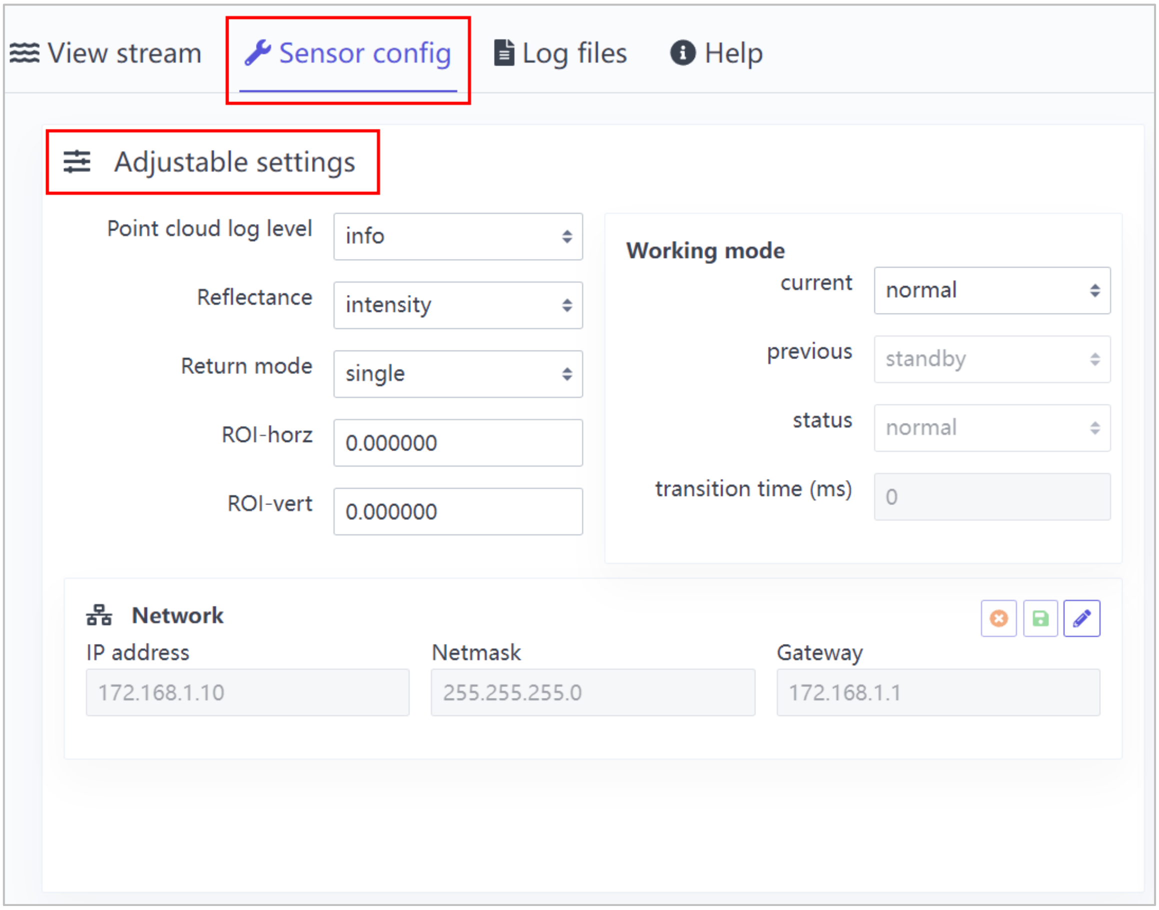
3. Click  to record the file.

Note

- Point cloud data recording starts immediately by default.
- The recorded data file is in the **Download** folder on the computer.
- The limitation of the file size is subject to change according to the file format.

4.1.5 Configure ROI

1. Go to **Sensor config** > **Adjustable settings**.



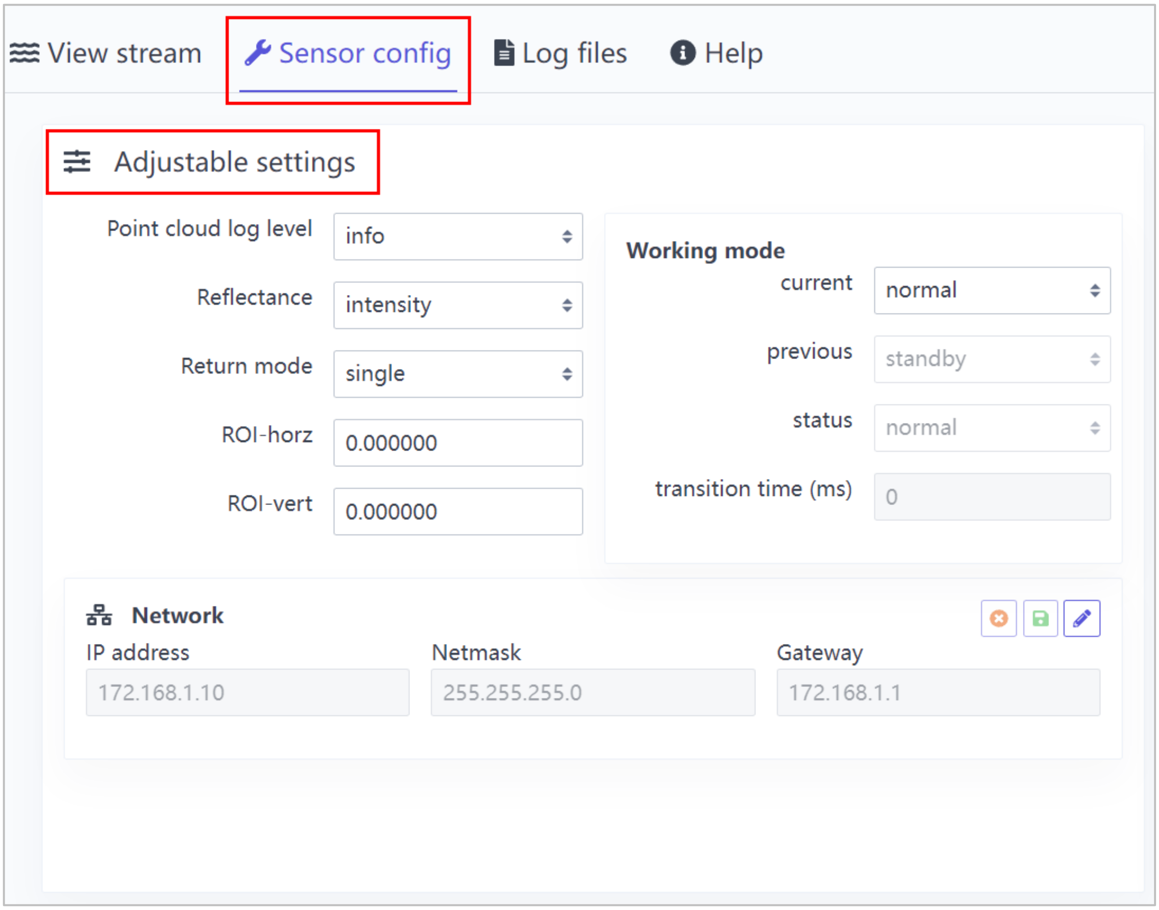
2. Enter the center position of the ROI. **ROI-horz** is the horizontal angle of the ROI’s center, ranging from -60 to 60. **ROI-vert** is the vertical angle of the ROI’s center, ranging from -25 to 25. The units of **ROI-horz** and **ROI-vert** are degrees (°).

Note

Users can also set the position of the ROI in **View stream** > **Region of Interest**.

4.1.6 Configure reflectance

1. Go to **Sensor config** > **Adjustable settings**.

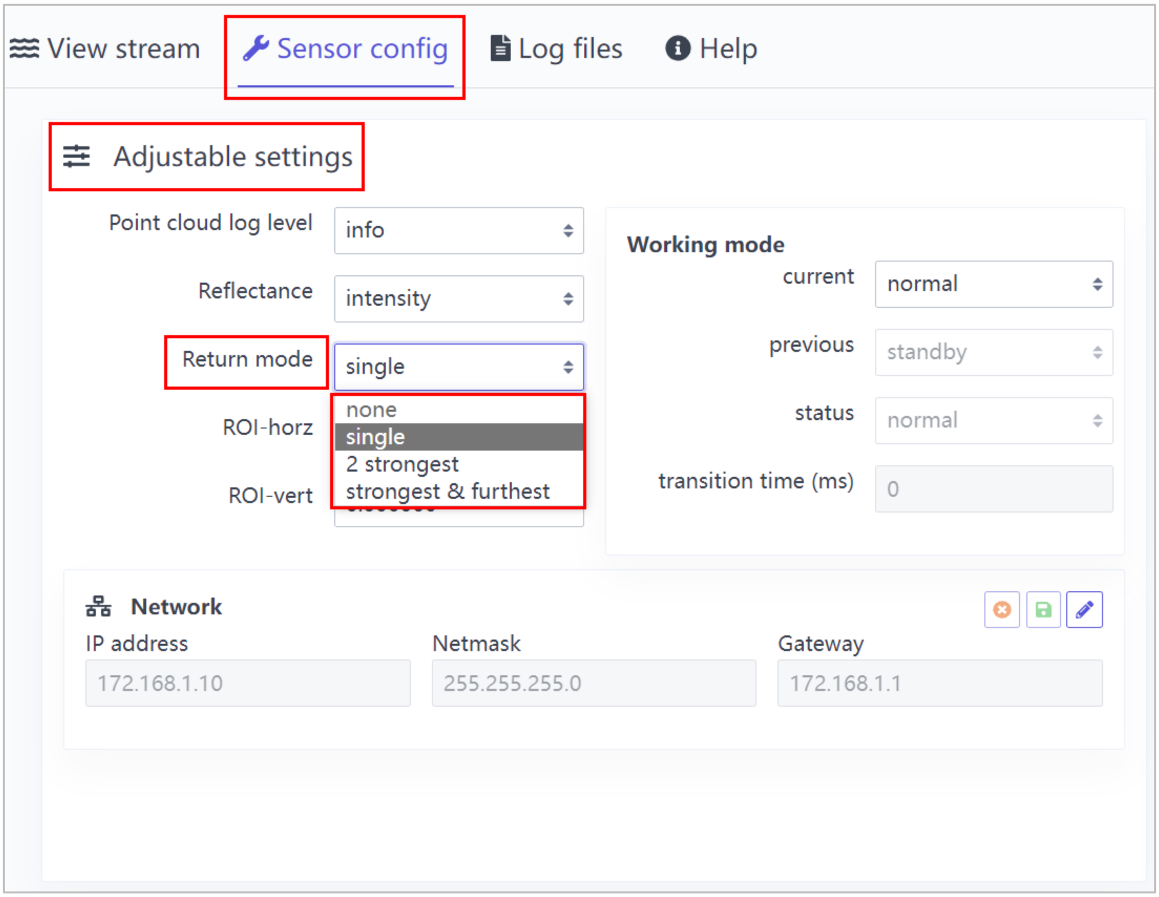


2. Change the reflectance of the LiDAR. The reflectance can be either **intensity** or **reflectivity**.

- **intensity** is the intensity of the echo read directly by LiDAR. The intensity varies with the influence of the factors including object distance, object reflectivity, beam angle, etc.
- **reflectivity** is the calculated result based on the intensity and rectified with the object distance, beam angle and other parameters.

4.1.7 Select return mode

1. Go to **Sensor config** > **Adjustable settings**.

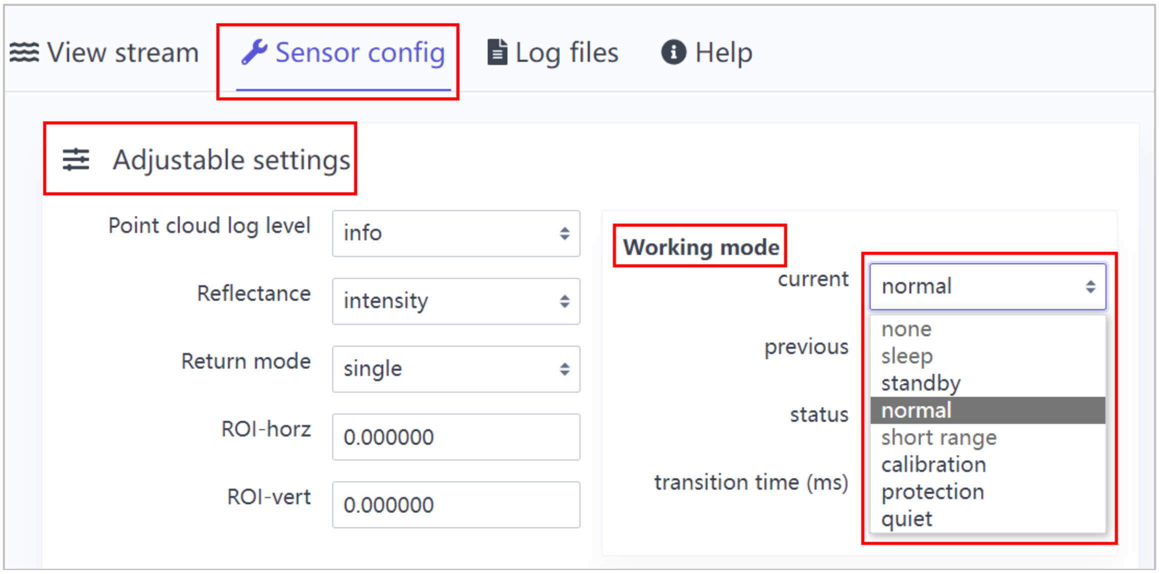


2. Configure the return mode received when a laser is emitted once.

   Either single return mode or dual return mode can be selected, and the dual return mode has two options: **strongest + 2 strongest** and **strongest & furthest**. The single return mode is chosen by default.

4.1.8 Select working mode

1. Go to **Sensor config** > **Adjustable settings** > **Working mode**.

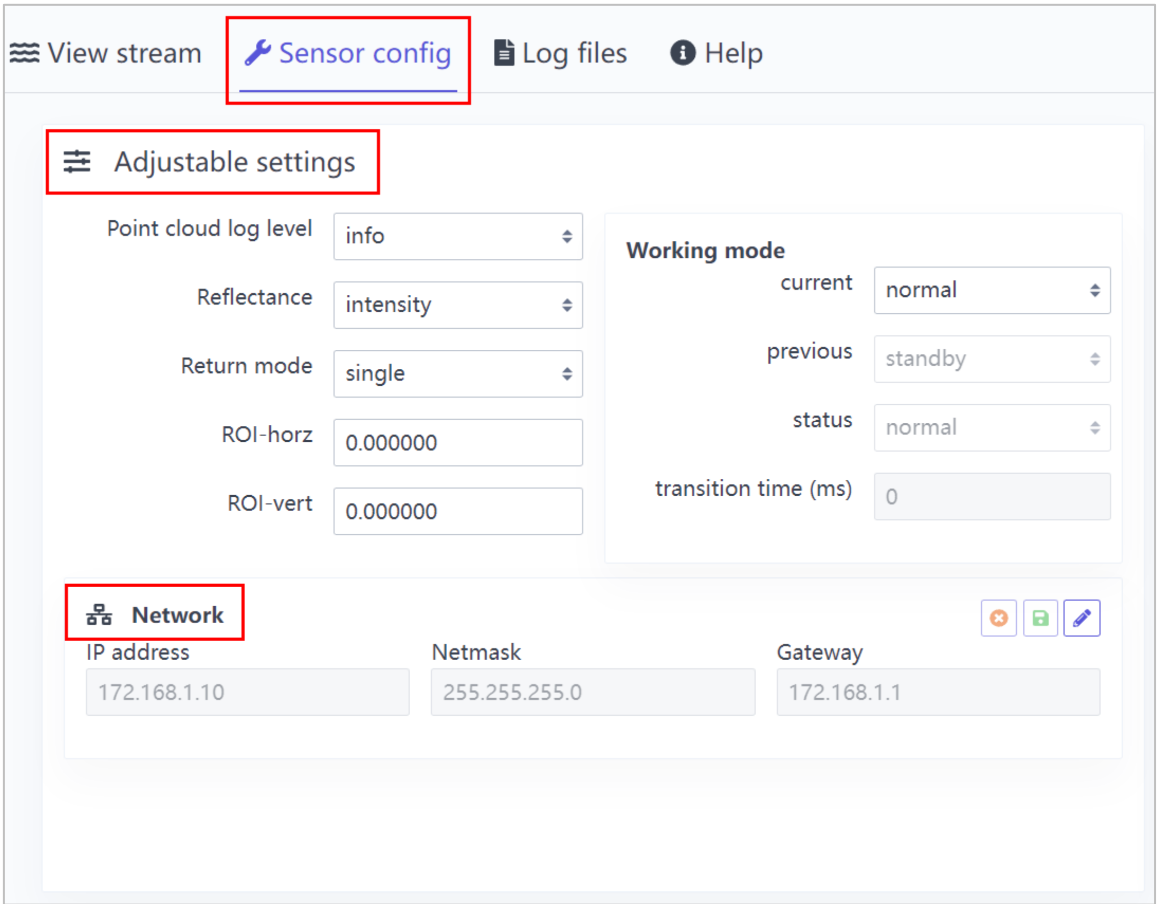


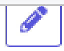
2. Select the working mode of the LiDAR in **current**.

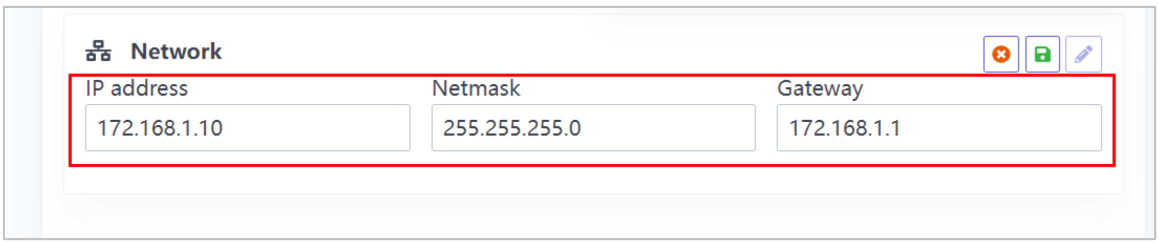
- none: The initialization value.
- sleep: The sleep mode. The mode can only be used when the LiDAR is configured to wake from CAN messages.
- standby: The standby mode. In this mode, components such as the laser, polygon and Galvo are disabled. The other internal components continue to function for fast switching to normal mode.
- normal: The normal mode.
- short range: This mode is no longer in use.
- calibration: The calibration mode.
- protection: The protection mode is intended to prevent LiDAR from overheating risk. The LiDAR will be in this mode when the laser temperature exceeds the threshold. and return to normal mode after the laser temperature drops below the threshold.
- quiet: The quiet mode. In this mode, the noise will be reduced with possible trade-offs to performance.

4.1.9 Change LiDAR IP address

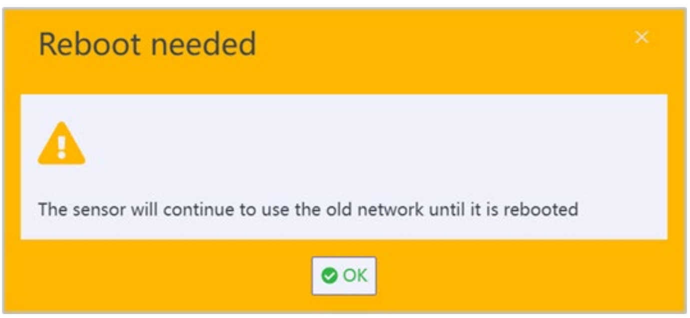
1. Go to **Sensor config** > **Adjustable settings** > **Network**.



2. Click  to change the IP address, netmask address, and gateway address of the LiDAR according to your needs.



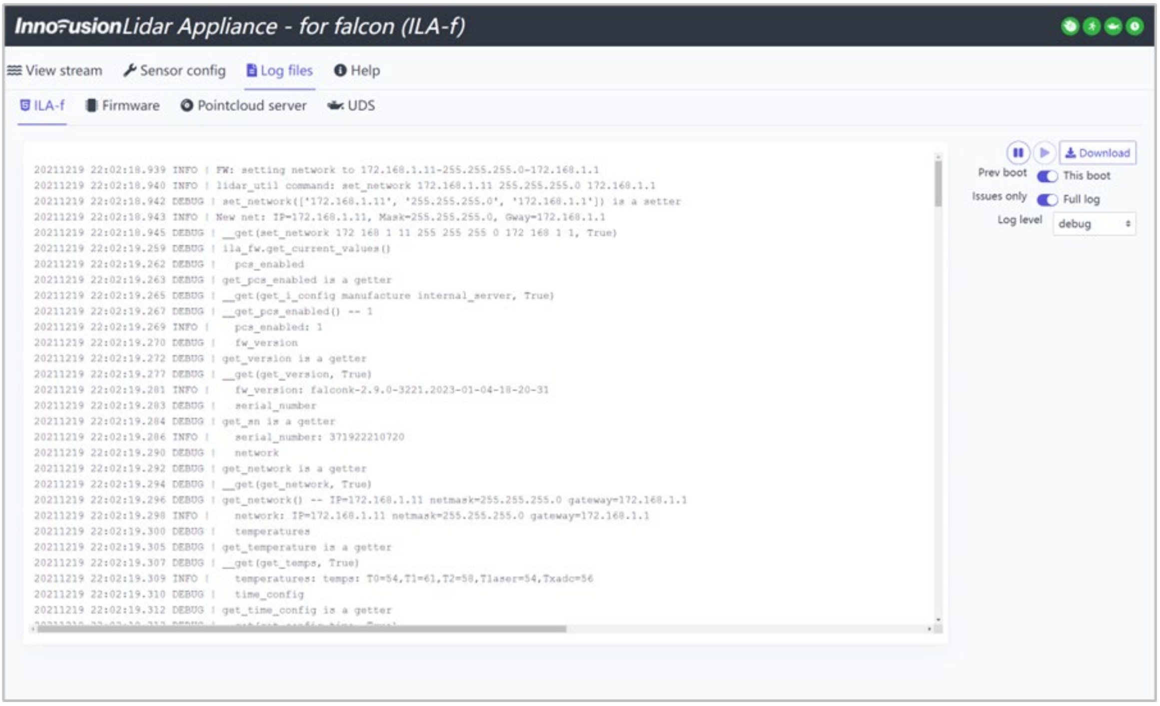
3. Click  to save the changes.
4. Power off and restart the LiDAR to enable the new IP address.



4.1.10 View/download logs

You can view and download logs related to different components to confirm operations and alarm information.

1. Go to Log files.
2. Select the log type. You can select four types of logs: **ILA-f**, **Firmware**, **Pointcloud server,** and **UDS**.



3. (Optional) Set the filtering criteria of the logs.

- Time range

  Select **Prev boot** or **This boot** to choose whether to display the last 100 log messages generated before this boot or those generated after this boot.

- Type of log

  Select **Issues only** or **Full log** to choose whether to display only the problem or all logs.

- Log level

  Click **log level** and select the log level to be displayed. The log level is described in the following table.

4. You can view logs on the left panel. You can also click **Download** to download the corresponding logs.

Other operations

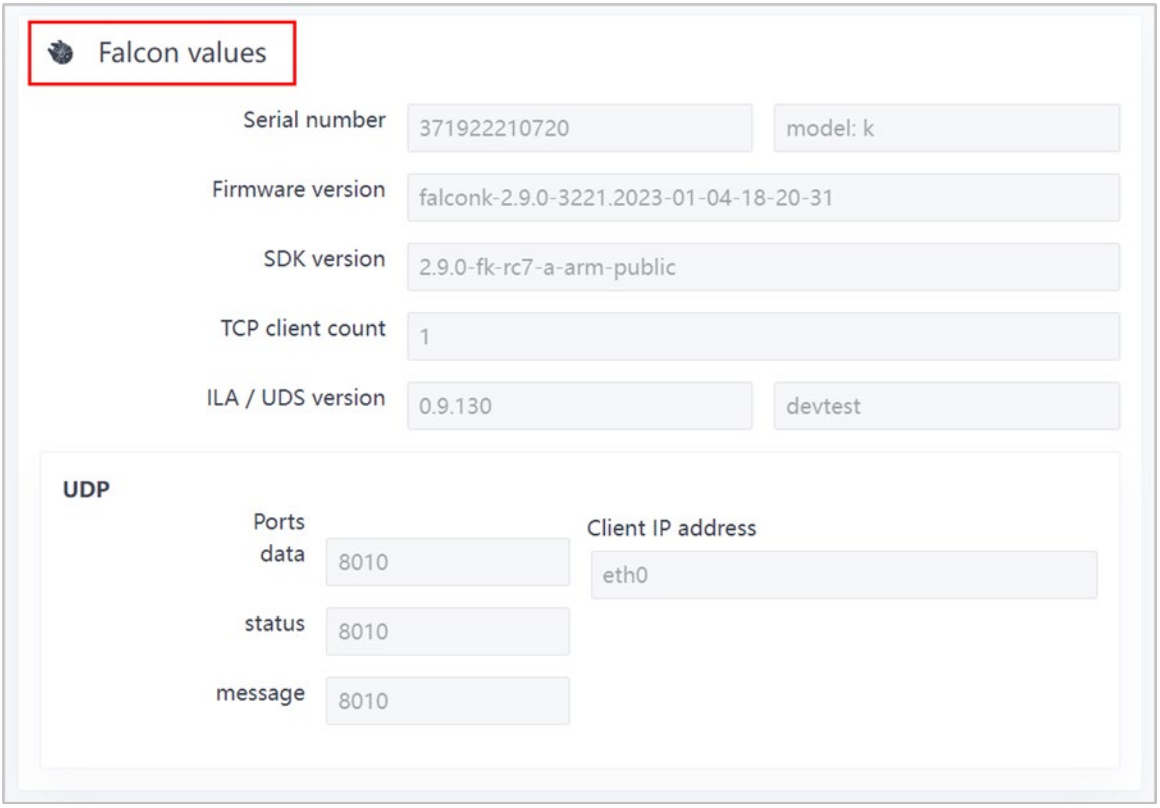
5. (Optional) Click  to pause logging.
6. (Optional) Click  to restart logging.

Table 9 Log level description

Log level	Description
fatal	Fatal and unrecoverable error. The fatal error will cause the exit of the application.
critical	Critical and irreversible error. When this error occurs, the system cannot work properly.
error	Error message.
temp	Not used yet.
warning	Warning message. This message is used to alert that there may be potential errors.
debug	Debug message. The message helps debug the application.
info	Message about the application running.
trace	Trace message. The message is used for outputting details
detail	Detail message. The message is used for outputting more detailed messages than the trace message.
verbose	Verbose message. The message is used for outputting more detailed messages than the trace message.
exception	Exceptional message.

4.1.11 View version information

You can check the device serial number, hardware version, and SDK version information in the **Sensor Config** > **Falcon Values**.



4.1.12 Shut down the LiDAR

Disconnect the power supply to shut down the LiDAR.

4.2 Operate in ROS

The operations in this chapter are based on the Ubuntu 18.04. The version of ROS is melodic.

4.2.1 Start the LiDAR

1. Connect the power supply to start the LiDAR.
2. The LiDAR completes initialization and generates data after powering on for 11 to 18 seconds.

Note

The LiDAR does not have a power switch. It will become operational when power is applied.

4.2.2 Obtain point cloud data

Note

- The ROS driver needs to be restarted after the LiDAR is shut down or the software is restarted.
- For the installation method of ROS, please refer to http://wiki.ros.org/.

1. Connect the computer to the LiDAR.
2. Change the computer IP address to the same subnet with the LiDAR.

Note

- The default LiDAR IP address is 172.168.1.10.
- It is recommended to check the access to the LiDAR IP address via the ping command. The return value is shown in the figure below.

```
demo@demo-OMEN-by-HP-Laptop-16-b0xxx:~$ ping 172.168.1.10
PING 172.168.1.10 (172.168.1.10) 56(84) bytes of data.
64 bytes from 172.168.1.10: icmp_seq=70 ttl=64 time=0.448 ms
64 bytes from 172.168.1.10: icmp_seq=71 ttl=64 time=0.222 ms
64 bytes from 172.168.1.10: icmp_seq=72 ttl=64 time=0.200 ms
64 bytes from 172.168.1.10: icmp_seq=73 ttl=64 time=0.208 ms
64 bytes from 172.168.1.10: icmp_seq=74 ttl=64 time=0.200 ms
64 bytes from 172.168.1.10: icmp_seq=75 ttl=64 time=0.219 ms
64 bytes from 172.168.1.10: icmp_seq=76 ttl=64 time=0.255 ms
64 bytes from 172.168.1.10: icmp_seq=77 ttl=64 time=0.212 ms
64 bytes from 172.168.1.10: icmp_seq=78 ttl=64 time=0.206 ms
64 bytes from 172.168.1.10: icmp_seq=79 ttl=64 time=0.170 ms
64 bytes from 172.168.1.10: icmp_seq=80 ttl=64 time=0.207 ms
64 bytes from 172.168.1.10: icmp_seq=81 ttl=64 time=0.207 ms
64 bytes from 172.168.1.10: icmp_seq=82 ttl=64 time=0.145 ms
64 bytes from 172.168.1.10: icmp_seq=83 ttl=64 time=0.168 ms
64 bytes from 172.168.1.10: icmp_seq=84 ttl=64 time=0.316 ms
64 bytes from 172.168.1.10: icmp_seq=85 ttl=64 time=0.192 ms
64 bytes from 172.168.1.10: icmp_seq=86 ttl=64 time=0.309 ms
64 bytes from 172.168.1.10: icmp_seq=87 ttl=64 time=0.295 ms
^C
--- 172.168.1.10 ping statistics ---
87 packets transmitted, 18 received, 79% packet loss, time 88040ms
rtt min/avg/max/mdev = 0.145/0.232/0.448/0.069 ms
demo@demo-OMEN-by-HP-Laptop-16-b0xxx:~$
```

3. View the system details and obtain the corresponding driver. Copy the driver to the root directory of the system. Execute the following command to install the driver.

```
sudo dpkg -i <package.deb>
```

Note

package.deb is the driver's name of the LiDAR. Obtain the latest driver version based on the actual conditions.

Table 10 Available system

System Version	CPU
Ubuntu 16.0.4	ARM
	X86
Ubuntu 18.0.4	ARM
	X86
Ubuntu 20.0.4	ARM
	X86

4. Start ROS. The return value is shown in the figure below.

```
roscore
```

5. Obtain the point cloud data of the LiDAR. The return value is shown in the figure below.

- Obtain the point cloud data via the TCP port.

```
source /opt/ros/melodic/setup.bash
roslaunch innovusion_pointcloud innovusion_points.launch device_ip:= <device_ip> port:=< TCP_port > processed:= <Processed_number>
```

- Obtain the point cloud data via the UDP port.

```
source /opt/ros/melodic/setup.bash
roslaunch innovusion_pointcloud innovusion_points.launch device_ip:= <device_ip> udp_port:= <UDP_port> processed:= <Processed_number>
```

Note

- The default value of device_ip is 172.168.1.10. By default, the UDP port number is 8010.
- The value of processed_number can be 0 or 1. When the process_number is set to 1, the point cloud data is obtained from ROS client. When the process_number is set to 0, the point cloud data is obtained from external PCS.

```
  customer: normal
[ INFO] [1679565063.399176909]:  8959 net_manager.cpp:93 Requesting /command/?get_lidar_id from 172.168.1.10:8010
[ INFO] [1679565063.399817013]:  8959 lidar_client.cpp:894 innovusion_nodelet_manager remote lidar_id: 0
[ INFO] [1679565063.399824240]:  8959 net_manager.cpp:93 Requesting /command/?get_debug from 172.168.1.10:8010
[ INFO] [1679565063.400252594]:  8959 lidar_client.cpp:894 innovusion_nodelet_manager remote debug: 6
[ INFO] [1679565063.400259688]:  8959 net_manager.cpp:93 Requesting /command/?get_udp_ports_ip from 172.168.1.10:8010
[ INFO] [1679565063.400691698]:  8959 lidar_client.cpp:894 innovusion_nodelet_manager remote udp_ports_ip: 8010,8010,8010,eth0,172.168.1.255
[ INFO] [1679565063.400700942]:  8959 net_manager.cpp:93 Requesting /command/?get_status_interval_ms from 172.168.1.10:8010
[ INFO] [1679565063.401117911]:  8959 lidar_client.cpp:894 innovusion_nodelet_manager remote status_interval_ms: 50
[ INFO] [1679565063.401126444]:  8959 net_manager.cpp:93 Requesting /command/?get_sn from 172.168.1.10:8010
[ INFO] [1679565063.402341843]:  8959 lidar_client.cpp:906 innovusion_nodelet_manager serial number: 371922210720
[ INFO] [1679565063.402366560]:  8959 net_manager.cpp:93 Requesting /command/?get_frame_rate from 172.168.1.10:8010
[ INFO] [1679565063.409042175]:  8959 lidar_client.cpp:915 innovusion_nodelet_manager frame_rate: 10.000000
[ INFO] [1679565063.409081497]:  8959 net_manager.cpp:93 Requesting /command/?get_udp_ports_ip from 172.168.1.10:8010
[ INFO] [1679565063.410094869]:  8959 stage_client_read.cpp:355 read udps: data:8010 message:8010 status:8010 ip=eth0,172.168.1.255 my_ip=
[ INFO] [1679565063.410111857]:  8959 stage_client_read.cpp:369 set_server_udp_ports_ip(8010)
[ INFO] [1679565063.410133210]:  8959 net_manager.cpp:93 Requesting /command/?set_udp_ports_ip=8010,8010,8010 from 172.168.1.10:8010
[ INFO] [1679565063.410636763]:  8959 net_manager.cpp:93 Requesting /command/?get_udp_ports_ip from 172.168.1.10:8010
[ INFO] [1679565063.411191583]:  8959 stage_client_read.cpp:355 read udps: data:8010 message:8010 status:8010 ip=eth0,172.168.1.255 my_ip=
[ INFO] [1679565063.411256491]:  8959 stage_client_read.cpp:242 innovusion_nodelet_manager wait for state 1
[ INFO] [1679565063.411424818]:  8960 stage_client_read.cpp:255 recvfrom UDP 8010
[ INFO] [1679565063.457292022]: frame #1: points=1275 (391 365 291 228) blocks=412
[ INFO] [1679565064.353766300]:  8958 stage_client_deliver.cpp:264 StageClientDeliever: convert_xyz mean/std/max/total=0.00ms/0.00/0.00/0 callback mean/std/max/total=0.03ms/0.03/0.30/244 total=273 total_dropped=0 data=244 message=10 status=19 points=21735 frames=10 points_2nd_return=0
[ INFO] [1679565064.353901492]:  8958 consumer_producer.cpp:495 deliver queue#0 added=273 finished=272 dropped=0 blocked=0 wait=68us process=31us drop=0us pid=8958 elapsed_time=957/957ms active_time=8/8ms ratio=0.90%/0.90%
[ INFO] [1679565064.432722366]: REMOTE-[ INFO] level=6, code=0, message=[ INFO] CODE=0 [FROM_FW]  2021-12-28 23:41:35.646 [ INFO] delay set config done

[ INFO] [1679565064.954081151]: frame #16: points=2270 (553 492 601 624) blocks=923
[ INFO] [1679565065.774826466]: REMOTE-[ INFO] level=6, code=0, message=[ INFO] CODE=0 2021-12-28 23:41:36.987  24508 stage_signal_reference.cpp:204 ref channel: 0, roi_type: 1, locked (3209729/0) half_w=(32/32) center=843, bound limited: 0/0
[ INFO] [1679565065.898364595]:  8960 net_manager.cpp:93 Requesting /command/?get_mode_status from 172.168.1.10:8010
[ WARN] [1679565065.902604909]:  8960 resource_stats.cpp:150 <READ> bandwidth too low: 0.33046, counter: 1
[ INFO] [1679565065.902723231]:  8960 resource_stats.cpp:160 innovusion_nodelet_manager pid=8926, uptime=2.50s, #RESOURCE_STATS# <READ> packets=723/723, bytes=807K/807K, bandwidth=0.33M/s; <DATA> packets=645/645, bytes=772K/772K, bandwidth=0.32M/s; <POINT> frames=25/25, points=55K/55K, point-rate=0.02M/s, ref_intensity_sum = 0, ref_count_total = 0; <MESSAGE> packets=27/27, bytes=4K/4K, bandwidth=0.00M/s; <STATUS> packets=50/50, bytes=29K/29K, bandwidth=0.01M/s;
[ INFO] [1679565065.902996838]: REMOTE-[ INFO] level=6, code=0, message=[ INFO] CODE=0 2021-12-28 23:41:37.116  26973 server_ws.cpp:104 Server: Connect connection 0x7f78010cd0. path: ^/command/?.*$ query: get_mode_status. From 172.168.1.111
[ INFO] [1679565065.903080731]: REMOTE-[ INFO] level=6, code=0, message=[ INFO] CODE=0 2021-12-28 23:41:37.116  26973 pcs.cpp:896 command get_mode_status ret=0 result= 3,2,2,0
[ INFO] [1679565065.903128779]: REMOTE-[ INFO] level=6, code=0, message=[ INFO] CODE=0 2021-12-28 23:41:37.116  26973 server_ws.cpp:51 Server: Event in connection 0x7f78010cd0. message: End of file
[ INFO] [1679565066.466663089]: frame #31: points=2285 (552 494 608 631) blocks=931
[ INFO] [1679565066.694277172]: REMOTE-[ INFO] level=6, code=0, message=[ INFO] CODE=0 2021-12-28 23:41:37.908  24501 status_report.cpp:372 counters: point_data_packet_sent=45809543 point_sent=2048759418 message_packet_sent=1790840 raw_data_read=653265338368 total_frame=3581679 total_polygon_rotation=14326721 total_polygon_facet=71633602 power_up_time_in_second=791689 process_up_time_in_second=791684 lose_ptp_sync=0 bad_data[4]=0/0/0/0 data_drop[8]=0/0/0/0/0/0/0/0 signals[8]=28959065/2617744848/0/2768625976/0/2579918756/2696291965/3372156756 latency_10us_average[6]=0/261/69/1048/0/16 latency_10us_variation[6]=0/0/62/16/0/1 latency_10us_max[6]=0/3153/1326/3401/0/230 big_latency_frame=0 bad_frame=0 big_gap_frame=0 small_gap_frame=0 cpu_percentage=100 mem_percentage=4 netstat_rx speed/drop/err=0kBps/0/0 netstat_tx speed/drop/err=338kBps/0/0 sys_cpu_percentage=28/20/27/25 motor[5]=35582/6872/12216/39784/9685 galvo[5]=36134/28976/62880/41784/28125 in_faults=0x137438953496;sensor_readings: t_fpga=520 t_laser=510 t_adc=580 t_board=0 t_det=550/550/560/560 t_other=0/0/0 heater_ma=0 m_rpm=4801000 g_fpm=10000 m_total=0 g_total=281 moisture=0/0 window_blockage=0/0 motor=220/0/0/0/0/0 galvo=22/0/0/0/0/0 laser=2700/0/0/0/0/0 galvo_client=0x0/0
```

4.2.3 View LiDAR point cloud data

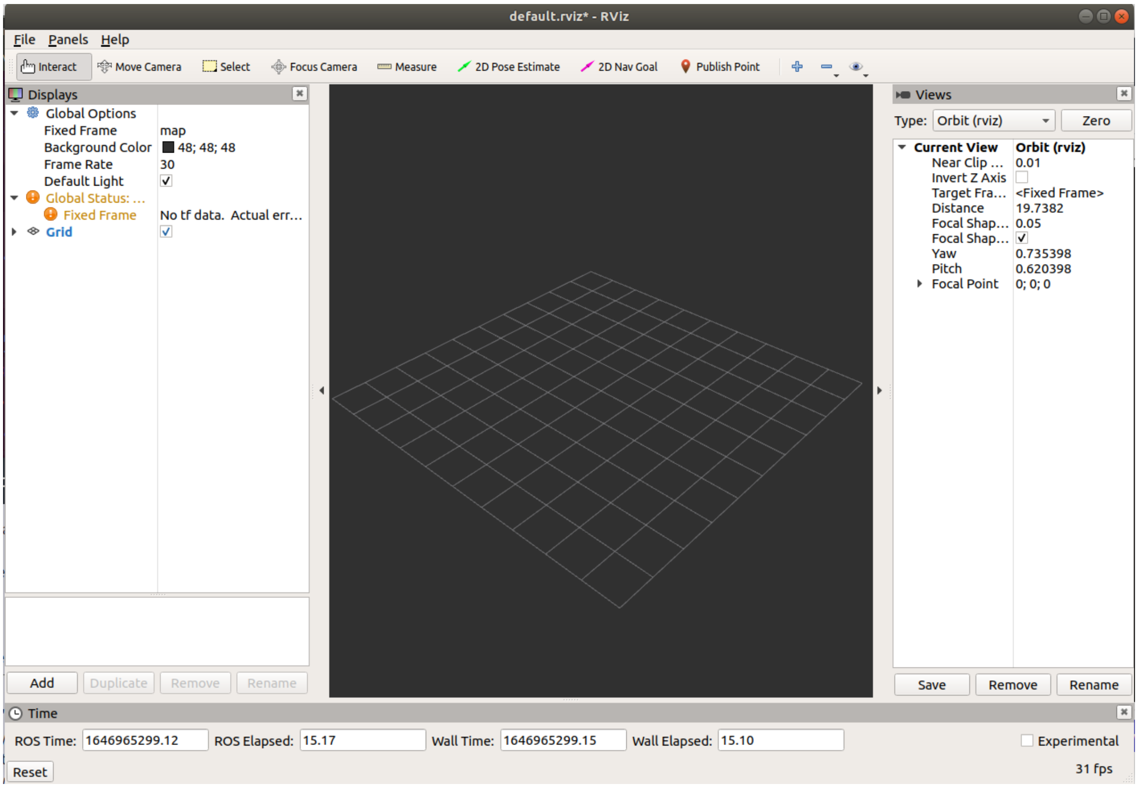
Note

Before viewing the point cloud data, please confirm that the point cloud data has been obtained.

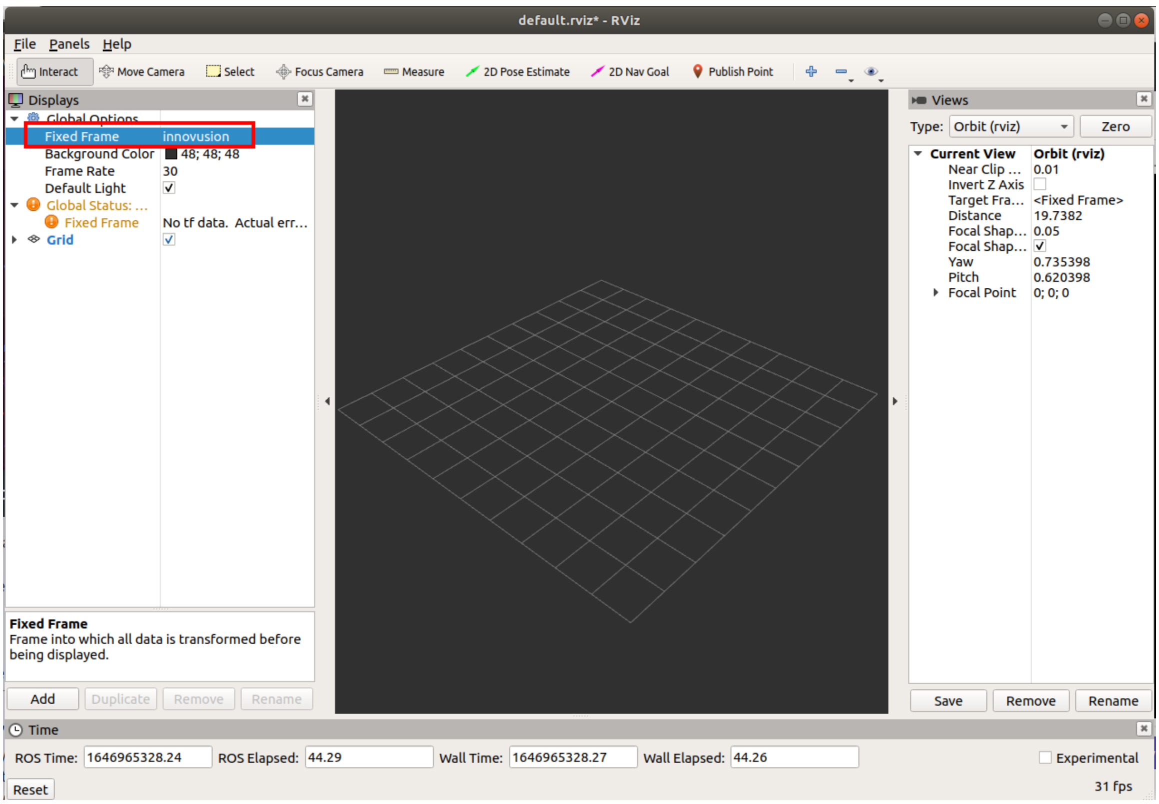
1. Start the graphical tool **rviz**. The return value and **rviz** interface are shown below.

```
rviz
```

```
demo@demo-OMEN-by-HP-Laptop-16-b0xxx:~$ rviz rviz
[ INFO] [1646808931.535449917]: rviz version 1.13.21
[ INFO] [1646808931.535492615]: compiled against Qt version 5.9.5
[ INFO] [1646808931.535502544]: compiled against OGRE version 1.9.0 (Ghadamon)
[ INFO] [1646808931.539157206]: Forcing OpenGl version 0.
[ INFO] [1646808932.122437501]: Stereo is NOT SUPPORTED
[ INFO] [1646808932.122552411]: OpenGL device: NVIDIA GeForce RTX 3060 Laptop GPU/PCIe/SSE2
[ INFO] [1646808932.122633840]: OpenGl version: 4.6 (GLSL 4.6).
QObject::connect: Cannot queue arguments of type 'QVector<int>'
(Make sure 'QVector<int>' is registered using qRegisterMetaType().)
QObject::connect: Cannot queue arguments of type 'QVector<int>'
(Make sure 'QVector<int>' is registered using qRegisterMetaType().)
```



2. Select **Global Options** > **Fixed Frames**. Set the Fixed Frames value to **innovusion**.

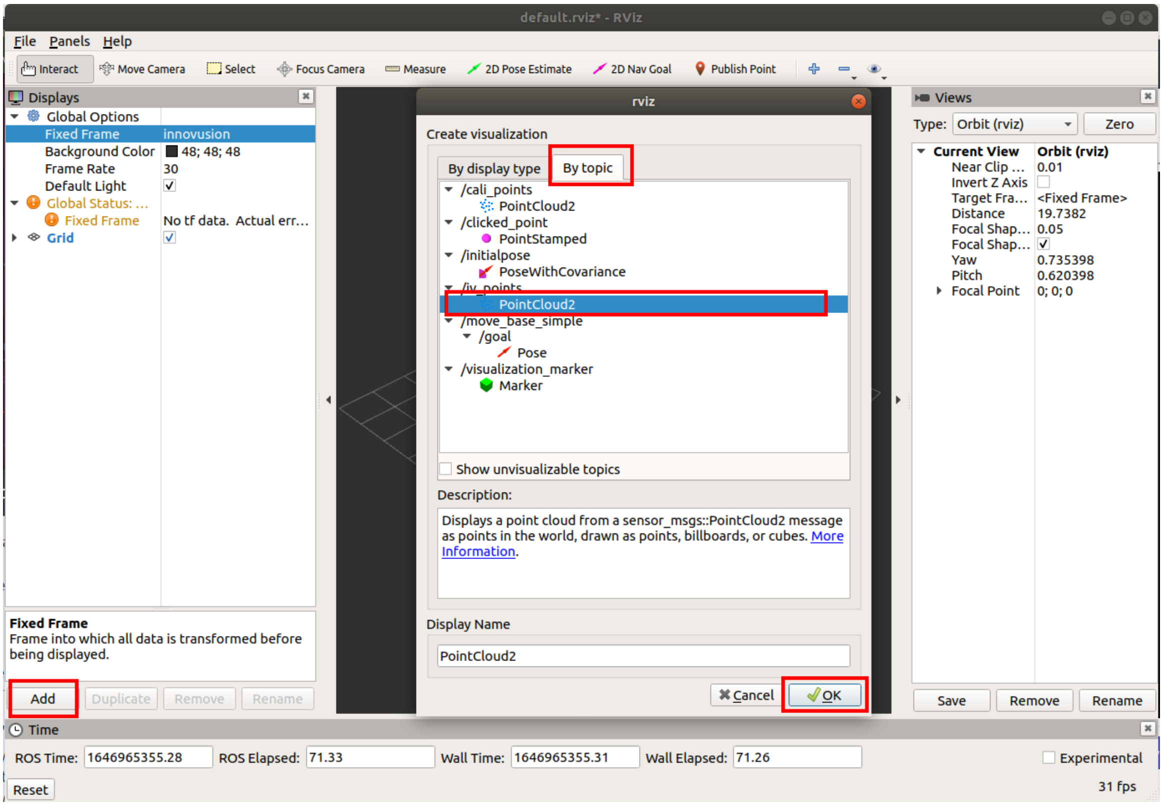


i. Add PointCloud2 to Displays.

