

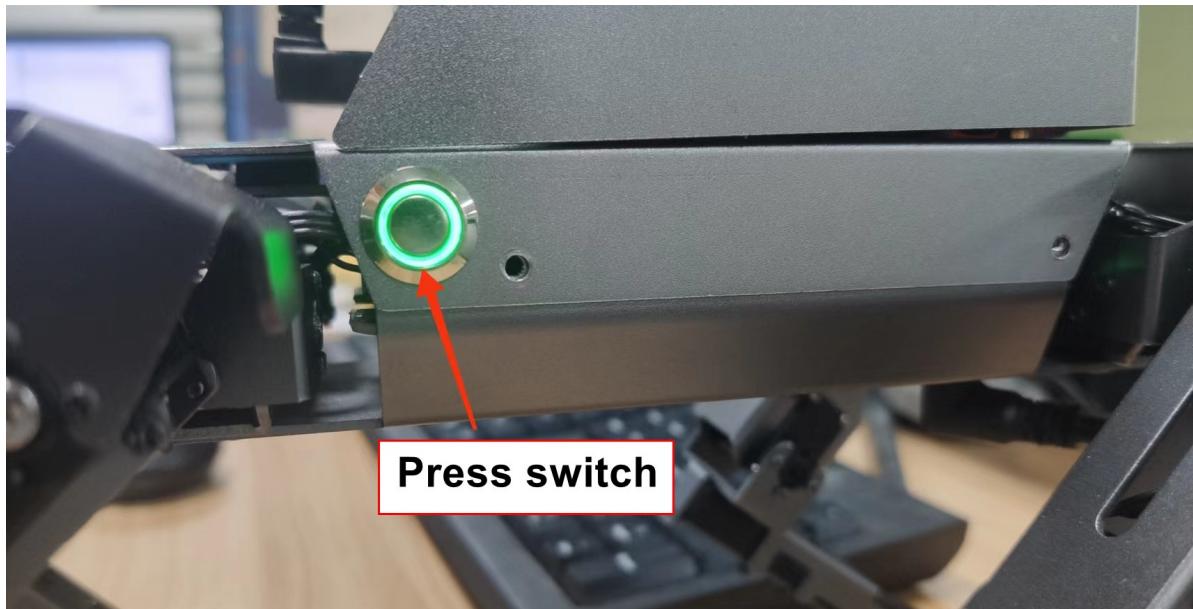
ROS2 environment entity robot dog mapping

Quick use

1. Power on the robot dog

PI4 version steps:

Press the power switch on the side of the robot dog and wait for the robot dog to start up. And connect the robot dog's WiFi to the same LAN environment as the computer.



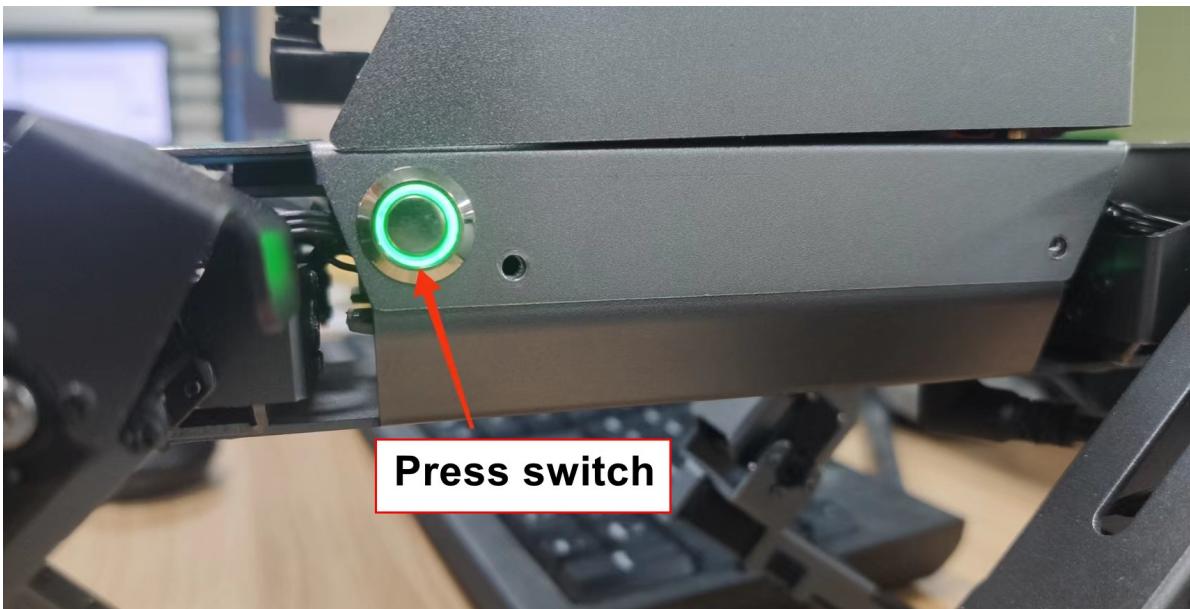
After the robot dog is started, the laser radar, imu, and robot dog joint status nodes have been automatically started.

If you find that you cannot obtain lidar and other data, please close the robot dog program and restart the chassis program.

