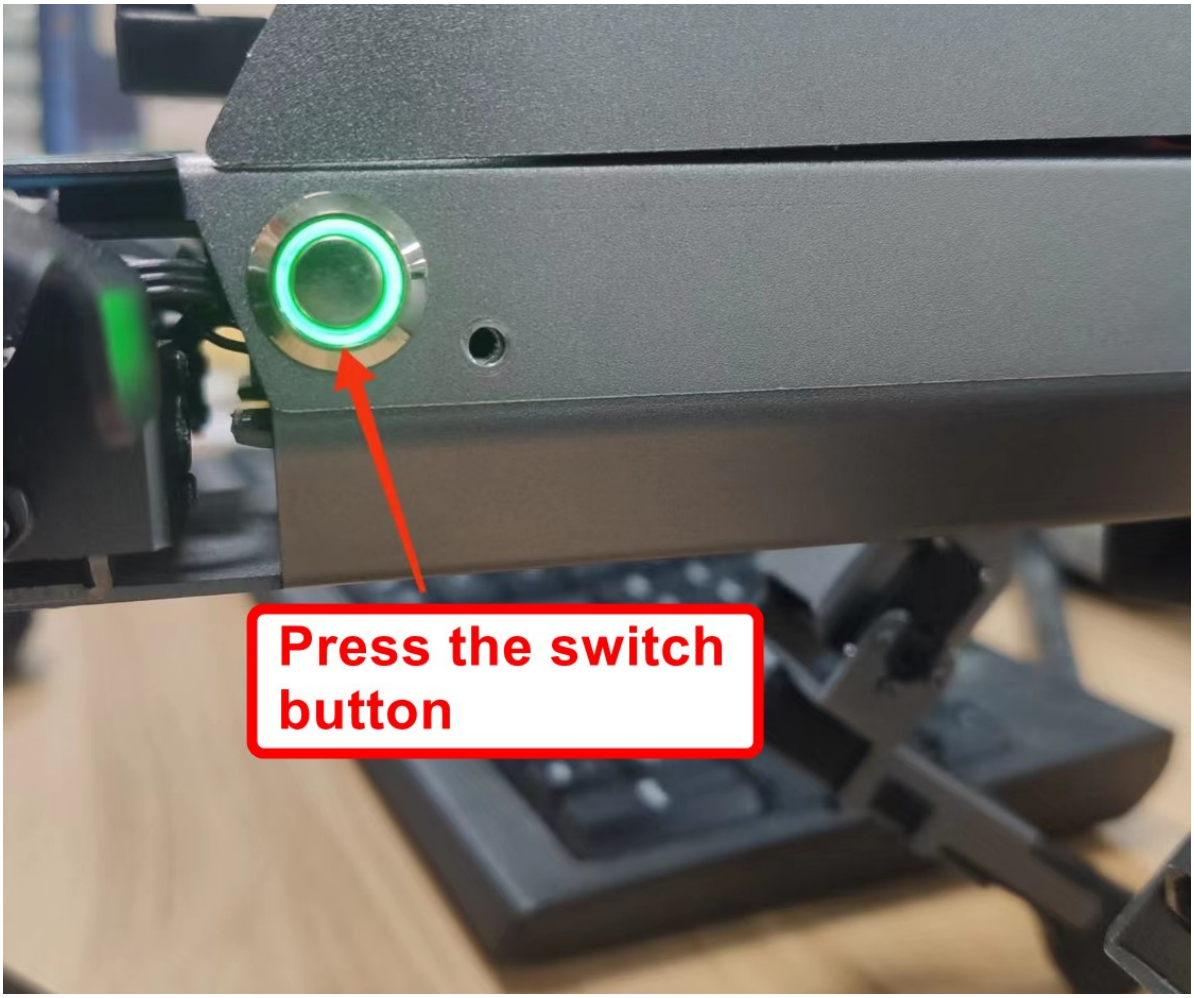
Voice control mechanical dog multi-point navigation

Quick use

1. Power on DOGZILLA

First, we turn on the switching power supply of the mechanical dog and start the mechanical dog



After starting, we can view the IP address on the small screen of the robot dog.

2. Start DOGZILLA chassis

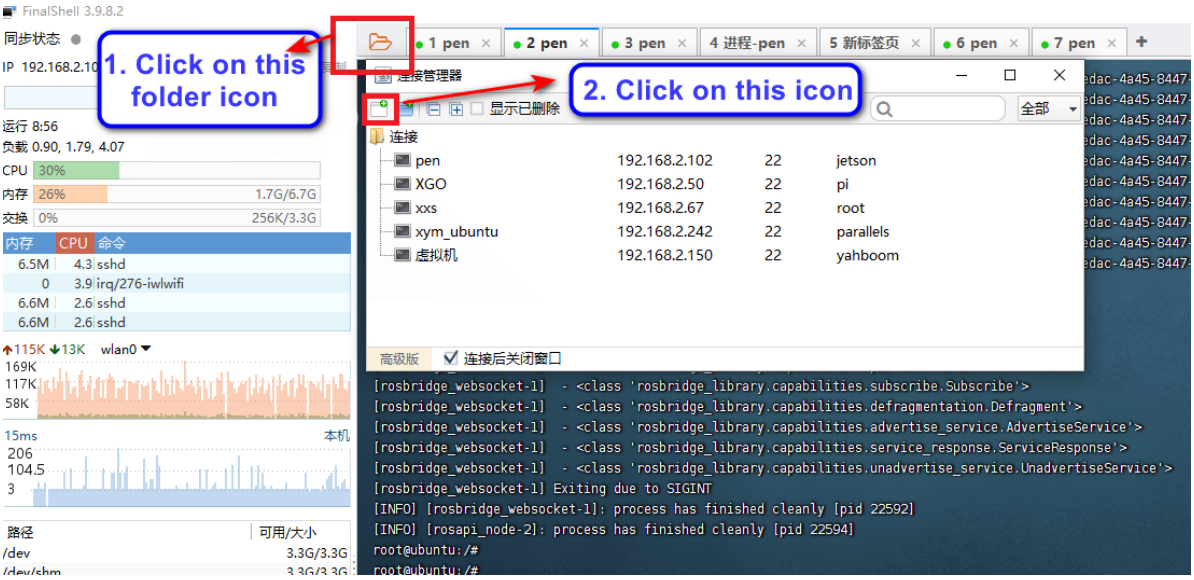
PI4 version steps:

For multi-machine communication ID modification, please refer to the tutorial: 14. Radar mapping navigation\6. Obtaining the status of the mechanical dog in the ROS2 environment\Acquiring the real joint data of the mechanical dog in the ROS2 environment.pdf

Then use the ssh terminal to connect to the robot dog.