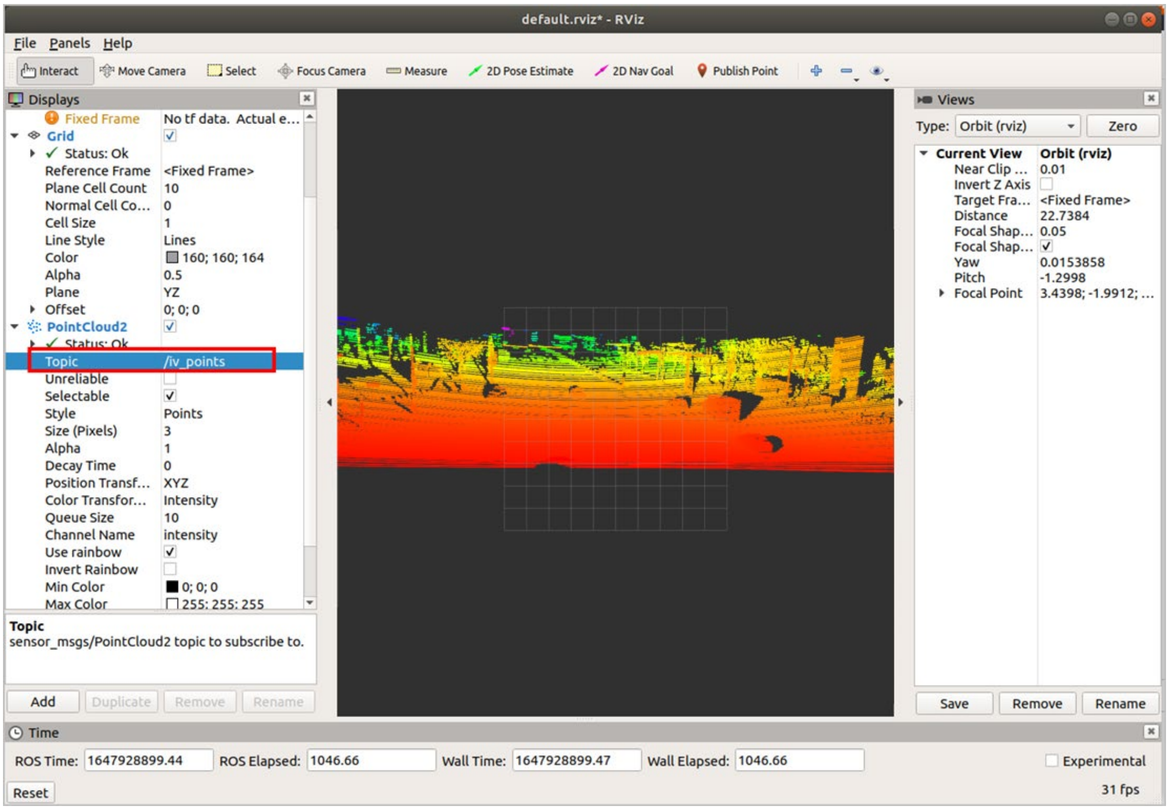
a. Click **Add.**

b. Select **By topic** > **iv_points** > **PointCloud2**.

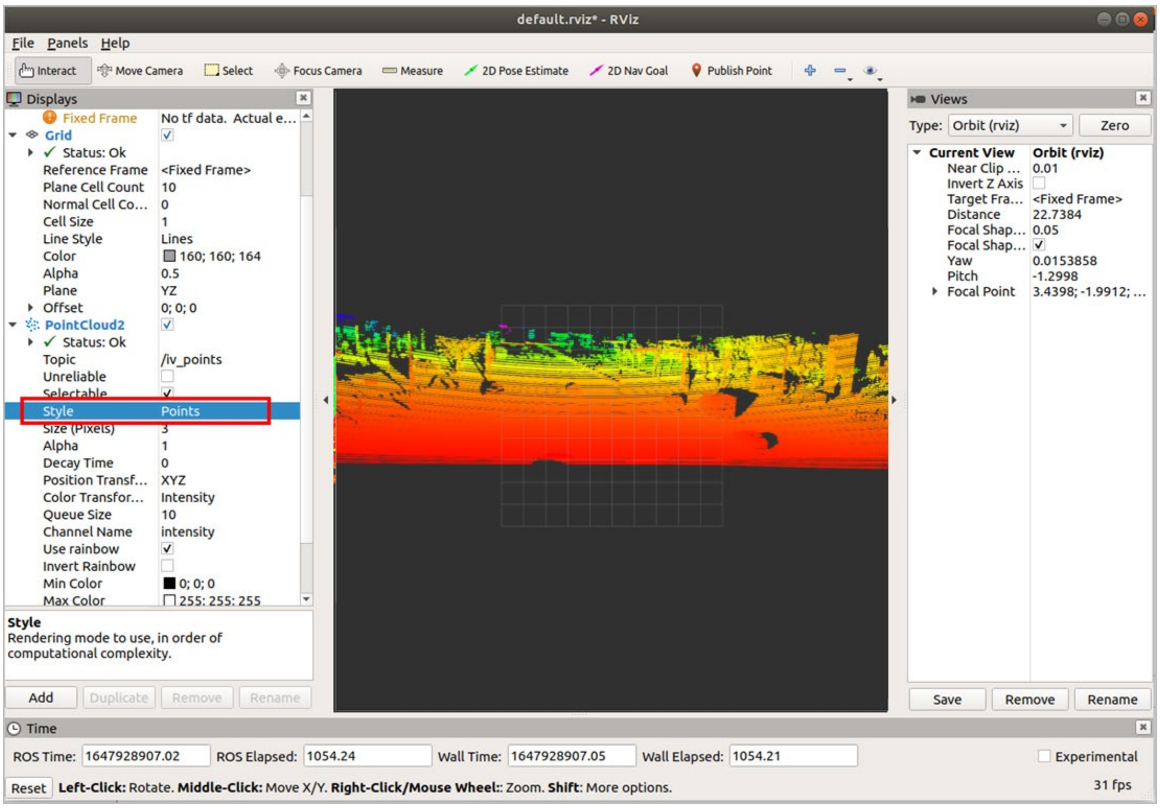
c. Click **OK**.



ii. Select **PointCloud2** > **Topic**. Set the Topic value to **/iv_points**.



iii. Select **PointCloud2** > **Style**. Set the Style value to **Points**.

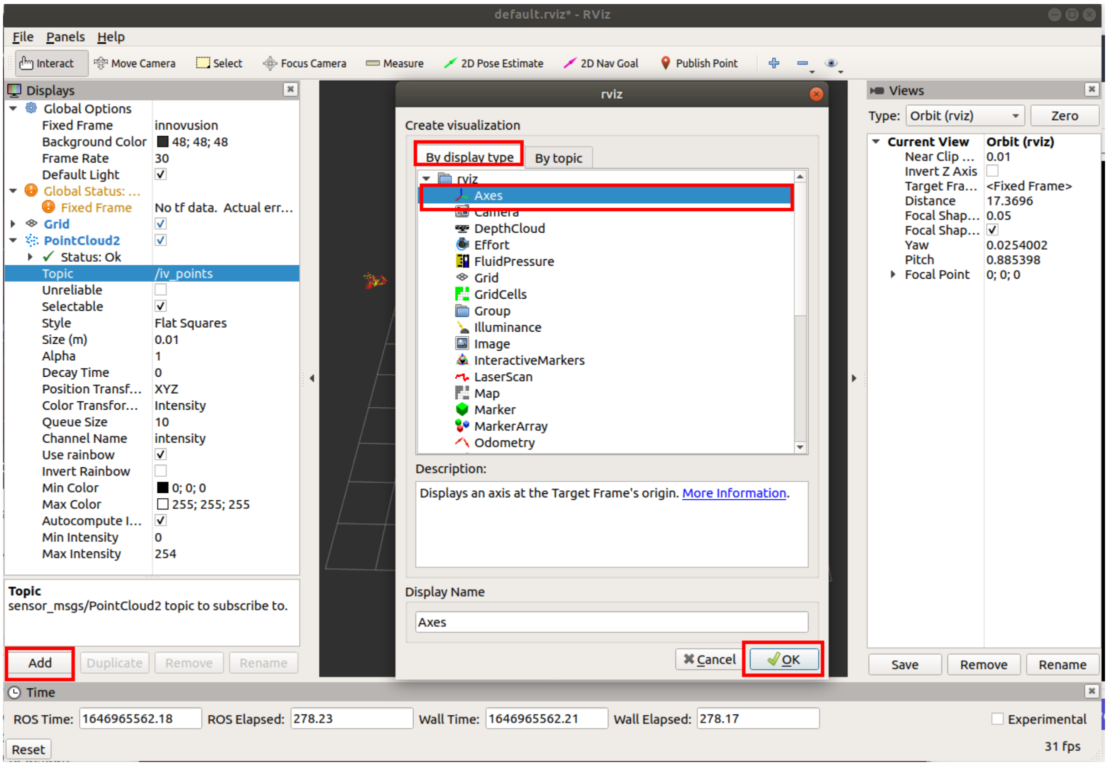


3. (Optional) You can change the angle and distance of the real-time point cloud status and get more information as needed.

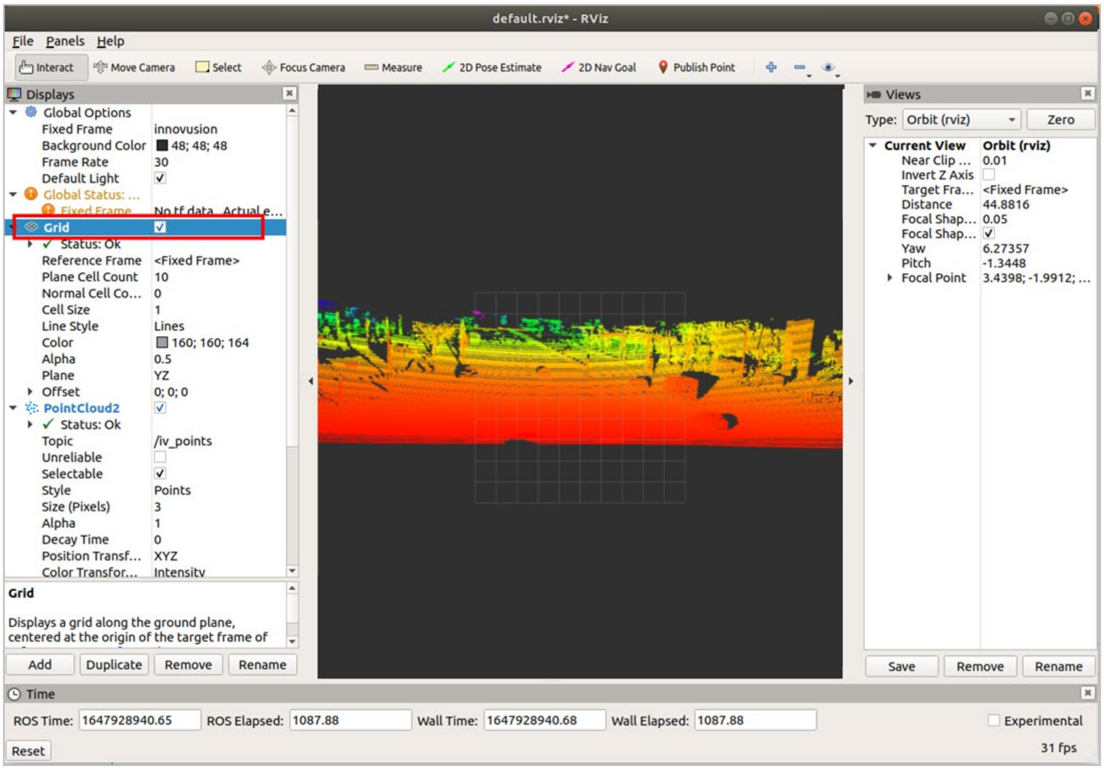
- You can select **Axes** to add the coordinate system to the diagram as a reference.

  a. Select **Add** > **By display type** > **Axes**.

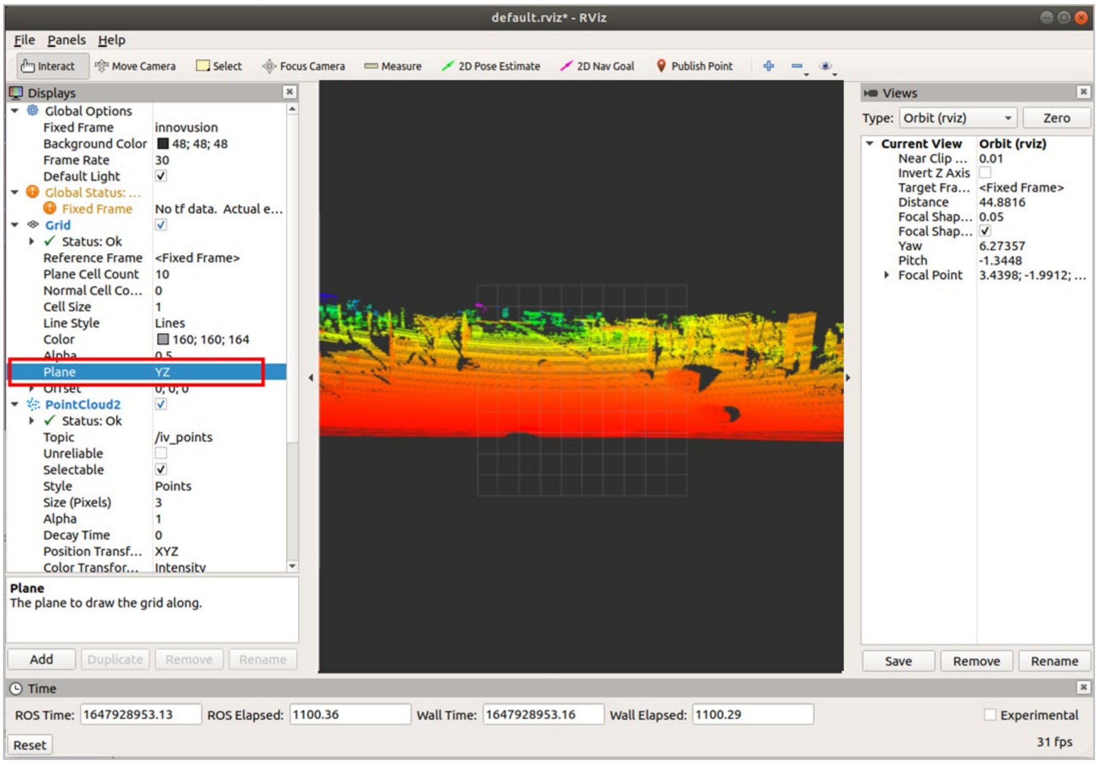
  b. Click **OK**.



- Select **Grid** to add the grid to the diagram as a reference. Grid is enabled by default when rviz is started.



- Set the **plane** value to view the point cloud status under different coordinate systems. There are three options: **XY**, **XZ**, and **YZ**.



4.2.4 Record LiDAR point cloud data

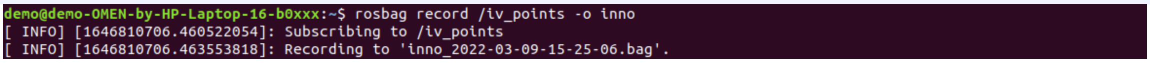
You can record the point cloud data of the LiDAR in bag format in ROS.

Note

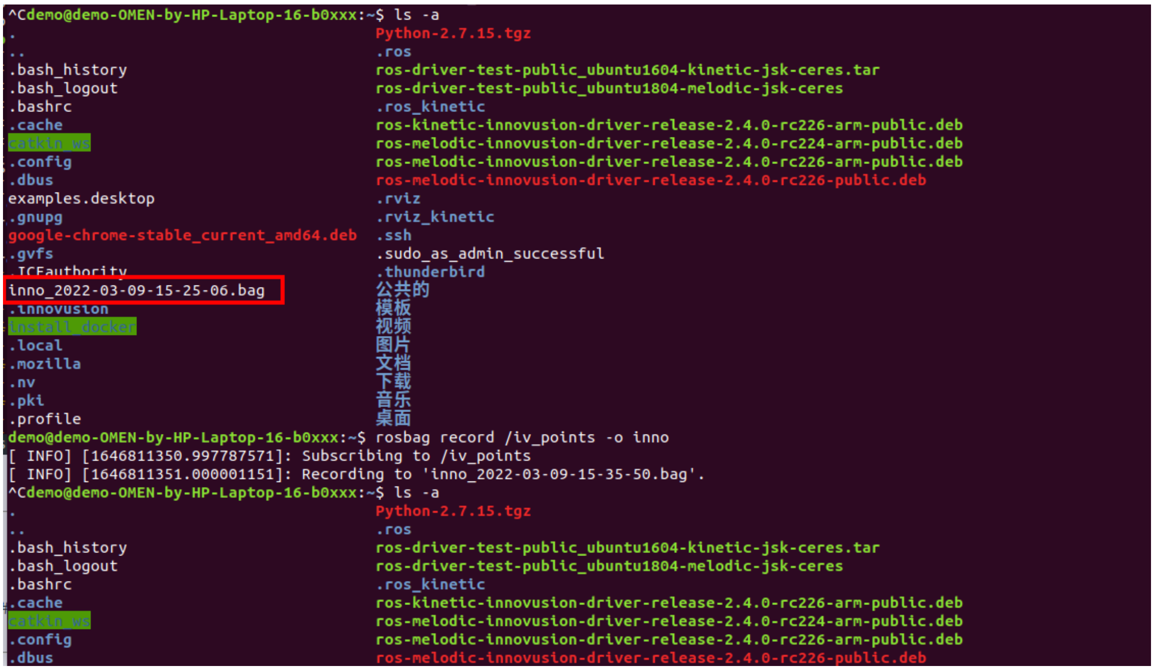
Before recording the point cloud data of the LiDAR, please confirm that the point cloud data has been obtained correctly in ROS. For information on how to get the point cloud data, see 4.2.2 Obtain point cloud data.

1. Record point cloud data in bag format. Recording starts at the execution time.

```
rosbag record /iv_points -o inno //Start to record the point cloud data in bag format. The file is saved in the current path and the file name is “inno-Year-Mon-Day-Hr-Min.bag”
```



2. Press **Ctrl+C** to stop recording point cloud data.
3. (Optional) Execute ls - a command to check the directory of recorded point cloud data.



4.2.5 Replay LiDAR point cloud data

You can replay the point cloud data in bag format in ROS environment.

Note

Before replaying LiDAR point cloud data, please confirm that the recorded point cloud data file has been obtained.

1. Run ROS. The return value is shown in the figure.

```
roscore
```

```
demo@demo-OMEN-by-HP-Laptop-16-b0xxx:~$ roscore
... logging to /home/demo/.ros/log/a09b36de-9f71-11ec-874a-c85acfaa1d16/roslaunch-demo-OMEN-by-HP-Laptop-16-b0xxx-9812.log
Checking log directory for disk usage. This may take a while.
Press Ctrl-C to interrupt
Done checking log file disk usage. Usage is <1GB.

started roslaunch server http://demo-OMEN-by-HP-Laptop-16-b0xxx:42677/
ros_comm version 1.14.12


SUMMARY
========

PARAMETERS
 * /rosdistro: melodic
 * /rosversion: 1.14.12

NODES

auto-starting new master
process[master]: started with pid [9822]
ROS_MASTER_URI=http://demo-OMEN-by-HP-Laptop-16-b0xxx:11311/

setting /run_id to a09b36de-9f71-11ec-874a-c85acfaa1d16
process[rosout-1]: started with pid [9833]
started core service [/rosout]
```

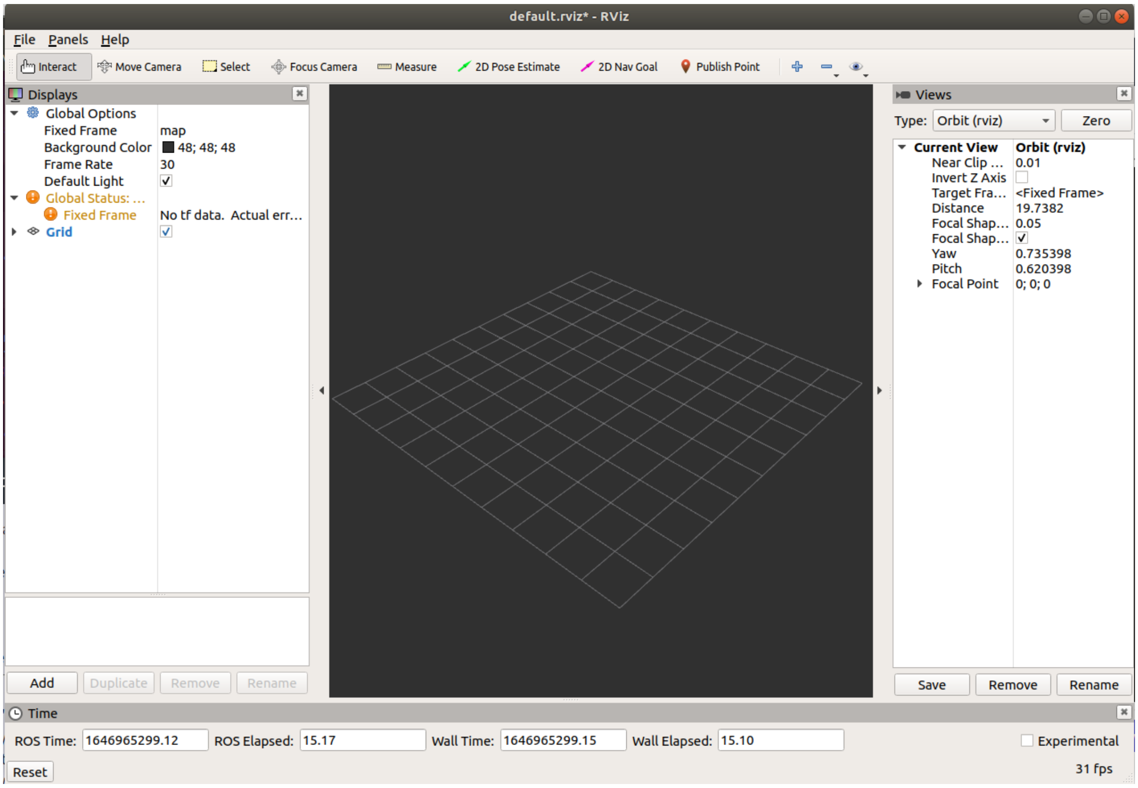
2. Start the graphical tool **rviz**. The return value and **rviz** interface are shown below.

```
rviz
```

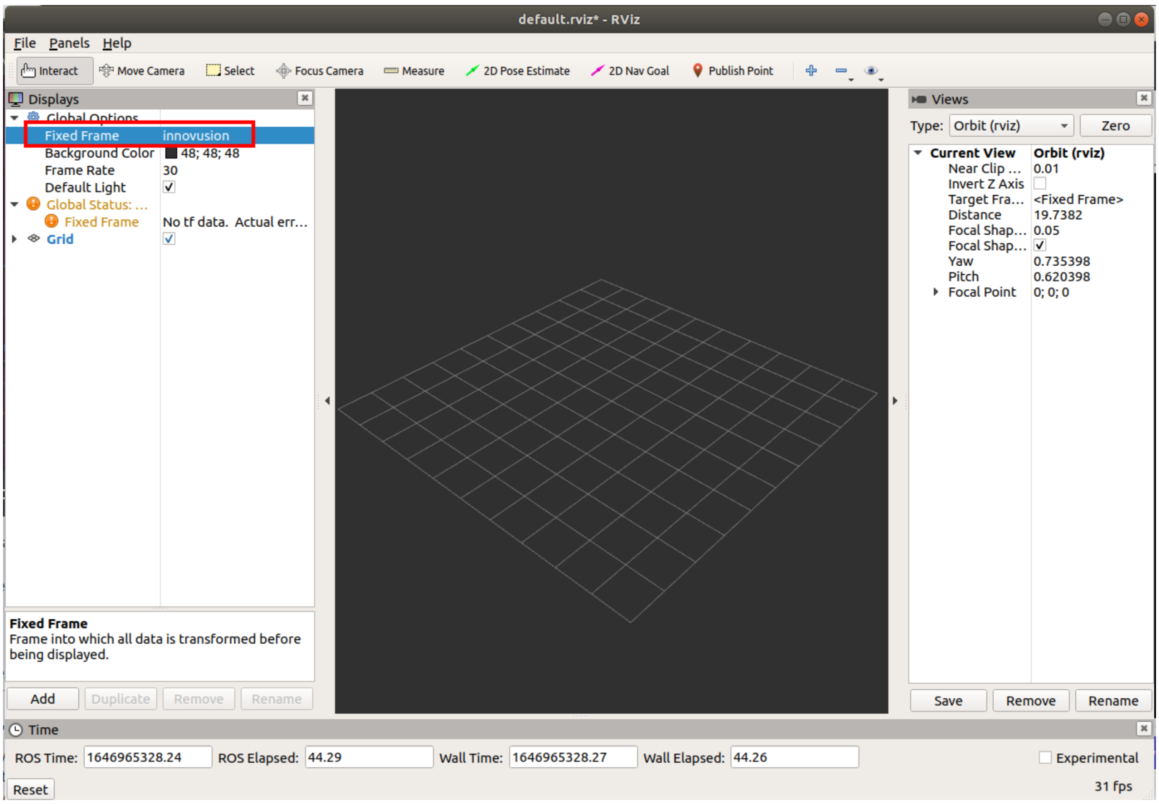
```
demo@demo-OMEN-by-HP-Laptop-16-b0xxx:~$ rviz rviz
[ INFO] [1646808931.535449917]: rviz version 1.13.21
[ INFO] [1646808931.535492615]: compiled against Qt version 5.9.5
[ INFO] [1646808931.535502544]: compiled against OGRE version 1.9.0 (Ghadamon)
[ INFO] [1646808931.539157206]: Forcing OpenGl version 0.
[ INFO] [1646808932.122437501]: Stereo is NOT SUPPORTED
[ INFO] [1646808932.122552411]: OpenGL device: NVIDIA GeForce RTX 3060 Laptop GP
U/PCIe/SSE2
[ INFO] [1646808932.122633840]: OpenGl version: 4.6 (GLSL 4.6).
QObject::connect: Cannot queue arguments of type 'QVector<int>'
(Make sure 'QVector<int>' is registered using qRegisterMetaType().)
QObject::connect: Cannot queue arguments of type 'QVector<int>'
(Make sure 'QVector<int>' is registered using qRegisterMetaType().)
```

3. Replay LiDAR point cloud data in **rviz**.

```
rosbag play <filename.bag>
```



4. Select **Global Options** > **Fixed Frames**. Set the **Fixed Frames** value to **innovusion**.



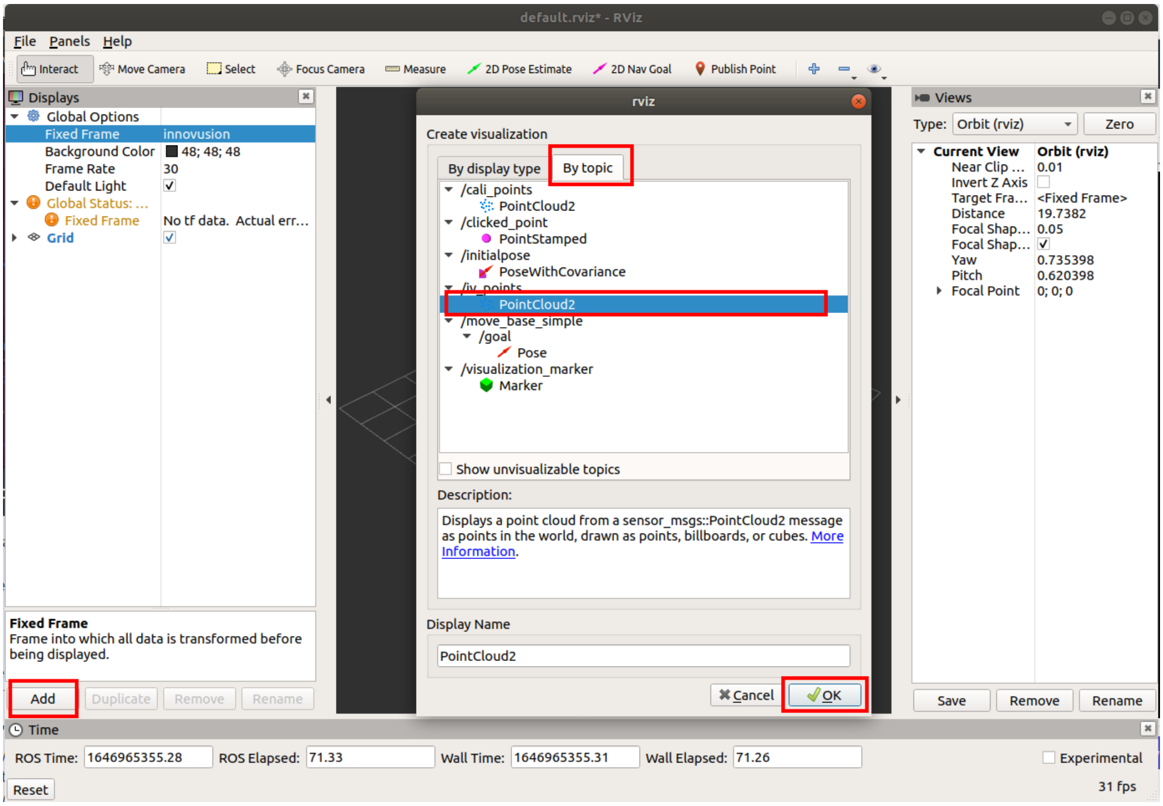
5. Add and configure **PointCloud2**.

i. Add **PointCloud2** to the Displays.

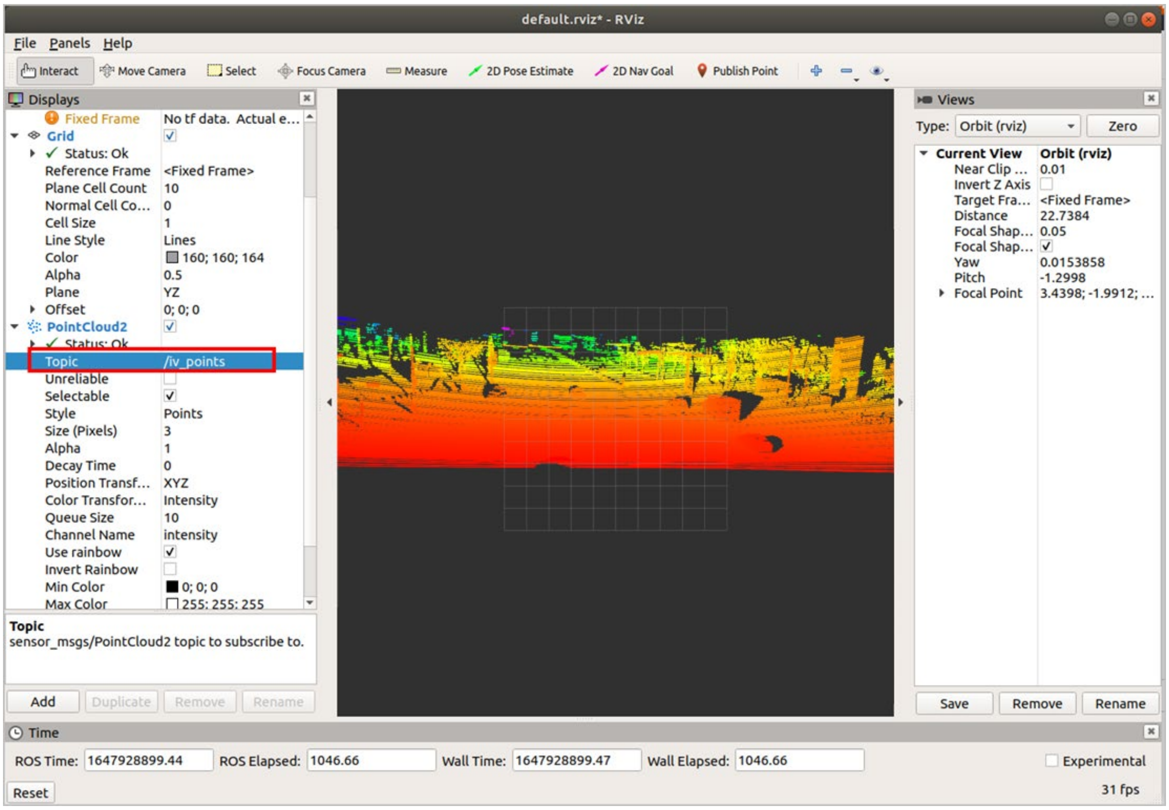
a. Click **Add.**

b. Select **By topic** > **display type** > **PointCloud2**.

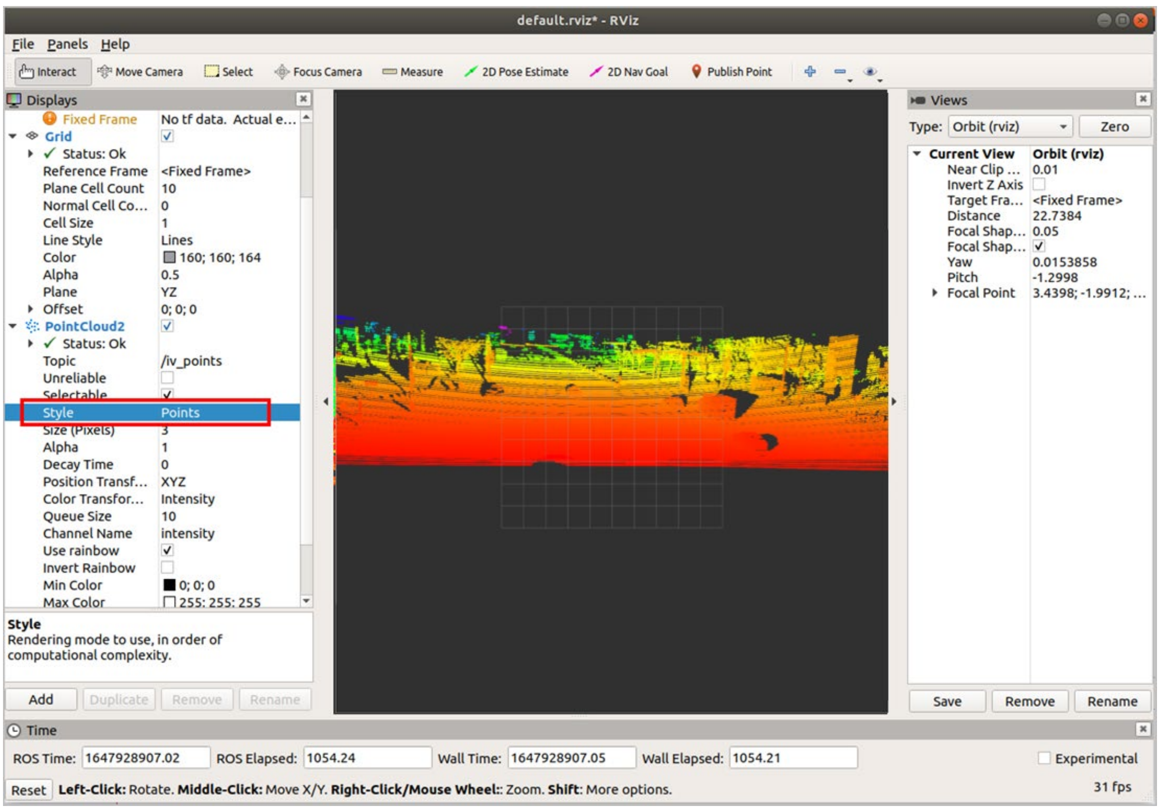
c. Click **OK**.



ii. Select PointCloud2 > Topic. Set the Topic value to /iv_points.



iii. Select **PointCloud2** > **Style**. Set the Style value to **Points**.

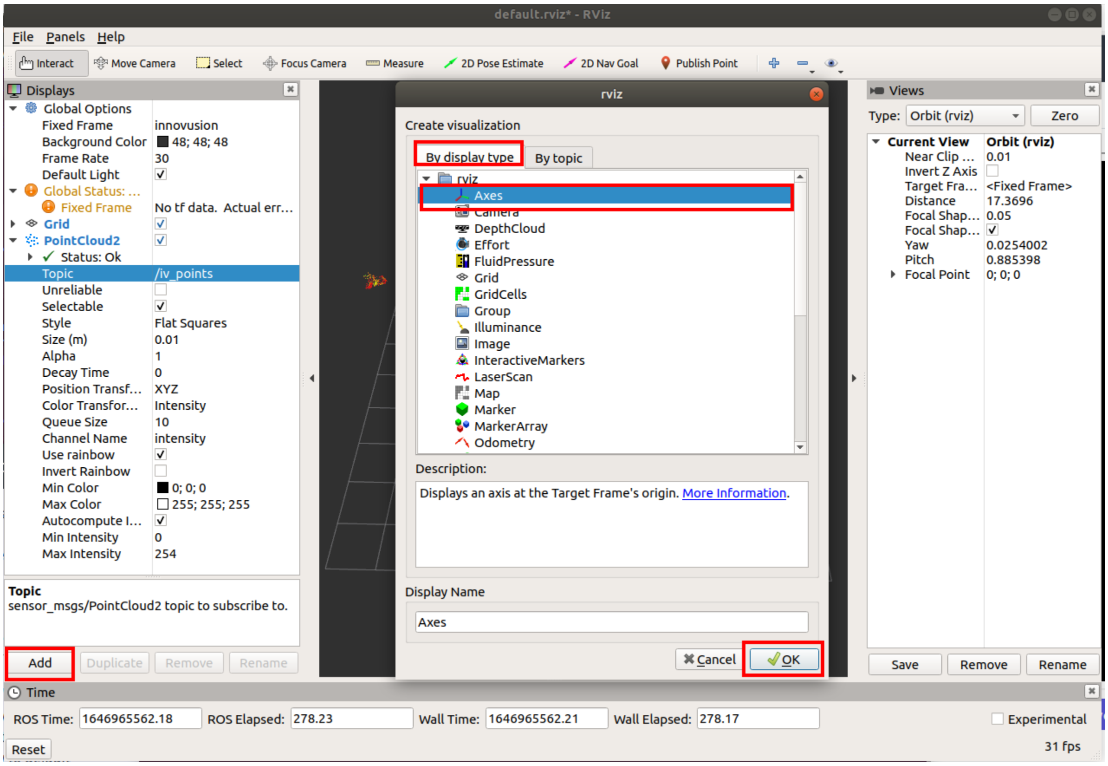


6. Press **Space** to pause playback of the point cloud data file.
7. (Optional) You can change the angle and distance of the real-time point cloud status and get more information as needed.

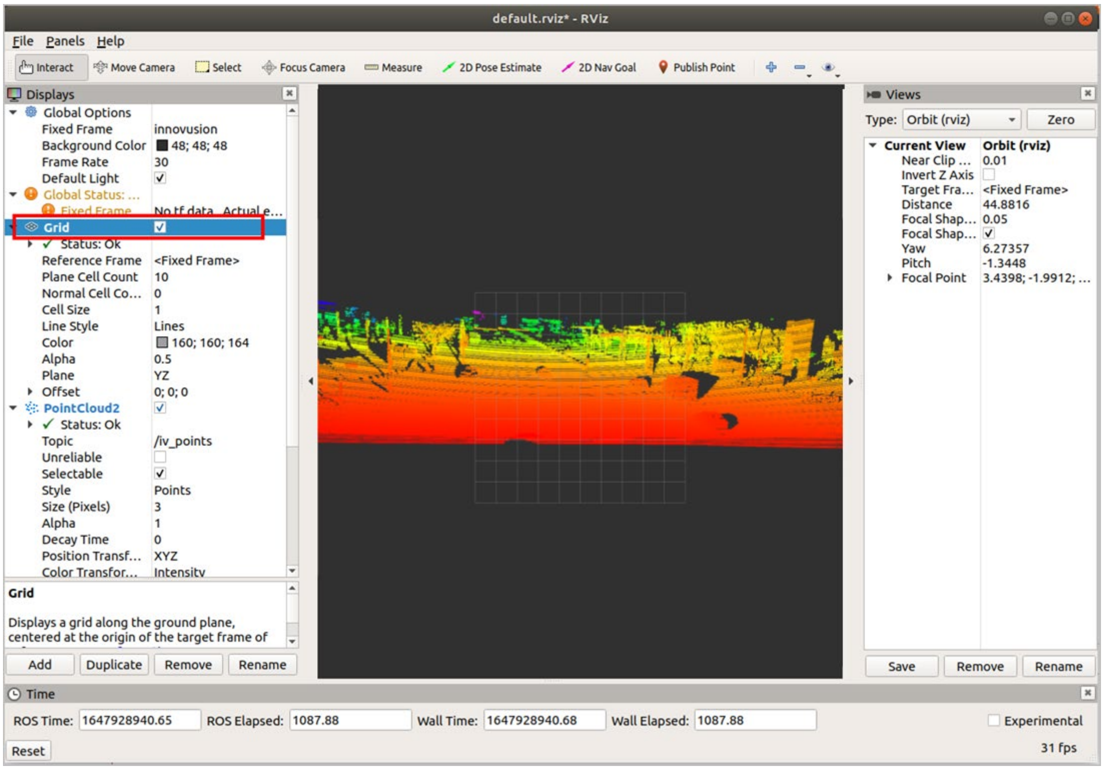
- You can select **Axes** to add the coordinate system to the diagram as a reference.

  a. Select **Add** > **By display type** > **Axes**.