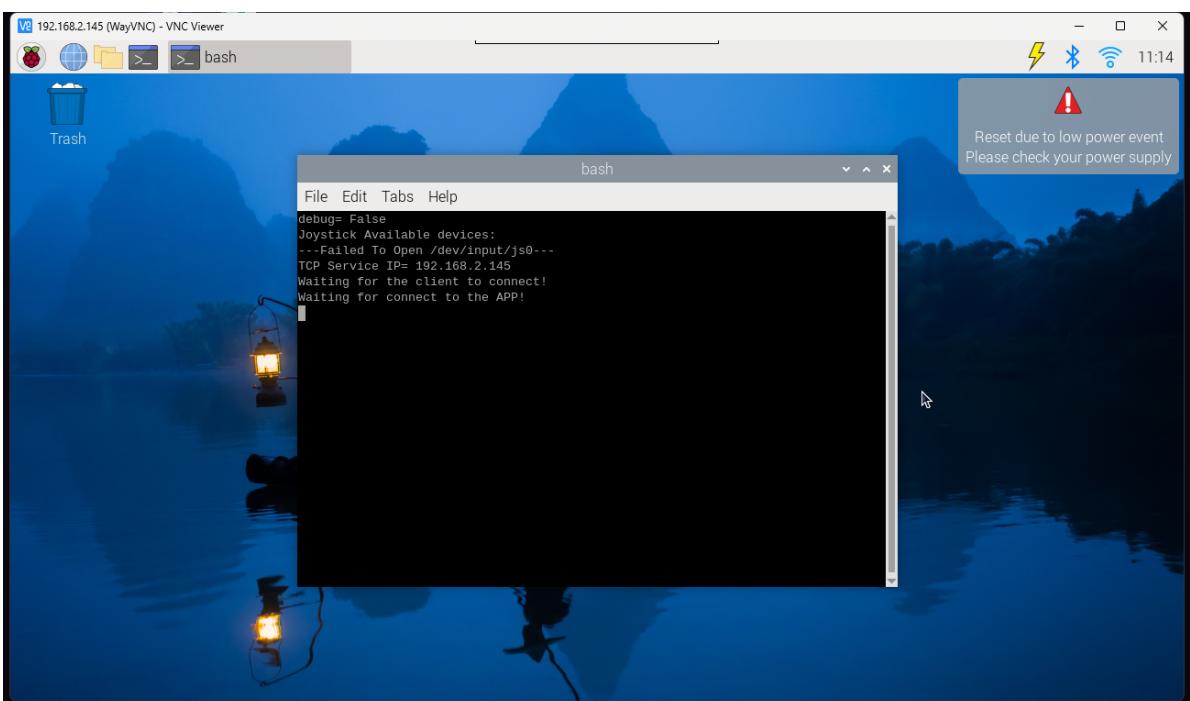
To close the large program, restart the chassis and modify the multi-level communication ID, please refer to the tutorial: 14. Radar mapping navigation\6. Obtaining the status of the physical robot dog in the ROS2 environment\Acquiring the real joint data of the robot dog in the ROS2 environment.pdf

PI5 version steps:

Press the power switch on the side of the robot dog and wait for the robot dog to start up. And connect the robot dog's WiFi to the same LAN environment as the computer.



After the robot dog is started, remotely connect to the robot dog through the IP address on the OLED.



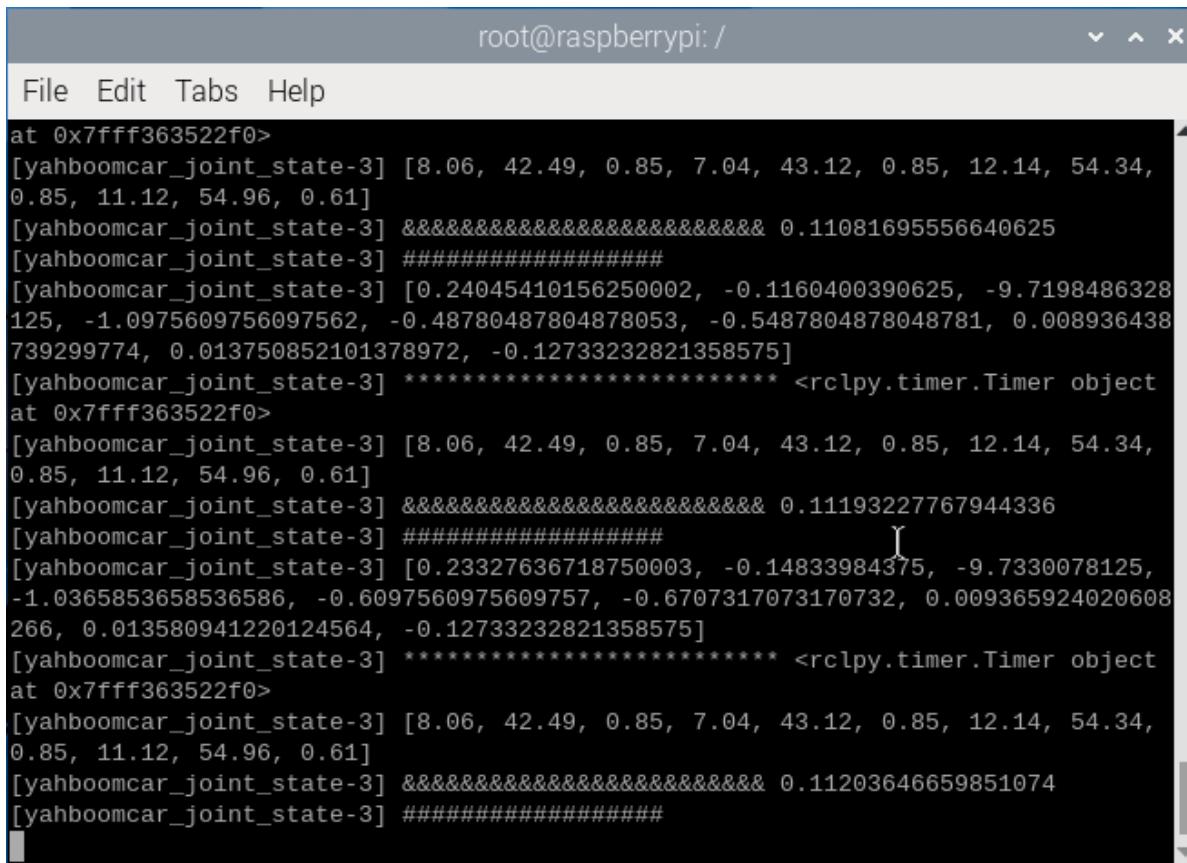
Then **ctrl+c** closes the large program and enter the following command to enter docker:

```
./run_humble.sh
```

```
TCP Service IP= 192.168.2.145
Waiting for the client to connect!
Waiting for connect to the APP!
^CKeyboardInterrupt
2024-04-28T10:17:27Z
-----program end-----
pi@raspberrypi:~ $ ./run_humble.sh
access control disabled, clients can connect from any host
root@raspberrypi:/#
```