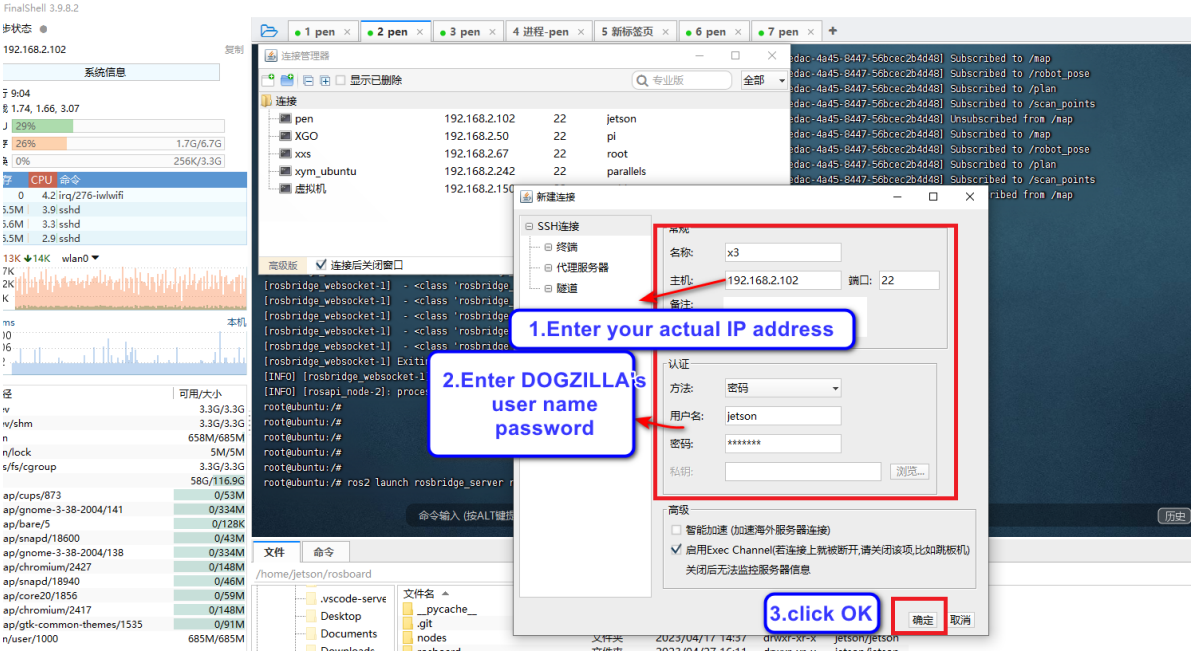
Note: The IP address used when writing this tutorial: 192.168.2.102 User name: pi Password: yahboom The actual IP address shall prevail when used.

Open the shell tool. The shell tool I use here is FinalShell. Enter username, password, port, connection name and other information.



Select ssh connection to create a new ssh connection

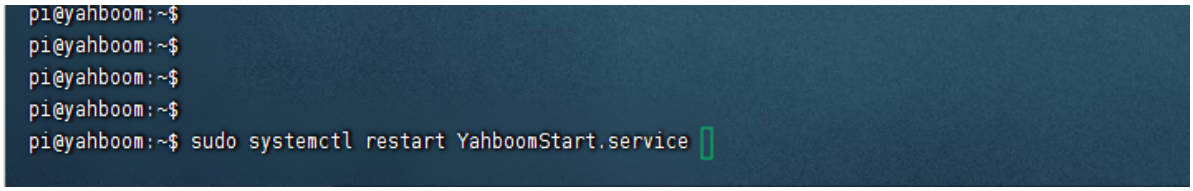
Here the username is pi, the password is yahboom, and the ip address is the IP address of the real robot dog.



Select the ssh connection you just created here.

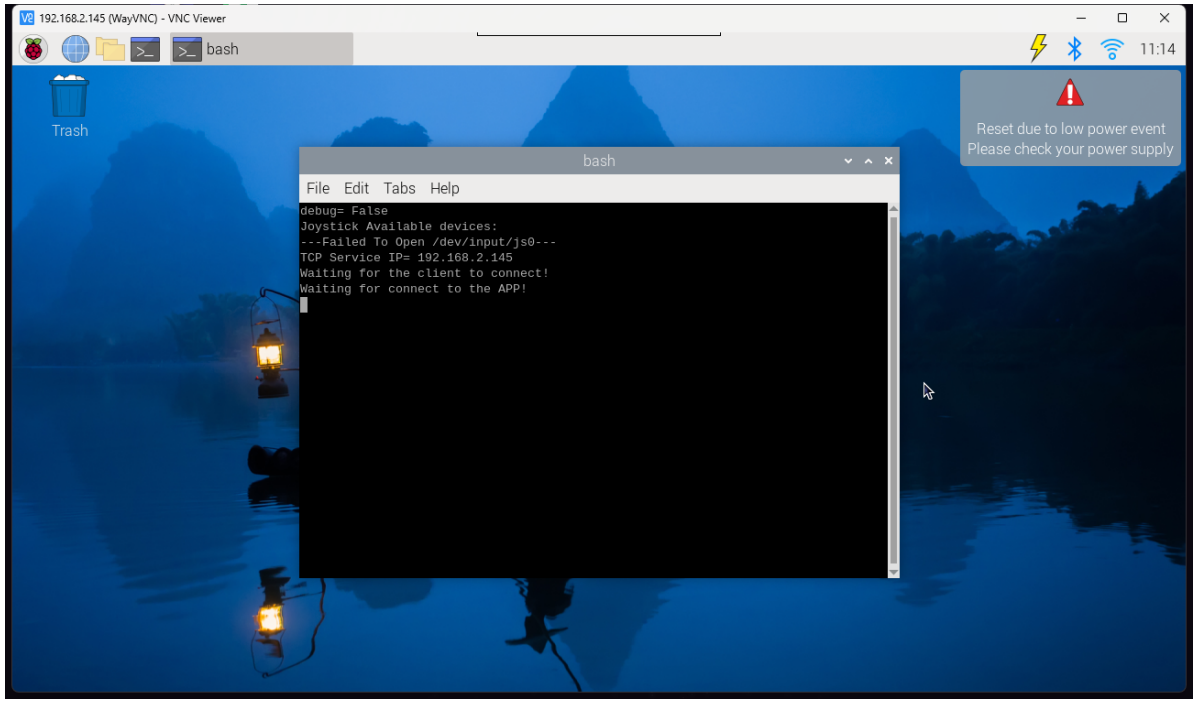
Then enter the command in the terminal to start the chassis task.

```
sudo systemctl restart YahboomStart.service
```



PI5 version steps:

After the mechanical dog is started, use the vnc software to remotely connect to the mechanical dog through the IP address on the OLED (**For specific steps, please see "Remote Login Operation"**).