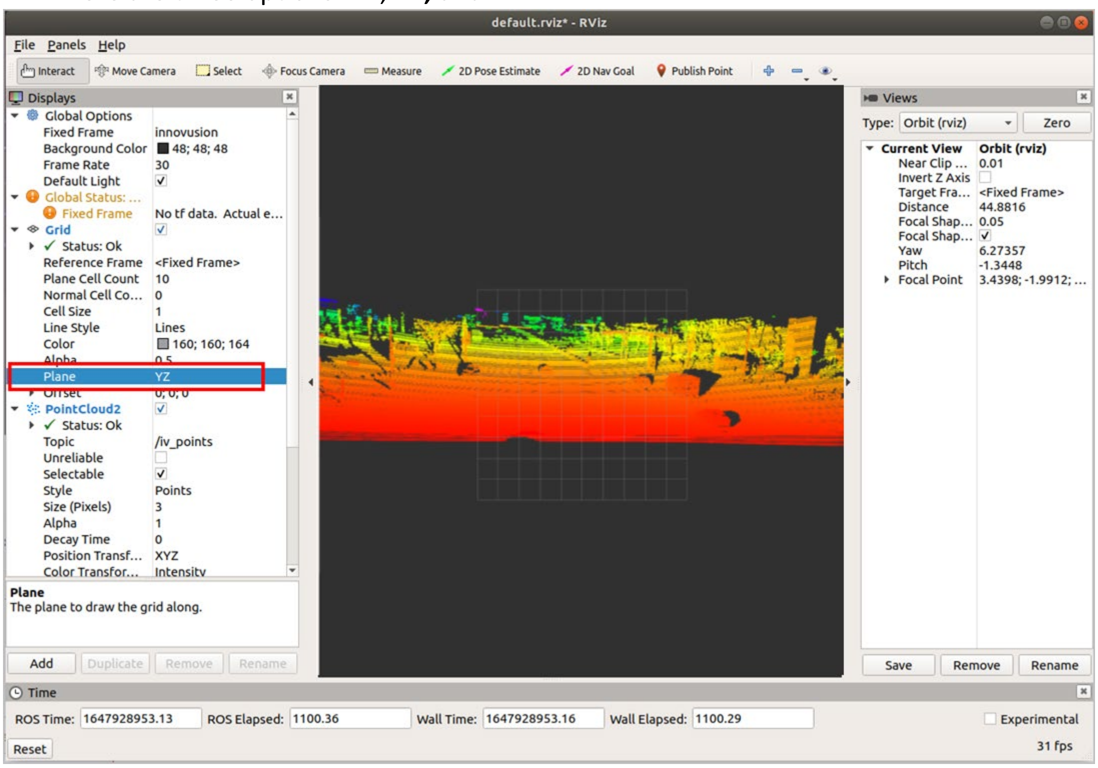
  b. Click **OK**.



- Select **Grid** to add the grid to the diagram as a reference. Grid is enabled by default when **rviz** is started.



- Set the **plane** value to view the point cloud status under different coordinate systems. There are three options: **XY**, **XZ,** and **YZ**.



4.2.6 (Optional) Convert a file in rosbag format to a file in pcd format

1. Run ROS. The return value is shown in the figure below.

```
roscore
```

2. Convert the file in bag format to a file in pcd format.

```
rosrun pcl_ros pointcloud_to_pcd input:=/iv_points
```

3. Play a file in bag format.

```
rosbag play <filename.bag>
```

4.2.7 Shut down the LiDAR

Disconnect the power supply to shut down the LiDAR.

4.3 Operate on MetaView

You can use MetaView on multiple operating systems including Linux, MacOS and Windows. This chapter takes operations on Windows as an example.

4.3.1 Start the LiDAR

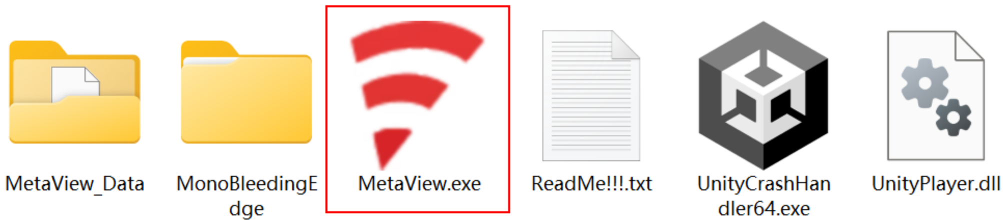
1. Connect the power supply to start the LiDAR.
2. The LiDAR completes initialization and generates data after powering on for 11 to 18 seconds.

Note

The LiDAR does not have a power switch. It will become operational when power is applied.

4.3.2 Open MetaView

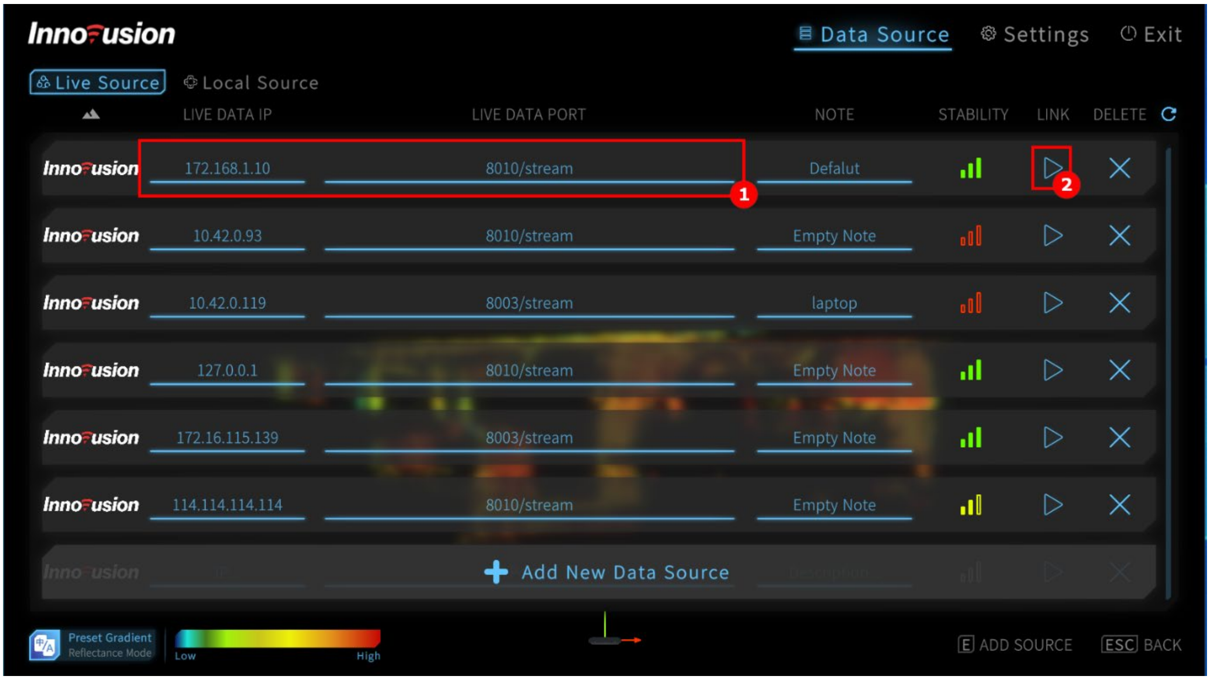
Unzip the MetaView package, double-click to open MetaView.exe.



4.3.3 Add a LiDAR

1. Connect the computer to LiDAR.
2. Change the computer IP address to the same subnet with the LiDAR.
3. Go to **Data Source** > **Live Source**. Click **Add New Data Source**.

4. Enter LiDAR IP address and port number. Click  to connect the LiDAR. The default port number is 8010.



Note

- The default LiDAR IP address is 172.168.1.10.
- It is recommended to check the access to the LiDAR IP address via the ping command. The return value is shown in the figure below.

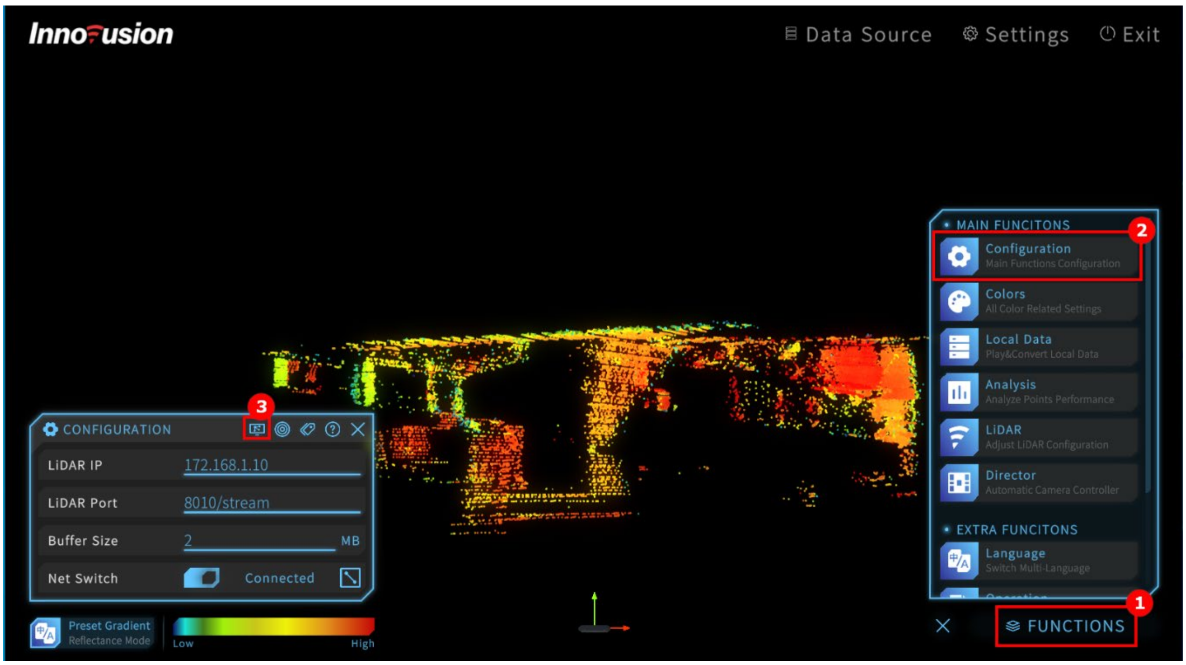
```
demo@demo-OMEN-by-HP-Laptop-16-b0xxx:~$ ping 172.168.1.10
PING 172.168.1.10 (172.168.1.10) 56(84) bytes of data.
64 bytes from 172.168.1.10: icmp_seq=70 ttl=64 time=0.448 ms
64 bytes from 172.168.1.10: icmp_seq=71 ttl=64 time=0.222 ms
64 bytes from 172.168.1.10: icmp_seq=72 ttl=64 time=0.200 ms
64 bytes from 172.168.1.10: icmp_seq=73 ttl=64 time=0.208 ms
64 bytes from 172.168.1.10: icmp_seq=74 ttl=64 time=0.200 ms
64 bytes from 172.168.1.10: icmp_seq=75 ttl=64 time=0.219 ms
64 bytes from 172.168.1.10: icmp_seq=76 ttl=64 time=0.255 ms
64 bytes from 172.168.1.10: icmp_seq=77 ttl=64 time=0.212 ms
64 bytes from 172.168.1.10: icmp_seq=78 ttl=64 time=0.206 ms
64 bytes from 172.168.1.10: icmp_seq=79 ttl=64 time=0.170 ms
64 bytes from 172.168.1.10: icmp_seq=80 ttl=64 time=0.207 ms
64 bytes from 172.168.1.10: icmp_seq=81 ttl=64 time=0.207 ms
64 bytes from 172.168.1.10: icmp_seq=82 ttl=64 time=0.145 ms
64 bytes from 172.168.1.10: icmp_seq=83 ttl=64 time=0.168 ms
64 bytes from 172.168.1.10: icmp_seq=84 ttl=64 time=0.316 ms
64 bytes from 172.168.1.10: icmp_seq=85 ttl=64 time=0.192 ms
64 bytes from 172.168.1.10: icmp_seq=86 ttl=64 time=0.309 ms
64 bytes from 172.168.1.10: icmp_seq=87 ttl=64 time=0.295 ms
^C
--- 172.168.1.10 ping statistics ---
87 packets transmitted, 18 received, 79% packet loss, time 88040ms
rtt min/avg/max/mdev = 0.145/0.232/0.448/0.069 ms
demo@demo-OMEN-by-HP-Laptop-16-b0xxx:~$
```

4.3.4 Delete a LiDAR

1. Go to **Data Source** > **Live Source**.
2. Click to delete the LiDAR information.

4.3.5 Configure LiDAR network information

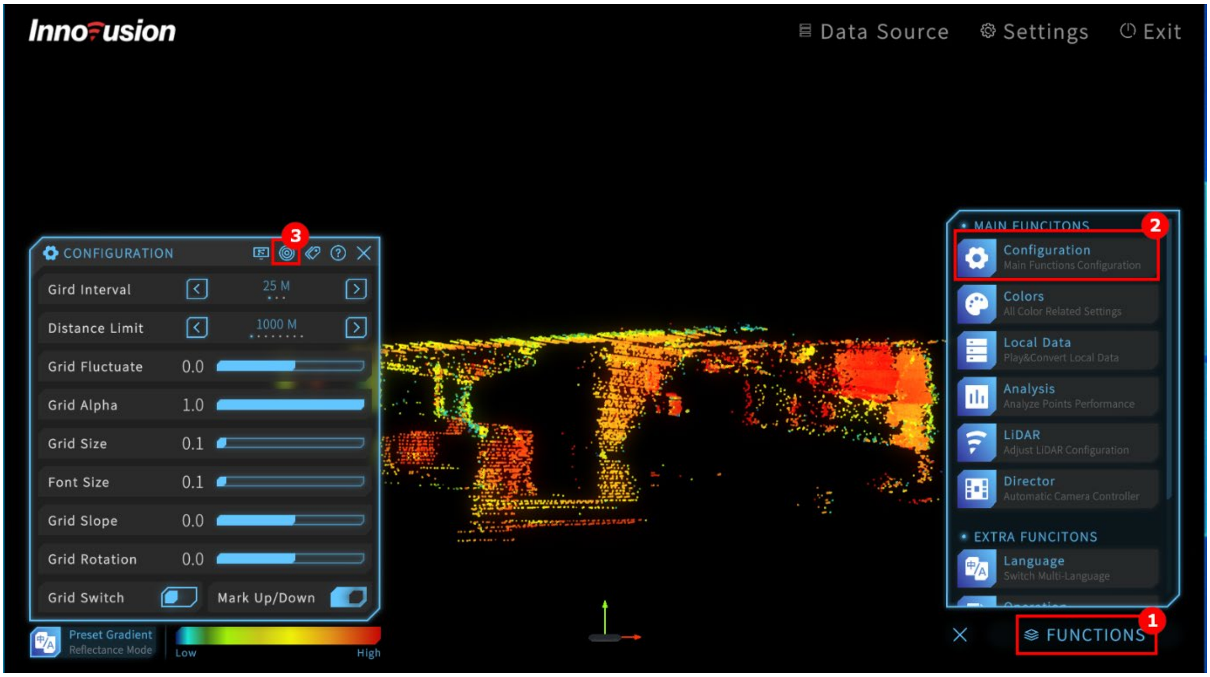
1. Go to **FUNCTIONS** > **Configuration**. Click .



2. If the LiDAR’ s network information has been changed, you can update the settings on this page or on the **Live Source** page.

4.3.6 Configure grid

1. Go to **FUNCTIONS** > **Configuration**. Click .



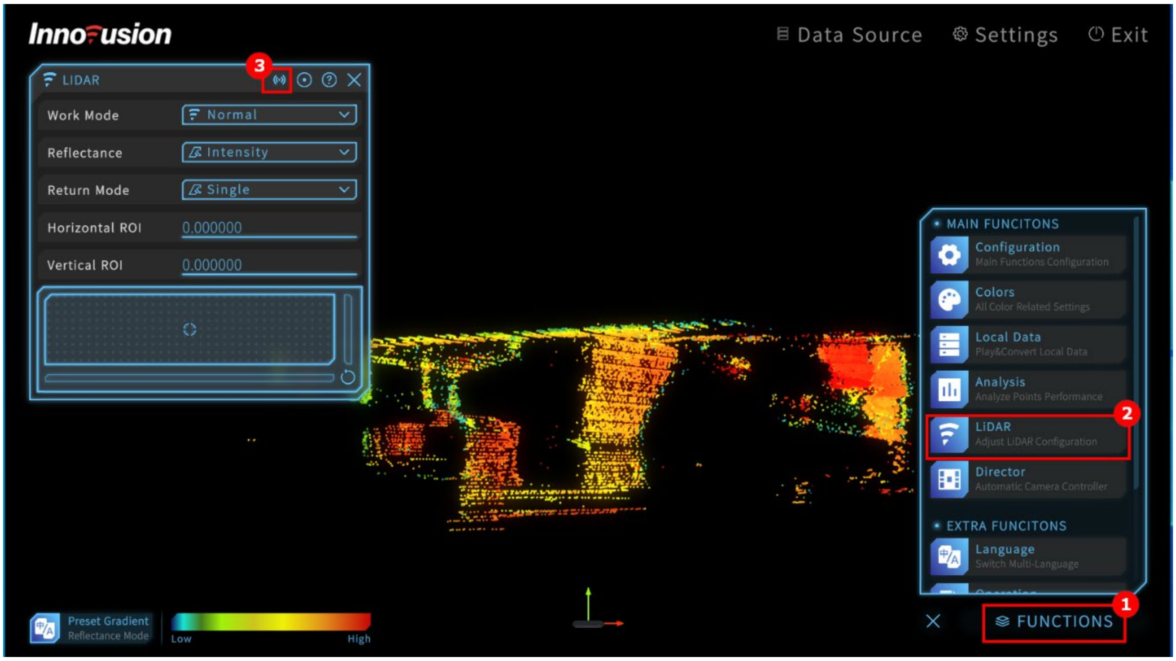
2. You can change the grid configuration according to the actual needs.

Table 11 Parameter description

Parameter	Description
Grid Interval	The interval between adjacent grids.
Distance Limit	The maximum distance between the adjacent grid lines.
Grid Fluctuate	The vertical position of the grid.
Grid Alpha	The transparency of the grid.
Grid Size	The thickness of the grid line.
Font Size	The font size in the grid.
Grid Slope	The vertical angle of the grid.
Grid Rotation	The horizontal angle of the grid.

4.3.7 Configure LiDAR reflectance

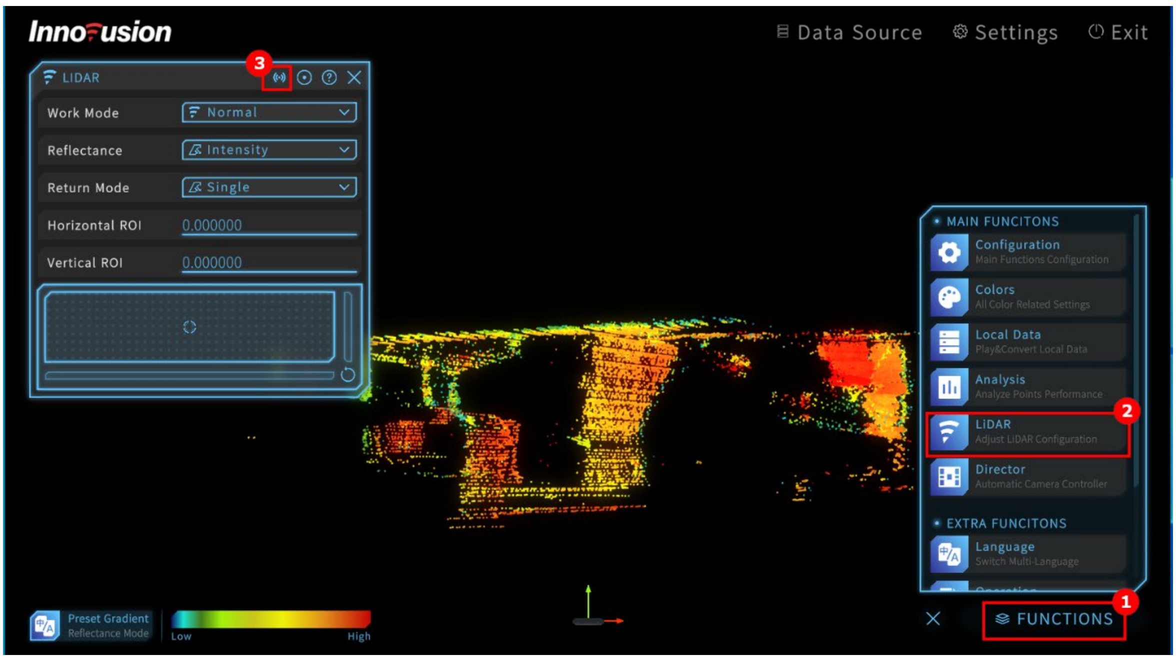
1. Go to **FUNCTIONS** > **LiDAR**.



2. Change the reflectance of the LiDAR. The reflectance can be either **intensity** or **reflectivity**.

4.3.8 Configure return mode

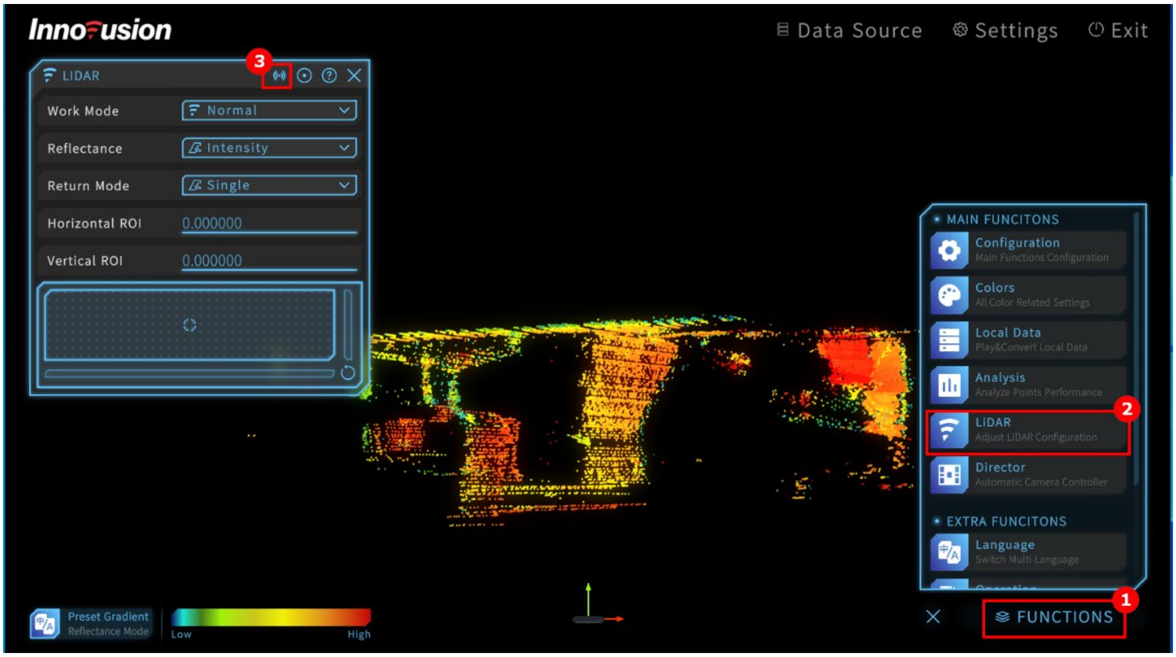
1. Select **FUNCTIONS** > **LiDAR**.



2. Configure the return mode received by the LiDAR when a laser is emitted once. Either single return mode or dual return mode can be selected, and the dual return mode has two options: "strongest + 2 strongest" and " strongest & furthest". The default is single return mode.

4.3.9 Configure work mode

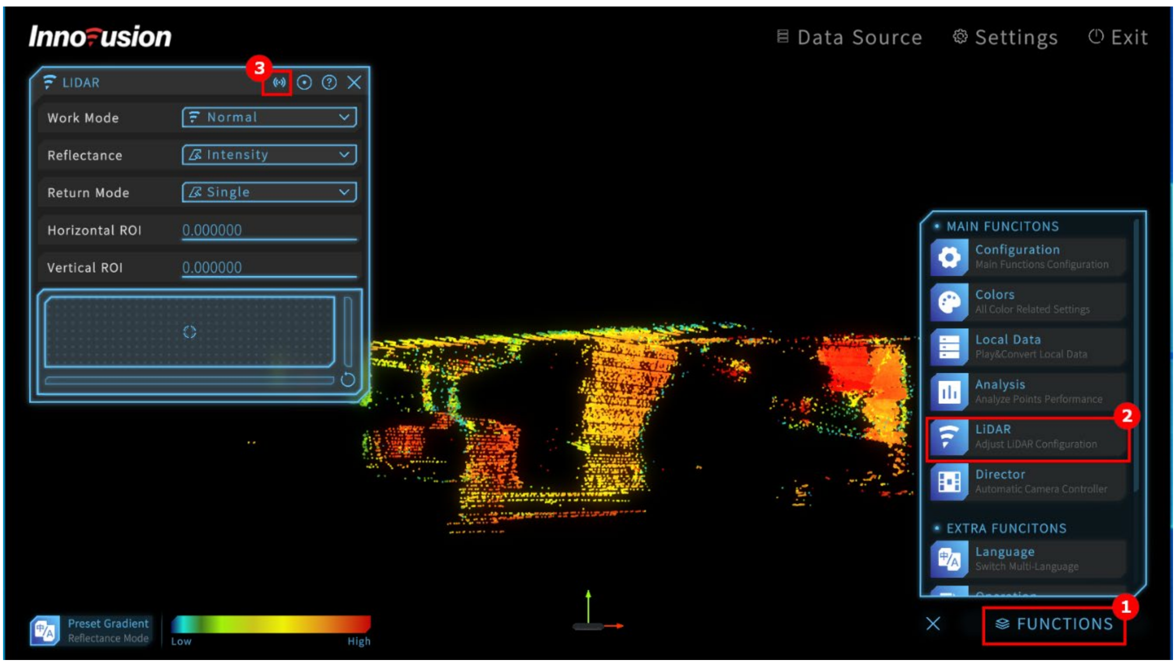
1. Go to **FUNCTIONS** > **LiDAR**.



2. Select the work mode of LiDAR. Normal and calibration work modes are available.

4.3.10 Configure ROI

1. Go to **FUNCTIONS** > **LiDAR**.



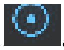
2. Enter the horizontal and vertical positions of the ROI center. ROI can also be adjusted in the

bottom FOV representative console.

The range of horizontal ROI is between -60 and 60. The range of vertical ROI is between -25 and 25. The unit is degree (°).

4.3.11 Record point cloud data

You can record LiDAR point cloud data in different formats.

1. Go to **FUNCTIONS** > **LiDAR**. Click .

2. Select the file format and size of the data to be recorded.

- Record a file in pcd format.

  Select **.pcd** in **Save Format**. The PCD format is ASCII based. Enter the size of the file in **Frame(s)**.

- Record a file in pcd_binary format.

  Select **.pcd_binary** in **Save Format**. Enter the size of the file in **Frame(s)**.

- Record a file in inno_pc format.

  Select **.inno_pc** in **Save Format**. Enter the size of the file in **Frame(s)**.

  inno_pc is the proprietary format of Innovusion point cloud files. The points in inno_pc files are in spherical coordinates.

- Record a file in inno_pc_xyz format.

  Select **.inno_pc_xyz** in **Save Format**. Enter the size of the file in **Frame(s)**.

  inno_pc_xyz is the proprietary format of Innovusion point cloud files. The points in inno_pc_xyz files are in Cartesian coordinates. inno_pc_xyz is also saved in ASCII form.

- Record a file in inno_raw format.

  Select **.inno_raw** in **Save Format**. Enter the size of the file in **MiB**.

  inno_raw is the proprietary format of Innovusion point cloud files.
- Record a file in inno_raw_raw format.

  Select **.inno_raw_raw** in **Save Format**. Enter the channel number in **Channel #**.
- Record a file in png format.

  Select **.png** in **Save Format**. Enter the number of pictures in **ImageCount**.
- Record a file in csv format.

  Select **.csv** in **Save Format**. Enter the size of the file in **Frame(s)**.
- Record a file in yaml format.

  Select **.yaml** in **Save Format**.

3. Click **Start Record** to record the file.

Note

- Point cloud data recording starts immediately by default.
- The limitation of the file size is subject to change based on the file format.

4.3.12 Obtain point cloud data

1. Go to **FUNCTIONS** > **Analysis**. Click .

2. Press **Z** key and draw a region to obtain all point cloud data information in this region.
3. Check the point cloud data in the **ANALYSIS**.

4. (Optional) Click to save the point cloud data information in a csv file.

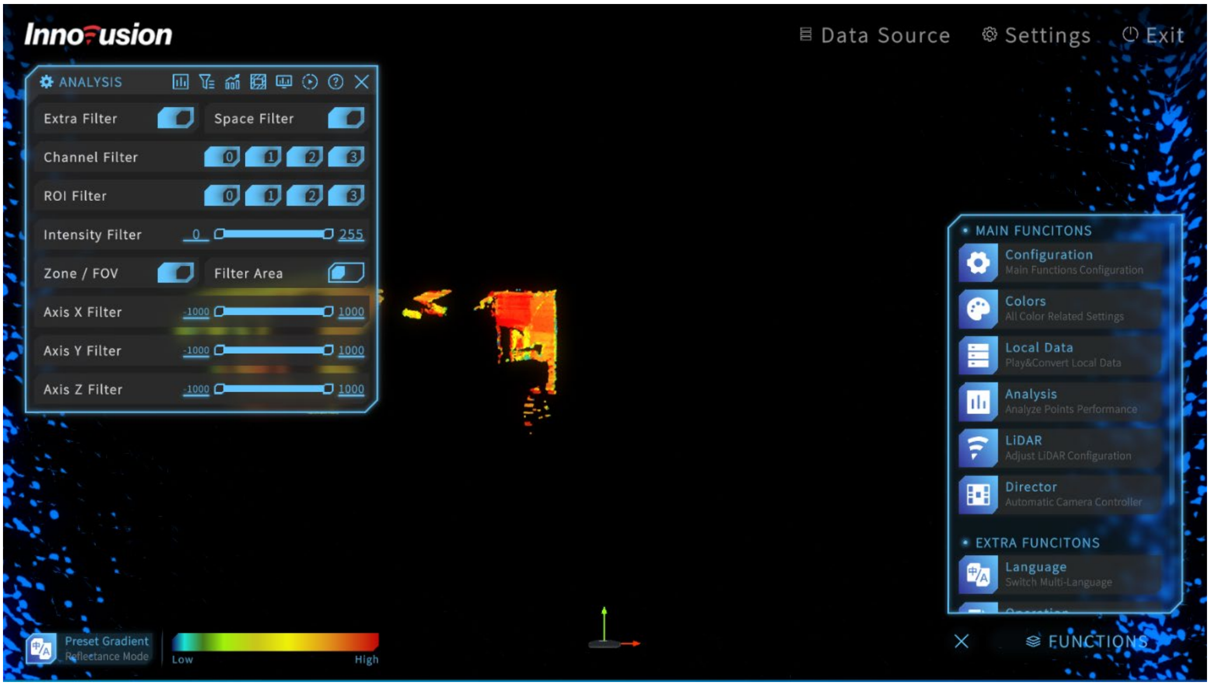
Note

You can select a maximum of 1000 points at a time.

4.3.13 Filter point cloud data

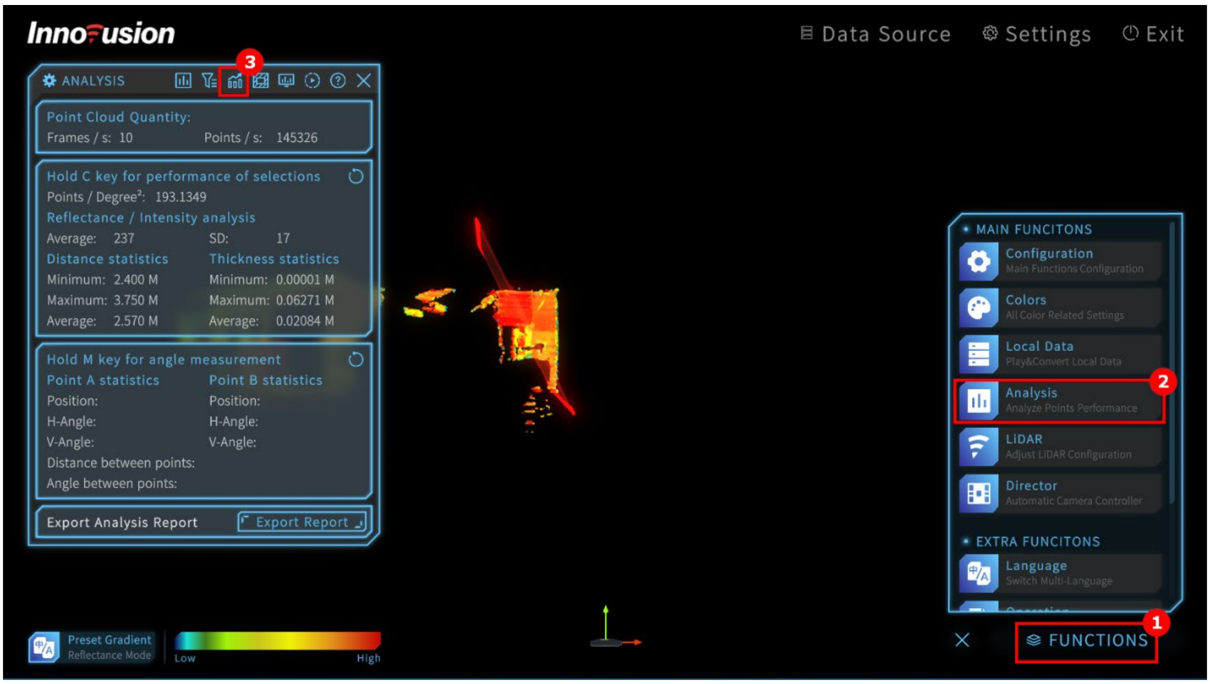
1. Go to **FUNCTIONS** > **Analysis**. Click .

2. Select the data to be filtered.



4.3.14 Analysis performance

1. Go to **FUNCTIONS** > **Analysis**. Click .



2. Select the point or region that needs to be analyzed.

- Test the performance of a region.

  Press **C** key and select the region to test its performance.

- Measure the angle and distance of points.

  Press **C** key and select two points to measure the distance between them and their angles from the origin.

3. (Optional) Click **Export Report** to export the analysis result.

4.3.15 Create a fitting plane area

1. Go to **FUNCTIONS** > **Analysis**. Click .

2. Press **C** key and select the region. The system will calculate the three fittest plane areas for the selected region, and the portion of final fit points, fitting noise points, and unrelated points.

3. (Optional) Click **RANSAC+OLS**, **RANSAC**, or **OLS** to switch the display of the corresponding fittest plane areas.

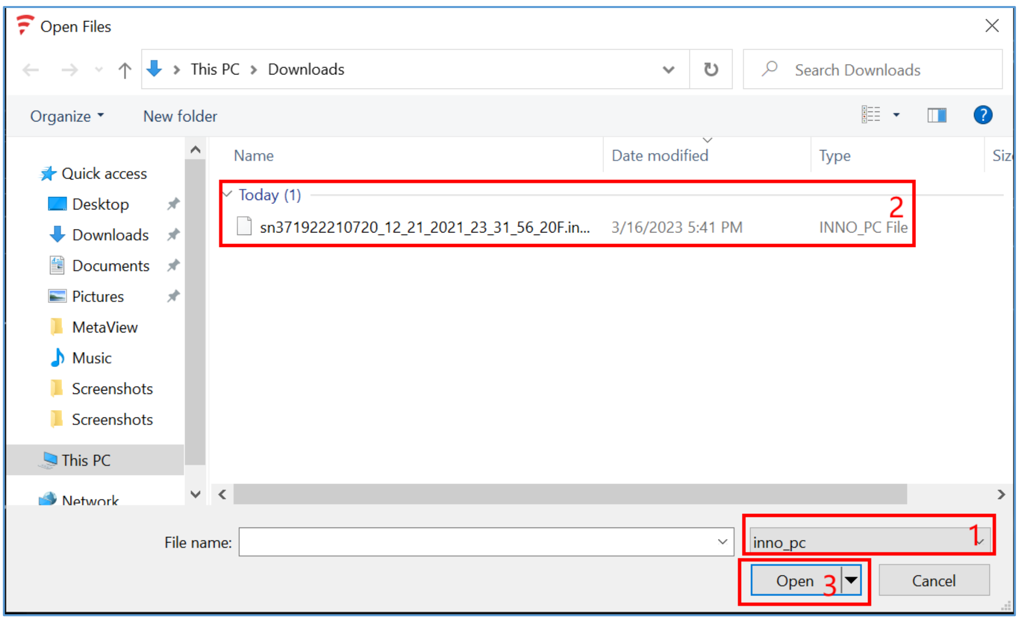
4.3.16 Data monitoring

1. Go to FUNCTIONS > **Analysis**. Click .
2. Check the data flow status in the window.

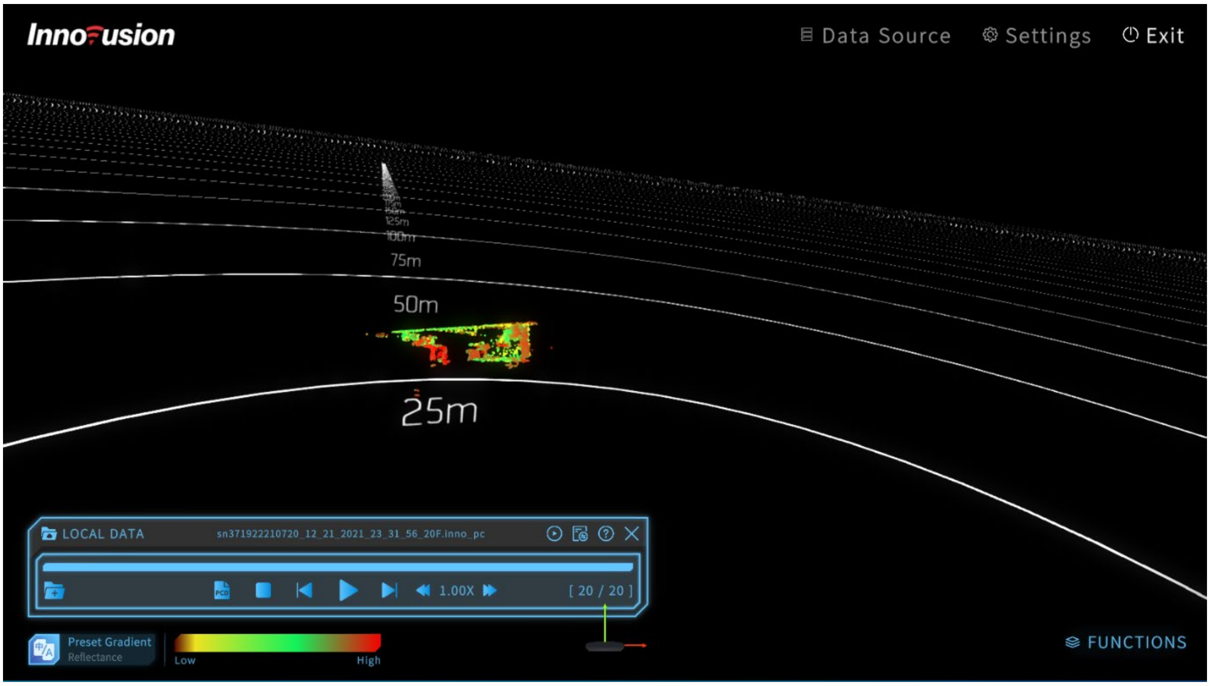
4.3.17 Replay point cloud data file

1. Go to **Data Source** > **Local Source**.
2. Click **Add New Local Data Source** to enter **Open Files**.

3. Select the file format and select the local data source. Click **OK**.

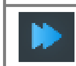


4. Click  to play the file.



5. (Optional) You can reduce the speed during replay.

Table12 Button Description

Button	Description
	Save the current frame as a single PCD file.
	Stop playing
	Previous frame
	Play/Pause
	Next frame
	Slower
	Faster
	Open the point cloud file

4.3.18 System configuration

4.3.18.1 Configure interface

1. Go to **Settings** > **Advanced** > **UI & LANGUAGE**.

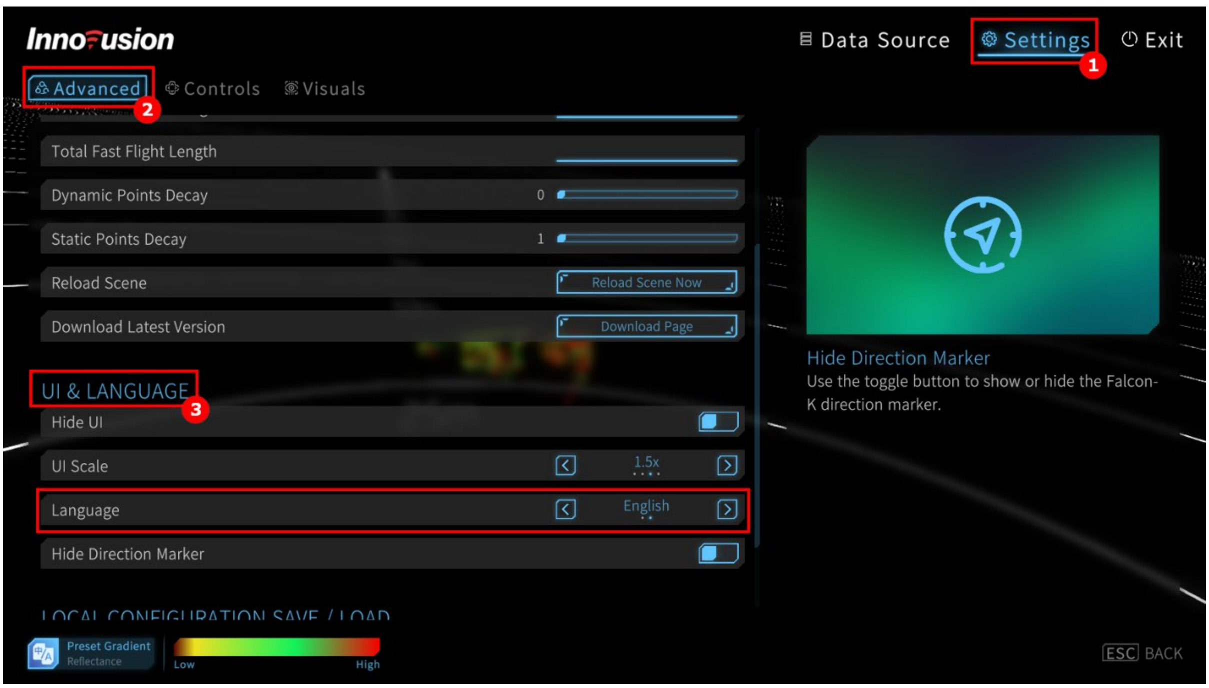
2. Change the interface scale according to the actual needs. The range is from 1 to 1.75.

4.3.18.2 Select language

There are three ways to select the language.

Method 1:

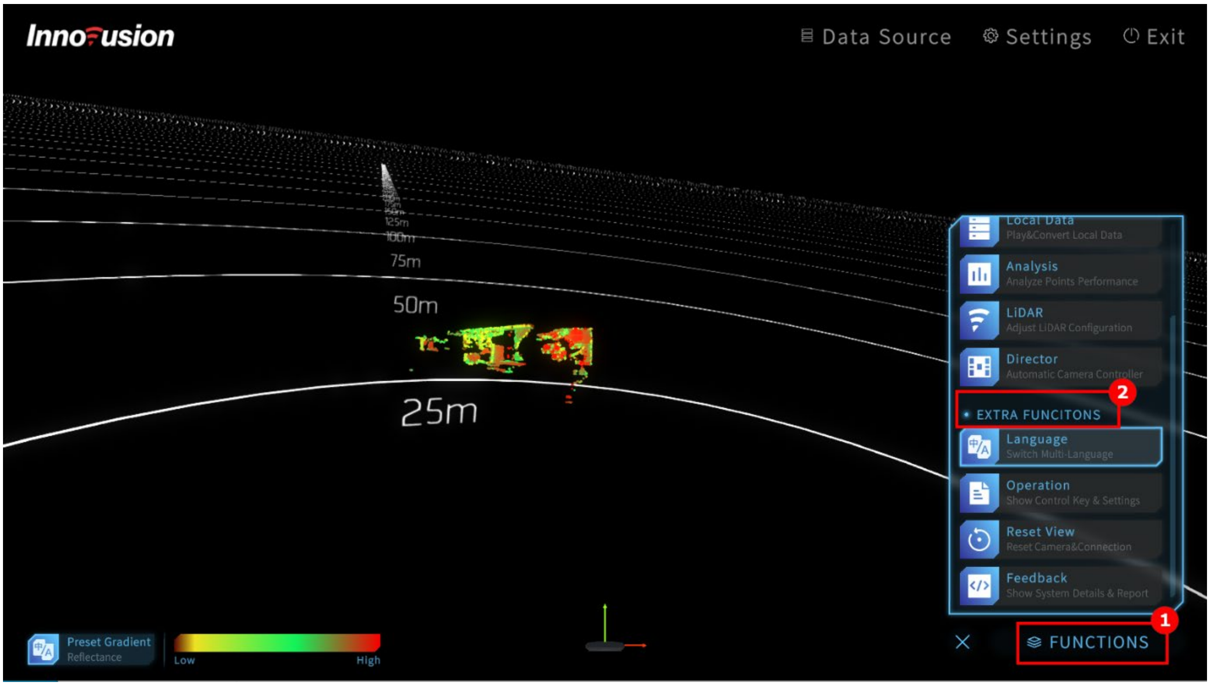
1. Go to **Settings** > **Advanced** > **UI & LANGUAGE**.