Then enter the following commands in the docker terminal to start the car radar, imu, and robot dog joint status nodes.

```
ros2 launch bringup Navigation_bringup.launch.py
```



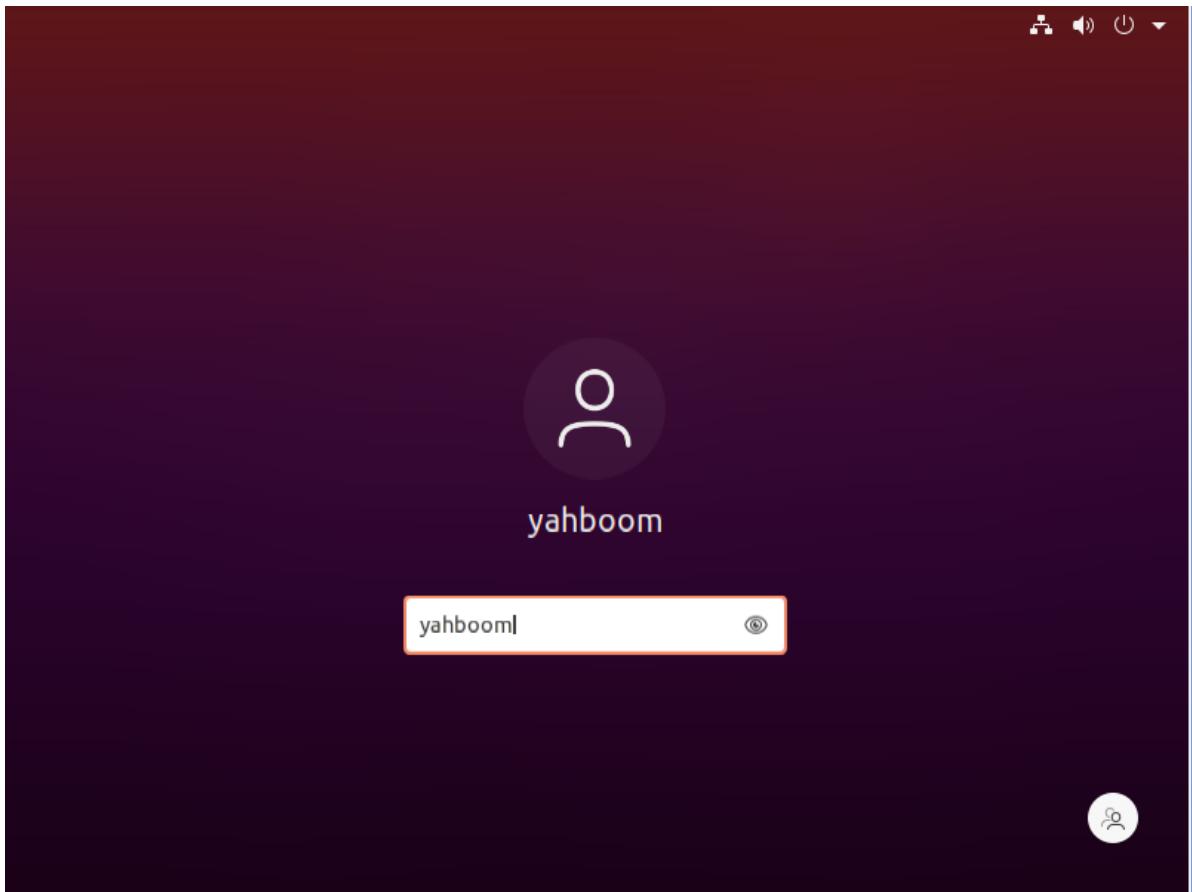
A terminal window titled "root@raspberrypi: /" showing the output of a ROS 2 launch command. The output lists multiple instances of the "yahboomcar_joint_state-3" topic, each publishing a message with various numerical values. The messages include coordinates and other sensor data.

```
root@raspberrypi: /  
File Edit Tabs Help  
at 0x7fff363522f0>  
[yahboomcar_joint_state-3] [8.06, 42.49, 0.85, 7.04, 43.12, 0.85, 12.14, 54.34,  
0.85, 11.12, 54.96, 0.61]  
[yahboomcar_joint_state-3] &&&&&&&&&&&&&&&&& 0.11081695556640625  
[yahboomcar_joint_state-3] #####  
[yahboomcar_joint_state-3] [0.24045410156250002, -0.1160400390625, -9.7198486328  
125, -1.0975609756097562, -0.48780487804878053, -0.5487804878048781, 0.008936438  
739299774, 0.013750852101378972, -0.12733232821358575]  
[yahboomcar_joint_state-3] ***** <rclpy.timer.Timer object  
at 0x7fff363522f0>  
[yahboomcar_joint_state-3] [8.06, 42.49, 0.85, 7.04, 43.12, 0.85, 12.14, 54.34,  
0.85, 11.12, 54.96, 0.61]  
[yahboomcar_joint_state-3] &&&&&&&&&&&&&&&&& 0.11193227767944336  
[yahboomcar_joint_state-3] #####  
[yahboomcar_joint_state-3] [0.23327636718750003, -0.14833984375, -9.7330078125,  
-1.0365853658536586, -0.6097560975609757, -0.6707317073170732, 0.009365924020608  
266, 0.013580941220124564, -0.12733232821358575]  
[yahboomcar_joint_state-3] ***** <rclpy.timer.Timer object  
at 0x7fff363522f0>  
[yahboomcar_joint_state-3] [8.06, 42.49, 0.85, 7.04, 43.12, 0.85, 12.14, 54.34,  
0.85, 11.12, 54.96, 0.61]  
[yahboomcar_joint_state-3] &&&&&&&&&&&&&&&&& 0.11203646659851074  
[yahboomcar_joint_state-3] #####
```

2. Open the virtual machine

Note: The default virtual machine has been installed here.