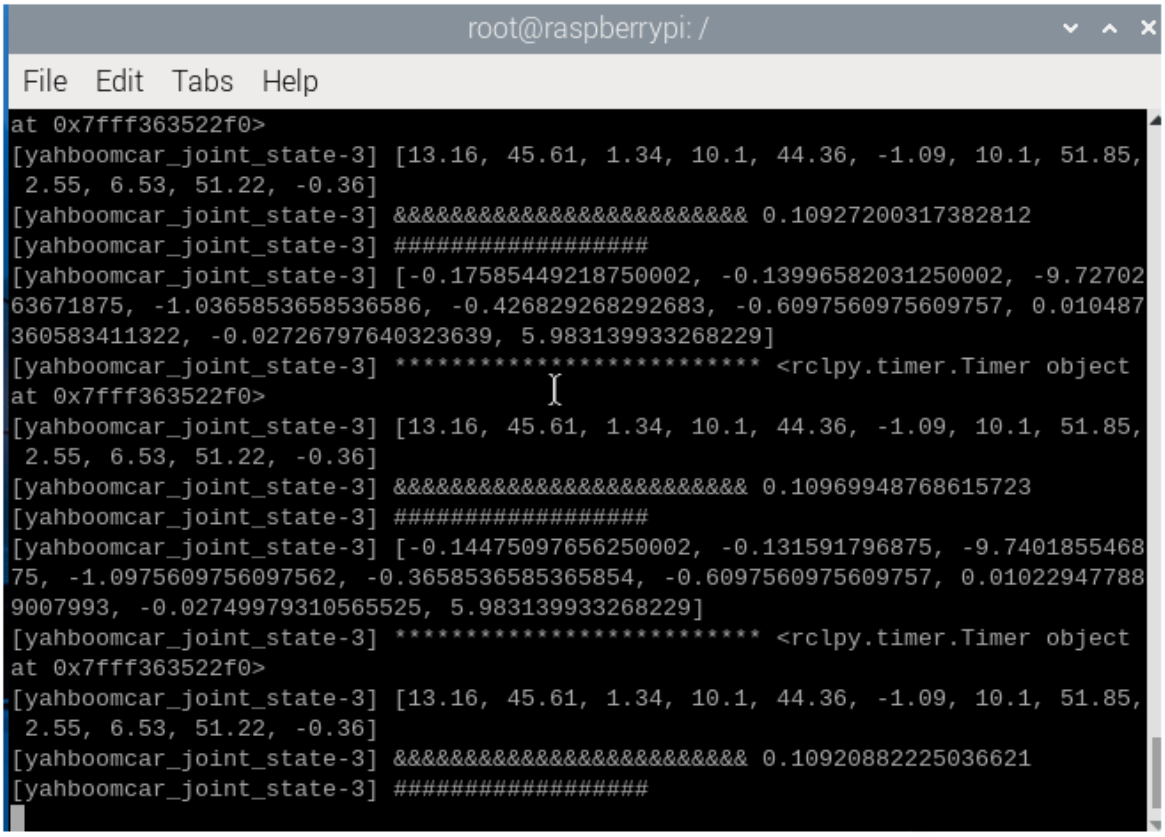
Then ctrl+c closes the large program and enter the following command to enter docker:

```
./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 mechanical dog joint status nodes.

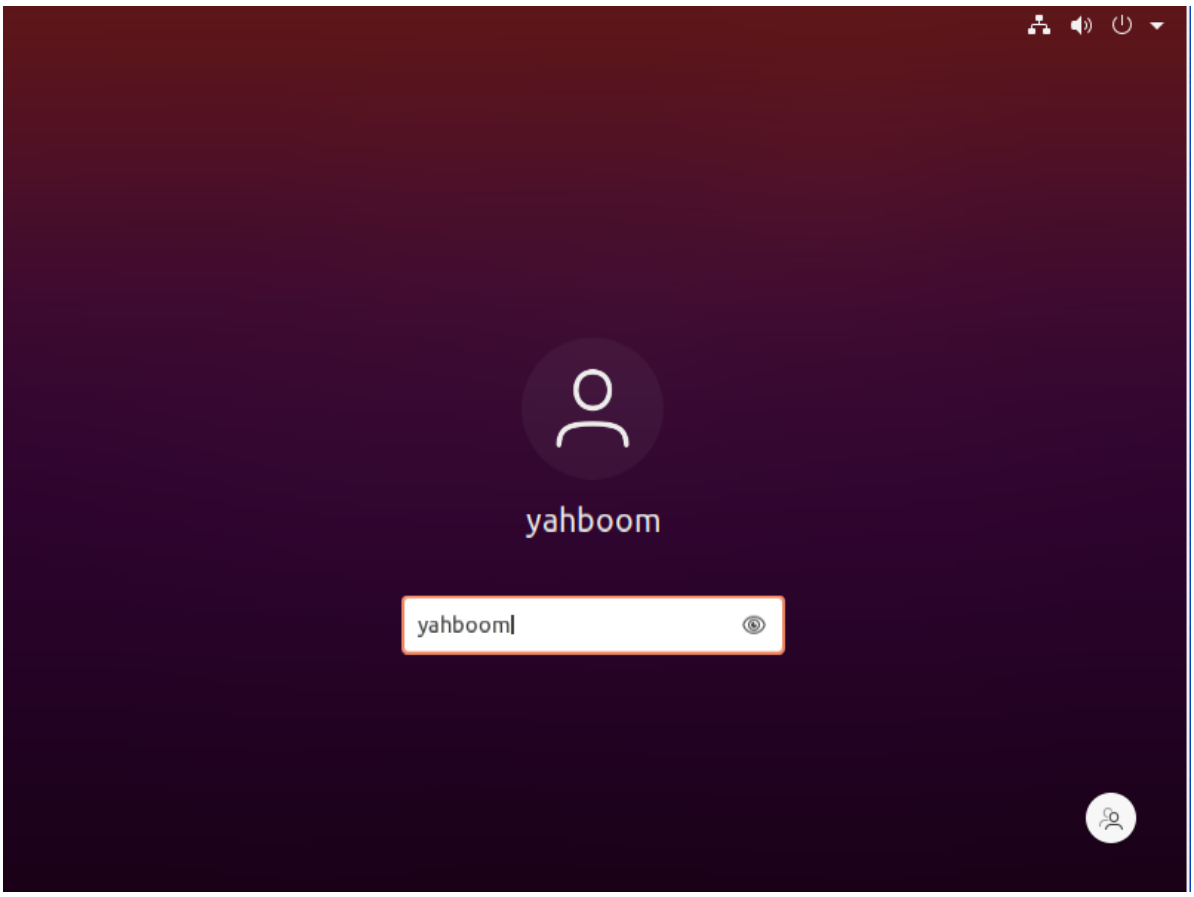
```
ros2 launch bringup Navigation_bringup.launch.py