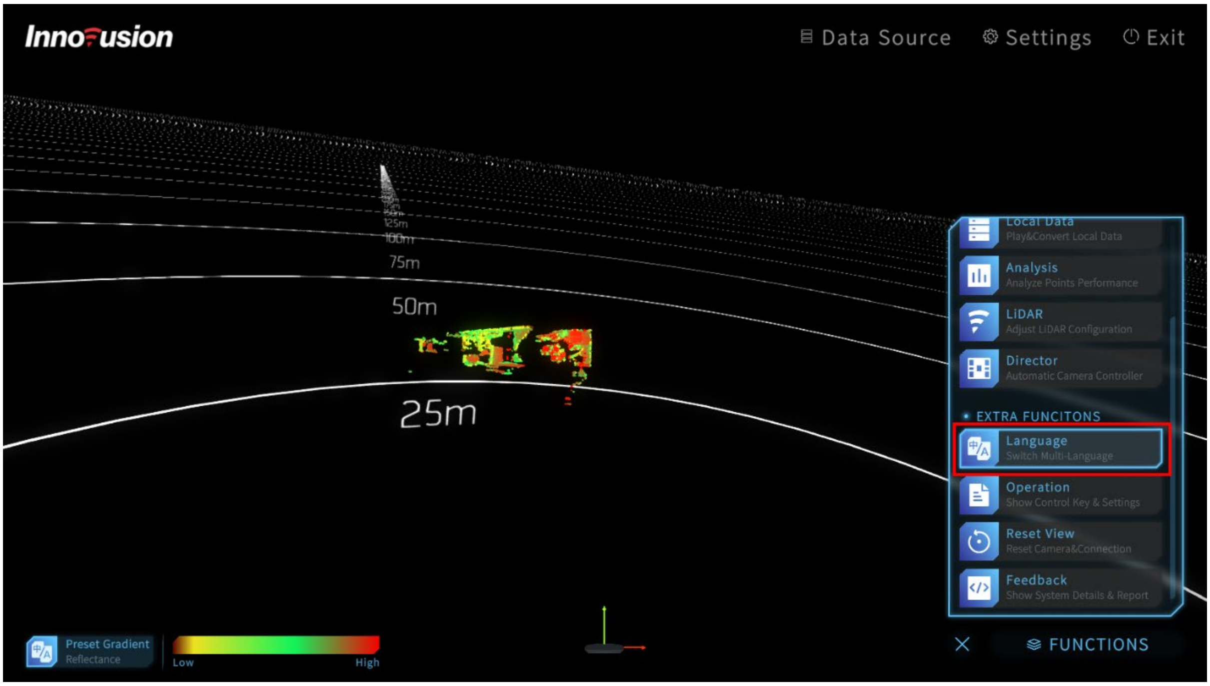
2. Select the language.

Method 2:

1. Go to **FUNCTIONS** > **EXTRA FUNCTIONS**.

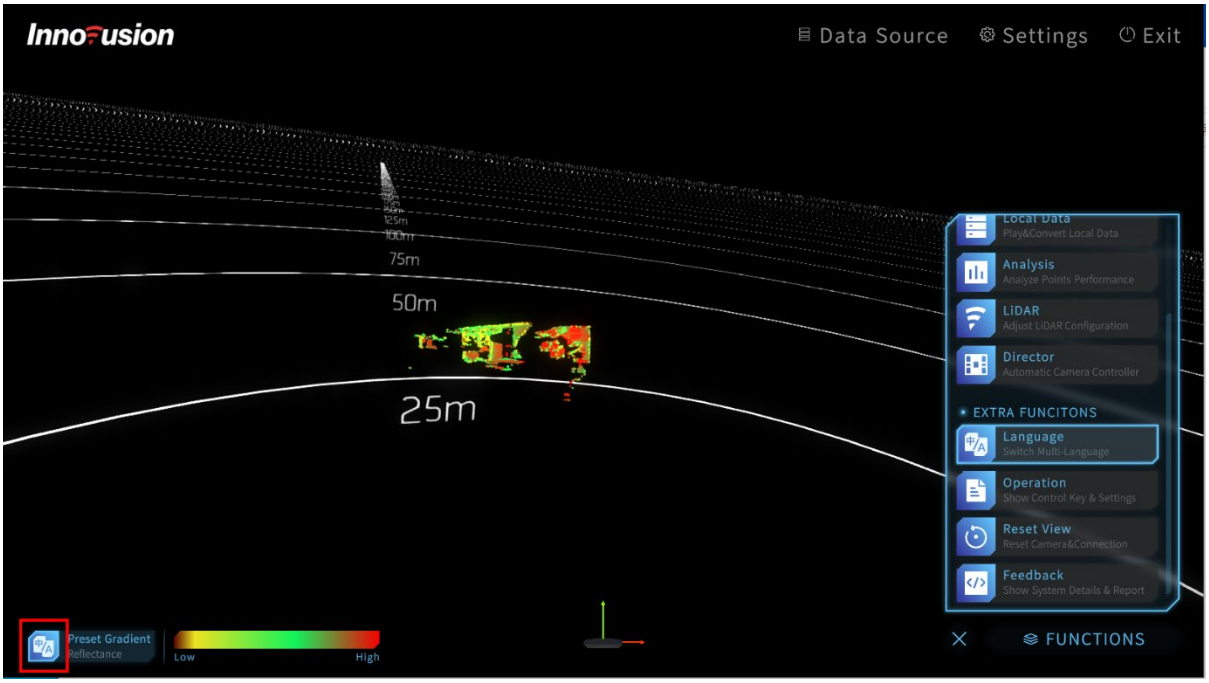


2. Click **Languages**.



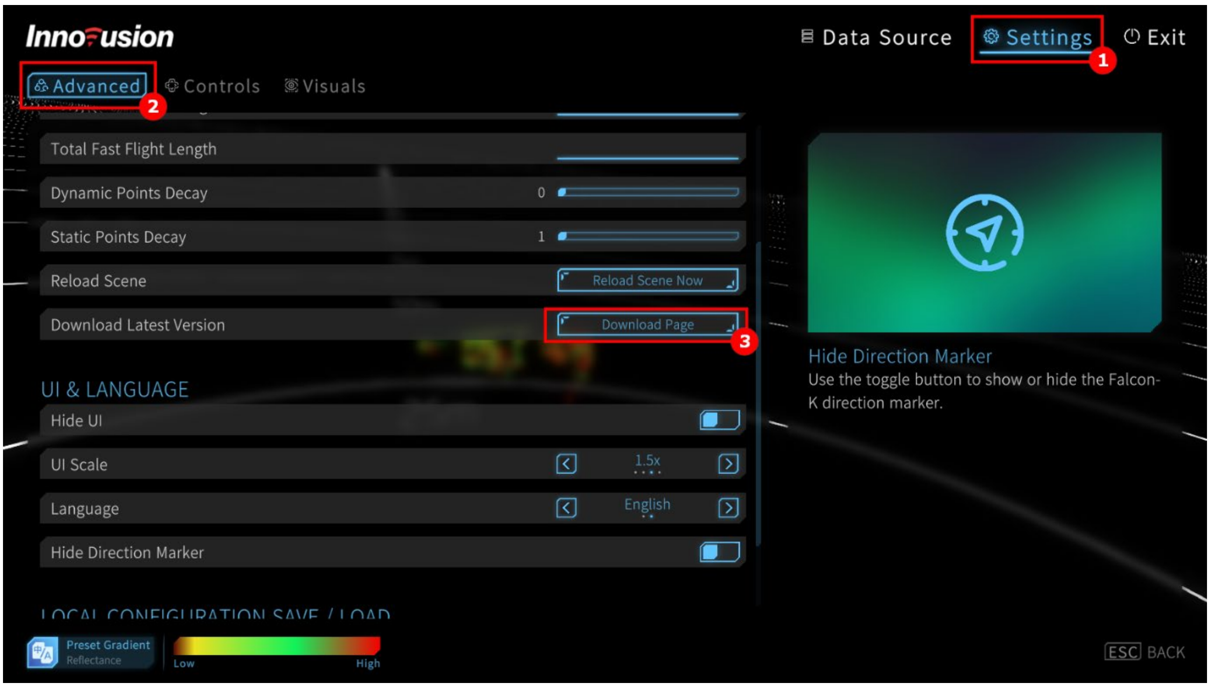
Method 3:

Click  to select the language.



4.3.18.3 Upgrade

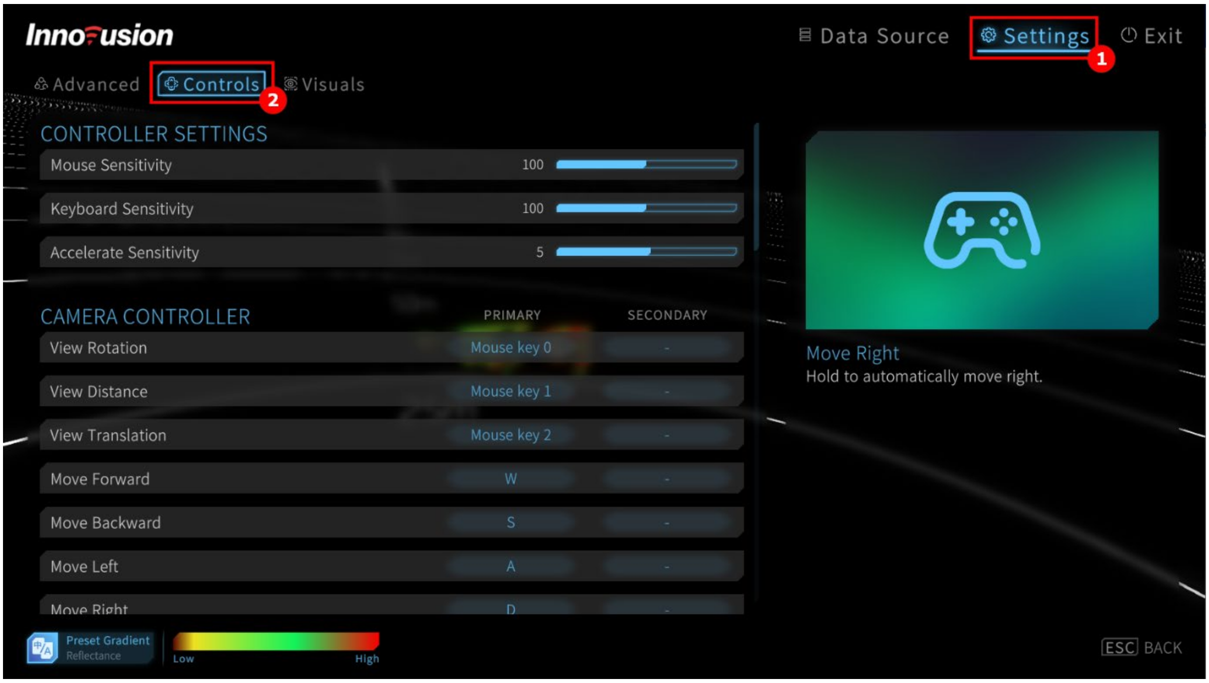
1. Go to **Settings** > **Advanced** > **GENERAL**. Click **Download Page**.



2. Download the latest version.

4.3.18.4 Control settings

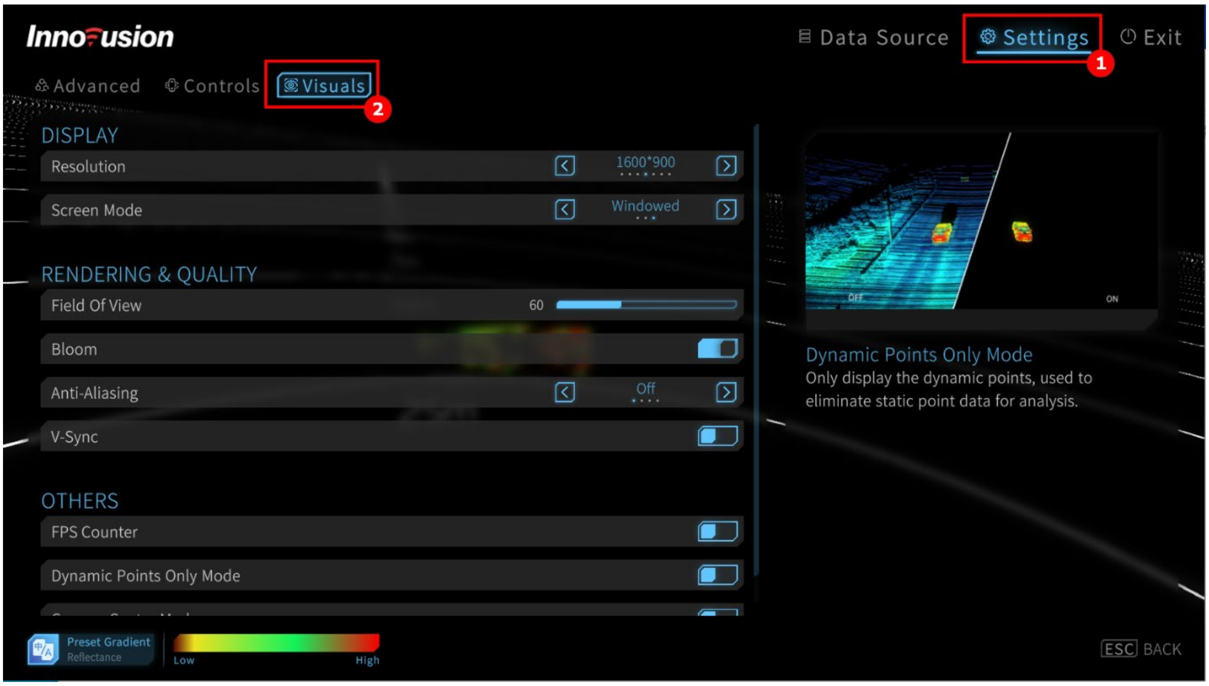
1. Go to **Settings** > **Controls**.



2. Set the controller sensitivity and the shortcut key according to the actual needs.

4.3.18.5 Set visual effect

1. Go to **Settings** > **Visuals**.

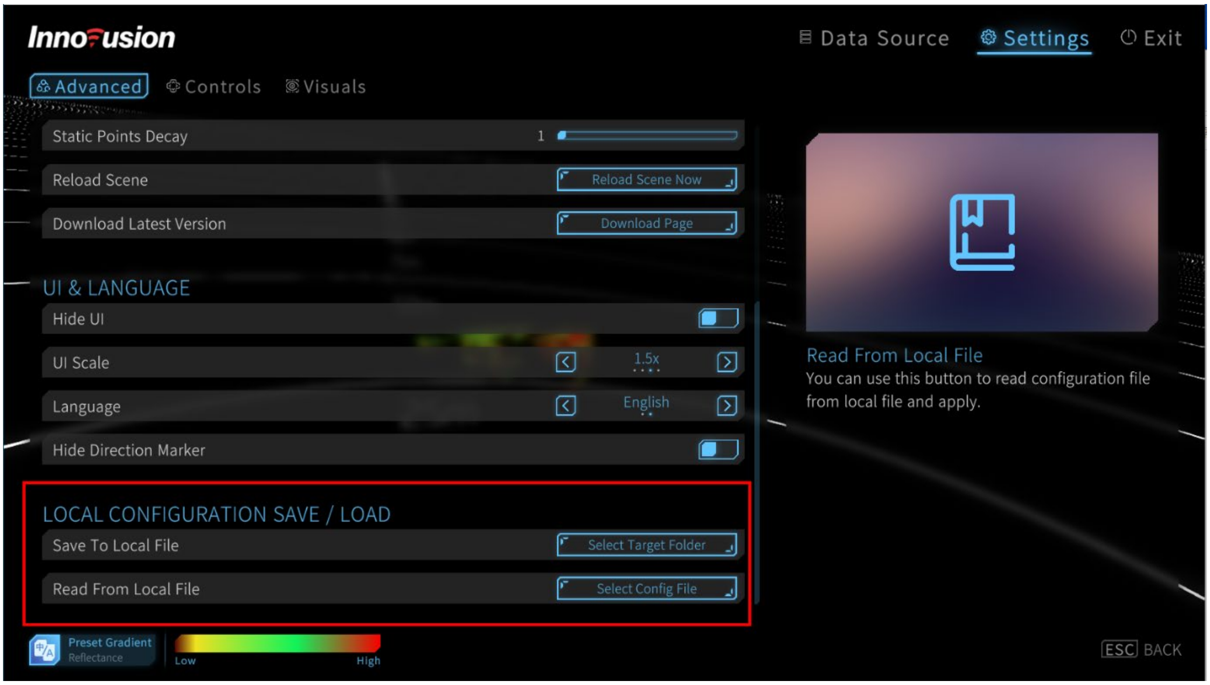


2. Configure resolution, FOV, and other parameters according to the actual needs.

4.3.18.6 Export a configuration file

You can export the configuration and save the file locally.

1. Go to **Settings** > **Advanced** > **LOCAL CONFIGURATION SAVE/LOAD**.

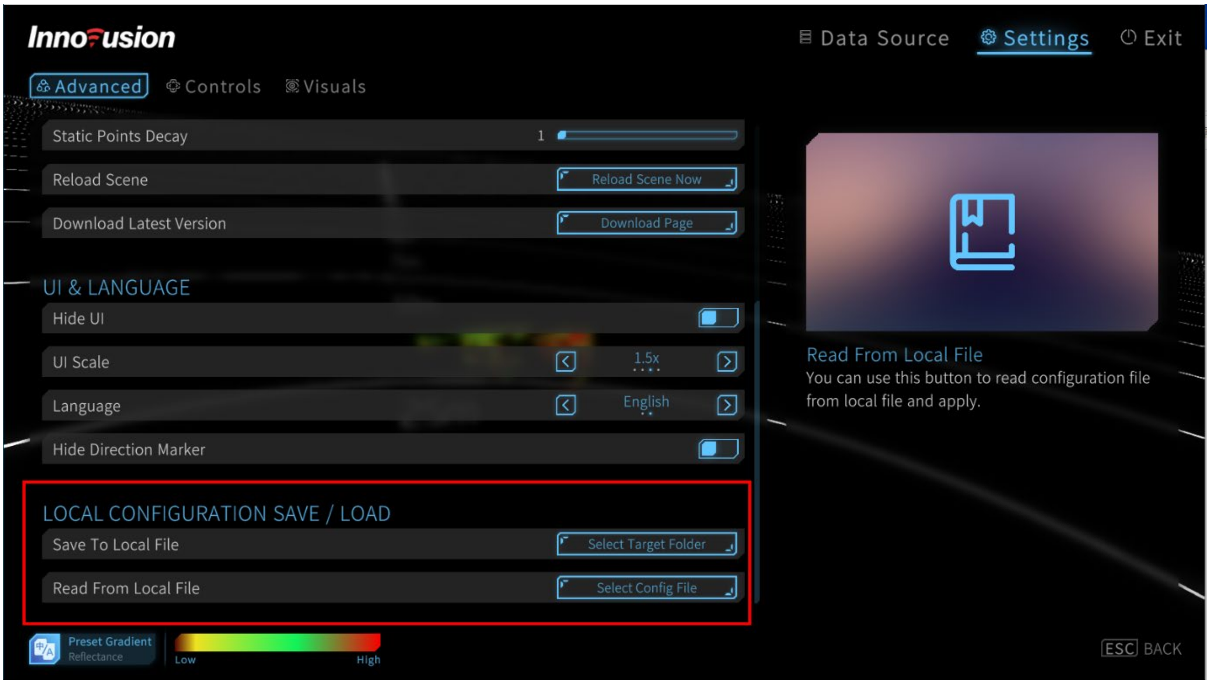


2. Click **Select Target Folder** to select the path of the configuration file.

4.3.18.7 Import a configuration file

Import the configuration to MetaView.

1. Go to **Settings** > **Advanced** > **LOCAL CONFIGURATION SAVE/LOAD**.



2. Click **Select Config File** and select the configuration file to be imported.

4.4 Operate in Docker

4.4.1 Start the LiDAR

1. Connect the power supply to start the LiDAR.
2. The LiDAR completes initialization and generates data after powering on for 11 to 18 seconds.

Note

The LiDAR does not have a power switch. It will become operational when power is applied.

4.4.2 View the LiDAR point cloud data

Note

For more information about Docker installation, please contact the innovusion staff.

1. Change the computer IP address to the same subnet with the LiDAR.

Note

- The default LiDAR IP address is 172.168.1.10.
- You can check the access to the LiDAR IP address via the ping command.

2. Enter the Docker path.
3. View the status of the LiDAR point cloud.

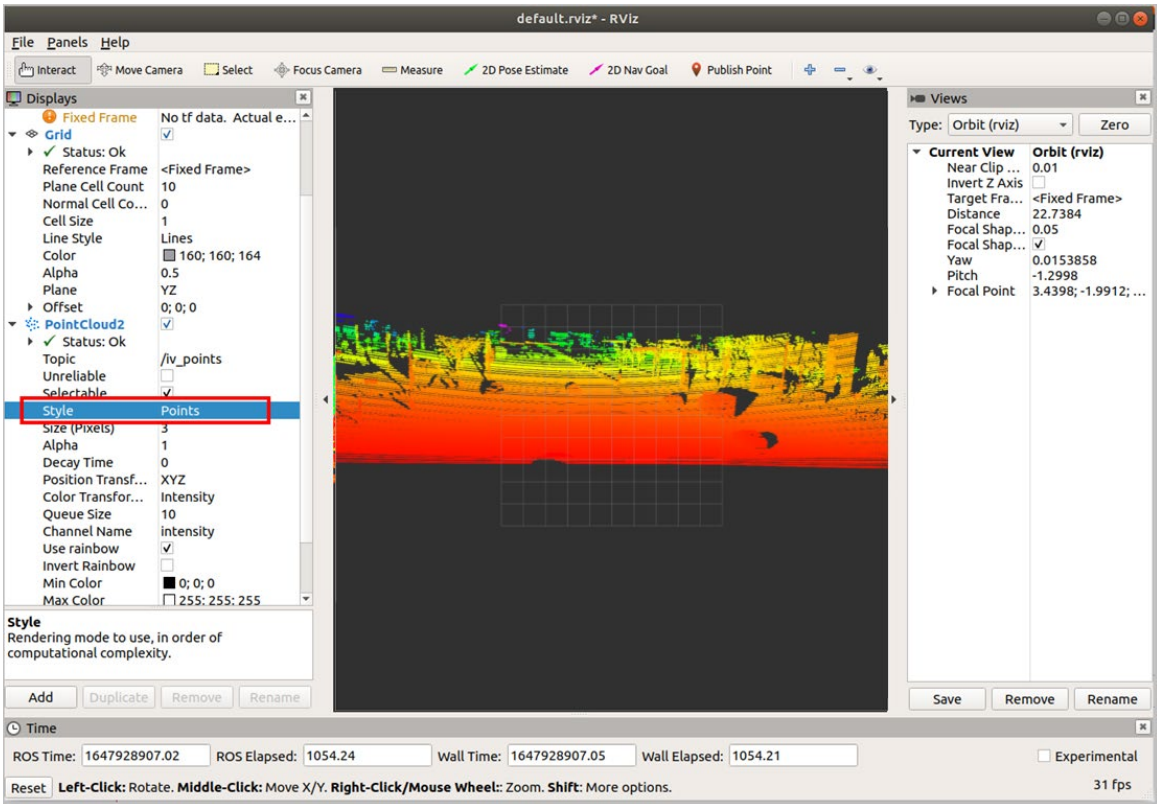
```
./launch-docker.py  --deb-file   <  package.deb>--lidar-ip  <INPUT_LIDAR_IP>   --ros-version
<ros version>
```

```
demo@demo-OMEN-by-HP-Laptop-16-b0xxx:~/install_docker/catkin_ws/src/rosprocessor/build/docker$ ./launch-docker.p
y --deb-file ros-kinetic-innovusion-driver-release-2.4.0-rc237-public.deb  --lidar-ip 172.168.1.10 --ros-version
 kinetic
Script is starting 2022May11-191440
docker launch command: xhost +local:root; nvidia-docker run -it --rm --label rosdocker  --env="DISPLAY" --env="Q
T_X11_NO_MITSHM=1" --env="IV_CWD=/home/demo/install_docker/catkin_ws/src/rosprocessor/build/docker"  -v=/home/de
mo/install_docker/catkin_ws/src/rosprocessor/build/docker/ros-kinetic-innovusion-driver-release-2.4.0-rc237-publ
ic.deb:/root/ros-kinetic-innovusion-driver-release-2.4.0-rc237-public.deb  -v=/home/demo/install_docker/catkin_w
s/src/rosprocessor/build/docker:/root/docker  -v=/home/demo/install_docker/catkin_ws/src/rosprocessor/build/dock
er:/home/demo/install_docker/catkin_ws/src/rosprocessor/build/docker -v="/home/demo/.rviz_kinetic:/root/.rviz" -
v="/home/demo/.ros_kinetic:/root/.ros"  -v=/home/demo/.innovusion:/root/.innovusion  -v=/home/demo/install_docke
r/catkin_ws/src/rosprocessor/build/docker/output:/root/output  --volume="/tmp/.X11-unix:/tmp/.X11-unix:rw" ivusw
/ros-driver-test-public:ubuntu1604-kinetic-jsk-ceres bash -c "(echo start roscore; rm -fr /root/.ros/log/; rosco
re; echo roscore done) & (echo start rviz; until rostopic list 2>/dev/null; do sleep 0.2; done; rosrun rviz rviz
 -f innovusion; echo rviz done) & dpkg -i "/root/ros-kinetic-innovusion-driver-release-2.4.0-rc237-public.deb" &
& echo dpkg install done && until rostopic list 2>/dev/null; do sleep 0.2; done; find /root/docker/cali_manager
-user root -exec chmod guo+rw {} \; && roslaunch innovusion_pointcloud innovusion_points.launch  device_ip:=172.
168.1.10  lidarhome:=//root/.innovusion "
=====
inside docker command: (echo start roscore; rm -fr /root/.ros/log/; roscore; echo roscore done) & (echo start rv
iz; until rostopic list 2>/dev/null; do sleep 0.2; done; rosrun rviz rviz -f innovusion; echo rviz done) & dpkg
-i "/root/ros-kinetic-innovusion-driver-release-2.4.0-rc237-public.deb" && echo dpkg install done && until rosto
pic list 2>/dev/null; do sleep 0.2; done; find /root/docker/cali_manager -user root -exec chmod guo+rw {} \; &&
roslaunch innovusion_pointcloud innovusion_points.launch  device_ip:=172.168.1.10  lidarhome:=//root/.innovusion

=====
'main' command: roslaunch innovusion_pointcloud innovusion_points.launch  device_ip:=172.168.1.10  lidarhome:=//
```

Note

- <package.deb> is the driver's name. Please get the latest driver based on the actual conditions of the computer. If the existing driver does not match the computer's version, please contact Innovusion staff.
- The default LiDAR IP address is 172.168.1.10.
- <ros version> is the version of the ROS environment.

4. **rviz** is started to show the current point cloud status. For a better display effect in rviz, refer to [4.2.3 View LiDAR point cloud data](#).



4.4.3 Replay LiDAR point cloud data

The point cloud data in inno_raw format can be replayed by Docker.

1. Enter the Docker path. View the status of the LiDAR point cloud.

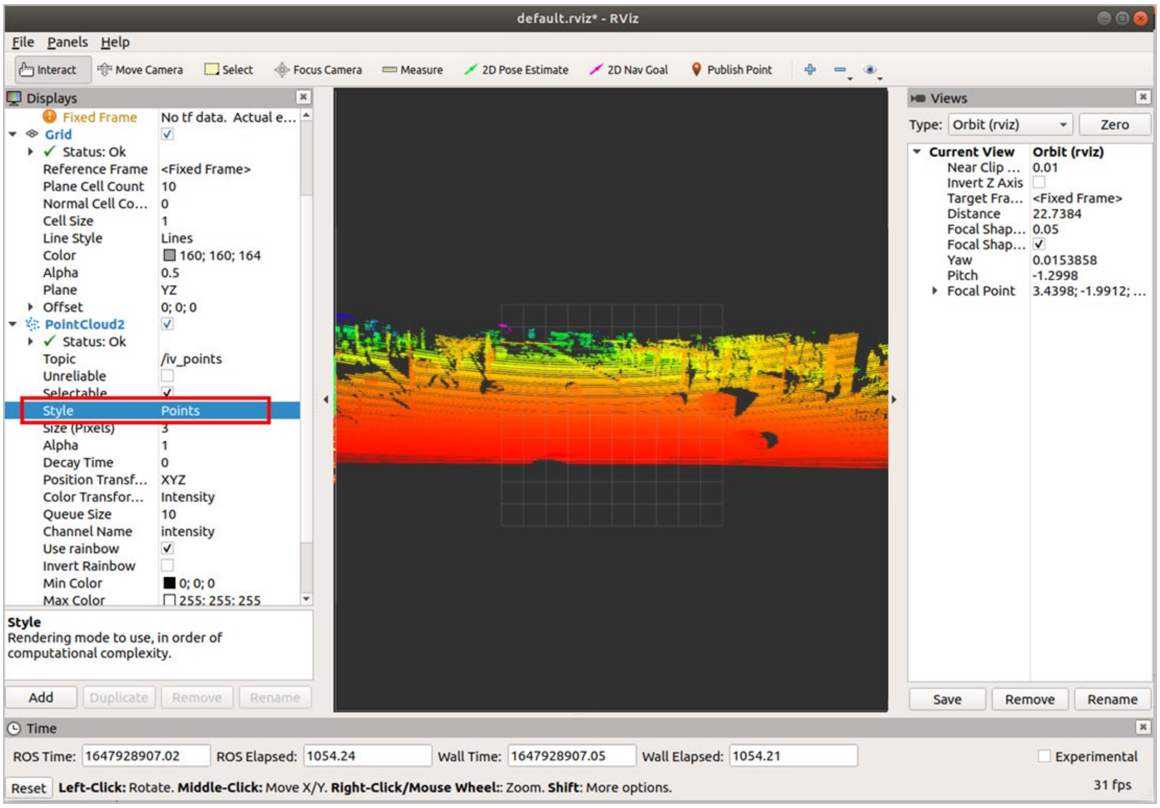
```
./launch-docker.py    --deb-file  <  package.deb>    --yaml  <file  name.yaml>    --pcap
<filename.inno_raw>   --ros-version <ros version>
```

```
demo@demo-OMEN-by-HP-Laptop-16-b0xxx:~/install_docker/catkin_ws/src/rosprocessor/build/docker$ ./launch-docker.py --deb-file ros-kinetic-innovusion-driver
-release-2.4.0-rc237-public.deb  --yaml FI0999_download.yaml --pcap 1_13_sn363222100999_01_20_2022_06_43_12.inno_raw --ros-version kinetic
Script is starting 2022May11-191631
use public yaml file FI0999_download.yaml
docker launch command: xhost +local:root; nvidia-docker run -it --rm --label rosdocker  --env="DISPLAY" --env="QT_X11_NO_MITSHM=1" --env="IV_CWD=/home/dem
o/install_docker/catkin_ws/src/rosprocessor/build/docker"  -v=/home/demo/install_docker/catkin_ws/src/rosprocessor/build/docker/ros-kinetic-innovusion-dri
ver-release-2.4.0-rc237-public.deb:/root/ros-kinetic-innovusion-driver-release-2.4.0-rc237-public.deb  -v=/home/demo/install_docker/catkin_ws/src/rosproce
ssor/build/docker:/root/docker  -v=/home/demo/install_docker/catkin_ws/src/rosprocessor/build/docker:/home/demo/install_docker/catkin_ws/src/rosprocessor/
build/docker  -v=/home/demo/install_docker/catkin_ws/src/rosprocessor/build/docker/FI0999_download.yaml:/root/FI0999_download.yaml -v="/home/demo/.rviz_ki
netic:/root/.rviz" -v="/home/demo/.ros_kinetic:/root/.ros"  -v=/home/demo/.innovusion:/root/.innovusion  -v=/home/demo/install_docker/catkin_ws/src/rospro
cessor/build/docker/output:/root/output  -v=/home/demo/install_docker/catkin_ws/src/rosprocessor/build/docker/1_13_sn363222100999_01_20_2022_06_43_12.inno
_raw:/root/1_13_sn363222100999_01_20_2022_06_43_12.inno_raw  --volume="/tmp/.X11-unix:/tmp/.X11-unix:rw" ivusw/ros-driver-test-public:ubuntu1604-kinetic-j
sk-ceres bash -c "(echo start roscore; rm -fr /root/.ros/log/; roscore; echo roscore done) & (echo start rviz; until rostopic list 2>/dev/null; do sleep 0
.2; done; rosrun rviz rviz -f innovusion; echo rviz done) & dpkg -i "/root/ros-kinetic-innovusion-driver-release-2.4.0-rc237-public.deb" && echo dpkg inst
all done && until rostopic list 2>/dev/null; do sleep 0.2; done; find /root/docker/cali_manager -user root -exec chmod guo+rw {} \; && roslaunch innovusio
n_pointcloud innovusion_points.launch  calibration:=/root/FI0999_download.yaml  device_ip:=172.168.1.10  lidarhome:=//root/.innovusion  pcap:=/root/1_13_s
n363222100999_01_20_2022_06_43_12.inno_raw  read_once:=0 "
=====
inside docker command: (echo start roscore; rm -fr /root/.ros/log/; roscore; echo roscore done) & (echo start rviz; until rostopic list 2>/dev/null; do sl
eep 0.2; done; rosrun rviz rviz -f innovusion; echo rviz done) & dpkg -i "/root/ros-kinetic-innovusion-driver-release-2.4.0-rc237-public.deb" && echo dpkg
 install done && until rostopic list 2>/dev/null; do sleep 0.2; done; find /root/docker/cali_manager -user root -exec chmod guo+rw {} \; && roslaunch inno
vusion_pointcloud innovusion_points.launch  calibration:=/root/FI0999_download.yaml  device_ip:=172.168.1.10  lidarhome:=//root/.innovusion  pcap:=/root/1
_13_sn363222100999_01_20_2022_06_43_12.inno_raw  read_once:=0
=====
'main' command: roslaunch innovusion_pointcloud innovusion_points.launch  calibration:=/root/FI0999_download.yaml  device_ip:=172.168.1.10  lidarhome:=//r
oot/.innovusion  pcap:=/root/1_13_sn363222100999_01_20_2022_06_43_12.inno_raw  read_once:=0
non-network local connections being added to access control list
start roscore
start rviz
Selecting previously unselected package ros-kinetic-innovusion-driver-public.
(Reading database ... 88654 files and directories currently installed.)
Preparing to unpack .../ros-kinetic-innovusion-driver-release-2.4.0-rc237-public.deb ...
Unpacking ros-kinetic-innovusion-driver-public (2.4.0-rc237-public) ...
... logging to /root/.ros/log/cfab6116-d11b-11ec-8360-0242ac110003/roslaunch-aa2c9dc92a1f-36.log
Checking log directory for disk usage. This may take awhile.
Press Ctrl-C to interrupt
Done checking log file disk usage. Usage is <1GB.

Setting up ros-kinetic-innovusion-driver-public (2.4.0-rc237-public) ...
started roslaunch server http://aa2c9dc92a1f:37743/
```

Note

- <package.deb> is the driver's name. Please get the latest driver based on the actual situation of the computer. If the existing driver does not match the computer's version, please contact Innovusion staff.
- <filename.yaml> is the configuration file. You can download this file using ILA.
- <file name.inno_raw> is LiDAR point cloud data in .inno_raw.
- <ros version> is the version of the ROS environment.

2. **rviz** is started to show the current point cloud status. For a better display effect in rviz, refer to 4.2.3 View LiDAR point cloud data.



4.4.4 (Optional) Convert a file in inno_raw format to a file in bag format

1. Enter the **Docker** path.
2. **Convert** a file in inno_raw format to a file in bag format.

```
./launch-docker.py --deb-file < package.deb> --yaml <file name.yaml> --pcap <file
name.inno_raw> --ros-version <ros version> –-record-bag-file
```

Note

- <package.deb> is the driver's name. Please get the latest driver based on the actual situation of the computer. If the existing driver does not match the computer's version, please contact Innovusion staff.
- <filename.yaml> is the configuration file. You can download this file using ILA.
- <file name.inno_raw> is LiDAR point cloud data in .inno_raw.
- <ros version> is the version of the ROS environment.

4.4.5 Shut down the LiDAR

Disconnect the power supply to shut down the LiDAR.

5 SDK configuration

You can operate the LiDAR using the following executable files. For more information, please contact Innovusion staff to obtain related manuals.

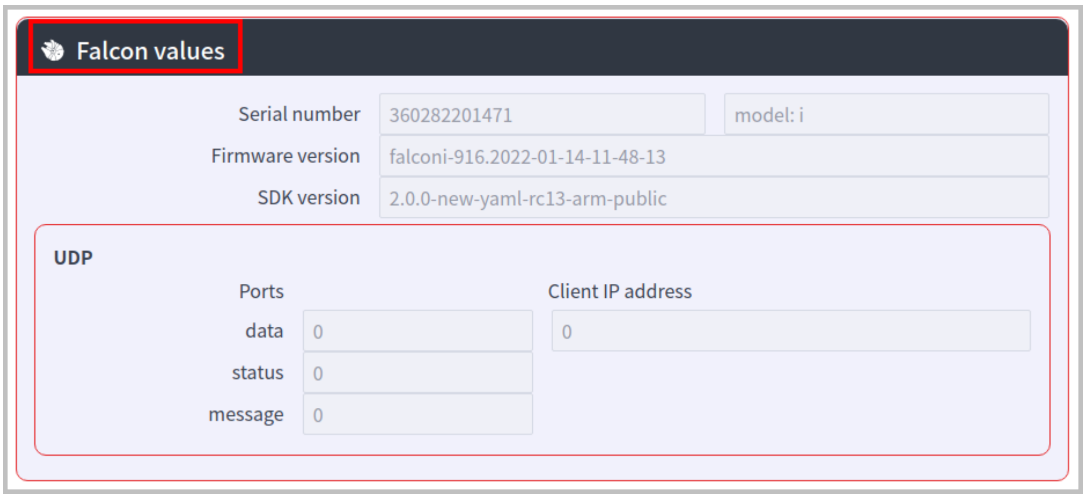
Table 13 executable file instructions

NO.	Software Interface	Description
1	get_pcd	This lightweight executable file allows the user to record data files and convert data between different formats. It is very useful for scripted/automated data collection.
2	Innovusion_lidar_util	The command line utility enables you to obtain and change the configuration of the LiDAR. You can use the commands to get the status of LiDAR and check the firmware logs.
3	SDK	The SDK offers the most flexibility for customization regarding data streaming and lidar control.
4	cURL commands	cURL commands provide a convenient way to adjust ROI location, download system logs, and start/stop the pointcloud server (PCS).

5.1 SDK download and installation

5.1.1 Download

Download the SDK package corresponding to the LiDAR version. You can log into ILA and select **Sensor Config** > **Falcon values** to check the SDK version information. For LiDAR upgrade, please refer to Appendix C Upgrade the LiDAR.



The download path of the SDK package is as follows.

https://innovusioncn.atlassian.net/wiki/spaces/STC/pages/1645970905/Falcon+K+FW+Release

2022.04.07 Release-2.5.0-rc263

inno*-public.tgz	ros*-public.deb
inno-lidar-sdk-release-2.5.0-rc263-arm-public.tgz	ros-kinetic-innovusion-driver-release-2.5.0-rc263-public.deb
inno-lidar-sdk-release-2.5.0-rc263-public.tgz	ros-kinetic-innovusion-driver-release-2.5.0-rc263-arm-public.deb
inno-lidar-sdk-release-2.5.0-rc263-mingw64-public.tgz	ros-melodic-innovusion-driver-release-2.5.0-rc263-public.deb
inno-lidar-embedded-release-2.5.0-rc263-arm-public.tgz	ros-melodic-innovusion-driver-release-2.5.0-rc263-arm-public.deb
inno-lidar-embedded-release-2.5.0-rc263-public.tgz	

5.1.2 Installation

Table 14 Installation environment requirements

Installation environment requirements	Architecture
Ubuntu 16.04 + GCC 5.4.0 and higher	x86/ARM
Ubuntu 18.04 + GCC 7.4.0 and higher	x86/ARM

5.2 Data format description

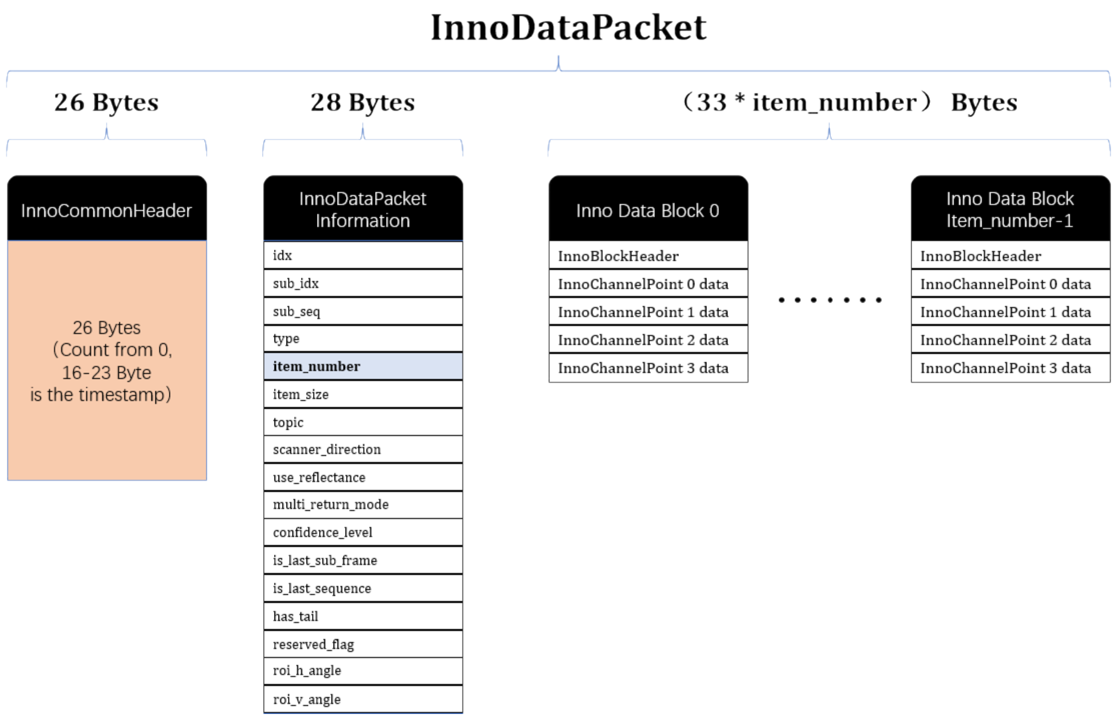
This chapter is mainly about the structures and variable definitions of the InnoDataPacket and InnoStatusPacket. Each frame of point cloud data is divided into multiple InnoDataPacket packets in accordance with the UDP packet size. An InnoStatusPacket is sent every 50 ms.

Each InnoDataPacket comprises InnoCommonHeader, InnoDataPacket Information, and Inno Data Blocks. Each InnoStatusPacket is composed of InnoCommonHeader and InnoStatusPacket Information. All data packets are in little-endian mode.

Note

If you want to change UDP packet size, please contact Innovusion staff.

5.2.1 InnoDataPacket



5.2.1.1 InnoCommonHeader

InnoCommonHeader: 26 Bytes.

Table 15 InnoCommonHeader Specification

Variable	Data type	Size (Bit)	Description
version	InnoCommonVersion	48	Data packet version information. For details, refer to 5.2.1.2 InnoCommonVersion.
checksum	uint32_t	32	Checksum, used for CRC checking.
size	uint32_t	32	Size of InnoDataPacket.
source_id	uint16_t	4	Identify each LiDAR when there are multiple LiDARs. The default value is 0 when there is only one LiDAR.
timestamp_sync_type	uint16_t	4	Time synchronization type.
reserved	uint16_t	8	Reserved field. Currently not used.
ts_start_us	InnoTimestampUs	64	The trigger time of the laser pulse in the 0th Block of InnoDataPacket.
lidar_mode	uint8_t	8	LiDAR working mode. lidar_mode=1: SLEEP lidar_mode=2: STANDBY lidar_mode=3: NORMAL

			lidar_mode=4: SHORT_RANGE (currently not used) lidar_mode=5: CALIBRATION lidar_mode=6: PROTECTION lidar_mode=7: WORK_QUIET
lidar_status	uint8_t	8	LiDAR status. lidar_status=0: Initial status lidar_status=1: TRANSITION lidar_status=2: NORMAL lidar_status=3: FAILED

5.2.1.2 InnoCommonVersion

InnoCommonVersion: 6 Bytes

Table 16 InnoCommonVersion Specification

Variable	Data type	Size (Bit)	Description
magic_number	uint16_t	16	Magic number, used for data verification.
major_version	uint8_t	8	Major version number.
minor_version	uint8_t	8	Minor version number.
fw_sequence	uint16_t	16	Reserved.

5.2.1.3 InnoDataPacket Information

InnoDataPacket Information: 28 Bytes

Table 17 InnoDataPacket Information Specification

Variable	Data type	Size (Bit)	Description
idx	uint64_t	64	Frame number, starting from 0.
sub_idx	uint16_t	16	Each frame is divided into multiple InnoDataPackets, and sub_ Idx is the sequence number of InnoDataPacket in a frame.
sub_seq	uint16_t	16	Currently not used. The default value is 0.
type	uint32_t	8	Data type in InnoDataPacket. type=1: Spherical coordinate point cloud type=2: message type=3: message log type=4: XYZ coordinate point cloud
item_number	uint32_t	24	The number of Blocks in the current InnoDataPacket.
item_size	uint16_t	16	Size of one single Block.
topic	uint32_t	32	Currently not used. The default value is 0.

scanner_direction	uint16_t	1	Laser scanning direction scanner_direction=0: from top to bottom scanner_direction=1: from bottom to top
use_reflectance	uint16_t	1	Use_reflectance=0: intensity mode Use_reflectance=1: reflectivity mode
multi_return_mode	uint16_t	3	multi_return_mode=1: single multi_return_mode=2: strongest + second strongest multi_return_mode=3: strongest + furthest
confidence_level	uint16_t	2	Confidence level of point cloud data. confidence_level=0: lowest confidence level. confidence_level=1: low confidence level confidence_level=2: high confidence level confidence_level=3: highest confidence level
is_last_sub_frame	uint16_t	1	Whether it is the last InnoDataPacket in a frame. is_last_sub_frame=0: No is_last_sub_frame=1: Yes
is_last_sequence	uint16_t	1	Currently not used. The default value is 1.
has_tail	uint16_t	1	Currently not used. The default value is 0.
reserved_flag	uint16_t	6	Currently not used. The default value is 0.
roi_h_angle	int16_t	16	Horizontal angle of the ROI center.
roi_v_angle	int16_t	16	Vertical angle of the ROI center.