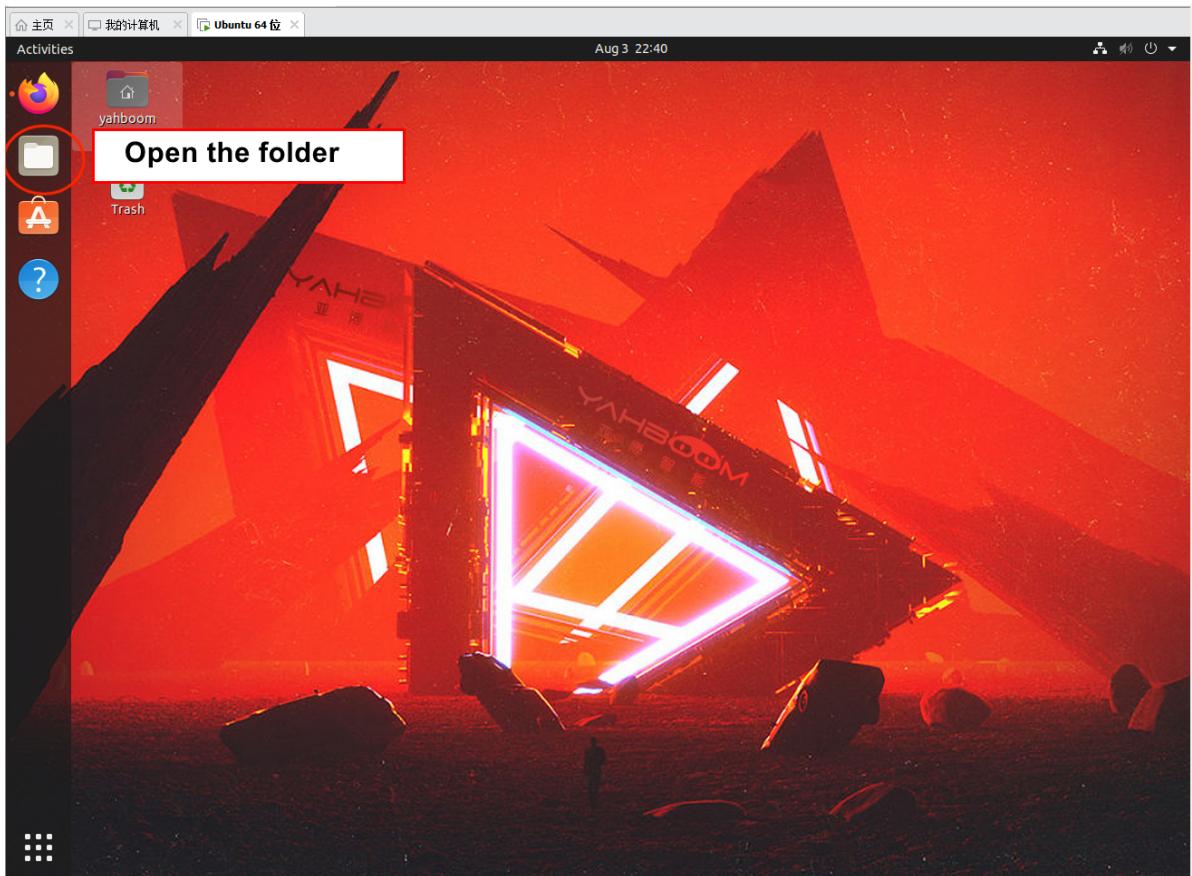
Open the virtual machine, enter the password: yahboom and press the Enter key to enter the system desktop.