```



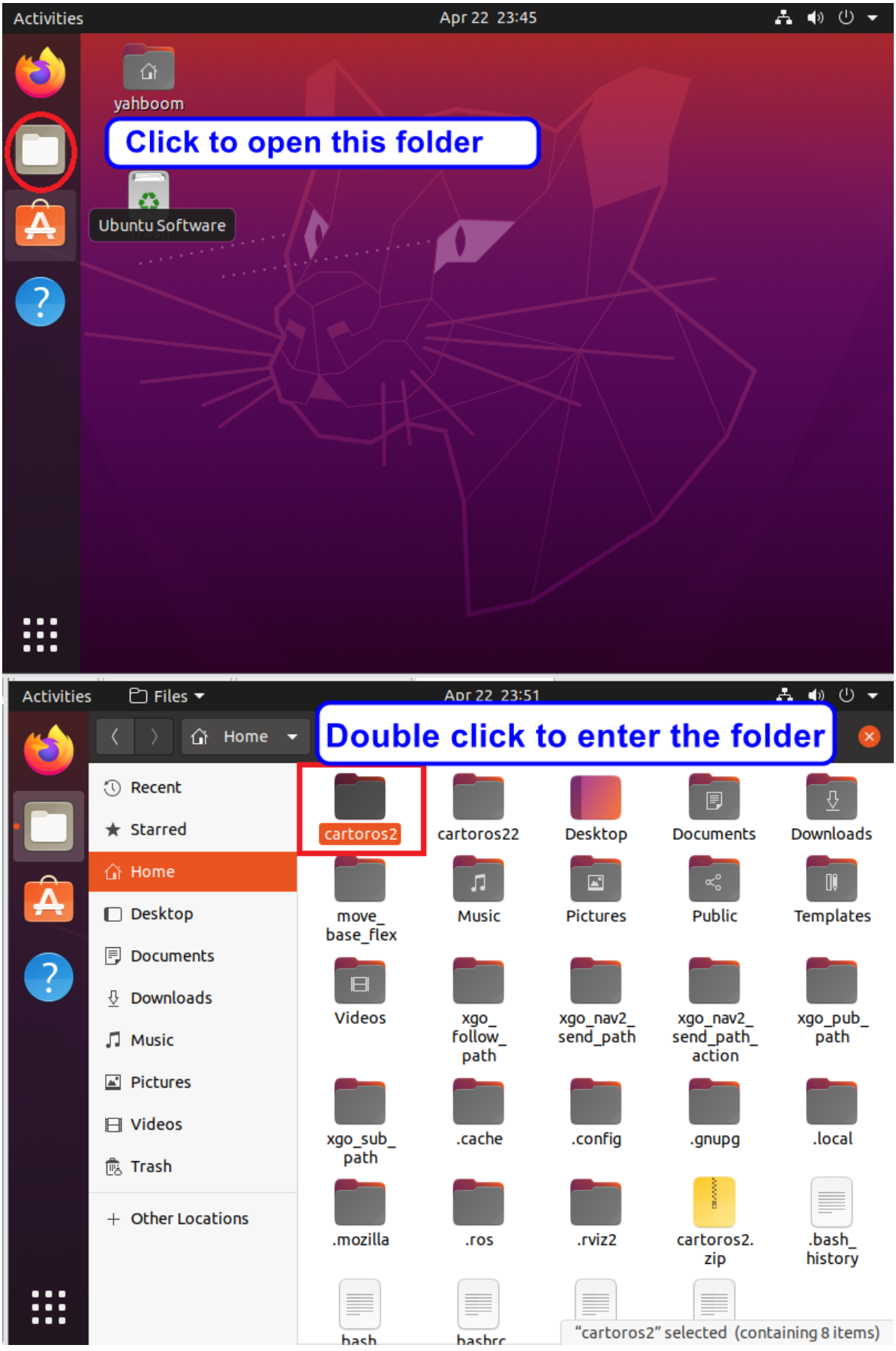
3. Start the navigation program

PI4 version steps:

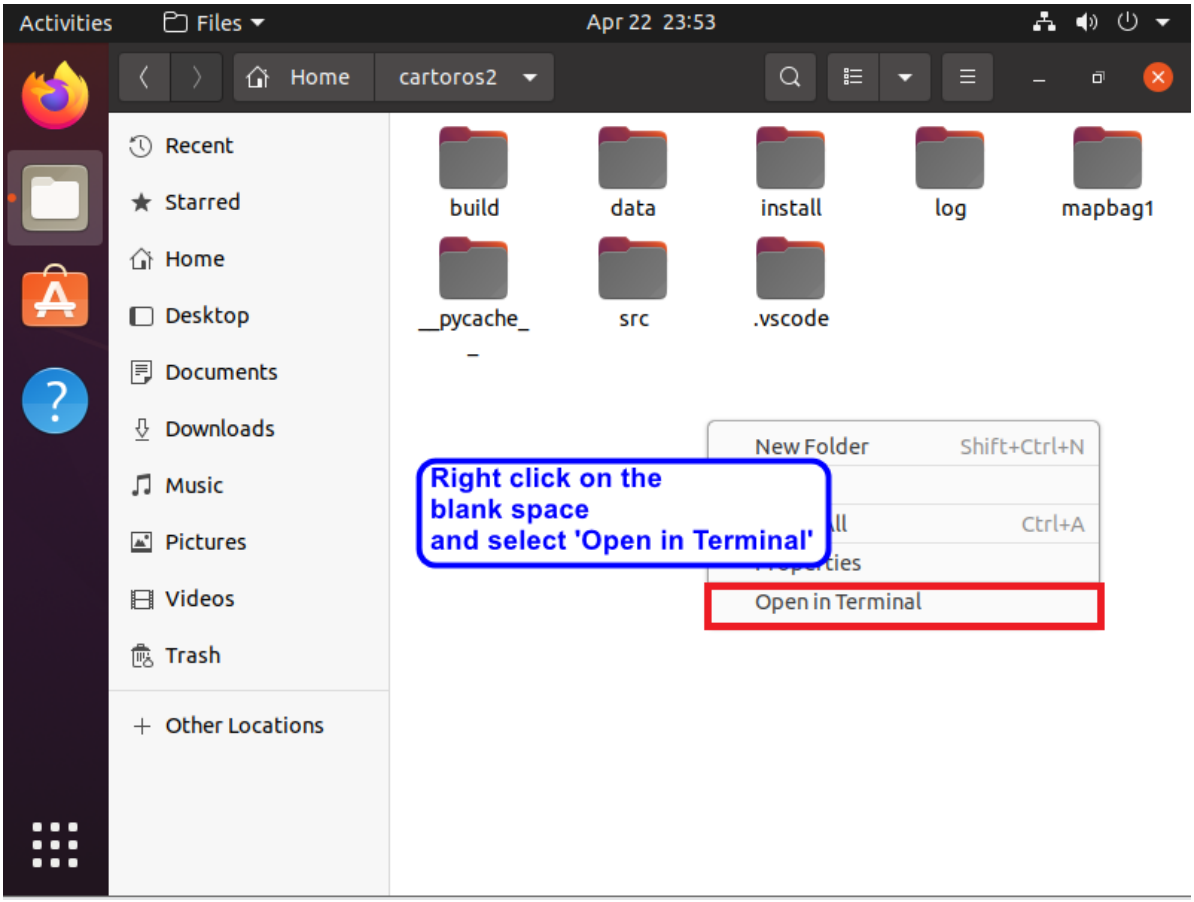
Open the virtual machine and enter the user name yahboom and the password yahboom.



Click on the folder to open the cartoros2 folder.

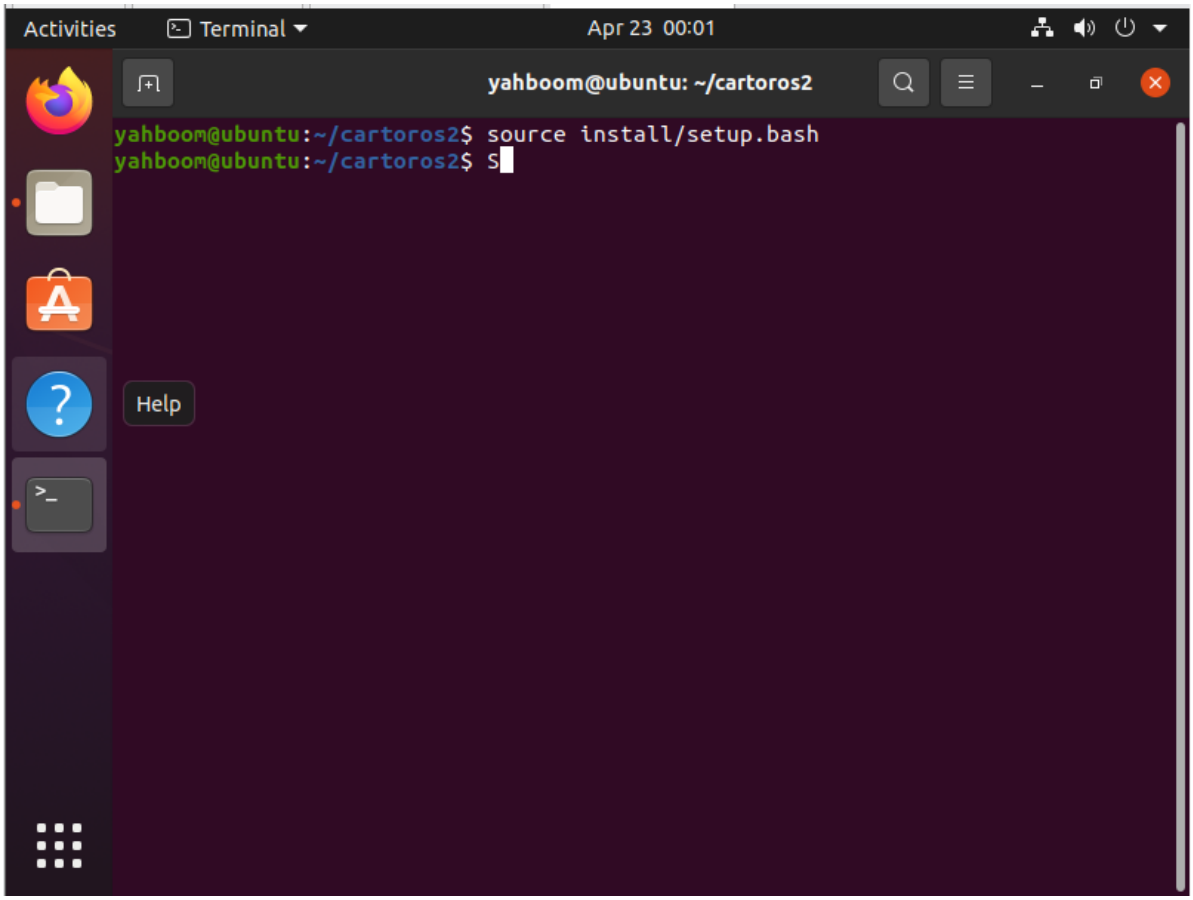