5.2.1.4 InnoBlockHeader

InnoBlockHeader: 17 Bytes

Table 18 InnoBlockHeader Specification

Variable	Data type	Size (Bit)	Description
h_angle	int16_t	16	Horizontal angle of channel 0. The range is -π to π.
v_angle	int16_t	16	Vertical angle of channel 0. The range is -π to π.
ts_10us	uint16_t	16	Time offset of this Block relative to the 0th Block in the same InnoDataPacket. The unit is 10 µs.
scan_idx	uint16_t	16	Block ID of the scanning line.
scan_id	uint16_t	9	ID of the scanning line.
h_angle_diff_1	int64_t	9	Horizontal angle offset of channel 1 relative

			to channel 0.
h_angle_diff_2	int64_t	10	Horizontal angle offset of channel 2 relative to channel 0.
h_angle_diff_3	int64_t	11	Horizontal angle offset of channel 3 relative to channel 0.
v_angle_diff_1	int64_t	8	Vertical angle offset of channel 1 relative to channel 0.
v_angle_diff_2	int64_t	9	Vertical angle offset of channel 2 relative to channel 0.
v_angle_diff_3	int64_t	9	Vertical angle offset of channel 3 relative to channel 0.
in_roi	uint64_t	2	in_roi=0x00: in spare region. in_roi=0x11: in center ROI.
facet	uint64_t	3	Polygon facet ID of the current Block (five facets in total, numbered from 0 to 4).
reserved_flags	uint64_t	2	Currently not used. The default value is 0.

5.2.1.5 InnoChannelPoint

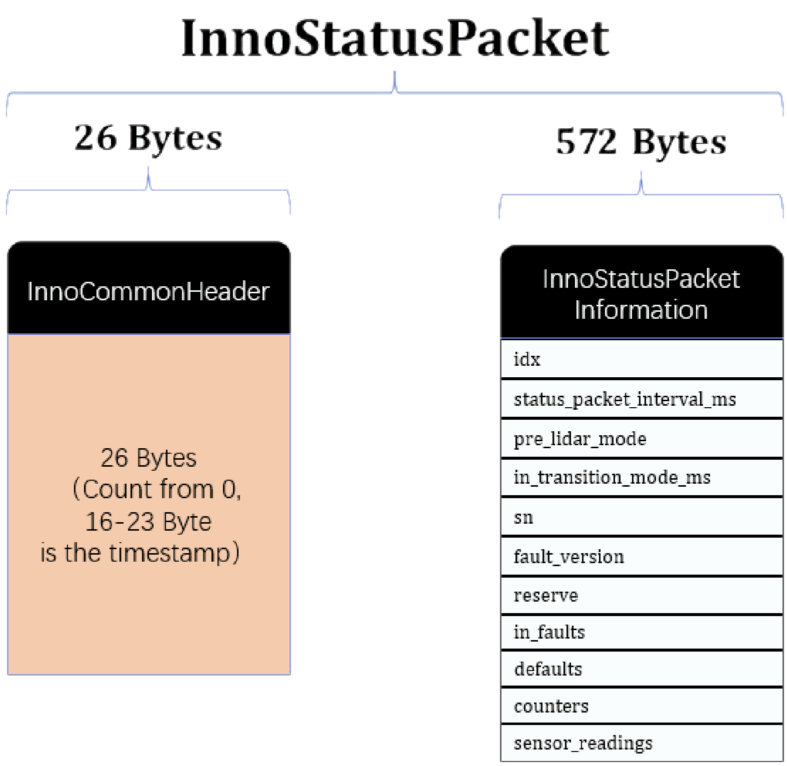
InnoChannelPoint: 4 Bytes

Table 19 InnoChannelPoint Specification

Variable	Data type	Size (Bit)	Description
radius	uint32_t	17	Distance between points. The value range is 0 to 655.35 m. The unit is 1/200 m.
refl	uint32_t	8	Reflectivity or intensity. The value range is 1 to 255.
is_2nd_return	unit32_t	1	is_2nd_return=0: single return mode. is_2nd_return=1: dual return mode.
type	uint32_t	2	type=0: normal. type=1: ground. type=2: fog.
elongation	uint32_t	4	Pulse width corresponds to the echoes.

5.2.2 InnoStatusPacket

This chapter mainly describes the InnoStatusPacket packet definition. The details of each variable in the packet are explained as follows.



5.2.2.1 InnoCommonHeader

InnoCommonHeader: 26 Bytes

Table 20 InnoCommonHeader Specification

Variable	Data type	Size (Bit)	Description
version	InnoCommonVersion	48	Data packet version information. For details, refer to 5.2.1.2 InnoCommonVersion.
checksum	uint32_t	32	Checksum, used for CRC checking.
size	uint32_t	32	Size of InnoDataPacket.
source_id	uint16_t	4	Identify each LiDAR when there are multiple LiDARs. The default value is 0 when there is only one LiDAR.
timestamp_sync_type	uint16_t	4	Time synchronization type.
reserved	uint16_t	8	Reserved field. Currently not used.
ts_start_us	InnoTimestampUs	64	The trigger time of the laser pulse in the 0th Block of InnoDataPacket.
lidar_mode	uint8_t	8	LiDAR working mode. lidar_mode=1: SLEEP lidar_mode=2: STANDBY lidar_mode=3: NORMAL

			lidar_mode=4: SHORT_RANGE (currently not used) lidar_mode=5: CALIBRIATION lidar_mode=6: PROTECTION lidar_mode=7: WORK_QUIET
lidar_status	uint8_t	8	LiDAR status. lidar_status=0: Initial status lidar_status=1: TRANSITION lidar_status=2: NORMAL lidar_status=3: FAILED

5.2.2.2 InnoCommonVersion

InnoCommonVersion： 6 Bytes.

Table 21 InnoCommonVersion Specification

Variable	Data type	Size (Bit)	Description
magic_number	uint16_t	16	Magic number, used for data verification.
major_version	uint8_t	8	Major version number.
minor_version	uint8_t	8	Minor version number.
fw_sequence	uint16_t	16	Reserved.

5.2.2.3 InnoStatusPacket Information

InnoStatusPacket Information: 572 Bytes

Table 22 InnoStatusPacket Information specification

Variable	Data type	Size (Bit)	Description
idx	uint64_t	64	The sequence number of status packet, starting from 0.
status_packet_interval_ms	uint8_t	8	Time interval for sending status packet. The unit is ms.
pre_lidar_mode	uint8_t	8	Previous mode of LiDAR.
in_transition_mode_ms	uint16_t	16	Time required for switching LiDAR mode. The unit is ms.
sn	char	128	LiDAR sequence number.
fault_version	uint16_t	16	The number of fault statuses changed. The initial value is 1.
reserve	uint16_t	112	Reserved. The default value is 0.
in_faults	InnoStatusInFaults	64	Internal fault code.
ex_faults	InnoStatusExFaults	64	Currently not used. The default value is 0.

counters	InnoStatusCounters	2560	Data statistics. For details, refer to 5.2.2.4 InnoStatusCounters.
sensor_readings	InnoStatusSensorReadings	1536	Sensor information. For details, refer to 5.2.2.5 InnoStatusSensorReadings.

5.2.2.4 InnoStatusCounters

InnoStatusCounters: 320 Bytes.

Table 23 InnoStatusCounters specification

Variable	Data type	Size (Bit)	Description
point_data_packet_sent	uint64_t	64	The number of InnoDataPackets sent.
point_sent	uint64_t	64	The number of InnoBlocks sent.
message_packet_sent	uint64_t	64	The number of messages sent.
raw_data_read	uint64_t	64	Size of raw_data read.
total_frame	uint64_t	64	The number of frames sent.
total_polygon_rotation	uint64_t	64	The number of polygon rotations.
total_polygon_facet	uint64_t	64	The number of polygon facet rotations.
power_up_time_in_second	uint32_t	32	Power up time. The unit is second.
process_up_time_in_second	uint32_t	32	Execution time. The unit is second.
lose_ptp_sync	uint32_t	32	The number of PTP time synchronization loss faults
bad_data	uint32_t	128	Bad data number. Currently not used. The default value is 0.
data_drop	uint32_t	256	The number of data losses.
in_signals	uint32_t	256	The number of signals.
latency_10us_average	uint16_t	96	Average delay. The unit is 10 µs.
latency_10us_variation	uint16_t	96	Fluctuating latency. The unit is 10 µs.
latency_10us_max	uint16_t	96	Maximum latency. The unit is 10 µs.
big_latency_frame	uint32_t	32	The number of long-latency frames. Currently not used. The default value is 0.
bad_frame	uint32_t	32	The number of bad frames. Currently not used. The default value is 0.
big_gap_frame	uint32_t	32	The number of long-interval frames. Currently not used. The default value is 0.
small_gap_frame	uint32_t	32	The number of short-interval frames. Currently not used. The default value is 0.
cpu_percentage	uint16_t	16	CPU utilization.
mem_percentage	uint16_t	16	Memory utilization.

motor	uint16_t	80	Polygon encodes time statistics.
galvo	uint16_t	80	Galvo encodes time statistics.
netstat_rx_speed_kBps	uint16_t	16	Packet receiving rate.
netstat_tx_speed_kBps	uint16_t	16	Packet sending rate.
netstat_rx_drop	uint16_t	16	The number of lost packets when receiving packets.
netstat_tx_drop	uint16_t	16	The number of lost packets when sending packets.
netstat_rx_err	uint16_t	16	The number of error packets when receiving packets.
netstat_tx_err	uint16_t	16	The number of error packets when sending packets.
sys_cpu_percentage	uint16_t	64	Total CPU utilization.
reserved	uint32_t	608	Reserved field. The default value is 0.

5.2.2.5 InnoStatusSensorReadings

InnoStatusSensorReadings:192 Bytes

Table 24 InnoStatusSensorReadings specification

Variable	Data type	Size (Bit)	Description
temperature_fpga_10th_c	int16_t	16	FPGA temperature. The unit is 0.1°C. For example, if the FPGA temperature is 30°C, then temperature_fpga_10th_c = 300.
temperature_laser_10th_c	int16_t	16	Laser temperature. The unit is 0.1°C. For example, if the laser temperature is 30°C, then temperature_laser_10th_c = 300.
temperature_adc_10th_c	int16_t	16	ADC module temperature. The unit is 0.1°C. For example, if the ADC module temperature is 30°C, then temperature_adc_10th_c = 300.
temperature_board_10th_c	int16_t	16	Mainboard temperature. The unit is 0.1°C. For example, if the mainboard temperature is 30°C, then temperature_board_10th_c = 300.
temperature_det_10th_c	int16_t	64	Detection board temperature. The unit is 0.1°C. It is used for fault detection. For example, if the detection board temperature is 30°C, then temperature_det_10th_c = 300.
temperature_other_10th_c	int16_t	48	Reserved.

heater_current_ma	uint16_t	16	Thermal current. The unit is mA.
motor_rpm_1000th	uint32_t	32	Rotation speed of the polygon, unit: Revolutions Per Minute (RPM) / 1000. For example, if the rotation speed of the polygon is 4200 RPM, motor_rpm_1000th = 4200000.
galvo_fpm_1000th	uint32_t	32	Scan speed of the Galvo, unit: RPM/1000. For example, if the rotation speed of the Galvo is 10 RPM, then galvo_fpm_1000th = 10000.
motor_rotation_total	uint64_t	64	Total number of polygon rotations.
galvo_round_total	uint64_t	64	Total number of Galvo scanning iterations.
moisture_index	uint16_t	32	Currently not used. The default value is 0.
window_blockage_index	uint16_t	32	Currently not used. The default value is 0.
motor	uint16_t	96	Polygon current. The unit is mA.
galvo	uint16_t	96	Galvo current. The unit is mA.
laser	uint16_t	96	Laser current. The unit is mA.
galvo_status_client	uint16_t	16	Customized parameter.
galvo_offset_angle_client	uint16_t	16	Customized parameter.
reserved	uint16_t	768	Currently not used. The default value is 0.

5.3 get_pcd

This chapter only contains the most common operations. You can run ./get_pcd -h command to obtain more information.

5.3.1 Start the LiDAR

1. Connect the power supply to start the LiDAR.
2. The LiDAR completes initialization and generates data after powering on for 11 to 18 seconds.

Note

The LiDAR does not have a power switch. It will become operational when power is applied.

5.3.2 Usage

1. Unzip the SDK tgz file.
2. Enter the get_pcd path.

```
cd /<SDK package path>/apps/example    // Go to the path of get_pcd.
```

3. Execute the innovusion_lidar_util command.

```
./get_pcd <parameter> // Run the get_pcd
```

5.3.3 Command

5.3.3.1 Set the working mode of the LiDAR

[Format]

```
./get_pcd --lidar-ip <LIDAR_IP> --lidar-mode <LIDAR_MODE>
```

[Parameter]

- <LIDAR_IP>: LiDAR IP address.
- <LIDAR_MODE>: The working mode of the LiDAR. A value of 3 means the normal mode. A value of 5 means the calibration mode.

[Sample]

Set the working mode of the LiDAR to the calibration mode.

```
./get_pcd --lidar-ip 172.168.1.10 --lidar-mode 5
```

5.3.3.2 Set the reflectance mode of the LiDAR

[Format]

```
./get_pcd --lidar-ip <LIDAR_IP> --reflectance <REFLECTANCE_MOOE>
```

[Parameter]

- <LIDAR_IP>: LiDAR IP address.
- <REFLECTANCE_MOOE>: The reflectance mode of the LiDAR. A value of 1 means the intensity mode. A value of 2 means the reflectivity mode.

[Sample]

Set the reflectance mode of the LiDAR to the intensity mode.

```
./get_pcd --lidar-ip 172.168.1.10 --reflectance 1
```

5.3.3.3 Set the return mode of the LiDAR

[Format]

```
./get_pcd --lidar-ip <LIDAR_IP> --multireturn <MULTI_RETURN_MODE>
```

[Parameter]

- <LIDAR_IP>: LiDAR IP address.
- <MULTI_RETURN_MODE>: The return of the LiDAR. A value of 1 means returning the single echo. A value of 2 means returning the two strongest echoes. A value of 3 means returning the strongest echo and the furthest echo.

[Sample]

Set the return mode of the LiDAR to the single return mode.

```
./get_pcd --lidar-ip 172.168.1.10 --multireturn 1
```

5.3.3.4 Set the position of the ROI

[Format]

```
./get_pcd --lidar-ip <LIDAR_IP> --falcon-eye <HORIZONTAL_DEGREE>,<VERTICAL_DEGREE>
```

[Parameter]

- <LIDAR_IP>: LiDAR IP address.
- <HORIZONTAL_DEGREE>: The horizontal position of the ROI center in degree, ranging from -60 to 60.
- <VERTICAL_DEGREE>: The vertical position of the ROI center in degree, ranging from -20 to 20.

[Sample]

Set the ROI center of the LiDAR to 40° in horizontal and 20° in vertical.

```
./get_pcd --lidar-ip 172.168.1.10 --falcon-eye 40,20
```

5.3.3.5 Record LiDAR point cloud data

[Format]

```
./get_pcd --lidar-ip <LIDAR_IP> {[--lidar-port <LIDAR_PORT>] | [--lidar-udp-port <LIDAR_UOP_PORT>]} [--use-tcp] [--file-number <NUMBER_OF_FILE>] [--frame-start <Nth_FRAME_TO_RECORD>] [--frame-number <NUMBER OF FRAME TO RECORD>] --output-filename <OUTPUT_FILENAME>
```

[Parameter]

- <LIDAR_IP>: LiDAR IP address.
- <LIDAR_PORT>: The TCP port of the LiDAR. The default is 8010. The command will read point cloud data through the port 8010 if the parameter is not specified.
- <LIDAR_UDP_PORT>: The UDP port of the LiDAR. The default is 8010. The command will read point cloud data through the port 8010 if the parameter is not specified.
- use-tcp: Specify the communication protocol. It defaults to the UDP port if the parameter is not specified.
- <NUMBER_OF_FILE>: The number of the recorded point cloud files.
- <Nth_FRAME_TO_RECORD>: The start frame of the recorded point cloud file.
- <NUMBER_OF_FRAME_TO_RECORD>: The size of each point cloud file in frame.
- <OUTPUT_FILENAME>: The name and format of the recorded file. The format of pcd, csv, inno_pc, inno_pc_xyz, inno_cframe, bag and png is supported by default.

[Sample]

Record 10 frames of the point cloud file in pcd format.

```
./get_pcd --lidar-ip 172.168.1.10 --frame-number 10 --output-filename test.pcd
```

5.3.3.6 Convert a inno_pc file

[Format]

```
./get_pcd --inno-pc-filename <INPUT_INNO PC_FILENAME> --output-filename <OUTPUT_FILENAME.pcd | csv | inno_pc | inno_pc_xyz | inno_cframe | bag | png>
```

[Parameter]

- <INPUT_INNO PC_FILENAME>: The inno_pc file name.
- <OUTPUT_FILENAME>: The name and format of the recorded file. The format of pcd, csv, inno_pc, inno_pc_xyz, inno_cframe, bag and png is supported by default.

[Sample]

Convert an inno_pc file into a pcd file.

```
./get_pcd --inno-pc-filename input.inno_pc --frame-start 0 --frame-number 1000 --output-filename test.pcd
```

Note

This command can also extract a specific range of frames during the file conversion by specifying values for frame-start and frame-number.

5.3.3.7 Split a file in inno_pc format

[Format]

```
./get_pcd --lidar-ip <LIDAR_IP> {[--lidar-port <LIDAR_PORT>] | [--lidar-udp-port <LIDAR_UOP_PORT>]} [--use-tcp] [--file-number <NUMBER_OF_FILE>] [--frame-start <Nth_FRAME_TO_RECORD>] [--frame-number <NUMBER OF FRAME TO RECORD>] --output-filename <OUTPUT_FILENAME> ./get_pcd --inno-pc-filename <INPUT_INNO PC_FILENAME> --frame-start <Nth_FRAME_TO_RECORO> --file-number <NUMBER_OF_FILE> --frame-number <NUMBER OF FRAME TO RECORD> --output-filename <OUTPUT_FILENAME.pcd | csv | inno_pc | inno_pc_xyz | inno_cframe | bag | png>
```

[Parameter]

- <INPUT_INNO PC_FILENAME>: The inno_pc file name.
- <Nth_FRAME_TO_RECORD>: The start frame of the recorded point cloud file.
- <NUMBER_OF_FILE>: The number of the generated files.
- <NUMBER OF FRAME TO RECORD>: The size of each point cloud file in frame.

- <OUTPUT_FILENAME>: The name and format of the recorded file. The format of pcd, csv, inno_pc, inno_pc_xyz, inno_cframe, bag and png is supported by default.

[Sample]

Split one inno_pc file into four files of five frames each.

```
./get_pcd --inno-pc-filename input.inno_pc --frame-start 0 --frame-number 5 --file-number 4
--output-filename test.inno_pc
```

5.4 Innovusion_lidar_util

5.4.1 Introduction

The Innovusion_lidar_util commands mainly provide the functions to change and get LiDAR-related parameters. For the detailed meaning of the symbols, refer to **Command line instructions.**

5.4.2 Usage

1. Unzip the SDK tgz file.
1. Enter the innovusion_lidar_util path.

```
cd /<SDK package path>/apps/lidar_util    // Go to the path of innovusion_lidar_util
```

2. Execute the innovusion_lidar_util command.

```
./innovusion_lidar_util <parameter> // Run the innovusion_lidar_util
```

5.4.2.1 download_cal_file

Download the LiDAR's yaml file to a local directory.

[Format]

```
./innovusion_lidar_util <ip of LIDAR> download_cal_file
```

[Parameter description]

- <ip of LIDAR> - LiDAR IP
- <file_id> - Calibration ID
- <filename> - Name of the yaml file

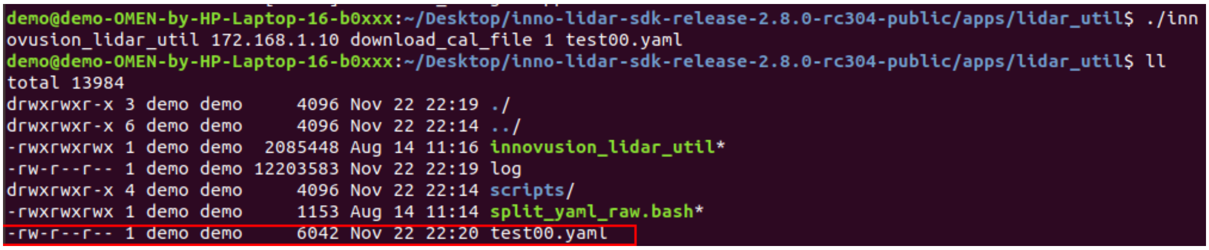
[Steps]

1. Enter the innovusion_lidar_util path.

```
cd /<SDK package path>/apps/lidar_util  // Go to the path of innovusion_lidar_util
```

2. Run the following command.

```
./innovusion_lidar_util <ip of LIDAR> download_cal_file <file_id> <filename>.
```



```
demo@demo-OMEN-by-HP-Laptop-16-b0xxx:~/Desktop/inno-lidar-sdk-release-2.8.0-rc304-public/apps/lidar_util$ ./innovusion_lidar_util 172.168.1.10 download_cal_file 1 test00.yaml
demo@demo-OMEN-by-HP-Laptop-16-b0xxx:~/Desktop/inno-lidar-sdk-release-2.8.0-rc304-public/apps/lidar_util$ ll
total 13984
drwxrwxr-x 3 demo demo     4096 Nov 22 22:19 ./
drwxrwxr-x 6 demo demo     4096 Nov 22 22:14 ../
-rwxrwxrwx 1 demo demo  2085448 Aug 14 11:16 innovusion_lidar_util*
-rw-r--r-- 1 demo demo 12203583 Nov 22 22:19 log
drwxrwxr-x 4 demo demo     4096 Nov 22 22:14 scripts/
-rwxrwxrwx 1 demo demo     1153 Aug 14 11:14 split_yaml_raw.bash*
-rw-r--r-- 1 demo demo     6042 Nov 22 22:20 test00.yaml
```

5.4.2.2 download_internal_file

Download the PCS/PTP configuration file of the LiDAR.

[Format]

```
./innovusion_lidar_util <ip of LIDAR> download_internal_file <file_id> <filename>
```

[Parameter description]

- <ip of LIDAR> - LiDAR IP.
- <file_id> - Configuration file type. The supported file types are as follows.
  - PCS_ENV: PCS configuration file.
  - PCS_CFG: PCS configuration file.
  - CFG: Firmware configuration file.
  - PTP: PTP configuration file.
  - PTP_WL: PTP filter whitelist. The whitelist is empty by default.
  - gPTP: gPTP configuration file.
- <filename> - Name of the PCS/PTP configuration file.

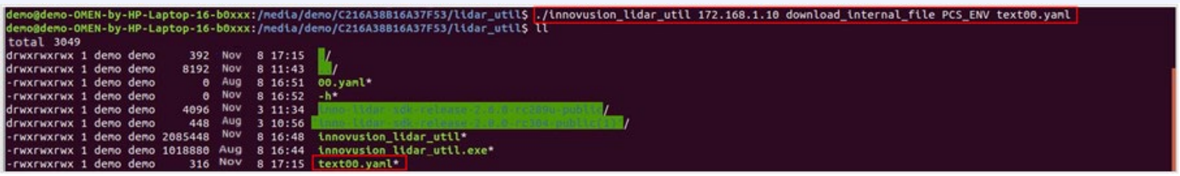
[Steps]

1. Enter the innovusion_lidar_util path.

```
cd /<SDK package path>/apps/lidar_util  // Go to the path of innovusion_lidar_util
```

2. Run the following command.

```
./innovusion_lidar_util <ip of LIDAR> download_internal_file <file_id> <filename>
```



```
demo@demo-OMEN-by-HP-Laptop-16-b0xxx:/media/demo/C216A38B16A37F53/lidar_util$ ./innovusion_lidar_util 172.168.1.10 download_internal_file PCS_ENV text00.yaml
demo@demo-OMEN-by-HP-Laptop-16-b0xxx:/media/demo/C216A38B16A37F53/lidar_util$ ll
total 3049
drwxrwxrwx 1 demo demo     392 Nov  8 17:15 ./
drwxrwxrwx 1 demo demo    8192 Nov  8 11:43 ../
-rwxrwxrwx 1 demo demo       0 Aug  8 16:51 00.yaml*
-rwxrwxrwx 1 demo demo       0 Nov  8 16:52 -h*
drwxrwxrwx 1 demo demo    4096 Nov  3 11:34 inno-lidar-sdk-release-2.6.0-rc289u-public/
drwxrwxrwx 1 demo demo     448 Aug  3 10:56 'inno-lidar-sdk-release-2.8.0-rc304-public(1)'/
-rwxrwxrwx 1 demo demo 2085448 Nov  8 16:48 innovusion_lidar_util*
-rwxrwxrwx 1 demo demo 1018880 Aug  8 16:44 innovusion_lidar_util.exe*
-rwxrwxrwx 1 demo demo     316 Nov  8 17:15 text00.yaml*
```

5.4.2.3 get_config

Get the LiDAR internal parameters.

[Format]

```
./innovusion_lidar_util <ip of LIDAR> get_config [section [key]]
```

[Parameter description]

- <section> - Manufacturer or time
- <key> - Parameter name
- <value> - Parameter value

[Steps]

1. Enter the innovusion_lidar_util path.

```
cd /<SDK package path>/apps/lidar_util  // Go to the path of innovusion_lidar_util
```

2. Run the following command.

```
/innovusion_lidar_util <ip of LIDAR> get_config [section [key]]
```

```
demo@demo-OMEN-by-HP-Laptop-16-b0xxx:/media/demo/C216A38B16A37F53/lidar_util/inno-lidar-sdk-release-2.8.0-rc304-public(1)/apps/lidar_util$ ./innovusion_lidar_util 172.168.1.10 get_config
[manufacture]
internal_server = 1
goto_worknormal = 1
reflectance_mode = 2
multiple_return_mode = 1
[time]
gps_pps_en = 0
gps_baudrate = 4800
ptp_en = 0
ptp_automotive = 1
ptp_delay_mode = 2
ptp_network_mode = 1
ntp_en = 1
ntp_server = "10.42.0.21"
demo@demo-OMEN-by-HP-Laptop-16-b0xxx:/media/demo/C216A38B16A37F53/lidar_util/inno-lidar-sdk-release-2.8.0-rc304-public(1)/apps/lidar_util$
```

5.4.2.4 get_config_changed

View the LiDAR's internal parameters that were changed.

[Format]

```
./innovusion_lidar_util <ip of LIDAR> get_config_changed [section [key]]
```

[Parameter description]

- <section> - Manufacturer or time
- <key> - Parameter name
- <value> - Parameter value

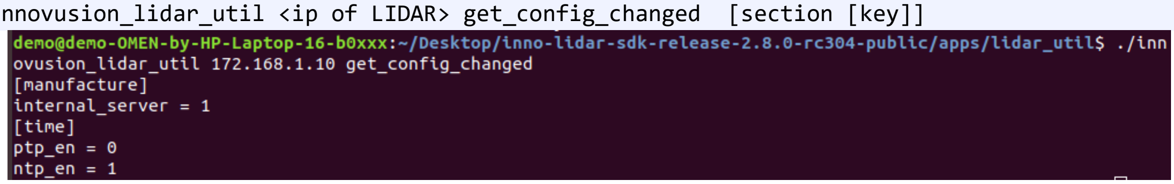
[Steps]

1. Enter the innovusion_lidar_util path.

```
cd /<SDK package path>/apps/lidar_util  // Go to the path of innovusion_lidar_util
```

2. Run the following command.

```
innovusion_lidar_util <ip of LIDAR> get_config_changed  [section [key]]
```



```
demo@demo-OMEN-by-HP-Laptop-16-b0xxx:~/Desktop/inno-lidar-sdk-release-2.8.0-rc304-public/apps/lidar_util$ ./inn
ovusion_lidar_util 172.168.1.10 get_config_changed
[manufacture]
internal_server = 1
[time]
ptp_en = 0
ntp_en = 1
```

5.4.2.5 get_detector_temps

Get the detector temperature.

[Format]

```
./innovusion_lidar_util <ip of LIDAR> get_detector_temps
```

[Parameter description]

- <ip of LIDAR> - LiDAR IP

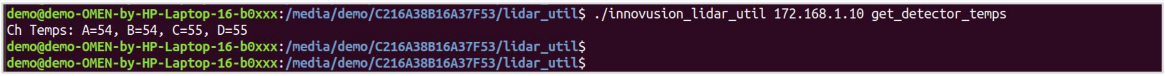
[Steps]

1. Enter the innovusion_lidar_util path.

```
cd /<SDK package path>/apps/lidar_util  // Go to the path of innovusion_lidar_util
```

2. Run the following command.

```
./innovusion_lidar_util <ip of LIDAR> get_detector_temps
```



5.4.2.6 get_error_code

Get the error code.

[Format]

```
./innovusion_lidar_util <ip of LIDAR> start <file_name> <capture_size_in_MB>
```

[Parameter description]

- <ip of LIDAR> - LiDAR IP

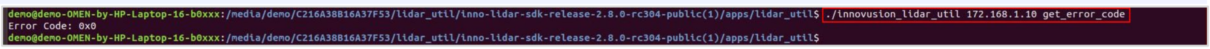
[Steps]

1. Enter the innovusion_lidar_util path.

```
cd /<SDK package path>/apps/lidar_util  // Go to the path of innovusion_lidar_util
```

2. Run the following command.

```
./innovusion_lidar_util <ip of LIDAR> get_error_code
```



5.4.2.7 get_framerate

Get the LiDAR frame rate information.

[Format]

```
./innovusion_lidar_util <ip of LIDAR> get_framerate
```

[Parameter description]

- <ip of LIDAR> - LiDAR IP

[Steps]

1. Enter the innovusion_lidar_util path.

```
cd /<SDK package path>/apps/lidar_util  // Go to the path of innovusion_lidar_util
```

2. Run the following command.

```
./innovusion_lidar_util <ip of LIDAR> get_framerate
```

```
demo@demo-OMEN-by-HP-Laptop-16-b0xxx:/media/demo/C216A38B16A37F53/lidar_util$ ./innovusion_lidar_util 172.168.1.10 get_framerate
10.00000
demo@demo-OMEN-by-HP-Laptop-16-b0xxx:/media/demo/C216A38B16A37F53/lidar_util$
```

5.4.2.8 get_log

Obtain all firmware log information.

[Format]

```
./innovusion_lidar_util <ip of LIDAR> get_log
```

[Parameter description]

- <ip of LIDAR> - LiDAR IP

[Steps]

1. Enter the innovusion_lidar_util path.

```
cd /<SDK package path>/apps/lidar_util  // Go to the path of innovusion_lidar_util
```

2. Run the following command.

```
./innovusion_lidar_util <ip of LIDAR> get_log
```

```
2021-12-20 00:15:11.688 [ INFO] dsp events: 0x10
2021-12-20 00:15:11.688 [ERROR] s...h.cc:1006 FAULT set: 15
2021-12-20 00:15:11.692 [ INFO] MCB TX: STW10800016ND
2021-12-20 00:15:11.794 [ INFO] MCB RX: AC
2021-12-20 00:15:11.795 [ INFO] dsp event interrupt
2021-12-20 00:15:11.795 [ INFO] MCB TX: STR108ND
2021-12-20 00:15:11.897 [ INFO] MCB RX: 00000
2021-12-20 00:15:11.897 [ INFO] dsp events: 0x0
2021-12-20 00:15:11.898 [ INFO] MCB TX: STR108ND
2021-12-20 00:15:12.000 [ INFO] MCB RX: 00016
2021-12-20 00:15:12.000 [ INFO] dsp events: 0x10
2021-12-20 00:15:12.000 [ERROR] s...h.cc:1006 FAULT set: 15
2021-12-20 00:15:12.004 [ INFO] MCB TX: STW10800016ND
2021-12-20 00:15:12.106 [ INFO] MCB RX: AC
2021-12-20 00:15:12.107 [ INFO] dsp event interrupt
2021-12-20 00:15:12.107 [ INFO] MCB TX: STR108ND
2021-12-20 00:15:12.209 [ INFO] MCB RX: 00000
2021-12-20 00:15:12.209 [ INFO] dsp events: 0x0
2021-12-20 00:15:12.210 [ INFO] MCB TX: STR108ND
2021-12-20 00:15:12.312 [ INFO] MCB RX: 00016
2021-12-20 00:15:12.313 [ INFO] dsp events: 0x10
```

5.4.2.9 get_log_file

Download the LiDAR's firmware log to a local directory.

[Format]

```
./innovusion_lidar_util <ip of LIDAR> get_log_file <filename>
```

[Parameter description]

- <ip of LIDAR> - LiDAR IP
- <filename> - Name of the log file.

[Steps]

1. Enter the innovusion_lidar_util path.

```
cd /<SDK package path>/apps/lidar_util  // Go to the path of innovusion_lidar_util
```

2. Run the following command.

```
./innovusion_lidar_util <ip of LIDAR> get_log_file <filename>
```

```
demo@demo-OMEN-by-HP-Laptop-16-b0xxx:~/Desktop/inno-lidar-sdk-release-2.8.0-rc304-public/apps/lidar_util$ ./inn
ovusion_lidar_util 172.168.1.10 get_log_file log
demo@demo-OMEN-by-HP-Laptop-16-b0xxx:~/Desktop/inno-lidar-sdk-release-2.8.0-rc304-public/apps/lidar_util$ ll
total 13976
drwxrwxr-x 3 demo demo     4096 Nov 22 22:19 ./
drwxrwxr-x 6 demo demo     4096 Nov 22 22:14 ../
-rwxrwxrwx 1 demo demo  2085448 Aug 14 11:16 innovusion_lidar_util*
-rw-r--r-- 1 demo demo 12203583 Nov 22 22:19 log
drwxrwxr-x 4 demo demo     4096 Nov 22 22:14 scripts/
-rwxrwxrwx 1 demo demo     1153 Aug 14 11:14 split_yaml_raw.bash*
```

5.4.2.10 get_model

Get the LiDAR's model information.

[Format]

```
./innovusion_lidar_util <ip of LIDAR> get_model
```

[Parameter description]

- <ip of LIDAR> - LiDAR IP

[Steps]

1. Enter the innovusion_lidar_util path.

```
cd /<SDK package path>/apps/lidar_util  // Go to the path of innovusion_lidar_util
```

2. Run the following command.

```
./innovusion_lidar_util <ip of LIDAR> get_model
```

```
demo@demo-OMEN-by-HP-Laptop-16-b0xxx:/media/demo/C216A38B16A37F53/lidar_util$ ./innovusion_lidar_util 172.168.1.10 get_model
model: k
```

5.4.2.11 get_motor_speeds

Get the rotation speed of the polygon.

[Format]

```
./innovusion_lidar_util <ip of LIDAR> get_motor_speeds
```

[Parameter description]

- <ip of LIDAR> - LiDAR IP

[Steps]

1. Enter the innovusion_lidar_util path.

```
cd /<SDK package path>/apps/lidar_util  // Go to the path of innovusion_lidar_util
```

2. Run the following command.

```
./innovusion_lidar_util <ip of LIDAR> get_motor_speeds
```

```
demo@demo-OMEN-by-HP-Laptop-16-b0xxx:/media/demo/C216A38B16A37F53/lidar_util$ ./innovusion_lidar_util 172.168.1.10 get_motor_speeds
SP: 4801
```

5.4.2.12 get_network

Obtain the LiDAR network information.

[Format]

```
./innovusion_lidar_util <ip of LIDAR> get_network
```

[Parameter description]

➢ <ip of LIDAR> - LiDAR IP

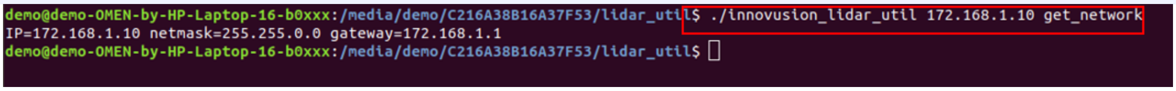
[Steps]

1. Enter the innovusion_lidar_util path.

```
cd /<SDK package path>/apps/lidar_util  // Go to the path of innovusion_lidar_util
```

2. Run the following command.

```
./innovusion_lidar_util <ip of LIDAR> get_network
```



```
demo@demo-OMEN-by-HP-Laptop-16-b0xxx:/media/demo/C216A38B16A37F53/lidar_util$ ./innovusion_lidar_util 172.168.1.10 get_network
IP=172.168.1.10 netmask=255.255.0.0 gateway=172.168.1.1
demo@demo-OMEN-by-HP-Laptop-16-b0xxx:/media/demo/C216A38B16A37F53/lidar_util$
```

5.4.2.13 get_registers

Get the register parameters.

[Format]

```
./innovusion_lidar_util <ip of LIDAR> get_registers
```

[Parameter description]

➢ <ip of LIDAR> - LiDAR IP

[Steps]

1. Enter the innovusion_lidar_util path.

```
cd /<SDK package path>/apps/lidar_util  // Go to the path of innovusion_lidar_util
```

2. Run the following command.

```
./innovusion_lidar_util <ip of LIDAR> get_registers
```

```
demo@demo-OMEN-by-HP-Laptop-16-b0xxx:/media/demo/C216A38B16A37F53/lidar_util$ ./innovusion_lidar_util 172.168.1.10 get_registers
HW/FW Version              0x00000000
Error Codes                0x00000000
Board Temperature & Alarms 0xdeadbeef
Laser Control & Status     0xdeadbeef
Laser Parameters           0xdeadbeef
Laser Temperature          0xdeadbeef
Laser Current              0xdeadbeef
Scan Head Control & Status 0xdeadbeef
Setting_1                  0xdeadbeef
Time_1                     0x00000000
Lock_1                     0x00080000
Setting_2                  0xdeadbeef
Time_2                     0xffffffff
Lock_2                     0x00080000
demo@demo-OMEN-by-HP-Laptop-16-b0xxx:/media/demo/C216A38B16A37F53/lidar_util$
```

5.4.2.14 get_sn

Get the serial number of LiDAR.

[Format]

```
./innovusion_lidar_util <ip of LIDAR> get_sn
```

[Parameter description]

➢ <ip of LIDAR> - LiDAR IP

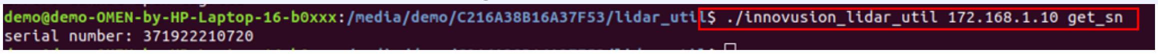
[Steps]

1. Enter the innovusion_lidar_util path.

```
cd /<SDK package path>/apps/lidar_util  // Go to the path of innovusion_lidar_util
```

2. Run the following command.

```
./innovusion_lidar_util <ip of LIDAR> get_sn
```



5.4.2.15 get_state

Obtain the LiDAR’s status.

[Format]

```
./innovusion_lidar_util <ip of LIDAR> get_state
```

[Parameter description]

➢ <ip of LIDAR> - LiDAR IP

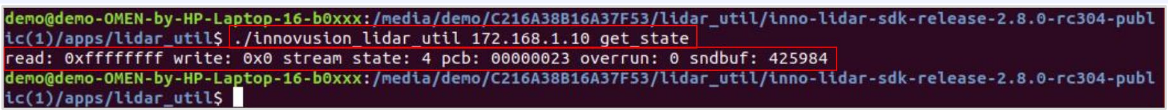
[Steps]

1. Enter the innovusion_lidar_util path.

```
cd /<SDK package path>/apps/lidar_util  // Go to the path of innovusion_lidar_util
```

2. Run the following command.

```
./innovusion_lidar_util <ip of LIDAR> get_state
```



5.4.2.16 get_status

Get the LiDAR's status information.

[Format]

```
./innovusion_lidar_util <ip of LIDAR> get_status
```

[Parameter description]

➢ <ip of LIDAR> - LiDAR IP

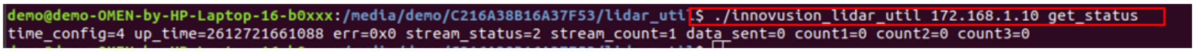
[Steps]

1. Enter the innovusion_lidar_util path.

```
cd /<SDK package path>/apps/lidar_util  // Go to the path of innovusion_lidar_util
```

2. Run the following command.

```
./innovusion_lidar_util <ip of LIDAR> get_status
```



5.4.2.17 get_temperature

Get the LiDAR temperature information.

[Format]

```
./innovusion_lidar_util <ip of LIDAR> get_temperature
```

[Parameter description]

- <ip of LIDAR> - LiDAR IP

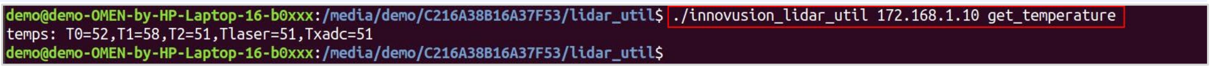
[Steps]

1. Enter the innovusion_lidar_util path.

```
cd /<SDK package path>/apps/lidar_util  // Go to the path of innovusion_lidar_util
```

2. Run the following command.

```
./innovusion_lidar_util <ip of LIDAR> get_temperature
```



5.4.2.18 get_uptime

View the total uptime of LiDAR.

[Format]

```
./innovusion_lidar_util <ip of LIDAR> get_uptime
```

[Parameter description]

- <ip of LIDAR> - LiDAR IP

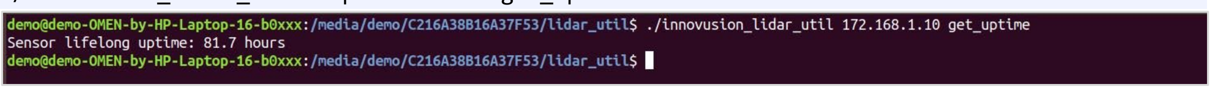
[Steps]

1. Enter the innovusion_lidar_util path.

```
cd /<SDK package path>/apps/lidar_util  // Go to the path of innovusion_lidar_util
```

2. Run the following command.

```
./innovusion_lidar_util <ip of LIDAR> get_uptime
```



5.4.2.19 get_version

Get detailed SW/FW version information of the LiDAR.

[Format]

```
./innovusion_lidar_util <ip of LIDAR> get_version
```

[Parameter description]

- <ip of LIDAR> - LiDAR IP

[Steps]

1. Enter the innovusion_lidar_util path.

```
cd /<SDK package path>/apps/lidar_util  // Go to the path of innovusion_lidar_util
```

2. Run the following command.

```
./innovusion_lidar_util <ip of LIDAR> get_version
```

```
demo@demo-OMEN-by-HP-Laptop-16-b0xxx:/media/demo/C216A38B16A37F53/lidar_util/inno-lidar-sdk-release-2.8.0-rc304-public(1)/apps/lidar_util$ ./innovusion_lidar_util 172.168.1.10 get_version
Software Version: 2.8.0-rc304-public
  build_tag: release-2.8.0-rc304
  build_time: 15:16:53 Aug 14 2022

App Version: BL131-fk-ptp-wl-rc1.861
build-time: 2022-10-10-22-45-11
FPGA Datecode: 0x220921f3
fpga-ver: 0x00
fpga-rev: 0x0
board-rev: 0x6
MCB-ver: FI36-C2-0.3.6.2
LASER: PN: ESFL-15C14Y-2P, SN: 022226632, HV: V1.0, FV: 35YMW, DATE: 2022.05.29
Firmware Version: falconk-2822.2022-10-26-15-37-05
  build-tag: falconk-2822
  build-seq: 2822
  build-time: 2022-10-26-15-37-05
  build-git-tag: 1.0.6
  board-type: falconib
demo@demo-OMEN-by-HP-Laptop-16-b0xxx:/media/demo/C216A38B16A37F53/lidar_util/inno-lidar-sdk-release-2.8.0-rc304-public(1)/apps/lidar_util$
```

5.4.2.20 ptp_diag

Note

The command is only supported by SDK 2.10.0 or above.

Obtain PTP diagnosis results.

[Format]

```
./innovusion_lidar_util <ip of LIDAR> ptp_diag [<status> | <trace> <period> <interval> | <stop> | <show>]
```

[Parameter description]

- <ip of LIDAR> - LiDAR IP.
- <status> - Obtain the status of PTP.
- <trace> -Start internal ptp_diag program of the LiDAR. <interval> is the time interval of the obtained PTP clock’s time deviation. <period> is the total number of the obtained time deviation.
- <period> - The total number of the obtained time deviation.
- <interval> - Time interval of the obtained PTP clock’s time deviation. The unit is second.
- <stop> - Stop internal ptp_diag program immediately.
- <show> - Show the result of the ptp_diag.