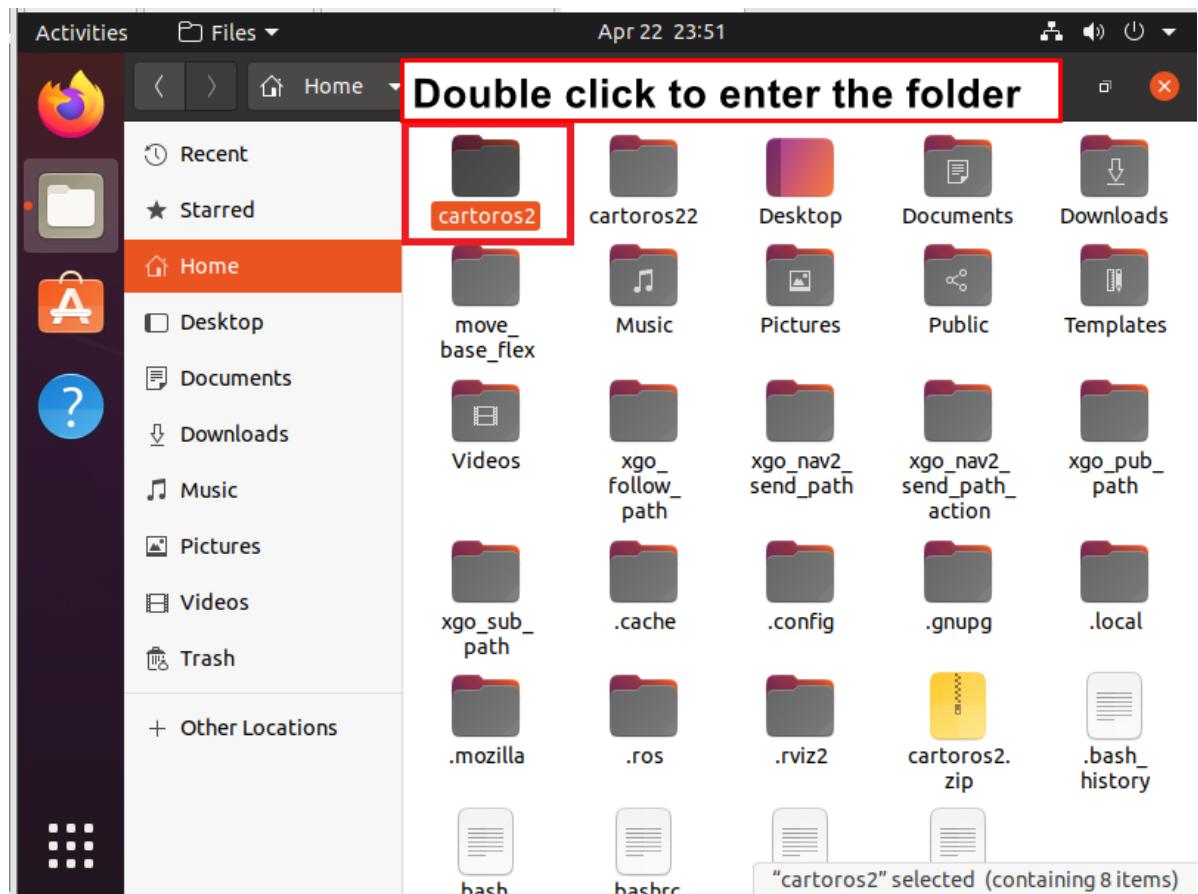
3. Start the robot dog mapping node

Note: This step is the same for virtual machines of PI4 and PI5 versions

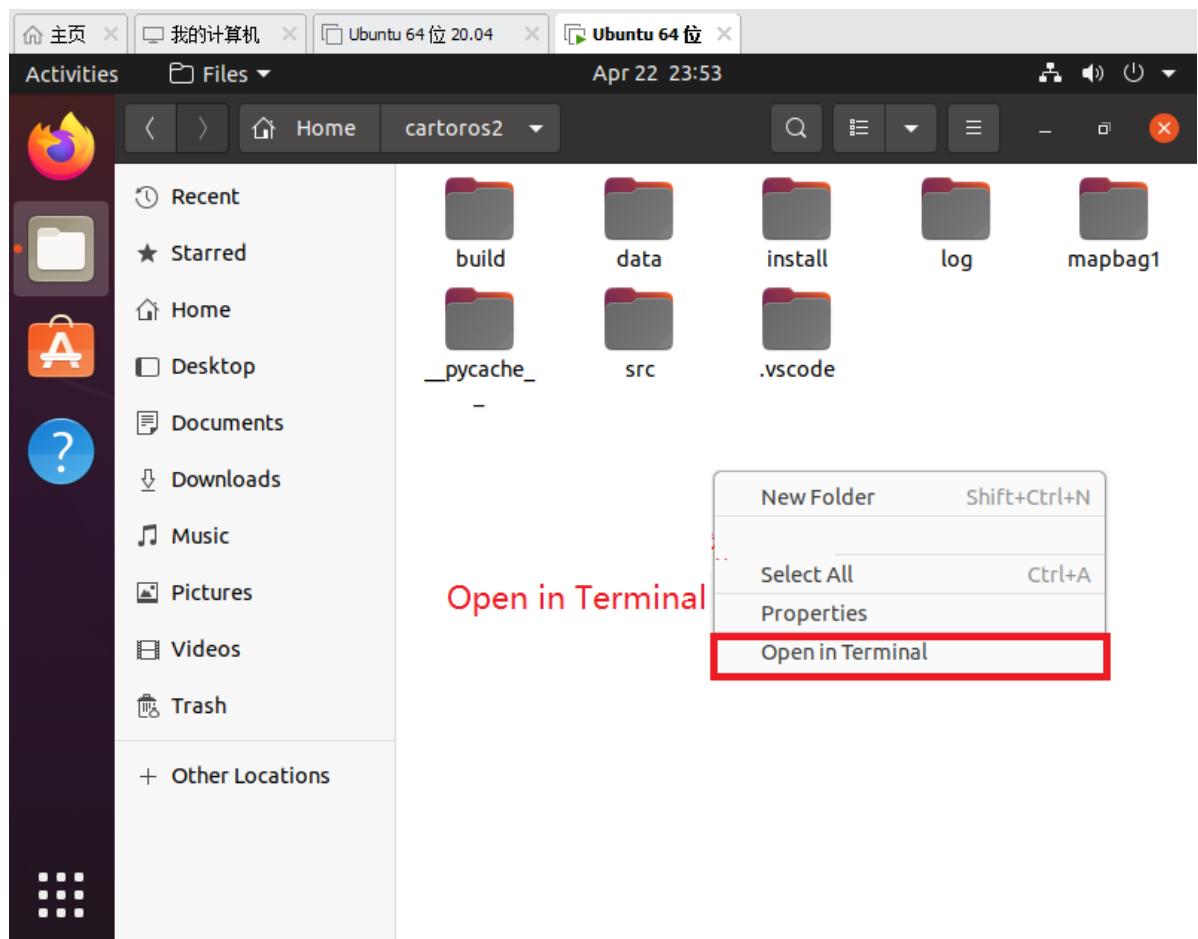
Enter the desktop system and open the folder.