Open the terminal under the folder



Then enter the following command

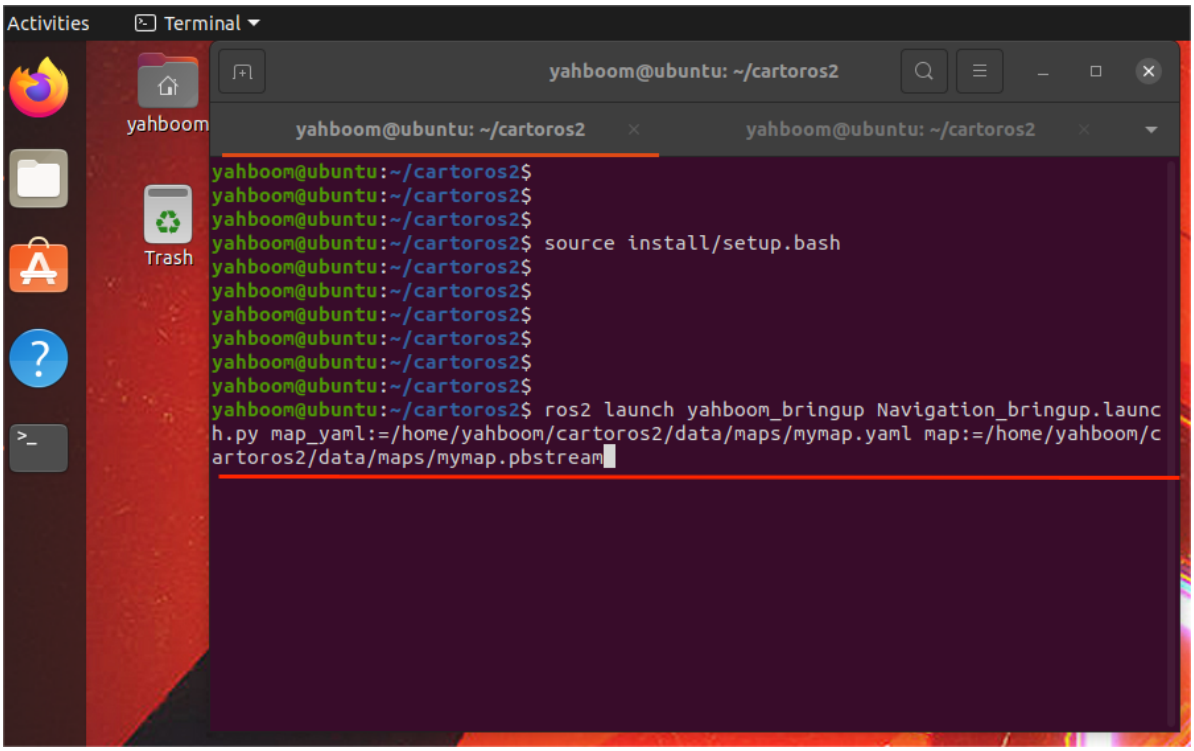
```
source install/setup.bash
```



Then start the navigation program, which is to first place the mechanical dog at the origin of the mapping. Then enter the command in the virtual machine terminal:

```
ros2 launch yahboom_bringup Navigation_bringup.launch.py
map_yaml:=/home/yahboom/cartoros2/data/maps/mymap.yaml
map:=/home/yahboom/cartoros2/data/maps/mymap.pbstream
```

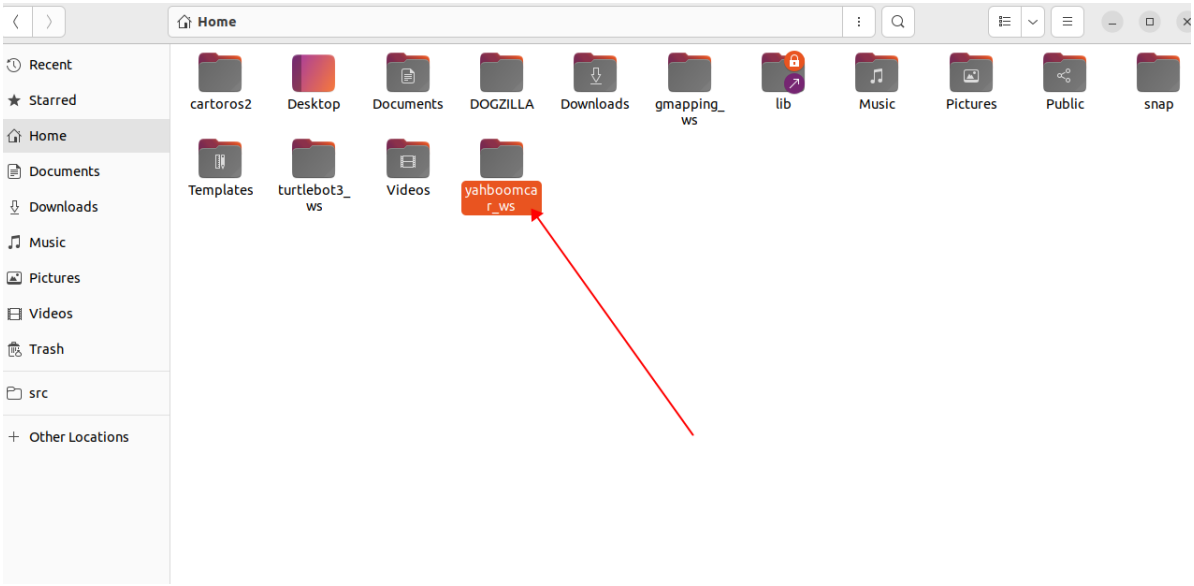
Note: The map files xxx.yaml and xxx.pbstream here are the two files we saved in the previous mapping tutorial.



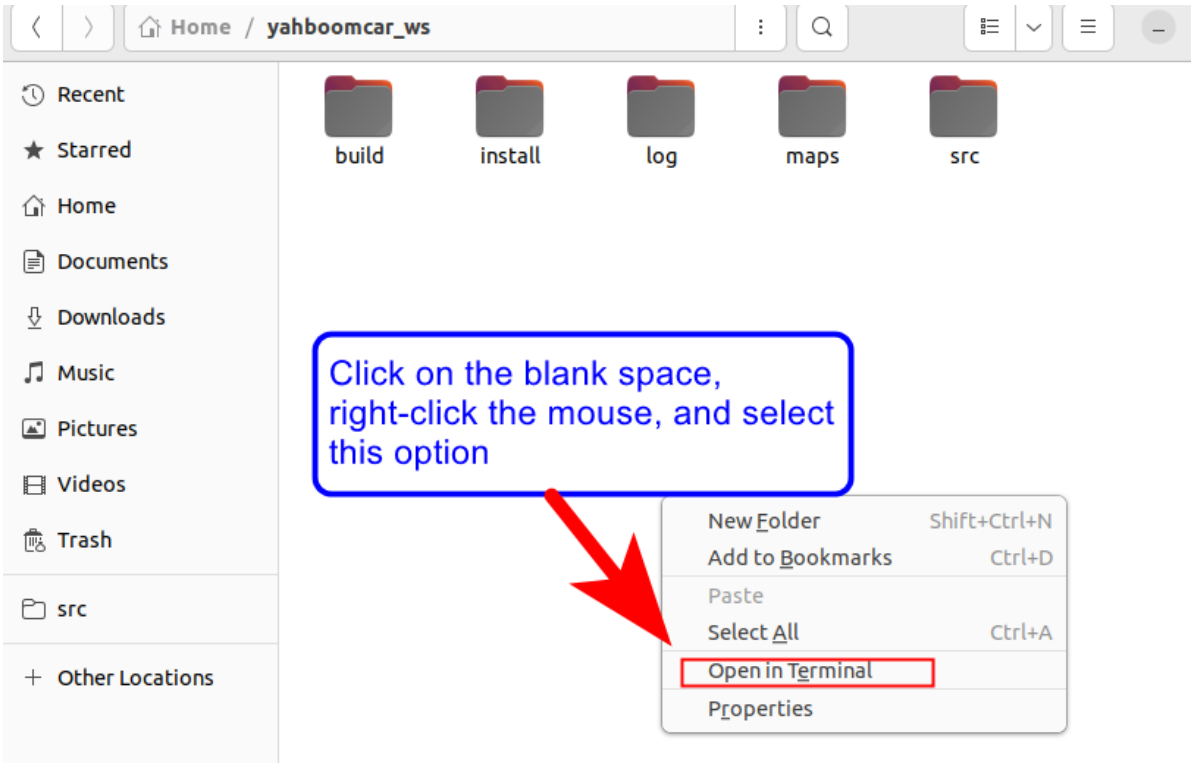
PI5 version steps:

Enter the desktop system and open the folder.

Double-click to open the yahboomcar_ws folder



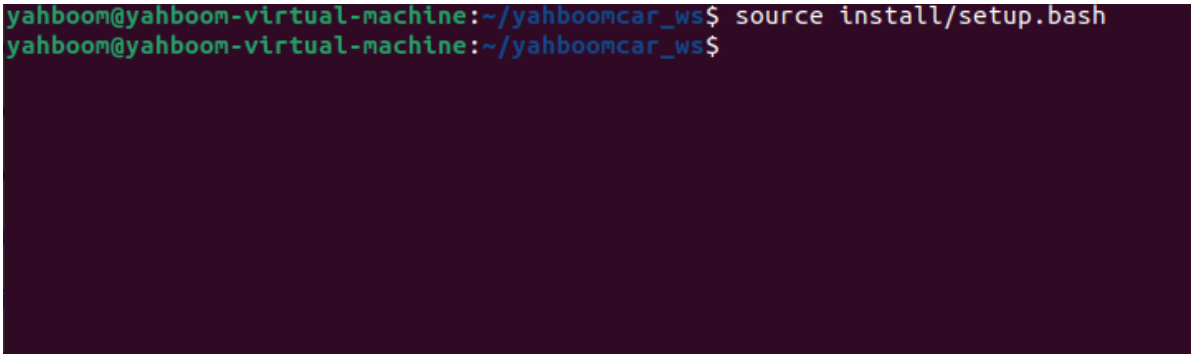
Then right-click in an empty space of the folder and select Open in Terminal



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

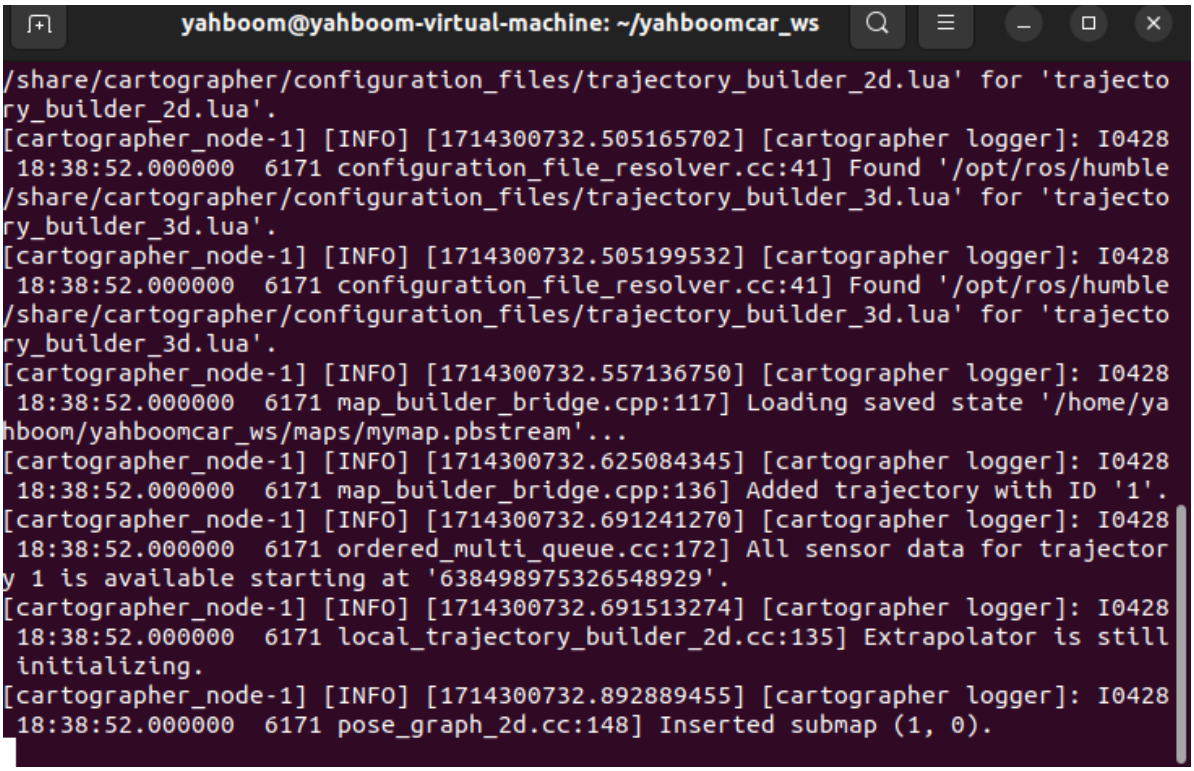
```
source install/setup.bash
```

After completing the input, press Enter.



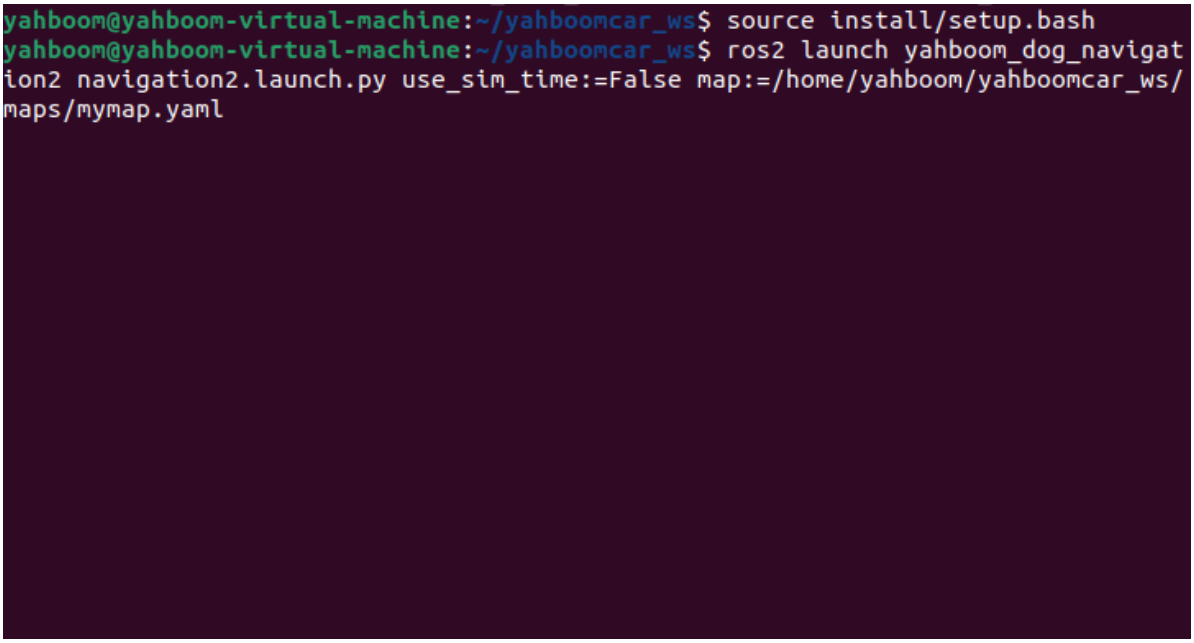
Then enter the command to start the relocation function