[Steps]

1. Enter the innovusion_lidar_util path.

```
cd /<SDK package path>/apps/lidar_util  // Go to the path of innovusion_lidar_util
```

2. Run the following command.

```
./innovusion_lidar_util <ip of LIDAR> ptp_diag [<status> | <trace> <period> <interval> | <stop> | <show>]
```

```
demo@demo-OMEN-by-HP-Laptop-16-b0xxx:/media/demo/C216A38B16A37F53/lidar_util/inno-lidar-sdk-release-2.8.0-rc304-public(1)/apps/lidar_util$ ./innovusion_lidar_util 172.168.1.10 ptp_diag status
start ptp diag

param:
loop 0
interval 0
ptp config /mnt/config_firmware/inno_internal_file_gPTP

sending: GET DEFAULT_DATA_SET
        04c3e6.fffe.433303-0 seq 0 RESPONSE MANAGEMENT DEFAULT_DATA_SET
                twoStepFlag             1
                slaveOnly               1
                numberPorts             1
                priority1               254
                clockClass              255
                clockAccuracy           0xfe
                offsetScaledLogVariance 0xffff
                priority2               254
                clockIdentity           04c3e6.fffe.433303
                domainNumber            0

sending: GET TIME_PROPERTIES_DATA_SET
        04c3e6.fffe.433303-0 seq 0 RESPONSE MANAGEMENT TIME_PROPERTIES_DATA_SET
                currentUtcOffset      37
                leap61                0
                leap59                0
                currentUtcOffsetValid 0
                ptpTimescale          0
                timeTraceable         0
                frequencyTraceable    0
                timeSource            0xa0

sending: GET GRANDMASTER_SETTINGS_NP
        04c3e6.fffe.433303-0 seq 0 RESPONSE MANAGEMENT GRANDMASTER_SETTINGS_NP
                clockClass              255
                clockAccuracy           0xfe
                offsetScaledLogVariance 0xffff
                currentUtcOffset        37
                leap61                  0
                leap59                  0
                currentUtcOffsetValid   0
```

5.4.2.21 raw_capture

Capture raw raw data.

[Format]

```
./innovusion_lidar_util <ip of LIDAR> raw_capture <unit> <channel> <capture_size> <0|1>
```

[Parameter description]

- <ip of LIDAR> - LiDAR IP
- <unit> - Name of the saved file
- <channel> - Channel number
- <capture_size> - Size of the captured file. The unit is MB.
- <0|1> - If the yaml file is included in the raw raw data. 0 means the yaml file is not included. 1 means the yaml file is included.

[Steps]

1. Enter the innovusion_lidar_util path.

```
cd /<SDK package path>/apps/lidar_util  // Go to the path of innovusion_lidar_util
```

2. Run the following command.

```
./innovusion_lidar_util <ip of LIDAR> raw_capture <unit> <channel> <capture_size> <0|1>
```

```
demo@demo-OMEN-by-HP-Laptop-16-b0xxx:/media/demo/C216A38B16A37F53/lidar_util/inno-lidar-sdk-release-2.8.0-rc304-public(1)/apps/lidar_util$ ./innovusion_lidar_util 172.168.1.10 raw_capture raw 1 5 0
Stopped.
capture_size: 5MB
Done
Done
id: 1
Stopped.
data recorded in raw_ADCData_11_04_2022_13_29_23_C1.raw
Done
demo@demo-OMEN-by-HP-Laptop-16-b0xxx:/media/demo/C216A38B16A37F53/lidar_util/inno-lidar-sdk-release-2.8.0-rc304-public(1)/apps/lidar_util$
```

5.4.2.22 set_config

Set the LiDAR parameters.

[Format]

```
./innovusion_lidar_util <ip of LIDAR> set_config <section> <key> <value> <…>
```

[Parameter description]

- <section> - Manufacturer or time
- <key> - Parameter name
- <value> - Parameter value

[Steps]

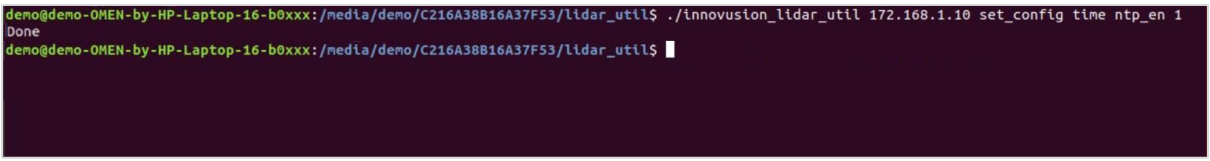
1. Enter the innovusion_lidar_util path.

```
cd /<SDK package path>/apps/lidar_util  // Go to the path of innovusion_lidar_util
```

2. Run the following command.

```
./innovusion_lidar_util <ip of LIDAR> set_config <section> <key> <value> <…>
```

3. You can use get_config_changed or get_config to confirm if the parameters have been changed.



5.4.2.23 set_ip_address

Change the LiDAR IP.

[Format]

```
./innovusion_lidar_util <ip of LIDAR> set_ip_address <new_ip_address>
```

[Parameter description]

- <ip of LIDAR> - LiDAR IP
- <new_ip_address> - New LiDAR IP

[Steps]

1. Enter the innovusion_lidar_util path.

```
cd /<SDK package path>/apps/lidar_util  // Go to the path of innovusion_lidar_util
```

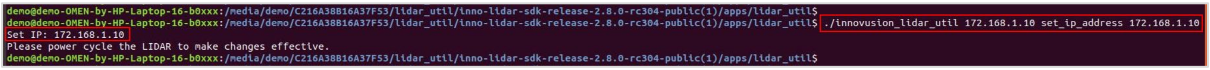
2. Run the following command.

```
./innovusion_lidar_util <ip of LIDAR> set_ip_address <new_ip_address>.
```

3. Reboot the LiDAR. You can reboot the LiDAR with the soft reboot command or the hard reboot (power reboot).

```
./innovusion_lidar_util <old ip address> soft_reboot
```

4. (Optional) You can check if the IP address has been changed by checking the LiDAR network information or logging into the ILA.



```
demo@demo-OMEN-by-HP-Laptop-16-b0xxx:/media/demo/C216A38B16A37F53/lidar_util/inno-lidar-sdk-release-2.8.0-rc304-public(1)/apps/lidar_util$
demo@demo-OMEN-by-HP-Laptop-16-b0xxx:/media/demo/C216A38B16A37F53/lidar_util/inno-lidar-sdk-release-2.8.0-rc304-public(1)/apps/lidar_util$ ./innovusion_lidar_util 172.168.1.10 set_ip_address 172.168.1.10
Set IP: 172.168.1.10
Please power cycle the LIDAR to make changes effective.
demo@demo-OMEN-by-HP-Laptop-16-b0xxx:/media/demo/C216A38B16A37F53/lidar_util/inno-lidar-sdk-release-2.8.0-rc304-public(1)/apps/lidar_util$
```

Note

The new IP address should be configured based on the LiDAR’ s gateway and should be in the same subnet with the LiDAR. About how to query the LiDAR gateway, see 5.4.2.12 get_network.

5.4.2.24 set_network

Change the LiDAR network information.

[Format]

```
./innovusion_lidar_util <ip of LIDAR> set_network <new_ip_address> <new_netmask_address> <new_gateway_address>
```

[Parameter description]

- <ip of LIDAR> - LiDAR IP
- <new_ip_address> - New LiDAR IP
- <new_netmask_address> - New subnet mask
- <new_gateway_address> - New gateway

[Steps]

1. Enter the innovusion_lidar_util path.

```
cd /<SDK package path>/apps/lidar_util  // Go to the path of innovusion_lidar_util
```

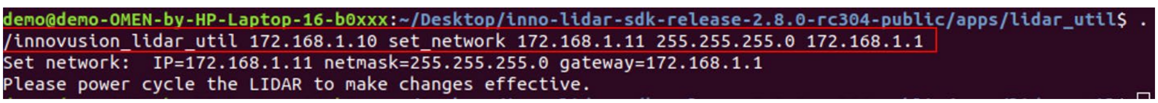
2. Run the following command.

```
./innovusion_lidar_util <ip of LIDAR> set_network <new_ip_address>  <new_netmask_address> <new_gateway_address>.
```

3. Reboot the LiDAR. You can reboot the LiDAR with the soft reboot command or the hard reboot (power reboot).

```
./innovusion_lidar_util <old ip address> soft_reboot
```

4. (Optional) You can check if the IP has been changed by viewing the LiDAR network information or logging into the ILA.



```
demo@demo-OMEN-by-HP-Laptop-16-b0xxx:~/Desktop/inno-lidar-sdk-release-2.8.0-rc304-public/apps/lidar_util$ ./innovusion_lidar_util 172.168.1.10 set_network 172.168.1.11 255.255.255.0 172.168.1.1
Set network:  IP=172.168.1.11 netmask=255.255.255.0 gateway=172.168.1.1
Please power cycle the LIDAR to make changes effective.
```

Note

The default IP address is 172.168.1.10. The default subnet mask is 255.255.255.0. The default gateway is 172.168.1.1.

5.4.2.25 set_vertical_roi

Set the vertical position of the ROI center.

[Format]

```
./innovusion_lidar_util <ip of LIDAR> set_vertical_roi <degree>
```

[Parameter description]

- <ip of LIDAR> - LiDAR IP
- <degree> - Vertical degree of the ROI center. The range is -25 to 25.

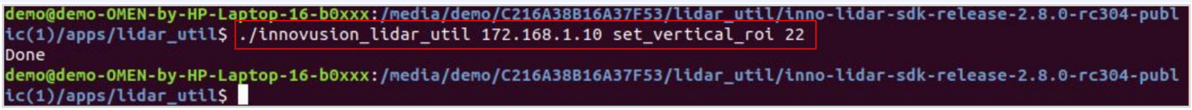
[Steps]

1. Enter the innovusion_lidar_util path.

```
cd /<SDK package path>/apps/lidar_util  // Go to the path of innovusion_lidar_util
```

2. Run the following command.

```
./innovusion_lidar_util <ip of LIDAR> set_vertical_roi <degree>
```



5.4.2.26 soft_reboot

Soft reboot.

[Format]

```
./innovusion_lidar_util <ip of LIDAR> soft_reboot
```

[Parameter description]

- <ip of LIDAR> - LiDAR IP

[Steps]

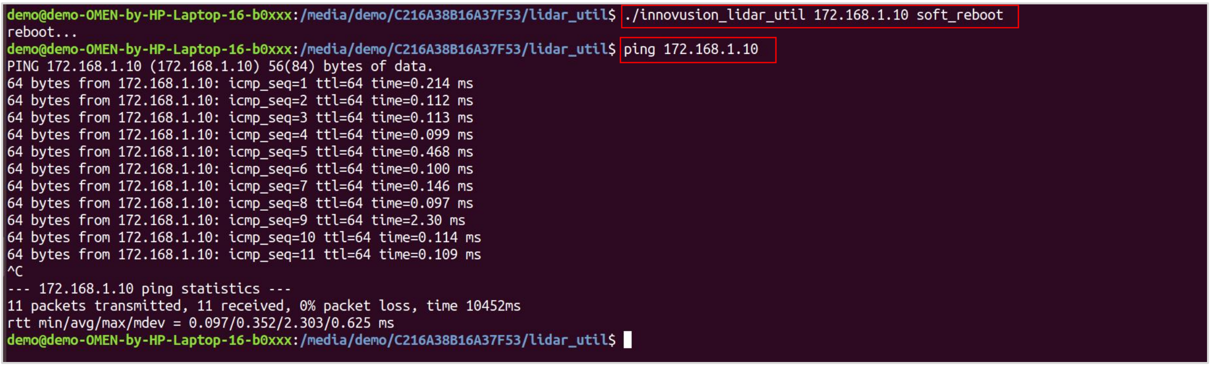
1. Enter the innovusion_lidar_util path.

```
cd /<SDK package path>/apps/lidar_util  // Go to the path of innovusion_lidar_util
```

2. Run the following command.

```
./innovusion_lidar_util <ip of LIDAR> soft_reboot
```

3. (Optional) After the reboot, you can check if the LiDAR point cloud is normal via the ILA or the ping command.



5.4.2.27 start

Download data.

[Format]

```
./innovusion_lidar_util <ip of LIDAR> start <file_name> <capture_size_in_MB>
```

[Parameter description]

- <ip of LIDAR> - LiDAR IP
- <file_name> - Name of the downloaded data file
- <capture_size_in_MB> - The unit is MB. (If it is not a positive number, it will be streamed indefinitely, but no data will be saved.)

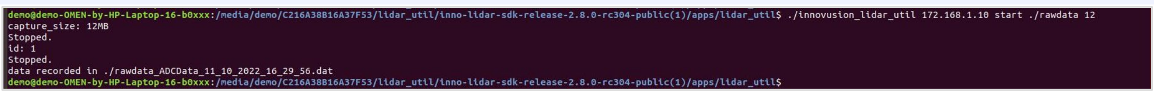
[Steps]

1. Enter the innovusion_lidar_util path.

```
cd /<SDK package path>/apps/lidar_util  // Go to the path of innovusion_lidar_util
```

2. Run the following command to capture and download the raw data. The result of <capture_size_in_MB> must be greater than zero.

```
./innovusion_lidar_util <ip of LIDAR> start <file_name> <capture_size_in_MB>
```



5.4.2.28 stop

Stop the internal point cloud service of the LiDAR.

[Format]

```
./innovusion_lidar_util <ip of LIDAR> stop
```

[Parameter description]

- <ip of LIDAR> - LiDAR IP

[Steps]

1. Enter the innovusion_lidar_util path.

```
cd /<SDK package path>/apps/lidar_util  // Go to the path of innovusion_lidar_util
```

2. Run the following command.

```
./innovusion_lidar_util <ip of LIDAR> stop
```

5.4.2.29 upload_cal_file

Upload the LiDAR's YAML file for calibration.

[Format]

```
./innovusion_lidar_util <ip of LIDAR> upload_cal_file
```

[Parameter description]

- <ip of LIDAR> - LiDAR IP
- <file_id> - Calibration ID. field_id is 0.
- <filename> - Name of the yaml file

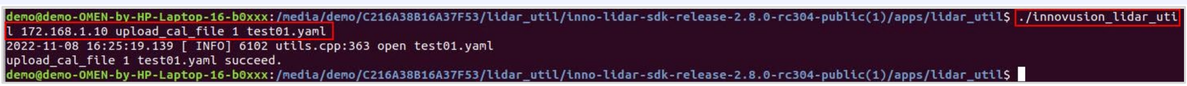
[Steps]

1. Enter the innovusion_lidar_util path.

```
cd /<SDK package path>/apps/lidar_util  // Go to the path of innovusion_lidar_util
```

2. Run the following command.

```
./innovusion_lidar_util <ip of LIDAR> upload_cal_file
```



5.4.2.30 upload_internal_file

Upload the PCS/PTP configuration file of the LiDAR.

[Format]

```
./innovusion_lidar_util <ip of LIDAR> upload_internal_file <file_id> <filename>
```

[Parameter description]

- <ip of LIDAR> - LiDAR IP.
- <file_id> - Configuration file type. The supported file types are as follows.
  - PCS_ENV: PCS configuration file.
  - PCS_CFG: PCS configuration file.
  - CFG: Firmware configuration file.
  - PTP: PTP configuration file.
  - PTP_WL: PTP filter whitelist. The whitelist is empty by default.
  - gPTP: gPTP configuration file.

- <filename> - Name of the PCS/PTP configuration file.

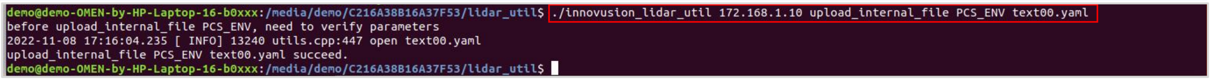
[Steps]

1. Enter the innovusion_lidar_util path.

```
cd /<SDK package path>/apps/lidar_util  // Go to the path of innovusion_lidar_util
```

2. Run the following command to upload the PCS/PTP configuration file in the specified path.

```
./innovusion_lidar_util <ip of LIDAR> upload_internal_file <file_id> <filename>
```



5.5 cURL commands

5.5.1 Introduction

The cURL commands mainly provide the functions get and set LiDAR-related parameters. For the detailed meaning of the symbols, refer to Appendix A Command format instruction.

The cURL commands are based on the 3692 version of the firmware with inno-lidar-sdk-release-2.11.0-fk-rc2.1134.

5.5.2 Usage

1. Connect the computer to LiDAR and ensure the Ethernet connection.
2. Change the computer IP address to the same subnet with the LiDAR.
3. Open a terminal to run the command.

5.5.3 Capture

[Description]

Capture the point cloud data and saves it as a file.

[Format]

```
curl  "http://<LIDAR_IP>:<LIDAR_PORT>/capture/?type=<capture_type>&duration=<capture_time>"
-O -J
```

[Parameter]

<capture_type> is the type of the captured data. The available data type is as follows.

- pcd
- pcd_binary
- csv
- inno_pc
- inno_pc_xyz

- png
- raw
- inno_raw
- raw_raw
- yaml

<capture_time> is the capturing data, in frames.

5.5.4 Getter

5.5.4.1 get_enabled

[Description]

Get if Falcon internal PCS is enabled.

[Format]

```
curl http://<LIDAR_IP>:<LIDAR_PORT>/command/?get_enabled
```

[Return value]

- A return value of 0 means the internal PCS is disabled.
- A return value of 1 means the internal PCS is enabled.

```
demo@demo-OMEN-by-HP-Laptop-16-b0xxx:~$ curl http://172.168.1.10:8010/command/?get_enabled
1.000000demo@demo-OMEN-by-HP-Laptop-16-b0xxx:~$
```

5.5.4.2 get_error_log

[Description]

Get most recent lines from error log, including warning information.

[Format]

```
curl http://<LIDAR_IP>:<LIDAR_PORT>/command/?get_error_log
```

[Return value]

```
demo@demo-OMEN-by-HP-Laptop-16-b0xxx:~$ curl http://172.168.1.10:8010/command/?get_error_log
2021-12-19 19:46:55.172 [ WARN]  913 stage_deliver.cpp:953 frame points too few(0) for 1 times,
 skipped blocks: 8081
2021-12-19 19:46:55.276 [ WARN]  913 stage_deliver.cpp:953 frame points too few(10) for 2 times
, skipped blocks: 22267
2021-12-19 19:46:55.367 [ WARN]  913 stage_deliver.cpp:953 frame points too few(2) for 3 times,
 skipped blocks: 22282
2021-12-19 19:46:55.474 [ WARN]  913 stage_deliver.cpp:953 frame points too few(0) for 4 times,
 skipped blocks: 22285
2021-12-19 19:46:55.578 [ WARN]  913 stage_deliver.cpp:953 frame points too few(3) for 5 times,
 skipped blocks: 22279
2021-12-19 19:46:55.670 [ WARN]  913 stage_deliver.cpp:953 frame points too few(5) for 6 times,
 skipped blocks: 22275
2021-12-19 19:46:55.775 [ WARN]  913 stage_deliver.cpp:953 frame points too few(0) for 7 times,
 skipped blocks: 22283
2021-12-19 19:46:55.864 [ WARN]  913 stage_deliver.cpp:953 frame points too few(0) for 8 times,
 skipped blocks: 22281
2021-12-19 19:46:55.973 [ WARN]  913 stage_deliver.cpp:953 frame points too few(3) for 9 times,
 skipped blocks: 22281
2021-12-19 19:46:56.078 [ WARN]  913 stage_deliver.cpp:953 frame points too few(0) for 10 times
, skipped blocks: 22279
2021-12-19 19:46:56.169 [ WARN]  913 stage_deliver.cpp:953 frame points too few(3) for 11 times
, skipped blocks: 22273
2021-12-19 19:46:56.275 [ WARN]  913 stage_deliver.cpp:953 frame points too few(0) for 12 times
, skipped blocks: 22282
2021-12-19 19:46:56.363 [ WARN]  913 stage_deliver.cpp:953 frame points too few(0) for 13 times
, skipped blocks: 22283
2021-12-19 19:46:56.472 [ WARN]  913 stage_deliver.cpp:953 frame points too few(5) for 14 times
, skipped blocks: 22274
2021-12-19 19:46:56.576 [ WARN]  913 stage_deliver.cpp:953 frame points too few(4) for 15 times
, skipped blocks: 22273
2021-12-19 19:46:56.666 [ WARN]  913 stage_deliver.cpp:953 frame points too few(2) for 16 times
, skipped blocks: 22279
2021-12-19 19:46:56.718 [ WARN]  919 resource_stats.cpp:150 <READ> bandwidth too low: 1.95825,
counter: 1
2021-12-19 19:46:56.773 [ WARN]  913 stage_deliver.cpp:953 frame points too few(0) for 17 times
, skipped blocks: 22278
2021-12-19 19:46:56.861 [ WARN]  913 stage_deliver.cpp:953 frame points too few(2) for 18 times
, skipped blocks: 22277
2021-12-19 19:46:56.971 [ WARN]  913 stage_deliver.cpp:953 frame points too few(4) for 19 times
, skipped blocks: 22274
2021-12-19 19:46:57.076 [ WARN]  913 stage_deliver.cpp:953 frame points too few(0) for 20 times
```

5.5.4.3 get_error_log_file

[Description]

Get the complete error log file.

[Format]

```
curl http://<LIDAR_IP>:<LIDAR_PORT>/command/?get_error_log_file
```

[Return value]

```
demo@demo-OMEN-by-HP-Laptop-16-b0xxx:~$ curl http://172.168.1.10:8010/command/?get_error_log_file
2021-12-31 01:34:03.587 [ WARN] 24509 resource_stats.cpp:150 <READ> bandwidth too low: 2.94847, counter: 1698
2021-12-31 01:34:53.595 [ WARN] 24509 resource_stats.cpp:150 <READ> bandwidth too low: 2.94865, counter: 1699
2021-12-31 01:35:43.596 [ WARN] 24503 resource_stats.cpp:150 <READ> bandwidth too low: 2.94775, counter: 1700
2021-12-31 01:36:33.605 [ WARN] 24509 resource_stats.cpp:150 <READ> bandwidth too low: 2.94859, counter: 1701
2021-12-31 01:37:23.617 [ WARN] 24509 resource_stats.cpp:150 <READ> bandwidth too low: 2.94841, counter: 1702
2021-12-31 01:38:13.632 [ WARN] 24509 resource_stats.cpp:150 <READ> bandwidth too low: 2.94824, counter: 1703
2021-12-31 01:39:03.636 [ WARN] 24509 resource_stats.cpp:150 <READ> bandwidth too low: 2.94888, counter: 1704
2021-12-31 01:39:53.646 [ WARN] 24501 resource_stats.cpp:150 <READ> bandwidth too low: 2.94722, counter: 1705
2021-12-31 01:40:43.662 [ WARN] 24509 resource_stats.cpp:150 <READ> bandwidth too low: 2.94949, counter: 1706
2021-12-31 01:41:33.670 [ WARN] 24509 resource_stats.cpp:150 <READ> bandwidth too low: 2.94865, counter: 1707
2021-12-31 01:42:23.680 [ WARN] 24509 resource_stats.cpp:150 <READ> bandwidth too low: 2.94853, counter: 1708
2021-12-31 01:43:13.681 [ WARN] 24503 resource_stats.cpp:150 <READ> bandwidth too low: 2.94775, counter: 1709
2021-12-31 01:44:03.689 [ WARN] 24509 resource_stats.cpp:150 <READ> bandwidth too low: 2.94996, counter: 1710
2021-12-31 01:44:53.692 [ WARN] 24509 resource_stats.cpp:150 <READ> bandwidth too low: 2.94769, counter: 1711
2021-12-31 01:46:33.693 [ WARN] 24503 resource_stats.cpp:150 <READ> bandwidth too low: 2.94775, counter: 1713
2021-12-31 01:47:23.703 [ WARN] 24509 resource_stats.cpp:150 <READ> bandwidth too low: 2.94853, counter: 1714
2021-12-31 01:48:13.713 [ WARN] 24509 resource_stats.cpp:150 <READ> bandwidth too low: 2.94722, counter: 1715
2021-12-31 01:49:03.714 [ WARN] 24503 resource_stats.cpp:150 <READ> bandwidth too low: 2.94775, counter: 1716
2021-12-31 01:49:53.726 [ WARN] 24509 resource_stats.cpp:150 <READ> bandwidth too low: 2.94972, counter: 1717
2021-12-31 01:50:43.728 [ WARN] 24509 resource_stats.cpp:150 <READ> bandwidth too low: 2.94900, counter: 1718
2021-12-31 01:51:33.729 [ WARN] 24509 resource_stats.cpp:150 <READ> bandwidth too low: 2.94906, counter: 1719
2021-12-31 01:52:23.730 [ WARN] 24508 resource_stats.cpp:150 <READ> bandwidth too low: 2.94906, counter: 1720
2021-12-31 01:53:13.732 [ WARN] 24509 resource_stats.cpp:150 <READ> bandwidth too low: 2.94906, counter: 1721
```

5.5.4.4 get_error_log_size

[Description]

Get the size of the error log (in bytes) in this boot, the lines of the warnings, and the lines of the errors.

[Format]

```
curl http://<LIDAR_IP>:<LIDAR_PORT>/command/?get_error_log_size
```

[Return value]

```
demo@demo-OMEN-by-HP-Laptop-16-b0xxx:~$ curl http://172.168.1.10:8010/command/?get_error_log_size
4624 34 1demo@demo-OMEN-by-HP-Laptop-16-b0xxx:~$
```

5.5.4.5 get_frame_rate

[Description]

Get the frames per second.

[Format]

```
curl http://<LIDAR_IP>:<LIDAR_PORT>/command/?get_frame_rate
```

[Return value]

```
demo@demo-OMEN-by-HP-Laptop-16-b0xxx:~$ curl http://172.168.1.10:8010/command/?get_frame_rate
10.000000demo@demo-OMEN-by-HP-Laptop-16-b0xxx:~$
```

5.5.4.6 get_frame_sync_stats

[Description]

Get statistics about frame sync since power on.

[Format]

```
curl http://<LIDAR_IP>:<LIDAR_PORT>/command/?get_frame_sync_stats
```

[Return value]

```
demo@demo-OMEN-by-HP-Laptop-16-b0xxx:~$ curl http://172.168.1.10:8010/command/?get_frame_sync_stats
{"settings":{"target_time_s":0.000000},"frame_sync_stats":{"sync_status":{"value":0,"time_sync_start_us"
:0,"cost_time_ms":0,"time_boot_to_sync_ms":0,"desc":"value: 0-init; 1-synced; 2-lost.cost_time_ms is tim
e from init to synced. time_boot_to_sync_ms is time cost from system boot up to frame sync success."},"f
rame_start_diff_ns":{"mean":0,"max":0,"std_dev":0,"count":0,"histogram_diff":[0, 0, 0, 0, 0, 0, 0, 0, 0,
 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0
, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0
, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0
, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0
, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0
, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0
, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
```

5.5.4.7 get_fw_version

[Description]

Get information about the currently installed firmware, including APP version, build time, firmware version, etc.

[Format]

```
curl http://<LIDAR_IP>:<LIDAR_PORT>/command/?get_fw_version
```

[Return value]

```
demo@demo-OMEN-by-HP-Laptop-16-b0xxx:~$ curl http://172.168.1.10:8010/command/?get_fw_version
App Version: release-2.11.0-fk-rc2.1134
build-time: 2023-03-30-10-32-03
FPGA Datecode: 0x230210f3
fpga-ver: 0x00
fpga-rev: 0x0
board-rev: 0x6
MCB-ver: FI36-C2-0.4.9.3
LASER: PN: ESFL-15C14Y-2P, SN: 022226632, HV: V1.0, FV: 35YMZ, DATE: 2022.05.29
Firmware Version: falconk-2.11.0-3692.2023-03-30-02-34-16
  build-tag: falconk-2.11.0-3692
  build-seq: 3692
  build-time: 2023-03-30-02-34-16
  build-git-tag: 1.0.13.1
  board-type: falconib
  customer: normal
demo@demo-OMEN-by-HP-Laptop-16-b0xxx:~$
```

5.5.4.8 get_mode

[Description]

Get the current working mode, the previous working mode, the status of working mode and the working mode transition time in ms.

[Format]

```
curl http://<LIDAR_IP>:<LIDAR_PORT>/command/?get_mode
```

[Return value]

The meaning of the return value corresponding to the working mode is as follows.

- A return value of 1 means the sleep mode.

- A return value of 2 means the standby mode.
- A return value of 3 means the normal mode.
- A return value of 4 means the short-range mode.
- A return value of 5 means the calibration mode.
- A return value of 6 means the protection mode.
- A return value of 7 means the quiet mode.

The meaning of the return value corresponding to the working mode status is as follows.

- A return value of 1 means the transition is in progress.
- A return value of 2 means the transition has been succeeded.
- A return value of 3 means the transition has been failed.

```
demo@demo-OMEN-by-HP-Laptop-16-b0xxx:~$ curl http://172.168.1.10:8010/command/?get_mode
3,2,2,0demo@demo-OMEN-by-HP-Laptop-16-b0xxx:~$
```

5.5.4.9 get_model

[Description]

Get the LiDAR model, e.g., i or k.

[Format]

```
curl http://<LIDAR_IP>:<LIDAR_PORT>/command/?get_model
```

[Return value]

```
demo@demo-OMEN-by-HP-Laptop-16-b0xxx:~$ curl http://172.168.1.10:8010/command/?get_model
kdemo@demo-OMEN-by-HP-Laptop-16-b0xxx:~$
```

5.5.4.10 get_mode_status

[Description]

Get the current working mode, the previous working mode, the status of working mode and the working mode transition time in ms.

[Format]

```
curl http://<LIDAR_IP>:<LIDAR_PORT>/command/?get_mode_status
```

[Return value]

The meaning of the return value corresponding to the working mode is as follows.

- A return value of 1 means the sleep mode.
- A return value of 2 means the standby mode.
- A return value of 3 means the normal mode.
- A return value of 4 means the short-range mode.

- A return value of 5 means the calibration mode.
- A return value of 6 means the protection mode.
- A return value of 7 means the quiet mode.

The meaning of the return value corresponding to the working mode status is as follows.

- A return value of 1 means the transition is in progress.
- A return value of 2 means the transition has been succeeded.
- A return value of 3 means the transition has been failed.

```
demo@demo-OMEN-by-HP-Laptop-16-b0xxx:~$ curl http://172.168.1.10:8010/command/?get_mode_status
3,2,2,0demo@demo-OMEN-by-HP-Laptop-16-b0xxx:~$
```

5.5.4.11 get_multiple_return

[Description]

Get the return mode of the LiDAR.

[Format]

```
curl http://<LIDAR_IP>:<LIDAR_PORT>/command/?get_mutliple_return
```

[Return value]

- A return value of 1 means the single return mode.
- A return value of 2 means the dual return mode (two strongest).
- A return value of 3 means the dual return mode (strongest and furthest).

```
demo@demo-OMEN-by-HP-Laptop-16-b0xxx:~$ curl http://172.168.1.10:8010/command/?get_multiple_return
1.000000demo@demo-OMEN-by-HP-Laptop-16-b0xxx:~$
```

5.5.4.12 get_reflectance_mode

[Description]

Get the reflectance mode of the LiDAR.

[Format]

```
curl http://<LIDAR_IP>:<LIDAR_PORT>/command/?get_reflectance_mode
```

[Return value]

- A return value of 1 means the intensity mode.
- A return value of 2 means the reflectivity mode.

```
demo@demo-OMEN-by-HP-Laptop-16-b0xxx:~$ curl http://172.168.1.10:8010/command/?get_reflectance_mode
2.000000demo@demo-OMEN-by-HP-Laptop-16-b0xxx:~$
```

5.5.4.13 get_reflectance_mode

[Description]

Get the reflectance mode of the LiDAR.

[Format]

```
curl http://<LIDAR_IP>:<LIDAR_PORT>/command/?get_reflectance_mode
```

[Return value]

- A return value of 1 means the intensity mode.
- A return value of 2 means the reflectivity mode.

```
demo@demo-OMEN-by-HP-Laptop-16-b0xxx:~$ curl http://172.168.1.10:8010/command/?get_reflectance_mode
2.000000demo@demo-OMEN-by-HP-Laptop-16-b0xxx:~$
```

5.5.4.14 get_return_mode

[Description]

Get the return mode of the LiDAR.

[Format]

```
curl http://<LIDAR_IP>:<LIDAR_PORT>/command/?get_return_mode
```

[Return value]

- A return value of 1 means the single return mode.
- A return value of 2 means the dual return mode (two strongest).
- A return value of 3 means the dual return mode (strongest and furthest).

```
demo@demo-OMEN-by-HP-Laptop-16-b0xxx:~$ curl http://172.168.1.10:8010/command/?get_return_mode
1.000000demo@demo-OMEN-by-HP-Laptop-16-b0xxx:~$
```

5.5.4.15 get_roi

[Description]

Get the center of ROI, including horizontal degree and vertical degree.

[Format]

```
curl http://<LIDAR_IP>:<LIDAR_PORT>/command/?get_roi
```

[Return value]

```
demo@demo-OMEN-by-HP-Laptop-16-b0xxx:~$ curl http://172.168.1.10:8010/command/?get_roi
0.000000,0.000000demo@demo-OMEN-by-HP-Laptop-16-b0xxx:~$
```

5.5.4.16 get_saved_roi_center_h

[Description]

Get the saved horizontal angle of ROI center.

[Format]

```
curl http://<LIDAR_IP>:<LIDAR_PORT>/command/?get_saved_roi_center_h
```

[Return value]

```
demo@demo-OMEN-by-HP-Laptop-16-b0xxx:~$ curl http://172.168.1.10:8010/command/?get_saved_roi_center_h
0.000000demo@demo-OMEN-by-HP-Laptop-16-b0xxx:~$
```

5.5.4.17 get_saved_roi_center_v

[Description]

Get the saved vertical angle of ROI center.

[Format]

```
curl http://<LIDAR_IP>:<LIDAR_PORT>/command/?get_saved_roi_center_v
```

[Return value]

```
demo@demo-OMEN-by-HP-Laptop-16-b0xxx:~$ curl http://172.168.1.10:8010/command/?get_saved_roi_center_v
0.000000demo@demo-OMEN-by-HP-Laptop-16-b0xxx:~$
```

5.5.4.18 get_saved_roi_enable

[Description]

Get if the ROI configuration is saved after the LiDAR reboot.

[Format]

```
curl http://<LIDAR_IP>:<LIDAR_PORT>/command/?get_saved_roi_enable
```

[Return value]

- A return value of 0 means the ROI configuration is not saved.
- A return value of 1 means the ROI configuration is saved.

```
demo@demo-OMEN-by-HP-Laptop-16-b0xxx:~$ curl http://172.168.1.10:8010/command/?get_saved_roi_enable
0.000000demo@demo-OMEN-by-HP-Laptop-16-b0xxx:~$
```

5.5.4.19 get_sdk_api_version

[Description]

Get the current SDK API version number.

[Format]

```
curl http://<LIDAR_IP>:<LIDAR_PORT>/command/?get_sdk_api_version
```

[Return value]

```
demo@demo-OMEN-by-HP-Laptop-16-b0xxx:~$ curl http://172.168.1.10:8010/command/?get_sdk_api_version
2.3.0.20230330020532demo@demo-OMEN-by-HP-Laptop-16-b0xxx:~$
```

5.5.4.20 get_sdk_build_tag

[Description]

Get the current SDK API build tag, usually the same as the SDK version number.

[Format]

```
curl http://<LIDAR_IP>:<LIDAR_PORT>/command/?get_sdk_build_tag
```

[Return value]

```
demo@demo-OMEN-by-HP-Laptop-16-b0xxx:~$ curl http://172.168.1.10:8010/command/?get_sdk_build_tag
release-2.11.0-fk-rc5-armdemo@demo-OMEN-by-HP-Laptop-16-b0xxx:~$
```

5.5.4.21 get_sdk_build_time

[Description]

Get the current SDK Build time and date.

[Format]

```
curl http://<LIDAR_IP>:<LIDAR_PORT>/command/?get_sdk_build_time
```

[Return value]

```
demo@demo-OMEN-by-HP-Laptop-16-b0xxx:~$ curl http://172.168.1.10:8010/command/?get_sdk_build_time
02:06:23 Mar 30 2023demo@demo-OMEN-by-HP-Laptop-16-b0xxx:~$
```

5.5.4.22 get_sdk_version

[Description]

Get the current SDK version number.

[Format]

```
curl http://<LIDAR_IP>:<LIDAR_PORT>/command/?get_sdk_version
```

[Return value]

```
demo@demo-OMEN-by-HP-Laptop-16-b0xxx:~$ curl http://172.168.1.10:8010/command/?get_sdk_version
release-2.11.0-fk-rc5-armdemo@demo-OMEN-by-HP-Laptop-16-b0xxx:~$
```

5.5.4.23 get_sn

[Description]

Get the sensor serial number.

[Format]

```
curl http://<LIDAR_IP>:<LIDAR_PORT>/command/?get_sn
```

[Return value]

```
demo@demo-OMEN-by-HP-Laptop-16-b0xxx:~$ curl http://172.168.1.10:8010/command/?get_sn
371922210720demo@demo-OMEN-by-HP-Laptop-16-b0xxx:~$
```

5.5.4.24 get_sw_version

[Description]

Get a conglomerate of information about SDK API version, API tag, API build time, SDK build time, SDK version, SDK tag.

[Format]

```
curl http://<LIDAR_IP>:<LIDAR_PORT>/command/?get_sw_version
```

[Return value]

```
demo@demo-OMEN-by-HP-Laptop-16-b0xxx:~$ curl http://172.168.1.10:8010/command/?get_sw_version
VERSION: 2.11.0-fk-rc5-arm-public
BUILD_TAG: release-2.11.0-fk-rc5-arm
BUILD_TIME: 02:08:54 Mar 30 2023
API: 2.3.0.20230330020532
API_BUILD_TAG: release-2.11.0-fk-rc5-arm
API_BUILD_TIME: 02:06:23 Mar 30 2023
```

5.5.4.25 get_time_sync_type

[Description]

Get current time synchronization type.

[Format]

```
curl http://<LIDAR_IP>:<LIDAR_PORT>/command/?get_time_sync_type
```

[Return value]

- A return value of 2 means the synchronized time source is the host PC.
- A return value of 3 means other time synchronization types have been switched to the beginning phase of the GPS.

- A return value of 4 means the GPS time synchronization is at a high confidence level.
- A return value of 5 means the GPS time synchronization is at a low confidence level.
- A return value of 6 means other time synchronization types have been switched to the beginning phase of the PTP.
- A return value of 7 means the PTP time synchronization is at a high confidence level.
- A return value of 8 means the PTP time synchronization is at a low confidence level.
- A return value of 9 means the playing file mode.
- A return value of 10 means other time synchronization types have been switched to the beginning phase of the NTP.
- A return value of 11 means the NTP time synchronization is at a high confidence level.
- A return value of 12 means the NTP time synchronization is at a low confidence level.
- A return value of 13 means the GTP time synchronization has been lost.
- A return value of 14 means the PTP time synchronization has been lost.
- A return value of 15 means the NTP time synchronization has been lost.

5.5.4.26 get_udp_ip

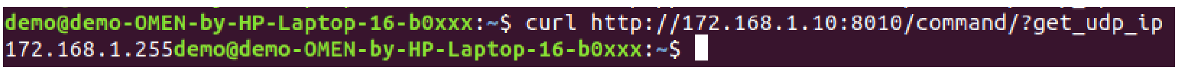
[Description]

Get the current UDP IP address.

[Format]

```
curl http://<LIDAR_IP>:<LIDAR_PORT>/command/?get_udp_ip
```

[Return value]



5.5.4.27 get_udp_ports_ip

[Description]

Get the current UDP port and IP address.

[Format]

```
curl http://<LIDAR_IP>:<LIDAR_PORT>/command/?get_udp_ports_ip
```

[Return value]

The return values are as follows in order.

- UDP data port #
- UDP status port #
- UDP message port #

- UDP client IP address
- UDP sender IP address, which is the IP address that submits the REST API request.

```
demo@demo-OMEN-by-HP-Laptop-16-b0xxx:~$ curl http://172.168.1.10:8010/command/?get_udp_ports_ip
8010,8010,8010,eth0,172.168.1.255demo@demo-OMEN-by-HP-Laptop-16-b0xxx:~$
```

5.5.5 Setter

5.5.5.1 set_clear_inner_fault

[Description]

Clear the specified inner faults.

[Format]

```
curl http://<LIDAR_IP>:<LIDAR_PORT>/command/?set_clear_inner_fault=<inner_fault_type>
```

[Parameter]

The range of inner_fault_type is from 0 to INNO_LIDAR_IN_FAULT_MAX. For the inner fault type, refer to InnoStatusInFaults.

Fault	Description
INNO_LIDAR_IN_FAULT_POWER_LOW=1	The battery undervoltage.
INNO_LIDAR_IN_FAULT_POWER_HIGH=2	The battery overvoltage.
INNO_LIDAR_IN_FAULT_WINDOW_BLOCKAGE1=3	The LiDAR window is obscured.
INNO_LIDAR_IN_FAULT_LASER_INTERLOCK=7	The laser is powering up. The cover is open while working. The polygon RPM is below the threshold.
INNO_LIDAR_IN_FAULT_COMM_LASER=8	UART communication failure between the laser and SoC: Checksum/Timeout.
INNO_LIDAR_IN_FAULT_LASER=9	Internal laser fault, such as overheating, pump overcurrent, trigger fault, and power fault.
INNO_LIDAR_IN_FAULT_COMM_DSP=10	UART/SPI communication failure between the DSP and SoC: Checksum/Timeout.
INNO_LIDAR_IN_FAULT_DSP =12	DSP chip fault, such as clock fault, memory fault, CPU fault, ADC fault, etc.
❖ INNO_LIDAR_IN_FAULT_POLYGON_CONTROL=13	• Motor control fault, such as varying rotation speed, motor blockage, U/V/W-phase current overcurrent and DC overvoltage. • Polygon driver fault, such as overcurrent, overheating, and performance failures.
INNO_LIDAR_IN_FAULT_POLYGON_SENSOR=14	• Polygon encoder fault, such as short circuit to the ground/ short circuit to the power/ open circuit of the signal lines, out of range/stuck/oscillated of the speed signal.

	• Faults related to the polygon hall.
INNO_LIDAR_IN_FAULT_GALVO_CONTROL=15	• Galvo motor control fault, such as varying rotation speed and Galvo blockage. • Galvo driver fault, such as overcurrent, overheating, and performance failure.
INNO_LIDAR_IN_FAULT_GALVO_SENSOR=16	Galvo encoder fault, such as short circuit to the ground/ short circuit to the power supply/ open circuit of the signal cable, inconsistency of dual output current signals.
INNO_LIDAR_IN_FAULT_OPTIC1=17	Refers to the open circuit of the optical path. (Such as the optical fiber disconnection of the reference optical path, detection board fault, etc.)
INNO_LIDAR_IN_FAULT_OPTIC2=18	The open circuit of the main optical path. (open circuit of the main optical path, detection board fault, etc.)
INNO_LIDAR_IN_FAULT_IIC_DSP=19	IIC communication failure between the DSP and peripherals (except SoC).
INNO_LIDAR_IN_FAULT_IIC_SoC=20	IIC communication failure between the SoC and peripherals (except DSP).
INNO_LIDAR_IN_FAULT_COMM_ADC=25	Communication failure between the ADC chip and SoC: SPI and JESD204C.
INNO_LIDAR_IN_FAULT_SoC=27	SoC chip fault, such as clock fault, memory fault, CPU fault, etc.
INNO_LIDAR_IN_FAULT_SOC_EXTWD=28	SoC external watchdog fault
INNO_LIDAR_IN_FAULT_RAWDATA_STREAM=29	Raw data order error, such as there is no Galvo encoder between two polygon encoder signals, etc.
INNO_LIDAR_IN_FAULT_POLYGON_TO=30	Polygon encoder signal timeout.
INNO_LIDAR_IN_FAULT_GALVO_TO=31	Galvo encoder signal timeout.
INNO_LIDAR_IN_FAULT_TRIGGER_TO=32	Laser trigger signal timeout.
INNO_LIDAR_IN_FAULT_POWSUPL1=33	DSP Safe PMIC chip fault.
INNO_LIDAR_IN_FAULT_POWSUPL2=34	SoC PMIC chip fault.
INNO_LIDAR_IN_FAULT_LPDDR4=35	LPDDR4 chip fault, such as reading/writing data errors, data damage.
INNO_LIDAR_IN_FAULT_FLASH=36	FLASH fault, such as reading/writing data errors, data damage, and SPI communication fault.
INNO_LIDAR_IN_FAULT_NETWORK1=37	Ethernet receiving message fault, PTP timeout, TCP request CRC fault
INNO_LIDAR_IN_FAULT_NETWORK2=38	EPHY chip fault.
INNO_LIDAR_IN_FAULT_OVERHEAT1=39	The laser temperature rose to the 1-level threshold

	of 95℃ (203°F).
INNO_LIDAR_IN_FAULT_OVERHEAT2=40	The laser temperature rose to the 2-level threshold of 99℃ (210.2°F).
INNO_LIDAR_IN_FAULT_OVERHEAT3=41	The laser temperature rose to the 3-level threshold of 100℃ (212°F).
INNO_LIDAR_IN_FAULT_CONFIG1=42	Software configuration data error, such as calibration data fault, upper layer application data fault and UDS configuration data fault.
INNO_LIDAR_IN_FAULT_CONFIG2=43	Firmware configuration data error.
INNO_LIDAR_IN_FAULT_ASSERT_FAILURE=44	Software assertion error.
INNO_LIDAR_IN_FAULT_CPULOAD_HIGH=45	Excessive CPU load.
INNO_LIDAR_IN_FAULT_LATENCY_LONG=46	Excessive system operation delay.
INNO_LIDAR_IN_FAULT_RESERVED07=47	Active data throwing in the Signal stage.
INNO_LIDAR_IN_FAULT_EXCESSIVE_NOISE=49	Excessive noise is found in the software.
INNO_LIDAR_IN_FAULT_DATA_DROP1 50	Software throws data actively in the noise filtering stage.
INNO_LIDAR_IN_FAULT_DATA_DROP2=51	Software throws data actively in the point cloud sending stage.
INNO_LIDAR_IN_FAULT_DATA_DROP3=52	Throw data actively in the SoC PL processing data stage.
INNO_LIDAR_TEMPHIGH_INHIBIT=53	LiDAR is in protection mode.
INNO_LIDAR_IN_FAULT_REFINTENSITY=55	Refers to abnormal light intensity.