Then double click on the cartoros2 folder



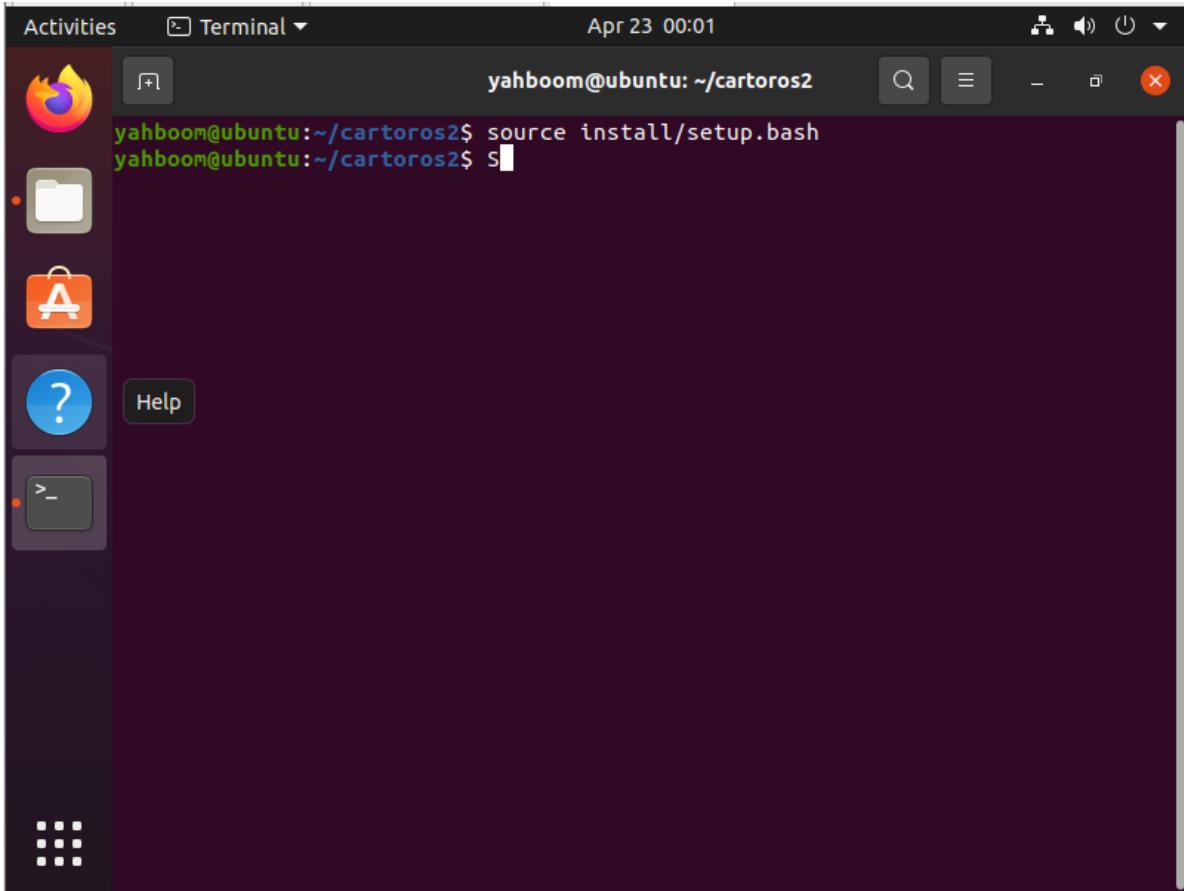
Then right-click in an empty space of the folder and select Open in Terminal



Then enter the following command in the terminal to activate the environment

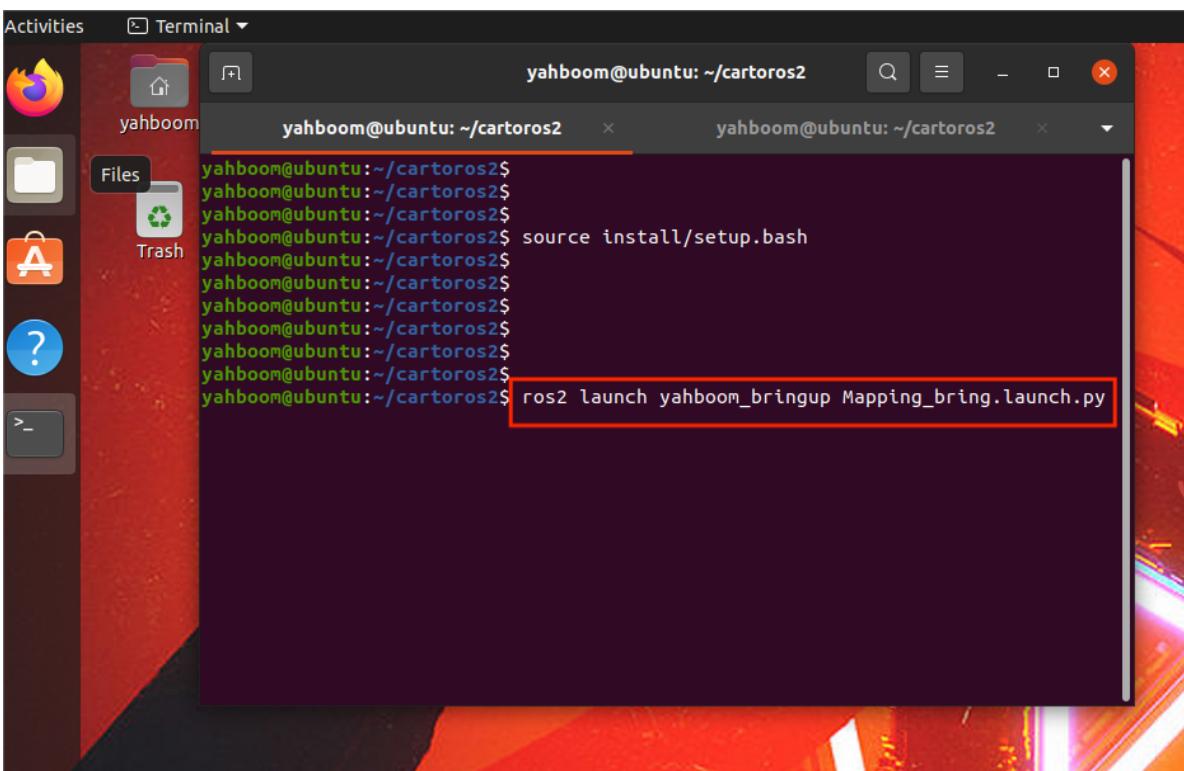
```
source install/setup.bash
```

After completing the input, press the Enter key.