```
ros2 launch yahboom_dog_cartographer localization_imu_odom.launch.py
load_state_filename:=/home/yahboom/yahboomcar_ws/maps/mymap.pbstream
```

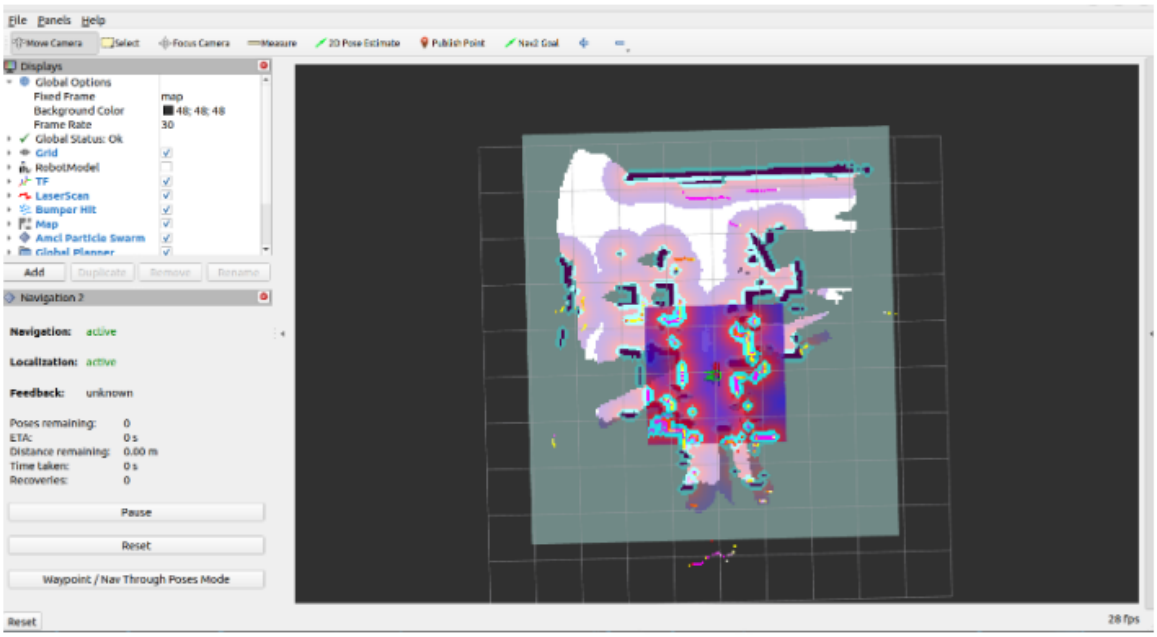


Repeat the above steps to open the terminal, reopen a terminal and enter navigation commands.