[Return value]

```
demo@demo-OMEN-by-HP-Laptop-16-b0xxx:~$ curl http://172.168.1.10:8010/command/?get_inner_fault
s
[{"ctr":2,"fault_status":139,"fid":3,"name":"INNO_LIDAR_IN_FAULT_WINDOW_BLOCKAGE1","no_fault_c
ycle_so_far":0,"ts":"15-c-13-13-2f-c"},{"ctr":3,"fault_status":139,"fid":4,"name":"INNO_LIDAR_
demo@demo-OMEN-by-HP-Laptop-16-b0xxx:~$ curl http://172.168.1.10:8010/command/?set_clear_inner
_fault=3
demo@demo-OMEN-by-HP-Laptop-16-b0xxx:~$ curl http://172.168.1.10:8010/command/?get_inner_fault
demo@demo-OMEN-by-HP-Laptop-16-b0xxx:~$
```

5.5.5.2 set_clear_inner_faults

[Description]

Clear all the inner faults.

[Format]

```
curl http://<LIDAR_IP>:<LIDAR_PORT>/command/?set_clear_inner_faults
```

[Parameter]

The range of inner_fault_type is from 0 to INNO_LIDAR_IN_FAULT_MAX. For the inner fault type, refer to InnoStatusInFaults.

Fault	Description
INNO_LIDAR_IN_FAULT_POWER_LOW=1	The battery undervoltage.
INNO_LIDAR_IN_FAULT_POWER_HIGH=2	The battery overvoltage.
INNO_LIDAR_IN_FAULT_WINDOW_BLOCKAGE1=3	The LiDAR window is obscured.
INNO_LIDAR_IN_FAULT_LASER_INTERLOCK=7	The laser is powering up. The cover is open while working. The polygon RPM is below the threshold.
INNO_LIDAR_IN_FAULT_COMM_LASER=8	UART communication failure between the laser and SoC: Checksum/Timeout.
INNO_LIDAR_IN_FAULT_LASER=9	Internal laser fault, such as overheating, pump overcurrent, trigger fault, and power fault.
INNO_LIDAR_IN_FAULT_COMM_DSP=10	UART/SPI communication failure between the DSP and SoC: Checksum/Timeout.
INNO_LIDAR_IN_FAULT_DSP =12	DSP chip fault, such as clock fault, memory fault, CPU fault, ADC fault, etc.
❖ INNO_LIDAR_IN_FAULT_POLYGON_CONTROL=13	• Motor control fault, such as varying rotation speed, motor blockage, U/V/W-phase current overcurrent and DC overvoltage. • Polygon driver fault, such as overcurrent, overheating, and performance failures.
INNO_LIDAR_IN_FAULT_POLYGON_SENSOR=14	• Polygon encoder fault, such as short circuit to the ground/ short circuit to the power/ open circuit of the signal lines, out of range/stuck/oscillated of the speed signal. • Faults related to the polygon hall.
INNO_LIDAR_IN_FAULT_GALVO_CONTROL=15	• Galvo motor control fault, such as varying rotation speed and Galvo blockage. • Galvo driver fault, such as overcurrent, overheating, and performance failure.
INNO_LIDAR_IN_FAULT_GALVO_SENSOR=16	Galvo encoder fault, such as short circuit to the ground/ short circuit to the power supply/ open circuit of the signal cable, inconsistency of dual output current signals.
INNO_LIDAR_IN_FAULT_OPTIC1=17	Refers to the open circuit of the optical path. (Such as the optical fiber disconnection of the reference optical path, detection board fault, etc.)
INNO_LIDAR_IN_FAULT_OPTIC2=18	The open circuit of the main optical path. (open circuit of the main optical path, detection board fault, etc.)

INNO_LIDAR_IN_FAULT_IIC_DSP=19	IIC communication failure between the DSP and peripherals (except SoC).
INNO_LIDAR_IN_FAULT_IIC_SoC=20	IIC communication failure between the SoC and peripherals (except DSP).
INNO_LIDAR_IN_FAULT_COMM_ADC=25	Communication failure between the ADC chip and SoC: SPI and JESD204C.
INNO_LIDAR_IN_FAULT_SoC=27	SoC chip fault, such as clock fault, memory fault, CPU fault, etc.
INNO_LIDAR_IN_FAULT_SOC_EXTWD=28	SoC external watchdog fault
INNO_LIDAR_IN_FAULT_RAWDATA_STREAM=29	Raw data order error, such as there is no Galvo encoder between two polygon encoder signals, etc.
INNO_LIDAR_IN_FAULT_POLYGON_TO=30	Polygon encoder signal timeout.
INNO_LIDAR_IN_FAULT_GALVO_TO=31	Galvo encoder signal timeout.
INNO_LIDAR_IN_FAULT_TRIGGER_TO=32	Laser trigger signal timeout.
INNO_LIDAR_IN_FAULT_POWSUPL1=33	DSP Safe PMIC chip fault.
INNO_LIDAR_IN_FAULT_POWSUPL2=34	SoC PMIC chip fault.
INNO_LIDAR_IN_FAULT_LPDDR4=35	LPDDR4 chip fault, such as reading/writing data errors, data damage.
INNO_LIDAR_IN_FAULT_FLASH=36	FLASH fault, such as reading/writing data errors, data damage, and SPI communication fault.
INNO_LIDAR_IN_FAULT_NETWORK1=37	Ethernet receiving message fault, PTP timeout, TCP request CRC fault
INNO_LIDAR_IN_FAULT_NETWORK2=38	EPHY chip fault.
INNO_LIDAR_IN_FAULT_OVERHEAT1=39	The laser temperature rose to the 1-level threshold of 95℃ (203°F).
INNO_LIDAR_IN_FAULT_OVERHEAT2=40	The laser temperature rose to the 2-level threshold of 99℃ (210.2°F).
INNO_LIDAR_IN_FAULT_OVERHEAT3=41	The laser temperature rose to the 3-level threshold of 100℃ (212°F).
INNO_LIDAR_IN_FAULT_CONFIG1=42	Software configuration data error, such as calibration data fault, upper layer application data fault and UDS configuration data fault.
INNO_LIDAR_IN_FAULT_CONFIG2=43	Firmware configuration data error.
INNO_LIDAR_IN_FAULT_ASSERT_FAILURE=44	Software assertion error.
INNO_LIDAR_IN_FAULT_CPULOAD_HIGH=45	Excessive CPU load.
INNO_LIDAR_IN_FAULT_LATENCY_LONG=46	Excessive system operation delay.
INNO_LIDAR_IN_FAULT_RESERVED07=47	Active data throwing in the Signal stage.
INNO_LIDAR_IN_FAULT_EXCESSIVE_NOISE=4	Excessive noise is found in the software.

9	
INNO_LIDAR_IN_FAULT_DATA_DROP1 50	Software throws data actively in the noise filtering stage.
INNO_LIDAR_IN_FAULT_DATA_DROP2=51	Software throws data actively in the point cloud sending stage.
INNO_LIDAR_IN_FAULT_DATA_DROP3=52	Throw data actively in the SoC PL processing data stage.
INNO_LIDAR_TEMPHIGH_INHIBIT=53	LiDAR is in protection mode.
INNO_LIDAR_IN_FAULT_REFINTENSITY=55	Refers to abnormal light intensity.

[Return value]

```
demo@demo-OMEN-by-HP-Laptop-16-b0xxx:~$ curl http://172.168.1.10:8010/command/?get_inner_faults_info
[{"ctr":5,"fault_status":8,"fid":3,"name":"INNO_LIDAR_IN_FAULT_WINDOW_BLOCKAGE1","no_fault_cycle_so_far":
0,"ts":"15-c-13-13-2f-c"},{"ctr":7,"fault_status":139,"fid":4,"name":"INNO_LIDAR_IN_FAULT_WINDOW_BLOCKAGE
2","no_fault_cycle_so_far":0,"ts":"15-c-13-13-2f-c"},{"ctr":2,"fault_status":8,"fid":37,"name":"INNO_LIDA
R_IN_FAULT_NETWORK1","no_fault_cycle_so_far":0,"ts":"15-c-1a-16-d-36"},{"ctr":0,"fault_status":8,"fid":38
,"name":"INNO_LIDAR_IN_FAULT_NETWORK2","no_fault_cycle_so_far":0,"ts":"15-c-13-13-2f-a"},{"ctr":0,"fault_
status":8,"fid":44,"name":"INNO_LIDAR_IN_FAULT_ASSERT_FAILURE","no_fault_cycle_so_far":0,"ts":"15-c-14-11
-6-8"},{"ctr":15,"fault_status":8,"fid":45,"name":"INNO_LIDAR_IN_FAULT_CPULOAD_HIGH","no_fault_cycle_so_f
ar":0,"ts":"15-c-14-15-11-15"},{"ctr":0,"fault_status":8,"fid":49,"name":"INNO_LIDAR_IN_FAULT_EXCESSIVE_N
OISE","no_fault_cycle_so_far":0,"ts":"15-c-13-14-d-a"},{"ctr":0,"fault_status":8,"fid":52,"name":"INNO_LI
DAR_IN_FAULT_DATA_DROP3","no_fault_cycle_so_far":0,"ts":"15-c-13-14-20-a"},{"ctr":0,"fault_status":8,"fid
":55,"name":"INNO_LIDAR_IN_FAULT_REFINTENSITY","no_fault_cycle_so_far":0,"ts":"15-c-13-13-2f-a"}]demo@dem
o-OMEN-by-HP-Laptop-16-b0xxx:~$ curl http://172.168.1.10:8010/command/?set_clear_inner_faults
demo@demo-OMEN-by-HP-Laptop-16-b0xxx:~$ curl http://172.168.1.10:8010/command/?get_inner_faults_info
[{"ctr":1,"fault_status":131,"fid":4,"name":"INNO_LIDAR_IN_FAULT_WINDOW_BLOCKAGE2","no_fault_cycle_so_far
":0,"ts":"15-c-16-2-1a-18"}]demo@demo-OMEN-by-HP-Laptop-16-b0xxx:~$
```

5.5.5.3 set_frame_sync

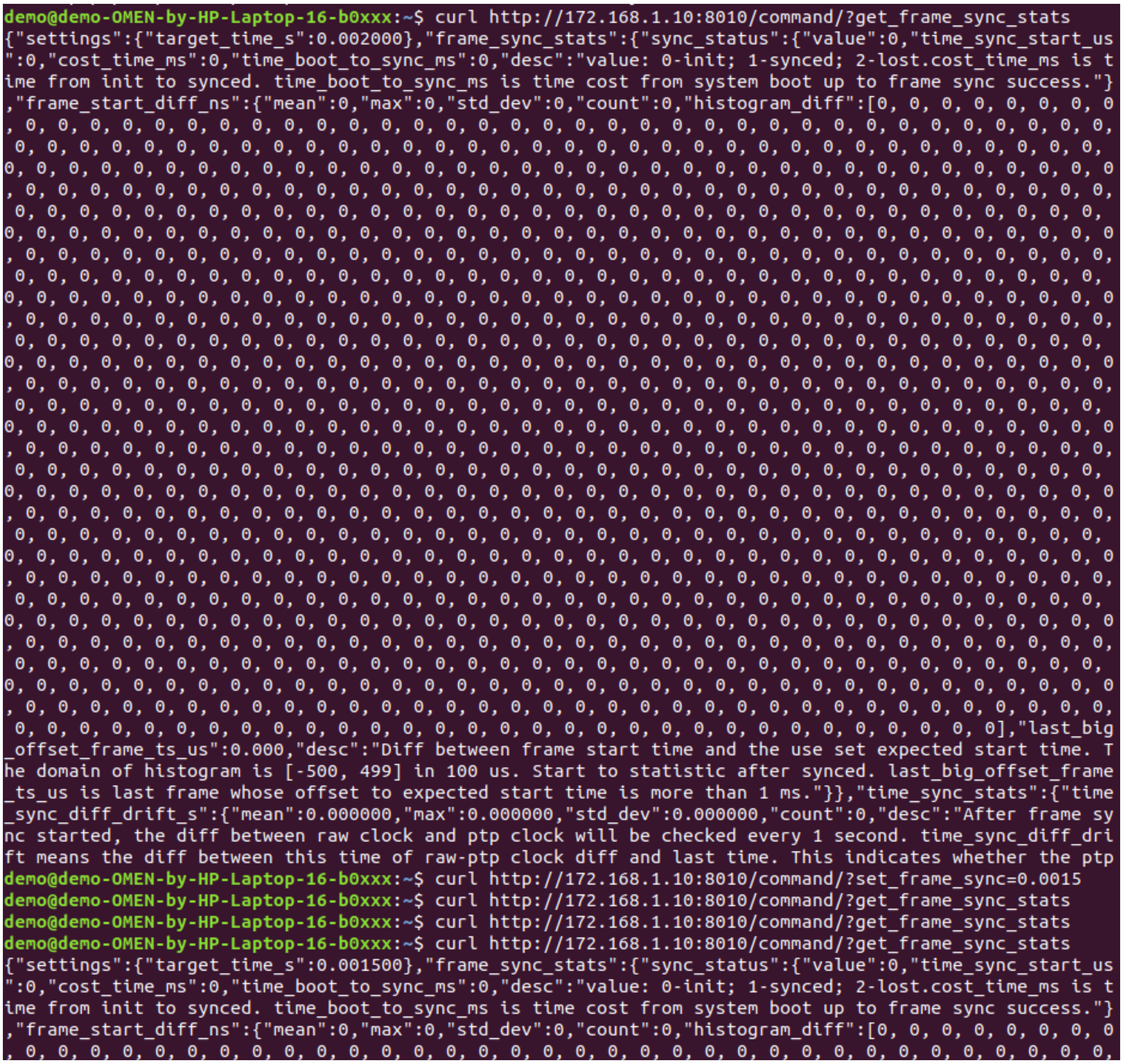
[Description]

Set the target value. Target value is recorded in seconds, ranging from 0 to 0.099.

[Format]

```
curl http://<LIDAR_IP>:<LIDAR_PORT>/command/?set_frame_sync=<frame_sync_offset>
```

[Return value]



```
demo@demo-OMEN-by-HP-Laptop-16-b0xxx:~$ curl http://172.168.1.10:8010/command/?get_frame_sync_stats
{"settings":{"target_time_s":0.002000},"frame_sync_stats":{"sync_status":{"value":0,"time_sync_start_us
":0,"cost_time_ms":0,"time_boot_to_sync_ms":0,"desc":"value: 0-init; 1-synced; 2-lost.cost_time_ms is t
ime from init to synced. time_boot_to_sync_ms is time cost from system boot up to frame sync success."}
,"frame_start_diff_ns":{"mean":0,"max":0,"std_dev":0,"count":0,"histogram_diff":[0, 0, 0, 0, 0, 0, 0, 0
, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0
, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0
, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0
, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0
, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0
, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0
, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0
, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0
, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0],"last_big
_offset_frame_ts_us":0.000,"desc":"Diff between frame start time and the use set expected start time. T
he domain of histogram is [-500, 499] in 100 us. Start to statistic after synced. last_big_offset_frame
_ts_us is last frame whose offset to expected start time is more than 1 ms."}},"time_sync_stats":{"time
_sync_diff_drift_s":{"mean":0.000000,"max":0.000000,"std_dev":0.000000,"count":0,"desc":"After frame sy
nc started, the diff between raw clock and ptp clock will be checked every 1 second. time_sync_diff_dri
ft means the diff between this time of raw-ptp clock diff and last time. This indicates whether the ptp
demo@demo-OMEN-by-HP-Laptop-16-b0xxx:~$ curl http://172.168.1.10:8010/command/?set_frame_sync=0.0015
demo@demo-OMEN-by-HP-Laptop-16-b0xxx:~$ curl http://172.168.1.10:8010/command/?get_frame_sync_stats
demo@demo-OMEN-by-HP-Laptop-16-b0xxx:~$ curl http://172.168.1.10:8010/command/?get_frame_sync_stats
demo@demo-OMEN-by-HP-Laptop-16-b0xxx:~$ curl http://172.168.1.10:8010/command/?get_frame_sync_stats
{"settings":{"target_time_s":0.001500},"frame_sync_stats":{"sync_status":{"value":0,"time_sync_start_us
":0,"cost_time_ms":0,"time_boot_to_sync_ms":0,"desc":"value: 0-init; 1-synced; 2-lost.cost_time_ms is t
ime from init to synced. time_boot_to_sync_ms is time cost from system boot up to frame sync success."}
,"frame_start_diff_ns":{"mean":0,"max":0,"std_dev":0,"count":0,"histogram_diff":[0, 0, 0, 0, 0, 0, 0, 0
, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
```

5.5.5.4 set_mode

[Description]

Set the Falcon to the corresponding working mode.

[Format]

```
curl http://<LIDAR_IP>:<LIDAR_PORT>/command/?set_mode=<working_mode>
```

[Parameter]

Value	Description
1	Sleep mode
2	Standby mode
3	Normal mode
5	Calibration mode
6	Protection mode

7	Quiet mode

[Return value]

```
demo@demo-OMEN-by-HP-Laptop-16-b0xxx:~$ curl http://172.168.1.10:8010/command/?set_mode=5
demo@demo-OMEN-by-HP-Laptop-16-b0xxx:~$ curl http://172.168.1.10:8010/command/?set_mode=3
demo@demo-OMEN-by-HP-Laptop-16-b0xxx:~$ curl http://172.168.1.10:8010/command/?get_mode
3,5,2,0demo@demo-OMEN-by-HP-Laptop-16-b0xxx:~$
```

5.5.5.5 set_reboot

[Description]

Reboot the LiDAR.

[Format]

```
curl http://<LIDAR_IP>:<LIDAR_PORT>/command/?set_reboot=<reboot_status>
```

[Parameter]

Value	Description
0	Do not reboot the LiDAR.
1	Reboot the LiDAR.

5.5.5.6 set_reflectance_mode

[Description]

Set the reflectance mode.

[Format]

```
curl http://<LIDAR_IP>:<LIDAR_PORT>/command/?set_reflectance_mode=<reflectance_mode>
```

[Parameter]

Value	Description
1	Intensity
2	Reflectivity

[Return value]

```
demo@demo-OMEN-by-HP-Laptop-16-b0xxx:~$ curl http://172.168.1.10:8010/command/?get_reflectance
_mode
2.000000demo@demo-OMEN-by-HP-Laptop-16-b0xxx:~$ curl http://172.168.1.10:8010/command/?set_ref
lectance_mode=1
demo@demo-OMEN-by-HP-Laptop-16-b0xxx:~$ curl http://172.168.1.10:8010/command/?get_reflectance
_mode
1.000000demo@demo-OMEN-by-HP-Laptop-16-b0xxx:~$
```

5.5.5.7 set_return_mode

[Description]

Set the return mode.

[Format]

```
curl http://<LIDAR_IP>:<LIDAR_PORT>/command/?set_return_mode=<return_mode>
```

[Parameter]

Value	Description
1	Single
2	2 strongest
3	Strongest & furthest

[Return value]

```
demo@demo-OMEN-by-HP-Laptop-16-b0xxx:~$ curl http://172.168.1.10:8010/command/?get_return_mode
1.000000demo@demo-OMEN-by-HP-Laptop-16-b0xxx:~$ curl http://172.168.1.10:8010/command/?set_return_mode=2
demo@demo-OMEN-by-HP-Laptop-16-b0xxx:~$ curl http://172.168.1.10:8010/command/?get_return_mode
2.000000
```

5.5.5.8 set_roi

[Description]

Set ROI horizontal degree offset and vertical degree offset.

[Format]

```
curl         http://<LIDAR_IP>:<LIDAR_PORT>/command/?set_roi=<ROI_horizontal_degree_offset>
<ROI_vertical_degree_offset>
```

[Parameter]

- The range of ROI horizontal degree offset is from -60 to 60.
- The range of ROI horizontal degree offset is from -20 to 20.

[Return value]

```
demo@demo-OMEN-by-HP-Laptop-16-b0xxx:~$ curl http://172.168.1.10:8010/command/?set_roi=3.3
demo@demo-OMEN-by-HP-Laptop-16-b0xxx:~$ curl http://172.168.1.10:8010/command/?set_roi=3,3
demo@demo-OMEN-by-HP-Laptop-16-b0xxx:~$ curl http://172.168.1.10:8010/command/?get_roi
3.000000,3.000000demo@demo-OMEN-by-HP-Laptop-16-b0xxx:~$
```

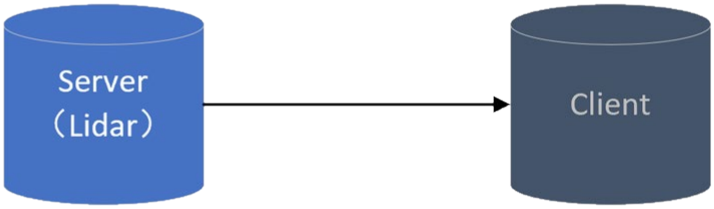
6 Communication protocol

You can obtain the point cloud data via TCP or UDP. This chapter describes how to transmit data between the server and the client and obtain LiDAR information.

6.1 Data transmission via TCP

6.1.1 Data transmission methodology

TCP (Transmission Control Protocol) is a connection-oriented unicast transmission protocol. The LiDAR serves as the server, and the client serves as the client in TCP. The client initiates a connection request to the server. After a reliable connection is established, the server will actively send data to the client.



6.1.2 Obtain point cloud data

You can obtain the point cloud data of the LiDAR via inno_pc_client.

- ❖ inno_pc_client is an executable file in the SDK file of LiDAR, which can be used to obtain LiDAR point cloud data. Please notice that inno_pc_client adopts TCP protocol by default.

1. Unzip SDK files.

```
tar -xzvf ~/<package.tgz>
```

Note

Please obtain the latest version of the SDK file according to the actual situation of the system.

2. Enter the inno_pc_client path.

```
cd ~/apps/pcs/
```

3. Obtain point cloud data.

```
./inno_pc_client --lidar-ip <INPUT_LIDAR_IP> --lidar-port <INPUT_LIDAR_TCP_PORT>
```

Note

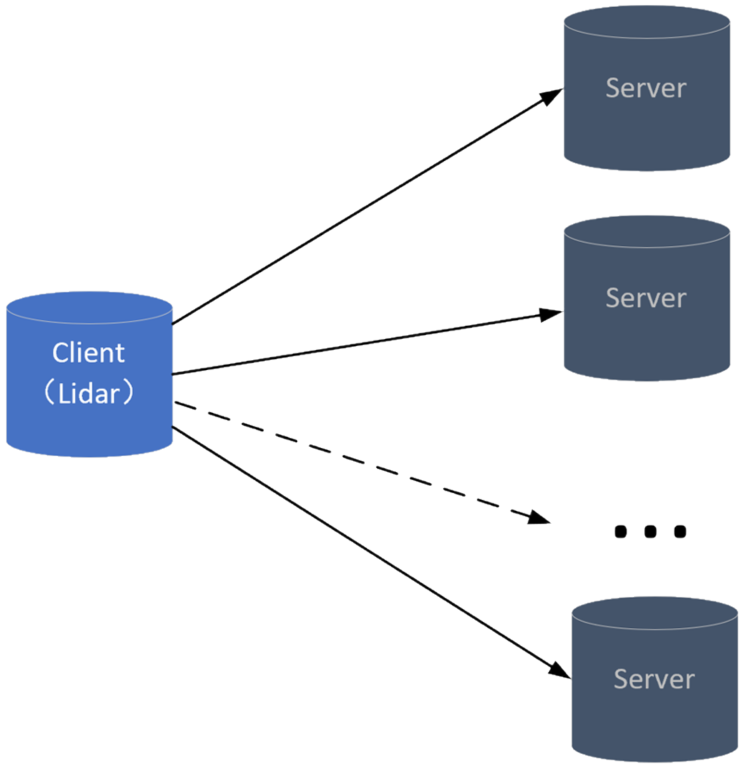
The default IP address of the LiDAR is 172.168.1.10, and the default port of the LiDAR is 8002.

6.2 Data transmission via UDP

6.2.1 Data transmission methodology

The LiDAR serves as the client, and the customer serves as the server in UDP (User Datagram Protocol). The client actively sends data to one or more servers.

Since UDP may lead to packet loss when transmitting mass data, it is only recommended to transmit a small amount of data. The connection diagram is shown in the figure below.



6.2.2 Change transmission mode

You can change the data transmission mode according to the actual situation in UDP.

1. Unzip SDK files.

```
tar -xzvf ~/<package.tgz>
```

Note

Please obtain the latest SDK file according to the actual situation of the system.

2. Enter the innovusion_lidar_util directory.

```
cd ~/apps/lidar_util/innovusion_lidar_util
```

3. Execute the following command to download the PCS_ENV file.

```
./innovusion_lidar_util lidar_ip download_internal_file PCS_ENV ./PCS_ENV
```

4. Open the PCS_ENV file and change UDP_IP as required.

```
TCP_SERVICE_PORT=8010
UDP_IP=eth0
UDP_PORT_DATA=8010
UDP_PORT_MESSAGE=8010
UDP_PORT_STATUS=8010
UDP_PORT_STATUS_LOCAL=8009
STATUS_INTERVAL_MS=50
REFLECTANCE=2
MULTIRETURN=1
LOG_OPTION="--log-filename /tmp/inno_pc_server.txt --log-file-rotate-
number 3 --log-file-max-size-k 2000"
MIN_RUN_TIME=5
MIN_RUN_TIME_SLEEP=5
```

- Broadcast mode

  UDP_IP=eth0. eth0 is the default value of the UDP IP. The device can obtain the point cloud data of the LiDAR via SDK in broadcast mode.
- Multicast mode

  UDP_IP=<Multicast_Address>. In multicast mode, the LiDAR sends data to all devices in the same subnet, and all devices in the subnet can obtain the point cloud data of the LiDAR via SDK.
- UDP unicast mode

  UDP_IP=<Device_IP>. <Device IP> is the IP address of the computer, which is in the same subnet with the LiDAR. In unicast mode, only the server can obtain the point cloud data through UDP mode.
- UDP data transmission channel is closed

  After the UDP_IP information is commented out, LiDAR does not actively send data through UDP.

  If the client needs to actively obtain data in UDP, UDP_IP should be the IP address of the client, and the transmission mode is unicast.

5. Upload the PCS_ENV file.

```
./innovusion_lidar_util lidar_ip upload_internal_file PCS_ENV ./PCS_ENV
```

6.2.3 Obtain point cloud data

You can obtain the point cloud data of the LiDAR via inno_pc_client.

❖ inno_pc_client is an executable file in the SDK file of LiDAR, which can be used to obtain LiDAR point cloud data.

1. Unzip SDK files.

```
tar -xzvf ~/<package.tgz>
```

Note

Please obtain the latest SDK file according to the actual situation of the system.

2. Enter the inno_pc_client directory.

```
cd ~/apps/pcs/
```

3. Obtain point cloud data.

```
./inno_pc_client --lidar-ip <INPUT_LIDAR_IP> --lidar-udp-port <INPUT_LIDAR_UDP_PORT>
```

Note

The default LiDAR IP address is 172.168.1.10. The port is 8010 by default.

7 Time synchronization

The point cloud data has associated timestamp information due to the measurement feature of LiDAR. Time information management is essential to facilitate device debugging, multi-sensor fusion, operation, maintenance, and other functions during the use of LiDAR in the network environment. The mainstream time synchronization methods include GPS, NTP, and PTP (including gPTP). Currently, Falcon supports PTP, gPTP and NTP as their time synchronization method.

Note

If you want to adopt NTP, please contact Innovision staff for technical support.

7.1 PTP time synchronization

7.1.1 PTP time synchronization introduction

Precision Time Protocol (PTP) is divided into Hardware timestamp synchronization (precision at sub-microsecond), Software timestamp synchronization (precision at tens of microseconds).

Table 25 Time Synchronization Method Instructions

Synchronization method		Synchronization principle	Synchronization precision
PTP	Hardware	Hardware timestamp synchronization	Sub microsecond
	Software	Software timestamp synchronization	Tens of microseconds

Note

Before sync time in PTP mode, you need to ensure the following conditions.

- The network between the computer and the Falcon is connected.
- You have obtained the innovusion_lidar_util tool.

7.1.2 Check the PTP time synchronization capability of the computer

Check if the computer has the capability of PTP time synchronization.

```
ethtool -T <network interface>
```

```
demo@demo-Zephyrus-M15-GU502LW-GU502LW:~$ ethtool -T eno2
Time stamping parameters for eno2:
Capabilities:
        software-transmit     (SOF_TIMESTAMPING_TX_SOFTWARE)
        software-receive      (SOF_TIMESTAMPING_RX_SOFTWARE)
        software-system-clock (SOF_TIMESTAMPING_SOFTWARE)
PTP Hardware Clock: none
Hardware Transmit Timestamp Modes: none
Hardware Receive Filter Modes: none
demo@demo-Zephyrus-M15-GU502LW-GU502LW:~$
```

Note

- You can execute the **sudo apt-get install** command to download and install the ethtool.
- <network interface> is the name of the physical network interface used for PTP time synchronization. You can obtain all physical network interfaces by the **ifconfig** command.

The following output should be contained if the computer supports hardware timestamp.

```
SOF_TIMESTAMPING_RAW_HARDWARE
SOF_TIMESTAMPING_TX_HARDWARE
SOF_TIMESTAMPING_RX_HARDWARE
```

The following output should be contained if the computer supports software timestamp.

```
SOF_TIMESTAMPING_SOFTWARE
SOF_TIMESTAMPING_TX_SOFTWARE
SOF_TIMESTAMPING_RX_SOFTWARE
```

7.1.3 Sync time based on PTP software timestamp

7.1.3.1 Install linuxptp

Install **linuxptp** service.

```
sudo apt install linuxptp
```

7.1.3.2 Configure ptp4l.service

1. Edit **ptp4l.service**.

```
sudo gedit /lib/systemd/system/ptp4l.service
```

2. Change the following content in **ptp4l.service**.

```
ExecStart=/usr/sbin/ptp4l -f /etc/linuxptp/ptp4l.conf -I eth2 -S -A
```

Table 26 Parameter descriptions

Parameter	Description
-S	Software timestamp. A hardware timestamp is used by default if there is no such option. This option must be added if the network interface does not support hardware timestamps.
-i	The physical interface for PTP time synchronization of the computer.
-P	P2P (peer-to-peer) delay measurement mechanism. P2P enables more precise delay measurement. However, P2P can only be used for specific

	network topologies. For example, each port can only exchange PTP messages with other ports on a one-to-one basis and all hardware (including transparent clocks) on the communication path must support the P2P mode. Therefore, this option is only recommended for practical deployments when you know a lot about the network topology.
-A	This option enables the automatic selection of the delay mechanism. Under this auto option, ptp4l runs in E2E mode. A peer delay request will automatically switch to P2P mode if it can be received.
-E	The E2E (end-to-end) delay measurement mechanism. It is the default and recommended option. The E2E mechanism is also called the "request-response" delay mechanism.

7.1.3.3 Configure ptp4l.conf

1. Enter **ptp4l.conf**.

```
sudo gedit /etc/linuxptp/ptp4l.conf     // Configure ptp4l.conf.
```

2. Change the following content in **ptp4l.conf**.

```
Change clock_servo pi to clock_servo linreg.
Change priority1 128 to priority1 127.
```

3. Save changes and exit.

7.1.3.4 Reload and restart the ptp41 service

```
sudo systemctl daemon-reload
sudo systemctl enable ptp4l.service
sudo systemctl restart ptp4l.service
```

7.1.3.5 Check the status of ptp4l service

Execute the following command to confirm the ptp4l.serive is running and 'assuming the grand master role' is displayed. Currently, the computer is in the grandmaster state (PTP master state) in this network environment.

```
systemctl status ptp4l.service
```

```
hikai@hikai-Default-string:~$ systemctl status ptp4l.service
● ptp4l.service - Precision Time Protocol (PTP) service
   Loaded: loaded (/lib/systemd/system/ptp4l.service; enabled; vendor preset: enabled)
   Active: active (running) since Thu 2021-11-11 15:47:11 CST; 1 months 11 days ago
     Docs: man:ptp4l
 Main PID: 1073 (ptp4l)
    Tasks: 1 (limit: 4915)
   CGroup: /system.slice/ptp4l.service
           └─1073 /usr/sbin/ptp4l -E -i enp2s0f0 -m

12月 23 11:14:07 hikai-Default-string ptp4l[1073]: [3612156.706] driver changed our HWTSTAMP options
12月 23 11:14:07 hikai-Default-string ptp4l[1073]: [3612156.706] tx_type   1 not 1
12月 23 11:14:07 hikai-Default-string ptp4l[1073]: [3612156.706] rx_filter 1 not 12
12月 23 11:14:07 hikai-Default-string ptp4l[1073]: [3612156.706] port 1: FAULTY to LISTENING on FAULT_CLEARED
12月 23 11:14:13 hikai-Default-string ptp4l[1073]: ptp4l[3612162.774]: port 1: LISTENING to MASTER on ANNOUNCE_RECEIPT_TIMEOUT_EXPIRES
12月 23 11:14:13 hikai-Default-string ptp4l[1073]: ptp4l[3612162.774]: selected best master clock 6cb311.fffe.347004
12月 23 11:14:13 hikai-Default-string ptp4l[1073]: ptp4l[3612162.774]: assuming the grand master role
12月 23 11:14:13 hikai-Default-string ptp4l[1073]: [3612162.774] port 1: LISTENING to MASTER on ANNOUNCE_RECEIPT_TIMEOUT_EXPIRES
12月 23 11:14:13 hikai-Default-string ptp4l[1073]: [3612162.774] selected best master clock 6cb311.fffe.347004
12月 23 11:14:13 hikai-Default-string ptp4l[1073]: [3612162.774] assuming the grand master role
hikai@hikai-Default-string:~$
```

7.1.3.6 Enable PTP time synchronization of the LiDAR

1. Enable the PTP time synchronization.

When ptp_en=1, the PTP f time synchronization is enabled.

When ptp_en=0, the PTP time synchronization is disabled.

```
./innovusion_lidar_util <LiDAR_IP> set_config time ptp_en 1
```

2. Set the PTP time synchronization customized mode.

   When ptp_automotive =0, the PTP time synchronization is in customized mode.

   When ptp_automotive =1, the PTP time synchronization is in gPTP mode.

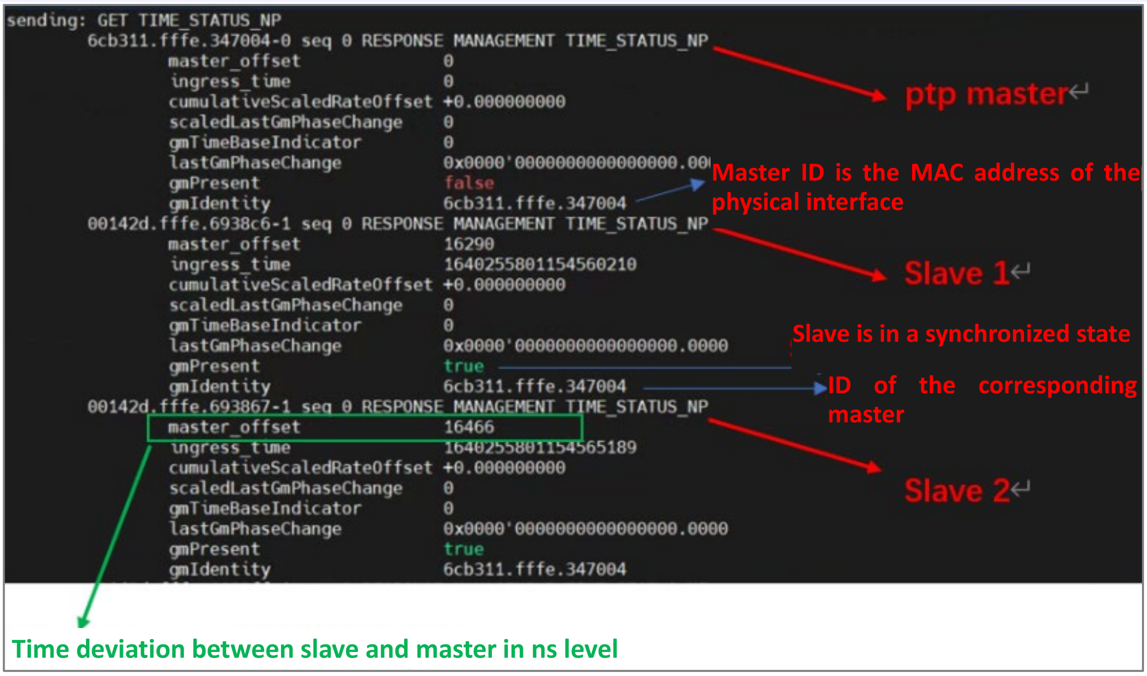
```
./innovusion_lidar_util <LiDAR_IP> set_config time ptp_automotive 0
```

7.1.3.7 Check the PTP status

Method 1

You can use the PMC tool to check the time deviation after time synchronization. The precision of the PMC query is ns level.

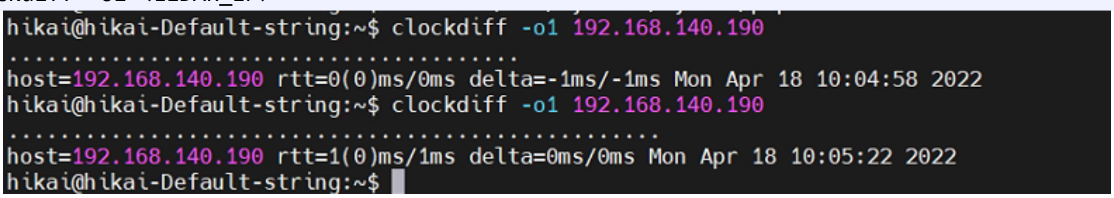
```
sudo pmc -u -b 1  'get time_status_np'
```



Method 2

You can use the clockdiff tool to check the time deviation after time synchronization. The precision of clockdiff query is ms level.

```
clockdiff -o1 <LiDAR_IP>
```



Method 3

You can download LiDAR logs from the ILA page to check the correct time.

7.1.4 Sync time based on PTP hardware timestamp

7.1.4.1 Install linuxptp

Install **linuxptp service**.

```
sudo apt install linuxptp
```

7.1.4.2 Configure ptp4l.service

1. Enter **ptp4l.service**.

```
sudo gedit /lib/142ystem/system/ptp4l.service
```

```
[Unit]
Description=Precision Time Protocol (PTP) service
Documentation=man:ptp4l

[Service]
Type=simple
ExecStart=/usr/sbin/ptp4l -E -i enp2s0f0 -m

[Install]
WantedBy=multi-user.target
~
~
```

2. Set the following parameters after the content of **ExecStart=/usr/sbin/ptp4l** in **ptp4l.service**.

Table 27 Parameter Descriptions

Parameter	Description
-P	P2P (peer-to-peer) delay measurement mechanism. P2P enables more precise delay measurement. However, P2P can only be used for specific network topologies. For example, each port can only exchange PTP messages with other ports on a one-to-one basis and all hardware (including transparent clocks) on the communication path must support the P2P mode. Therefore, this option is only recommended for practical deployments when you know a lot about the network topology.
-A	This option enables the automatic selection of the delay mechanism. Under this auto option, ptp4l runs in E2E mode. A peer delay request will automatically switch to P2P mode if it can be received.
-E	The E2E (end-to-end) delay measurement mechanism. It is the default and recommended option. The E2E mechanism is also called the “request-response” delay mechanism.

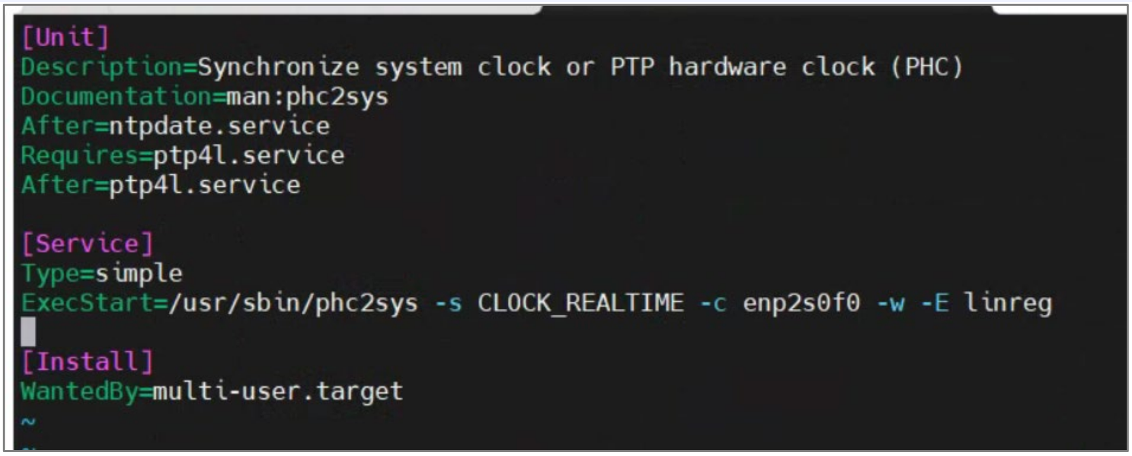
7.1.4.3 Configure phc2sys.service

1. Enter **phc2sys.service**.

```
sudo gedit /lib/143ystem/system/phc2sys.service
```

2. Set the following parameters after the content of **ExecStart=/usr/sbin/phc2sys** in **phc2sys.service**.

```
ExecStart=/usr/sbin/phc2sys -s CLOCK_REALTIME -c <network interface> -w -E linreg
```



```
[Unit]
Description=Synchronize system clock or PTP hardware clock (PHC)
Documentation=man:phc2sys
After=ntpdate.service
Requires=ptp4l.service
After=ptp4l.service

[Service]
Type=simple
ExecStart=/usr/sbin/phc2sys -s CLOCK_REALTIME -c enp2s0f0 -w -E linreg

[Install]
WantedBy=multi-user.target
~
```

Table 28 Parameter Descriptions

Parameter	Description
-s	CLOCK_REALTIME (fixed value)
-c	Network interface name of the computer used for time synchronization
-w	Waiting for ptp4l
-E	The synchronization algorithm between the master and slave. Linreg is recommended

7.1.4.4 Configure ptp4l.conf

1. Enter **ptp4l.conf**.

```
sudo gedit /etc/linuxptp/ptp4l.conf
```

2. Change the following content in **ptp4l.conf**.

```
Modify clock_servo pi to clock_servo linreg

Modify priority1 128 to priority1 127
```

3. Save changes and exit.

7.1.4.5 Reload and restart ptp4l/phc2sys services

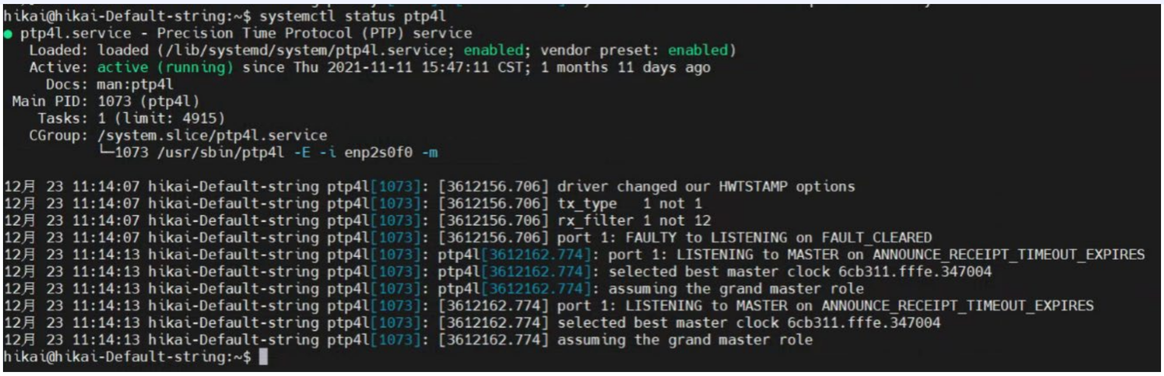
```
systemctl daemon-reload

systemctl enable ptp4l.service

systemctl restart ptp4l.service

systemctl enable phc2sys.service

systemctl restart phc2sys.service
```

7.1.4.6 Check the status of ptp4l and phc2sys services

1. Check the status of the **ptp4l service**.

   Execute the following command to confirm the ptp4l.serive is running and ‘assuming the grand master role’ is displayed. Currently, the computer is in the grandmaster state in this network environment.

```
systemctl status ptp4l.service
```

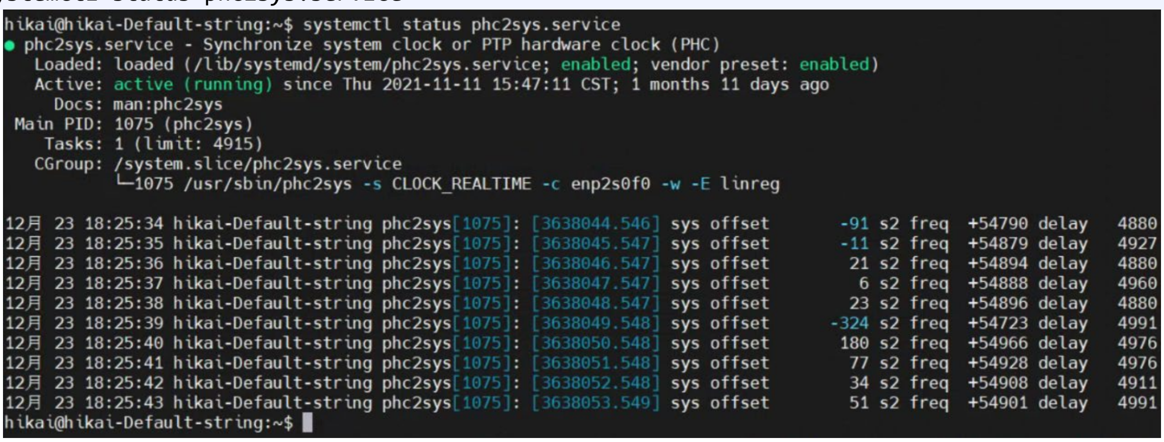


```
hikai@hikai-Default-string:~$ systemctl status ptp4l
● ptp4l.service - Precision Time Protocol (PTP) service
   Loaded: loaded (/lib/systemd/system/ptp4l.service; enabled; vendor preset: enabled)
   Active: active (running) since Thu 2021-11-11 15:47:11 CST; 1 months 11 days ago
     Docs: man:ptp4l
 Main PID: 1073 (ptp4l)
    Tasks: 1 (limit: 4915)
   CGroup: /system.slice/ptp4l.service
           └─1073 /usr/sbin/ptp4l -E -i enp2s0f0 -m

12月 23 11:14:07 hikai-Default-string ptp4l[1073]: [3612156.706] driver changed our HWTSTAMP options
12月 23 11:14:07 hikai-Default-string ptp4l[1073]: [3612156.706] tx_type   1 not 1
12月 23 11:14:07 hikai-Default-string ptp4l[1073]: [3612156.706] rx_filter 1 not 12
12月 23 11:14:07 hikai-Default-string ptp4l[1073]: [3612156.706] port 1: FAULTY to LISTENING on FAULT_CLEARED
12月 23 11:14:13 hikai-Default-string ptp4l[1073]: ptp4l[3612162.774]: port 1: LISTENING to MASTER on ANNOUNCE_RECEIPT_TIMEOUT_EXPIRES
12月 23 11:14:13 hikai-Default-string ptp4l[1073]: ptp4l[3612162.774]: selected best master clock 6cb311.fffe.347004
12月 23 11:14:13 hikai-Default-string ptp4l[1073]: ptp4l[3612162.774]: assuming the grand master role
12月 23 11:14:13 hikai-Default-string ptp4l[1073]: [3612162.774] port 1: LISTENING to MASTER on ANNOUNCE_RECEIPT_TIMEOUT_EXPIRES
12月 23 11:14:13 hikai-Default-string ptp4l[1073]: [3612162.774] selected best master clock 6cb311.fffe.347004
12月 23 11:14:13 hikai-Default-string ptp4l[1073]: [3612162.774] assuming the grand master role
hikai@hikai-Default-string:~$
```

2. Check the status of the **phc2sys service**.

   Execute the following command to confirm the ptp4l.serive is running and ‘assuming the grand master role’ is displayed. Please notice that the status for the phc2sys service may be in s2. (need to wait for a while). Currently, the computer is in the grandmaster state (PTP master state) in this network environment.

```
systemctl status phc2sys.service
```



```
hikai@hikai-Default-string:~$ systemctl status phc2sys.service
● phc2sys.service - Synchronize system clock or PTP hardware clock (PHC)
   Loaded: loaded (/lib/systemd/system/phc2sys.service; enabled; vendor preset: enabled)
   Active: active (running) since Thu 2021-11-11 15:47:11 CST; 1 months 11 days ago
     Docs: man:phc2sys
 Main PID: 1075 (phc2sys)
    Tasks: 1 (limit: 4915)
   CGroup: /system.slice/phc2sys.service
           └─1075 /usr/sbin/phc2sys -s CLOCK_REALTIME -c enp2s0f0 -w -E linreg

12月 23 18:25:34 hikai-Default-string phc2sys[1075]: [3638044.546] sys offset      -91 s2 freq  +54790 delay   4880
12月 23 18:25:35 hikai-Default-string phc2sys[1075]: [3638045.547] sys offset      -11 s2 freq  +54879 delay   4927
12月 23 18:25:36 hikai-Default-string phc2sys[1075]: [3638046.547] sys offset       21 s2 freq  +54894 delay   4880
12月 23 18:25:37 hikai-Default-string phc2sys[1075]: [3638047.547] sys offset        6 s2 freq  +54888 delay   4960
12月 23 18:25:38 hikai-Default-string phc2sys[1075]: [3638048.547] sys offset       23 s2 freq  +54896 delay   4880
12月 23 18:25:39 hikai-Default-string phc2sys[1075]: [3638049.548] sys offset     -324 s2 freq  +54723 delay   4991
12月 23 18:25:40 hikai-Default-string phc2sys[1075]: [3638050.548] sys offset      180 s2 freq  +54966 delay   4976
12月 23 18:25:41 hikai-Default-string phc2sys[1075]: [3638051.548] sys offset       77 s2 freq  +54928 delay   4976
12月 23 18:25:42 hikai-Default-string phc2sys[1075]: [3638052.548] sys offset       34 s2 freq  +54908 delay   4911
12月 23 18:25:43 hikai-Default-string phc2sys[1075]: [3638053.549] sys offset       51 s2 freq  +54901 delay   4991
hikai@hikai-Default-string:~$
```

7.1.4.7 Enable PTP time synchronization of the LiDAR

1. Enable the PTP time synchronization.

   When ptp_en=1, the PTP f time synchronization is enabled.