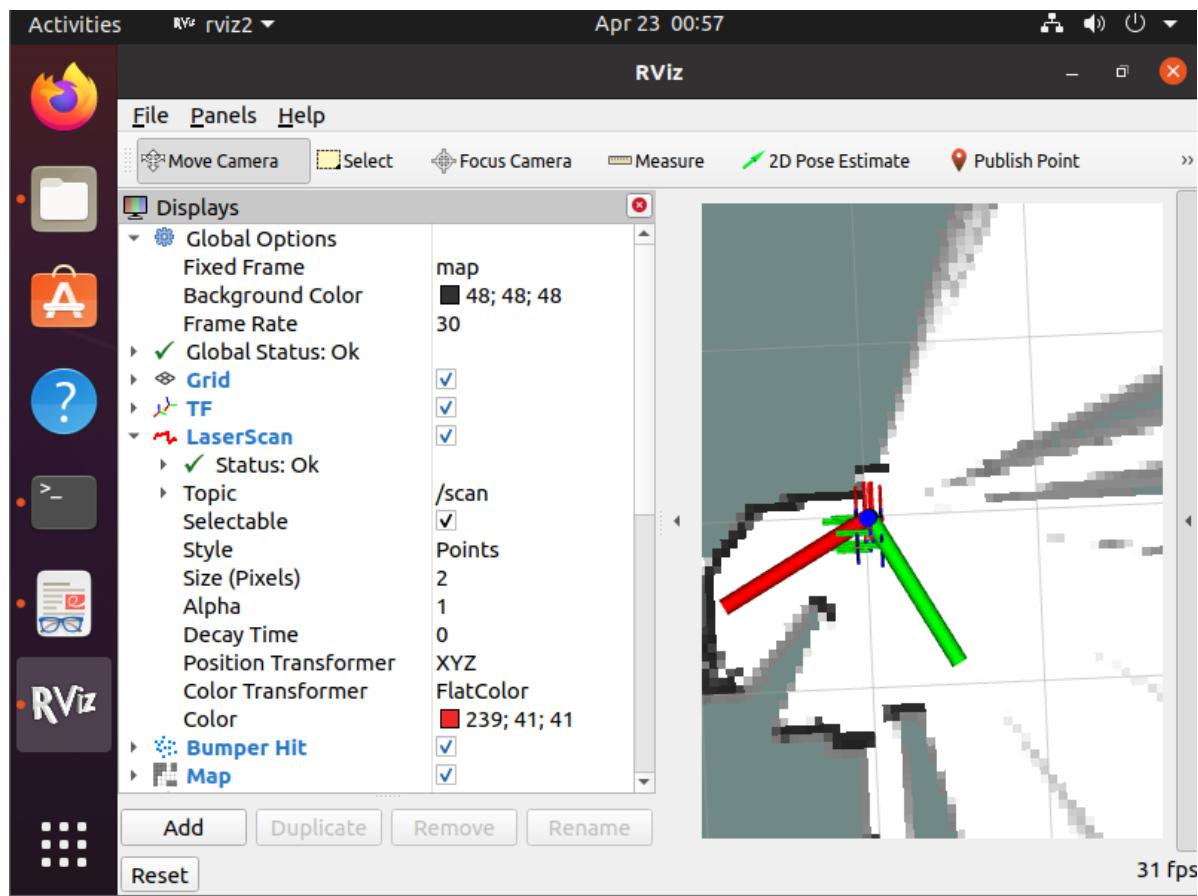


Then enter the command

```
ros2 launch yahboom_bringup Mapping Bring.launch.py
```



Then press the Enter key to start drawing.

