Go1-rshelios16p-operation

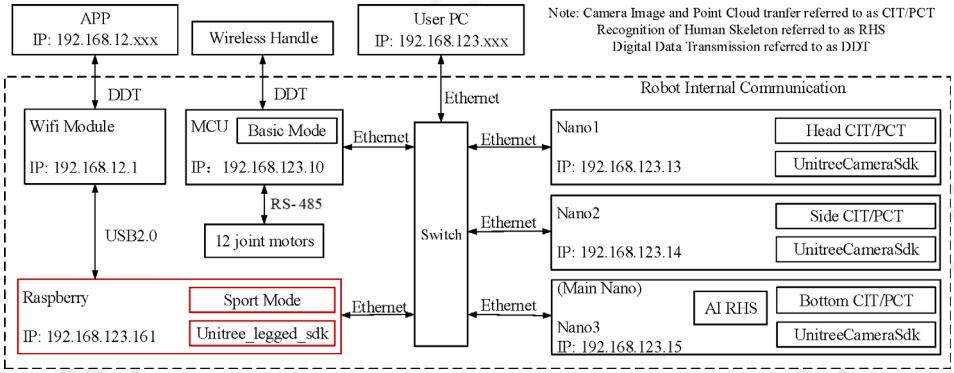
Introduction

This document focuses on how to start using 3d lidar on go1. If you need to do secondary development, please refer to the code and development guide by yourself
Video tutorial link:

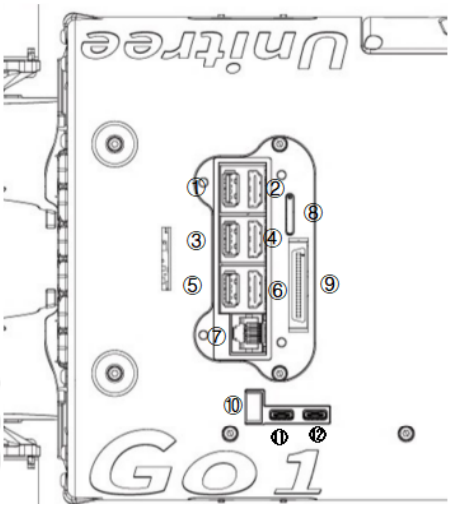
Introduction to robotics in general

1、After Go1-edu&Rshelios-16p are connected, it should look like the following figure

2、The software architecture of the robot is shown in the following diagram:



The 3d lidar program is located in the figure Main nano (nano3)(NX) deployment. Its IP address is 192.168.123.15
When we use the lidar locally, we need to connect the group of usb and hdmi interfaces(3&4) in the middle of the figure below



1. Nano2 USB
2. Nano2 HDMI
3. Nano/Nx3 USB
4. Nano/Nx3 HDMI
5. Raspberry USB
6. Raspberry HDMI
7. Gigabit Ethernet port
8. SIM
9. Integratedinterface
10. External power input (24V 30A)
11. Type-C
12. Type-C

When we need to remote control when we need to go to the 192.168.123.15 ip address to run the code

Preparation

1、Go1-edu&Rshelios-16p
2、Mouse keyboard usb docking station(hub) and monitor (Hdmi)
3、Computer with linux system,for remote control of robots
4、usb wireless network card(linux driver-free),Use in conjunction with the actual situation,for remote control of robots

Remark:

If you are purchasing Go1-edu&Rshelios-16p as a package from Unitree, then all configurations on the software side do not need to be changed, just run the startup command directly.

Local build use:

This step is to connect the robot to the monitor and then perform the map building task.

Remark:

There is some risk in doing so, because the robot is connected to the monitor in real time, so please pay attention to the length of the cable when operating the robot movement to ensure that the robot movement will not damage other equipment.

1. Connect the mouse and keyboard as well as the monitor according to the picture below

2. Run the following build command

```
cd UnitreeSlam

./build_map.sh
```

3. Re-open a terminal and run rviz

```
cd UnitreeSlam

./ build_map_rviz.sh
```

4. Send an arrow through rviz to run the robot to the location you want to patrol, then press X on the remote to save. After the point save is finished, control the robot back to the starting point then close both terminal programs
5. Execute patrol code instructions,the robot will automatically move to the point it just saved

```
cd UnitreeSlam

./start_patrol.sh

./start_patrol_rviz.sh
```

Remote use of lidar programs

This step is for us to use the lidar program on our own computer.
The basic logic is: first keep the robot's nx in communication with your own computer (under the same network segment), and then control the robot remotely via ROS_MASTER

1. Use tools such as Mouse keyboard usb docking station (hub) and monitor (Hdmi) to connect the robot's computer NX to your own network and get the ip address.Assume the following: `192.168.1.76`
2. Get your own computer to connect to your network and get the ip address.Assume the following: `192.168.1.75`
3. Open the file `~/.bashrc` on `NX` to configure the robot `NX` as `ROS Master` as follows.

```
export ROS_MASTER_URI=http://192.168.1.76:11311

export ROS_IP=192.168.1.76
```

4. Open the file `~/.bashrc` on your own computer and do the following to configure it as a slave.

```
export ROS_MASTER_URI=http://192.168.1.76:11311

export ROS_IP=192.168.1.75
```

5. Open a terminal on your computer, use `ssh` to log in to the robot `NX` remotely and go to the 3d slam program path
   Password is : **123**

```
ssh unitree@192.168.1.76
```

6. Run the following build map command

```
cd UnitreeSlam

./build_map.sh
```

7. Re-open a terminal and run rviz on your own computer()

```
rviz ./build_map_rviz
```

The rviz file is in the following path of nx, you can copy this file to your own computer by yourself

```
UnitreeSLAM\catkin_lidar_slam_3d\src\lidar_slam_3d\start\rviz
```

8. Send an arrow through rviz to run the robot to the location you want to patrol, then press X on the remote to save. After the point save is finished, control the robot back to the starting point then close both terminal programs

9. Execute patrol code instructions,the robot will automatically move to the point it just saved

```
ssh unitree@192.168.1.76

cd UnitreeSlam

./start_patrol_rviz.sh

rviz ./build_map_rviz
```

Script file explanation

These scripts are for the convenience of the customer to run the program, or if you don't want to run the script, you can just run the code directly as shown. Also multiple map builds can be named with their own map names:
For example, when performing a build task:
The original command:

```
roslaunch start build_map.launch map_name:=my_map_name
```

Modified command for the new map:

```
roslaunch start build_map.launch map_name:=new map
```

However, please note that the map name should correspond to the corresponding patrol command when it is executed
build_map.sh:

```
s#!/bin/bash

cd ~/UnitreeSLAM/catkin_lidar_slam_3d

source devel/setup.bash

roslaunch start build_map.launch map_name:=my_map_name
```

build_map_rviz:

```
s#!/bin/bash

cd ~/UnitreeSLAM/catkin_lidar_slam_3d

source devel/setup.bash

rosrun rviz rviz -d src/lidar_slam_3d/start/rviz/build_map.rviz
```

start_patrol.sh:

```
s#!/bin/bash

cd ~/UnitreeSLAM/catkin_lidar_slam_3d

source devel/setup.bash

roslaunch start start_patrol.launch map_name:=my_map_name
```

start_patrol_rviz:

```
s#!/bin/bash

cd ~/UnitreeSLAM/catkin_lidar_slam_3d

source devel/setup.bash

rosrun rviz rviz -d src/lidar_slam_3d/start/rviz/start_patrol.rviz
```