   When ptp_en=0, the PTP time synchronization is disabled.

```
./innovusion_lidar_util <LiDAR_IP> set_config time ptp_en 1
```

2. Set the PTP time synchronization customized mode.

When ptp_automotive =0, the PTP time synchronization is in customized mode.

When ptp_automotive =1, the PTP time synchronization is in gPTP mode.

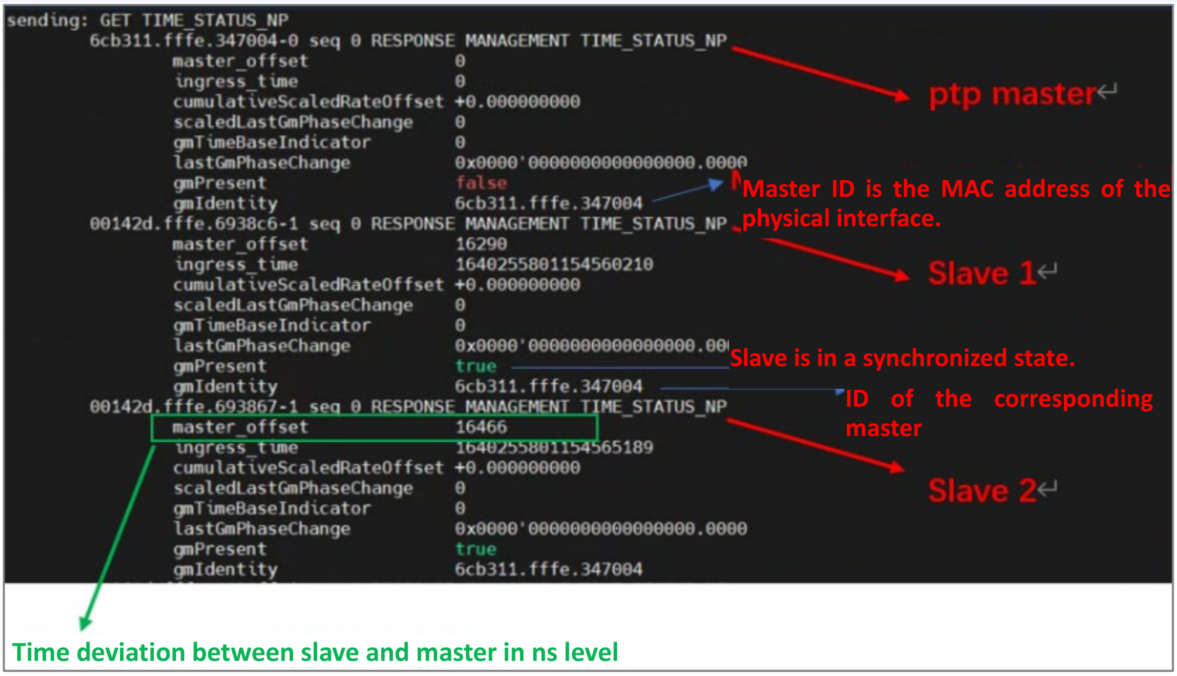
```
./innovusion_lidar_util <LiDAR_IP> set_config time ptp_automotive 0
```

7.1.4.8 Check the PTP status

Method 1

You can use the PMC tool to check the time deviation after time synchronization. The precision of the PMC query is ns level.

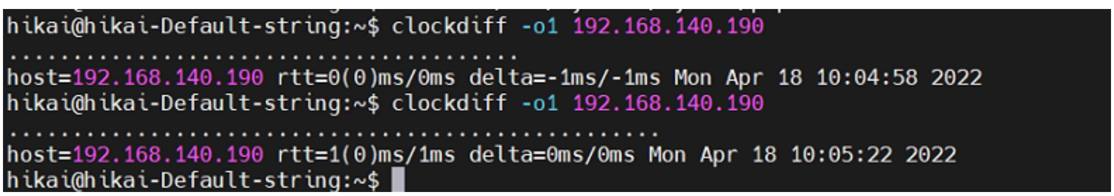
```
sudo pmc -u -b 1  'get time_status_np'
```



Method 2

You can use the clockdiff tool to check the time deviation after time synchronization. The precision of clockdiff query is ms level.

```
clockdiff -o1 <LiDAR_IP>
```



Method 3

You can download LiDAR logs from the ILA page to check whether correct time.

7.2 gPTP time synchronization

You need to confirm that all network nodes support PTP time synchronization before adopting gPTP as the time synchronization mode.

Note

Before sync time in gPTP mode, you need to ensure the following conditions.

- The network between the computer and the Falcon is connected.
- You have obtained the innovusion_lidar_util tool.

7.2.1 Sync time based on gPTP software timestamp

7.2.1.1 Install linuxptp

The Linuxptp installation package includes the **ptp4l** and **phc2sys** services. The computer for time synchronization needs to have a good connection with the Internet.

1. Install the **linuxptp**.

```
sudo apt install linuxptp
```

2. Confirm whether the version of linuxptp installed on the computer supports the gPTP service. The available version of linuxptp should be 3.1 or higher.

   i. Enter the **ptp.service** path.

```
cd /usr/sbin/
```

   ii. Check the **ptp4l** version.

```
ptp4l -v
```

```
demo@demo-OMEN-by-HP-Laptop-16-b0xxx:/usr/sbin$ ptp4l -v
3.1.1
```

7.2.1.2 Configure ptp4l service

1. Enter **ptp4l.service**.

```
sudo gedit /lib/systemd/system/ptp4l.service
```

```
[Unit]
Description=Precision Time Protocol (PTP) service
Documentation=man:ptp4l

[Service]
Type=simple
ExecStart=/usr/sbin/ptp4l -f /etc/linuxptp/automotive-master.conf -i enp2s0f0 -P -m

[Install]
WantedBy=multi-user.target
~
```

2. Change the following content in **ptp4l.service**.

```
ExecStart=/usr/sbin/ptp4l -f /etc/linuxptp/automotive-master.conf -i <network interface> -P
```

Table 29 Parameter Descriptions

Parameter	Description
-P	P2P delay measurement mechanism (The P parameter must be specified when using gPTP)
-i	The physical interface of the computer for PTP time synchronization.
-H	Time synchronization based on software timestamp (if the parameter is not specified, time synchronization will be based on hardware timestamp by default)
-m	Print information.

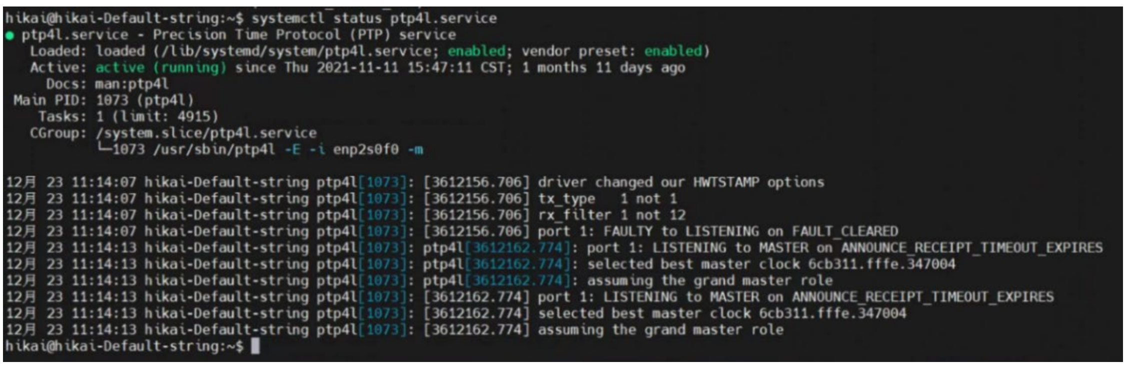
7.2.1.3 Reload and restart the ptp4l service

```
sudo systemctl daemon-reload
sudo systemctl enable ptp4l.service
sudo systemctl restart ptp4l.service
```

7.2.1.4 Check the status of ptp4l service

Verify that the service is in the 'active(running)' state with 'assuming the grand master role' displayed. Currently, the computer is in the grandmaster state (PTP master state) in this network environment.

```
systemctl status ptp4l.service
```



```
hikai@hikai-Default-string:~$ systemctl status ptp4l.service
● ptp4l.service - Precision Time Protocol (PTP) service
   Loaded: loaded (/lib/systemd/system/ptp4l.service; enabled; vendor preset: enabled)
   Active: active (running) since Thu 2021-11-11 15:47:11 CST; 1 months 11 days ago
     Docs: man:ptp4l
 Main PID: 1073 (ptp4l)
    Tasks: 1 (limit: 4915)
   CGroup: /system.slice/ptp4l.service
           └─1073 /usr/sbin/ptp4l -E -i enp2s0f0 -m

12月 23 11:14:07 hikai-Default-string ptp4l[1073]: [3612156.706] driver changed our HWTSTAMP options
12月 23 11:14:07 hikai-Default-string ptp4l[1073]: [3612156.706] tx_type   1 not 1
12月 23 11:14:07 hikai-Default-string ptp4l[1073]: [3612156.706] rx_filter 1 not 12
12月 23 11:14:07 hikai-Default-string ptp4l[1073]: [3612156.706] port 1: FAULTY to LISTENING on FAULT_CLEARED
12月 23 11:14:13 hikai-Default-string ptp4l[1073]: ptp4l[3612162.774]: port 1: LISTENING to MASTER on ANNOUNCE_RECEIPT_TIMEOUT_EXPIRES
12月 23 11:14:13 hikai-Default-string ptp4l[1073]: ptp4l[3612162.774]: selected best master clock 6cb311.fffe.347004
12月 23 11:14:13 hikai-Default-string ptp4l[1073]: ptp4l[3612162.774]: assuming the grand master role
12月 23 11:14:13 hikai-Default-string ptp4l[1073]: [3612162.774] port 1: LISTENING to MASTER on ANNOUNCE_RECEIPT_TIMEOUT_EXPIRES
12月 23 11:14:13 hikai-Default-string ptp4l[1073]: [3612162.774] selected best master clock 6cb311.fffe.347004
12月 23 11:14:13 hikai-Default-string ptp4l[1073]: [3612162.774] assuming the grand master role
hikai@hikai-Default-string:~$
```

7.2.1.5 Enable gPTP time synchronization of the LiDAR

1. Enable the PTP time synchronization.

   When ptp_en=1, the PTP f time synchronization is enabled.

   When ptp_en=0, the PTP time synchronization is disabled.

```
./innovusion_lidar_util <LiDAR_IP> set_config time ptp_en 1
```

2. Set the PTP time synchronization customized mode.

   When ptp_automotive =0, the PTP time synchronization is in customized mode.

   When ptp_automotive =1, the PTP time synchronization is in gPTP mode.

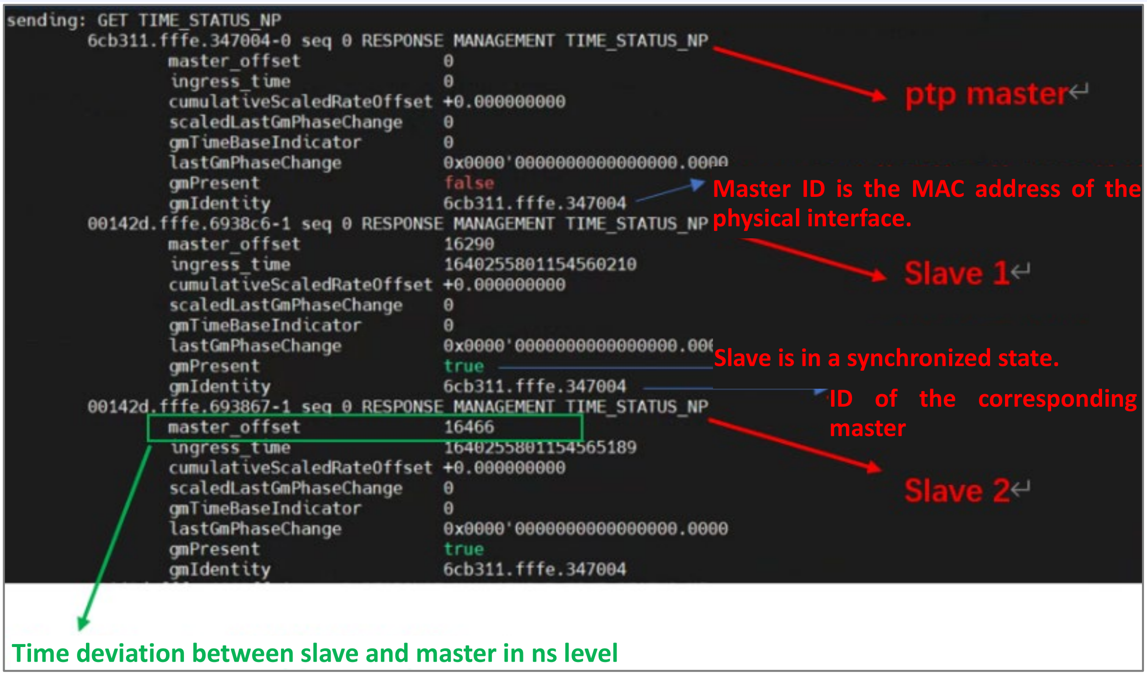
```
./innovusion_lidar_util <LiDAR_IP> set_config time ptp_automotive 1
```

7.2.1.6 Check the gPTP status

Method 1

You can use the PMC tool to check the time deviation after time synchronization. The precision of the PMC query is ns level.

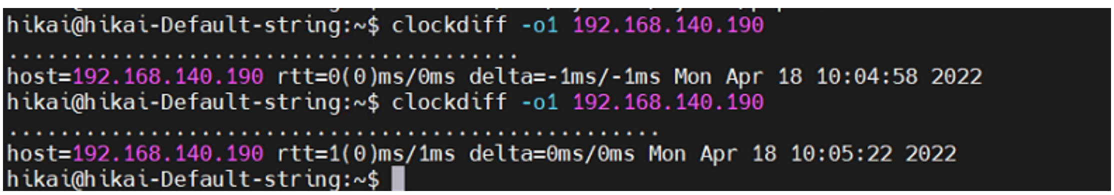
```
sudo pmc -u -b 1  'get time_status_np'
```



Method 2

You can use the clockdiff tool to check the time deviation after time synchronization. The precision of clockdiff query is ms level.

```
clockdiff -o1 <LiDAR_IP>
```



Method 3

You can download LiDAR logs from the ILA page to check the correct time.

7.2.2 Sync time based on gPTP hardware timestamp

7.2.2.1 Install linuxptp

The Linuxptp installation package includes the **ptp4l** and **phc2sys** services. The computer for time synchronization needs to have a good connection with the Internet.

1. Install the **linuxptp**.

```
sudo apt install linuxptp
```

2. Confirm whether the version of **ptp4l** installed on the computer supports the gPTP service. The available version of **ptp4l** should be 3.1 or higher.

   i. Enter the **ptp.service** path.

```
cd /usr/sbin/
```

   ii. Check the **ptp4l** version.

```
ptp4l -v
```

```
demo@demo-OMEN-by-HP-Laptop-16-b0xxx:/usr/sbin$ ptp4l -v
3.1.1
```

7.2.2.2 Configure ptp4l service

1. Enter **ptp4l.service**.

```
sudo gedit /lib/systemd/system/ptp4l.service
```

```
[Unit]
Description=Precision Time Protocol (PTP) service
Documentation=man:ptp4l

[Service]
Type=simple
ExecStart=/usr/sbin/ptp4l -f /etc/linuxptp/automotive-master.conf -i enp2s0f0 -P -m

[Install]
WantedBy=multi-user.target
~
```

2. Change the following content in **ptp4l.service**.

```
ExecStart=/usr/sbin/ptp4l -f /etc/linuxptp/automotive-master.conf -i <network interface> -P
```

Table 30 Parameter Descriptions

Parameter	Description
-P	P2P delay measurement mechanism (The P parameter must be specified when using gPTP)
-i	The physical interface of the computer for PTP time synchronization.
-H	Time synchronization based on software timestamp (if the parameter is not specified, time synchronization will be based on hardware timestamp by default)
-m	Print information.

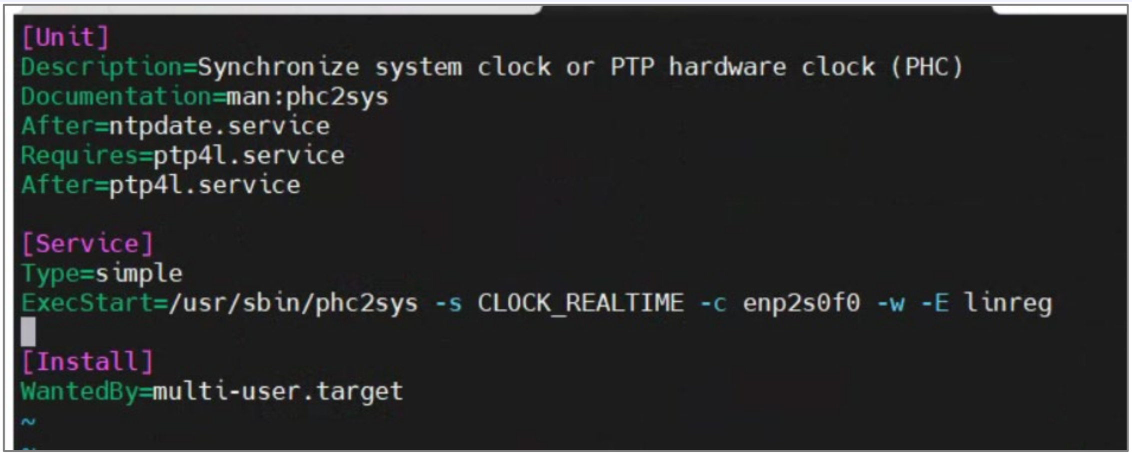
7.2.2.3 Configure phc2sys.service

1. Enter **phc2sys.service**.

```
sudo gedit /lib/systemd/system/phc2sys.service
```

2. Set the following parameters after the content of **ExecStart=/usr/sbin/phc2sys** in **phc2sys.service**.

```
ExecStart=/usr/sbin/phc2sys -f /etc/linuxptp/automotive-master.conf -s CLOCK_REALTIME -c
<network interface> -w -E linreg
```



```
[Unit]
Description=Synchronize system clock or PTP hardware clock (PHC)
Documentation=man:phc2sys
After=ntpdate.service
Requires=ptp4l.service
After=ptp4l.service

[Service]
Type=simple
ExecStart=/usr/sbin/phc2sys -s CLOCK_REALTIME -c enp2s0f0 -w -E linreg

[Install]
WantedBy=multi-user.target
~
```

Table 31 Parameter Descriptions

Parameter	Description
-s	CLOCK_REALTIME (fixed value)
-c	Network interface name of the computer used for time synchronization
-w	Waiting for ptp4l
-E	The synchronization algorithm between the master and slave. Linreg is recommended

7.2.2.4 Reload and restart ptp4l/phc2sys services

```
sudo systemctl daemon-reload
sudo systemctl enable ptp4l.service
sudo systemctl restart ptp4l.service
sudo systemctl enable phc2sys.service
sudo systemctl restart phc2sys.service
```

7.2.2.5 Check the status of ptp4l and phc2sys services

1. Check the status of the **ptp4l service**.

   Execute the following command to confirm the ptp4l.serive is running and 'assuming the grand master role' is displayed. Currently, the computer is in the grandmaster state in this network environment.

```
systemctl status ptp4l.service
```

```
hikai@hikai-Default-string:~$ systemctl status ptp4l
● ptp4l.service - Precision Time Protocol (PTP) service
   Loaded: loaded (/lib/systemd/system/ptp4l.service; enabled; vendor preset: enabled)
   Active: active (running) since Thu 2021-11-11 15:47:11 CST; 1 months 11 days ago
     Docs: man:ptp4l
 Main PID: 1073 (ptp4l)
    Tasks: 1 (limit: 4915)
   CGroup: /system.slice/ptp4l.service
           └─1073 /usr/sbin/ptp4l -E -i enp2s0f0 -m

12月 23 11:14:07 hikai-Default-string ptp4l[1073]: [3612156.706] driver changed our HWTSTAMP options
12月 23 11:14:07 hikai-Default-string ptp4l[1073]: [3612156.706] tx_type   1 not 1
12月 23 11:14:07 hikai-Default-string ptp4l[1073]: [3612156.706] rx_filter 1 not 12
12月 23 11:14:07 hikai-Default-string ptp4l[1073]: [3612156.706] port 1: FAULTY to LISTENING on FAULT_CLEARED
12月 23 11:14:13 hikai-Default-string ptp4l[1073]: ptp4l[3612162.774]: port 1: LISTENING to MASTER on ANNOUNCE_RECEIPT_TIMEOUT_EXPIRES
12月 23 11:14:13 hikai-Default-string ptp4l[1073]: ptp4l[3612162.774]: selected best master clock 6cb311.fffe.347004
12月 23 11:14:13 hikai-Default-string ptp4l[1073]: ptp4l[3612162.774]: assuming the grand master role
12月 23 11:14:13 hikai-Default-string ptp4l[1073]: [3612162.774] port 1: LISTENING to MASTER on ANNOUNCE_RECEIPT_TIMEOUT_EXPIRES
12月 23 11:14:13 hikai-Default-string ptp4l[1073]: [3612162.774] selected best master clock 6cb311.fffe.347004
12月 23 11:14:13 hikai-Default-string ptp4l[1073]: [3612162.774] assuming the grand master role
hikai@hikai-Default-string:~$
```

2. Check the status of the **phc2sys service**.

   Execute the following command to confirm the ptp4l.serive is running and 'assuming the grand master role' is displayed. Please notice that the status for the phc2sys service may be in s2. (need to wait for a while). Currently, the computer is in the grandmaster state (PTP master state) in this network environment.

```
systemctl status phc2sys.service
```

```
hikai@hikai-Default-string:~$ systemctl status phc2sys.service
● phc2sys.service - Synchronize system clock or PTP hardware clock (PHC)
   Loaded: loaded (/lib/systemd/system/phc2sys.service; enabled; vendor preset: enabled)
   Active: active (running) since Thu 2021-11-11 15:47:11 CST; 1 months 11 days ago
     Docs: man:phc2sys
 Main PID: 1075 (phc2sys)
    Tasks: 1 (limit: 4915)
   CGroup: /system.slice/phc2sys.service
           └─1075 /usr/sbin/phc2sys -s CLOCK_REALTIME -c enp2s0f0 -w -E linreg

12月 23 18:25:34 hikai-Default-string phc2sys[1075]: [3638044.546] sys offset      -91 s2 freq  +54790 delay   4880
12月 23 18:25:35 hikai-Default-string phc2sys[1075]: [3638045.547] sys offset      -11 s2 freq  +54879 delay   4927
12月 23 18:25:36 hikai-Default-string phc2sys[1075]: [3638046.547] sys offset       21 s2 freq  +54894 delay   4880
12月 23 18:25:37 hikai-Default-string phc2sys[1075]: [3638047.547] sys offset        6 s2 freq  +54888 delay   4960
12月 23 18:25:38 hikai-Default-string phc2sys[1075]: [3638048.547] sys offset       23 s2 freq  +54896 delay   4880
12月 23 18:25:39 hikai-Default-string phc2sys[1075]: [3638049.548] sys offset     -324 s2 freq  +54723 delay   4991
12月 23 18:25:40 hikai-Default-string phc2sys[1075]: [3638050.548] sys offset      180 s2 freq  +54966 delay   4976
12月 23 18:25:41 hikai-Default-string phc2sys[1075]: [3638051.548] sys offset       77 s2 freq  +54928 delay   4976
12月 23 18:25:42 hikai-Default-string phc2sys[1075]: [3638052.548] sys offset       34 s2 freq  +54908 delay   4911
12月 23 18:25:43 hikai-Default-string phc2sys[1075]: [3638053.549] sys offset       51 s2 freq  +54901 delay   4991
hikai@hikai-Default-string:~$
```

7.2.2.6 Enable gPTP time synchronization of the LiDAR

1. Enable the PTP time synchronization.

   When ptp_en=1, the PTP f time synchronization is enabled.

   When ptp_en=0, the PTP time synchronization is disabled.

```
./innovusion_lidar_util <LiDAR_IP> set_config time ptp_en 1
```

2. Set the PTP time synchronization customized mode.

   When ptp_automotive =0, the PTP time synchronization is in customized mode.

   When ptp_automotive =1, the PTP time synchronization is in gPTP mode.

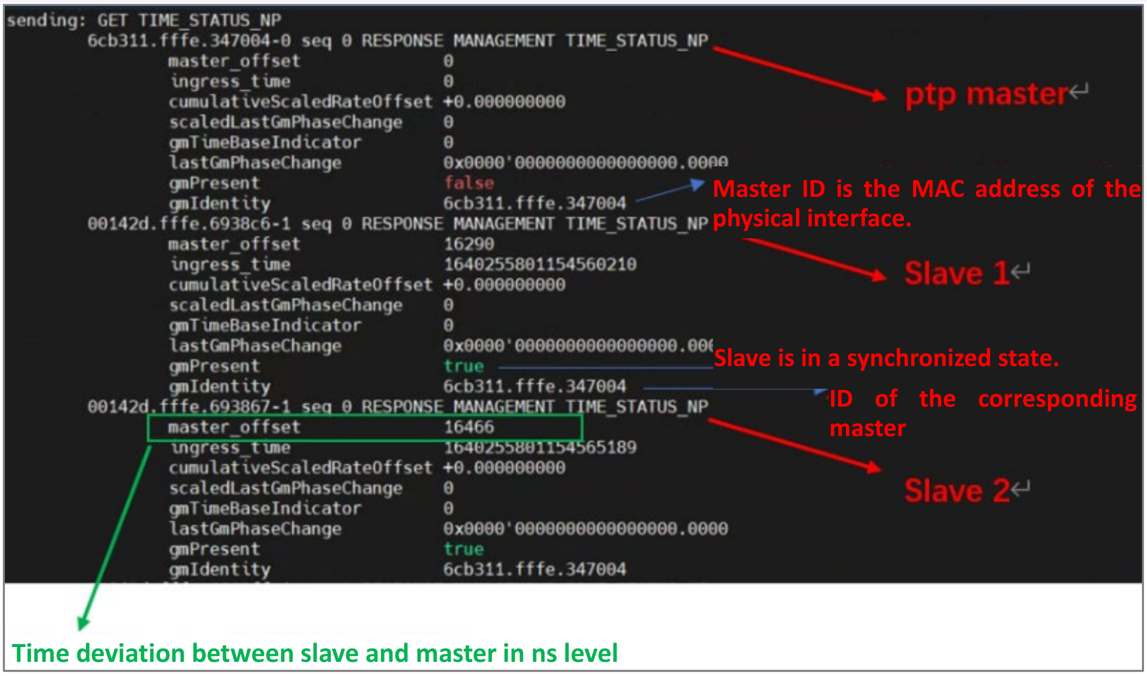
```
./innovusion_lidar_util <LiDAR_IP> set_config time ptp_automotive 1
```

7.2.2.7 Check the gPTP status

Method 1

You can use the PMC tool to check the time deviation after time synchronization. The precision of the PMC query is ns level.

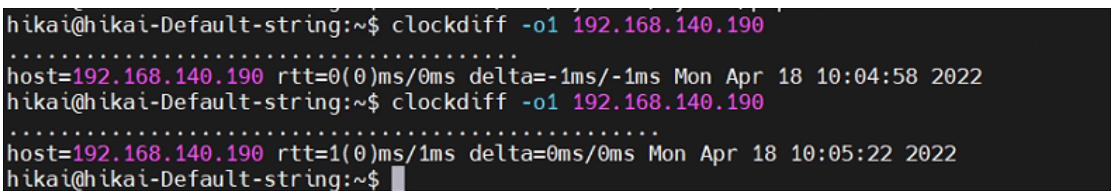
```
sudo pmc -u -b 1  'get time_status_np'
```



Method 2

You can use the clockdiff tool to check the time deviation after time synchronization. The precision of clockdiff query is ms level.

```
clockdiff -o1 <LiDAR_IP>
```



Method 3

You can download LiDAR logs from the ILA page to check the correct time.

8 Troubleshooting Guide

Table 32 Troubleshooting

Serial No.	Problem	Resolution
1	Network connection failure	1. Check whether the power supply voltage is normal.
		2. Check whether the power cable is tightly plugged in.
		3. Check whether the power supply current is normal, and the power consumption should be about 34W.
		4. Check whether the software configuration is correct.
		5. Power off for at least one minute, and power on again for testing.
		6. Check whether the network cable is plugged in.
		7. Check whether the network light flashes.
		8. Check whether the network card for the server is normal or change to another computer and retest.
		9. Check whether the network card for the server is a Gigabit network card.
		10. Check the IP address of the server and confirm that the IP address r is in the same subnet with the device.
		11. Power off for at least one minute, and power on again for testing.
2	Point cloud display failure/inappropriate	1. Check whether the firewall of the server is disabled.
		2. Use the Wireshark packet capture tool to check whether the data packet is complete.
		3. Check whether windows are blocked.
		4. Power off for at least one minute, and power on again for testing.
3	Noise points appear in the point cloud	1. Check whether windows are contaminated.
		2. Check whether the object is strongly reflective.
		3. Power off for at least one minute, and power on again for testing.
4	Incorrect point cloud FOV	1. Check whether windows are contaminated.
		2. Check whether windows are blocked.
		3. Check whether the software is configured correctly.
		4. Power off for at least one minute, and power on again for testing.
5	The ranging ability does not meet the standard.	1. Check whether windows are contaminated.
		2. Pay attention to the visibility of the weather.

		3. Check whether windows are blocked.
		4. Check whether the parameters of the mechanical hardware of the LiDAR are set correctly.
		5. Power off for at least one minute, and power on again for testing.
6	Time synchronization failure	1. Check whether the time synchronization interface is connected correctly.
		2. Check the time synchronization service for normal operation.

Appendix A. Computer configuration reference

The following table provides the reference for configuring the computer. Users can select suitable computers according to the table.

Note

- The computer configuration recommended in this table only enables you to view point cloud data. If you have other requirements, please consult Innovusion staff.
- This table is only for reference to the minimum configuration requirements of the server. Users can upgrade the computer' s configuration based on this table's requirements.

Table 33 Configuration reference for the computer

Attribute	**Configuration**
CPU	Dual-core CPU Intel I7-9th generation or other types of processors with equivalent performance or above
RAM	1 GB
Free hard drive space	≥ 1000 MB
Ethernet connection	1 G/s

Appendix B. Clockdiff installation

B.1 Online installation

Open a terminal. Execute the following command to install clockdiff online.

```
sudo apt install iputils-clockdiff
```

B.2 Offline installation

1. Download the installation package according to your actual computer environment. (The recommended installation package is shown in the figure below. Amd64, and i386 correspond to 64-bit and 32-bit Linux operating systems, respectively)

   The path of the installation package is as follows.

   http://archive.ubuntu.com/ubuntu/pool/universe/i/iputils/

Index of /ubuntu/pool/universe/i/iputils

Name	Last modified	Size
Parent Directory		-
iputils-clockdiff_20121221-4ubuntu1.1_amd64.deb	2014-05-07 22:03	23K
iputils-clockdiff_20121221-4ubuntu1.1_i386.deb	2014-05-07 22:03	23K
iputils-clockdiff_20121221-4ubuntu1_amd64.deb	2014-03-15 06:48	23K
iputils-clockdiff_20121221-4ubuntu1_i386.deb	2014-03-15 06:48	23K
iputils-clockdiff_20121221-5ubuntu2_amd64.deb	2014-05-07 19:53	24K
iputils-clockdiff_20121221-5ubuntu2_i386.deb	2014-05-07 19:53	24K
iputils-clockdiff_20161105-1ubuntu2_amd64.deb	2017-03-09 22:48	26K
iputils-clockdiff_20161105-1ubuntu2_i386.deb	2017-03-09 22:48	26K
iputils-clockdiff_20161105-1ubuntu3_amd64.deb	2019-07-10 19:58	27K
iputils-clockdiff_20161105-1ubuntu3_i386.deb	2019-07-10 19:58	27K
iputils-clockdiff_20190709-3_amd64.deb	2020-01-31 23:45	12K
iputils-clockdiff_20210202-1_amd64.deb	2021-02-02 23:34	13K
iputils-clockdiff_20211215-1_amd64.deb	2022-02-05 11:19	12K

Apache/2.4.29 (Ubuntu) Server at archive.ubuntu.com Port 80

2. Upload the installation package to a local directory.
3. Execute the command to install clockdiff.

```
sudo dpkg -i iputils-clockdiff_20161105-1ubuntu3_amd64.deb
```

❖

Appendix C. Upgrade the LiDAR

1. Connect a computer to LiDAR.
2. Obtain the upgrade package in img format and copy it to a local directory on the computer.

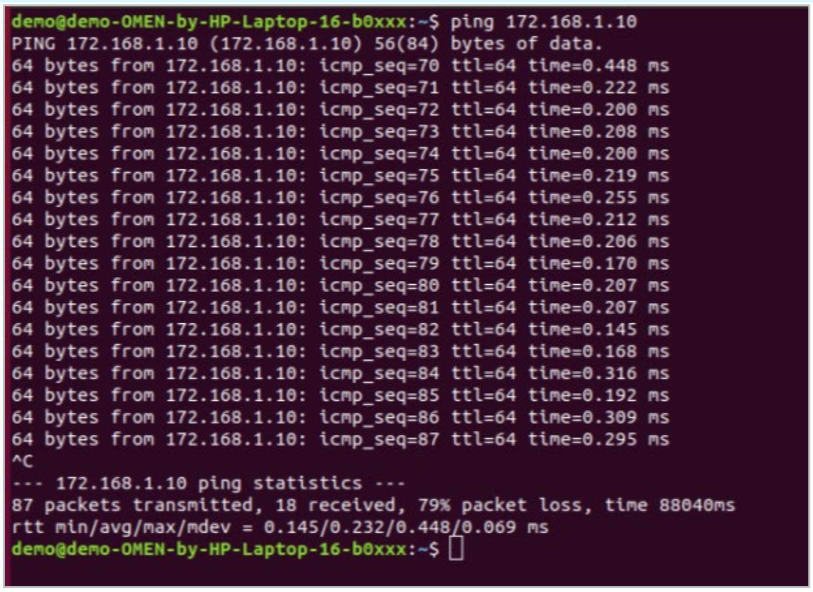
Note

If necessary, please contact Innovusion staff to obtain the upgrade package in img format. The upgrade package includes firmware and software upgrades.

3. Change the computer IP address to the same subnet with the LiDAR and confirm a good Ethernet connection between them.
4. Open the Chrome browser and enter the LiDAR IP address to access LiDAR.

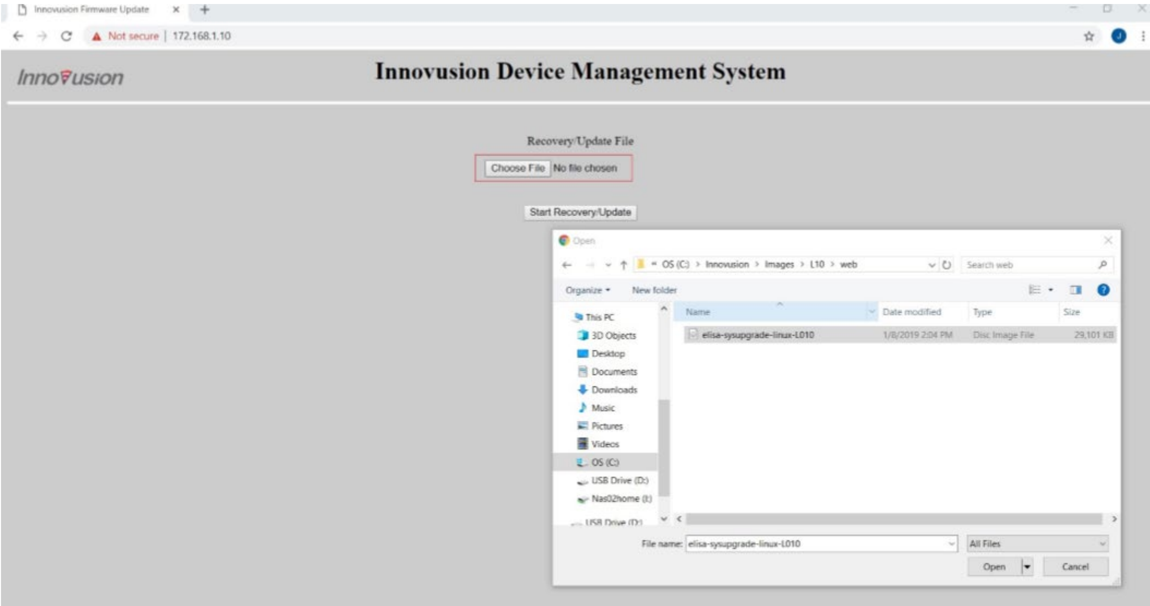
Note

- The default LiDAR IP address is 172.168.1.10.
- It is recommended to check the access to the LiDAR IP address via the ping command. You should make sure that the computer is connected to the LiDAR network. The return value is shown in the figure below.



5. Click **Recovery/Update File**.

6. Click **Choose File**.
7. Select the required upgrade package in the **Open** window.



8. Click Start Recovery/Update to start the upgrade.
9. Power off and restart the system after the upgrade.
10. (Optional) View the version information on the **System info**.

Appendix D. Command line instructions

Table 34 Command line instructions

Format	Description
< >	The content in “< >” needs to be replaced by the actual value. e.g., ./innovusion_lidar_util 172.168.1.10 get_version
[ ]	The contents in “[ ]” is optional.
[x | y | ...]	Only one option can be selected.
//	The content after “//” is a comment.

Appendix E.Abbreviations and terms

Table 35 Abbreviations

Abbreviations	Full name
AC	Alternating Current
DC	Direct Current
ETH	Ethernet
FAQ	Frequently Asked Questions
FOV	Field of View
GEN	Generation
GND	Ground
GPS	Global Positioning System
H × W × D	Height × Width × Depth
ILA	Innovusion LiDAR Appliance
IP	Internet Protocol
LiDAR	Light Detection and Ranging
MAC	Media Access Control
MEC	Multi-Access Edge Computing
NTP	Network Time Protocol
PPS	Pulse Per Second
PTP	Precision Time Protocol
PWR	Power
ROI	Region of Interest
ROS	Robot Operating System
SDK	Software Development Kit
SN	Serial Number
SW	Software
TCP	Transmission Control Protocol
TOF	Time of Flight
UDP	User Datagram Protocol

Table 36 Technical Terms

Terms	Definition
Class 1 laser products	Within the corresponding wavelength and emission duration, the exposure of personnel to laser radiation is not allowed to exceed Class 1 laser products that can reach the emission limit.
NTP	Network Time Protocol (NTP) is a protocol used to synchronize computer time. It is widely used to synchronize computers to Internet time servers, such as radio or satellite receivers or telephone modem services.

PTP	Precision Time Protocol (PTP) is a high-precision time synchronization protocol. It is used for high-precision time synchronization between devices but can also be used for frequency synchronization between devices.
Installer	Installers refer to those who have received professional training and have rich experience in the relevant field, fully understand the application of protective devices on the machine, and can assess its working safety state.
Commissioning personnel	Commissioning personnel have received professional training and have rich experience in the relevant field, fully understand the application of protective devices on the machine, and can assess its working safety state.
Time of flight (TOF)	The time-of-flight (TOF) realizes distance measurement by determining the time-of-flight interval between transmitting and receiving signals. For the formula, see the **Principles of operation** section.
Laser product	Combination of any products or components used to construct or prepare for use to construct a laser or a laser system. An electronic component sold as a component to another manufacturer is not a laser product.
Laser	An electromagnetic radiation device that mainly generates or amplifies the wavelength in the range of 180nm ~ 1mm through a controlled laser emission process.
Laser equipment	A combination of laser products or laser products containing lasers.
Server	A computer that can directly issue operation and control commands. The server sends commands first to the slave computer, and then the slave computer controls the device according to this command. The slave computer reads the device status data from time to time, converts it into a digital signal, and feeds it back to the server.
Configuration personnel	The configuration personnel should have expertise and experience in the relevant field and have sufficient experience to evaluate whether the machine is in a safe operation status after using protective equipment.
Eye safety	Although the product design meets the Class 1 eye safety standard, to protect your safety to the greatest extent, do not use amplification equipment (such as a microscope and magnifying glass) to look at the laser light in transmission directly.
Service personnel	Qualified service personnel refer to those who have received professional training and have rich experience in the relevant field, fully understand the application of protective devices on machines, and have received the guidance of the machine operation supervisor.

Appendix F. Revision history

Revision history

Version number	Revised content	Revision time
V1.2	**New** Getting started (#Section 2) cURL commands (#Section 5.5) **Update** get_pcd (#Section 5.3)	2023/7/25
V1.1	**New** Metaview (#Section 3.3) get_pcd (#Section 4.3) Data format (#Section 4.2) Innovusion_lidar_util (#Section 4.4) **Update** Installation (#Section 2) ILA operation (#Section 3.1) Time synchronization (#Section 6)	2023/4/19
V1.0	The first draft	2022/06/17