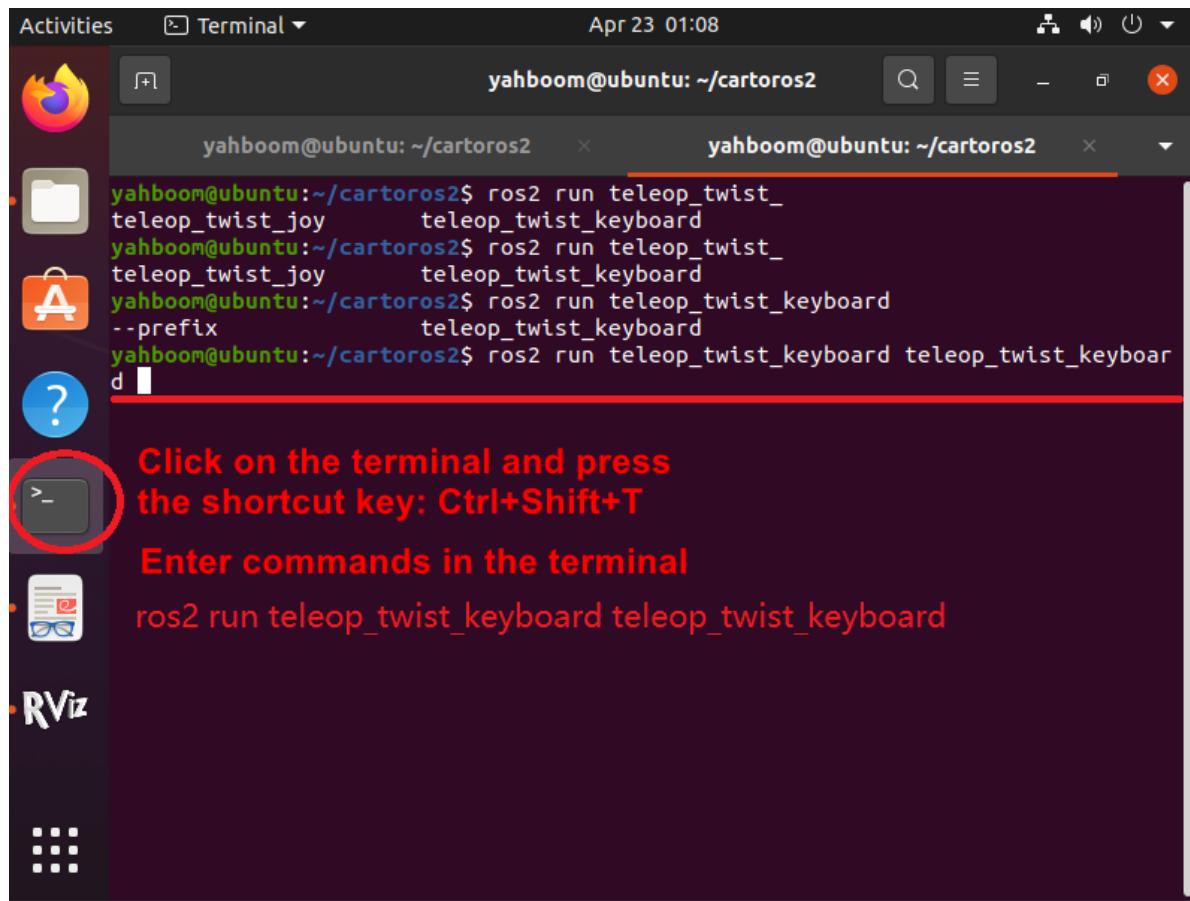


4. Control the robot dog to walk and build a map

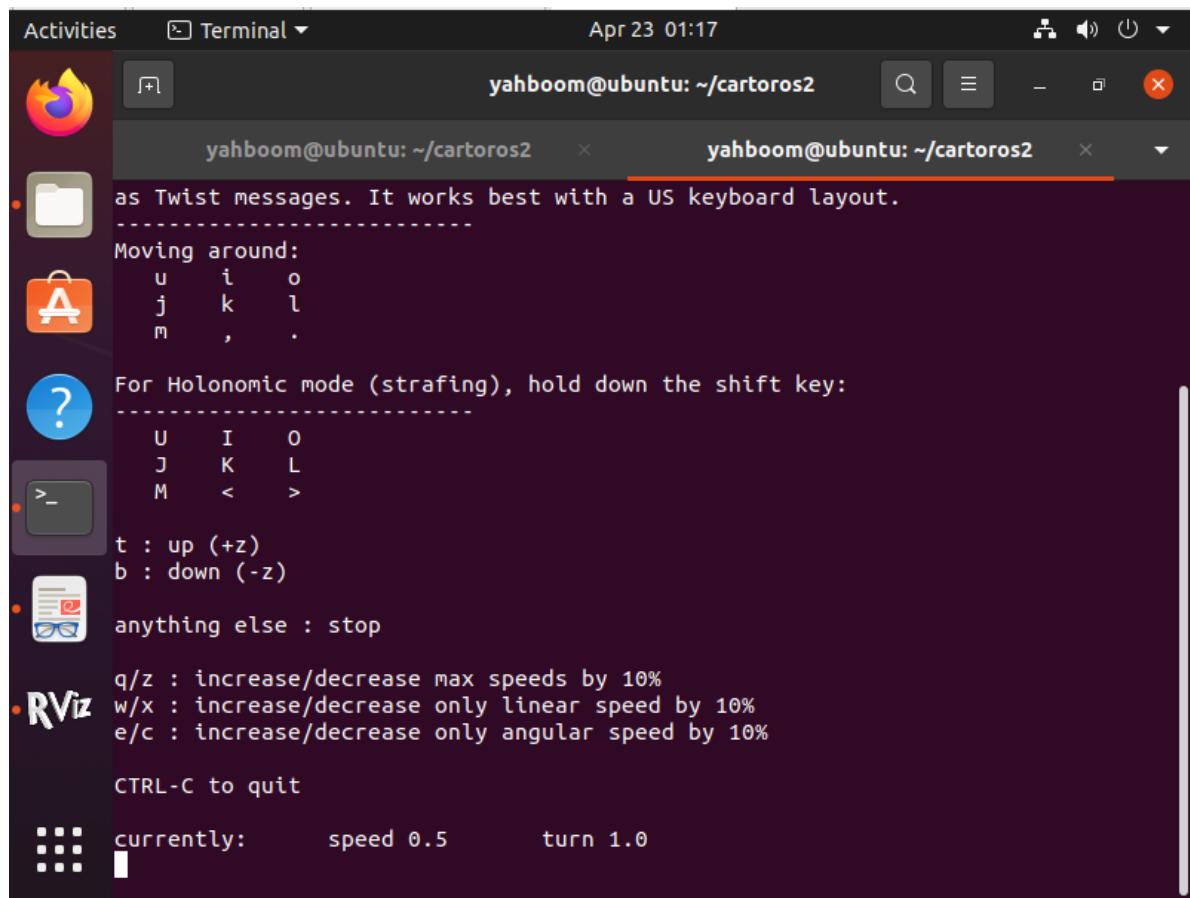
Click on the terminal and press the shortcut key: **ctrl + shift + t**

Enter the command in the terminal:

```
ros2 run teleop_twist_keyboard teleop_twist_keyboard
```



By pressing the Enter key, you can press the keyboard keys in the terminal to control the movement of the robot dog. Among them, I is forward, K is pause, and < is backward.



5. Save the map

When the map scan is completed, we press the shortcut key Ctrl shift t in the terminal to reopen a terminal window and enter the following command

First enter the following command to activate the environment

```
source install/setup.bash
```

Then enter the following command to stop mapping,

```
ros2 service call /finish_trajectory cartographer_ros_msgs/srv/FinishTrajectory "  
{trajectory_id: 0}"
```

Then enter the following command to save the pbstream file

```
#pi4  
ros2 service call /write_state cartographer_ros_msgs/srv/WriteState "{filename:  
'/home/yahboom/cartoros2/data/maps/mymap.pbstream'}"  
  
#pi5  
ros2 service call /write_state cartographer_ros_msgs/srv/WriteState "{filename:  
'/home/yahboom/yahboomcar_ws/maps/mymap.pbstream'}"
```

The path of the filename parameter is the path where the pbstream file of the map is saved.

Finally enter the following command to convert the pbstream file to a pgm file.

```
#pi4  
ros2 run cartographer_ros pbstream_to_ros_map_node -  
map_filestem=/home/yahboom/cartoros2/data/maps/mymap -  
pbstream_filename=/home/yahboom/cartoros2/data/maps/mymap.pbstream -  
resolution=0.05  
  
#pi5  
ros2 run cartographer_ros cartographer_pbstream_to_ros_map -  
map_filestem=/home/yahboom/yahboomcar_ws/maps/mymap -  
pbstream_filename=/home/yahboom/yahboomcar_ws/maps/mymap.pbstream -  
resolution=0.05
```