```
ros2 launch yahboom_dog_navigation2 navigation2.launch.py use_sim_time:=False
map:=/home/yahboom/yahboomcar_ws/maps/mymap.yaml
```



Then press the Enter key to navigate. **(Because the mechanical dog itself does not provide odom, we need to position the mechanical dog at the coordinate origin when constructing the map)**



4. Multi-point navigation node startup

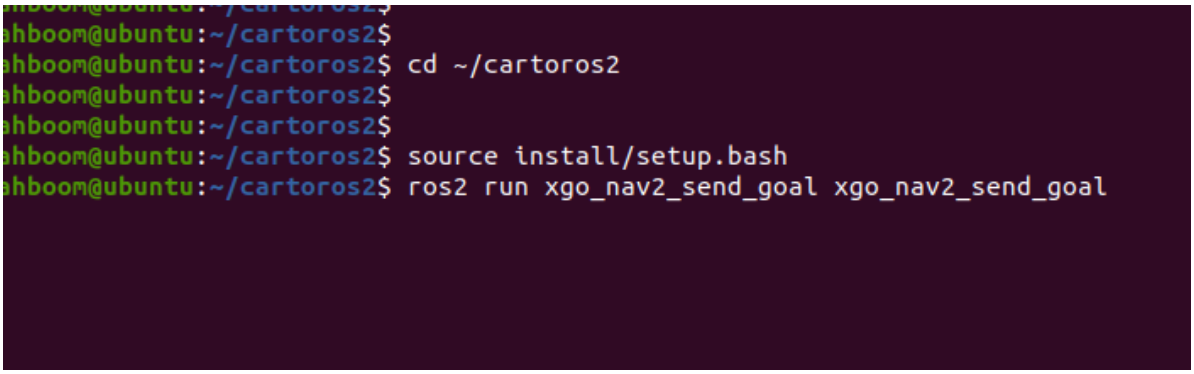
The steps are the same for PI4 and PI5 versions:

After the navigation module is started, we open a terminal and start the multi-point navigation node. Enter the command in the newly opened terminal:

```
cd ~/cartoros2
```

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

```
ros2 run xgo_nav2_send_goal xgo_nav2_send_goal
```

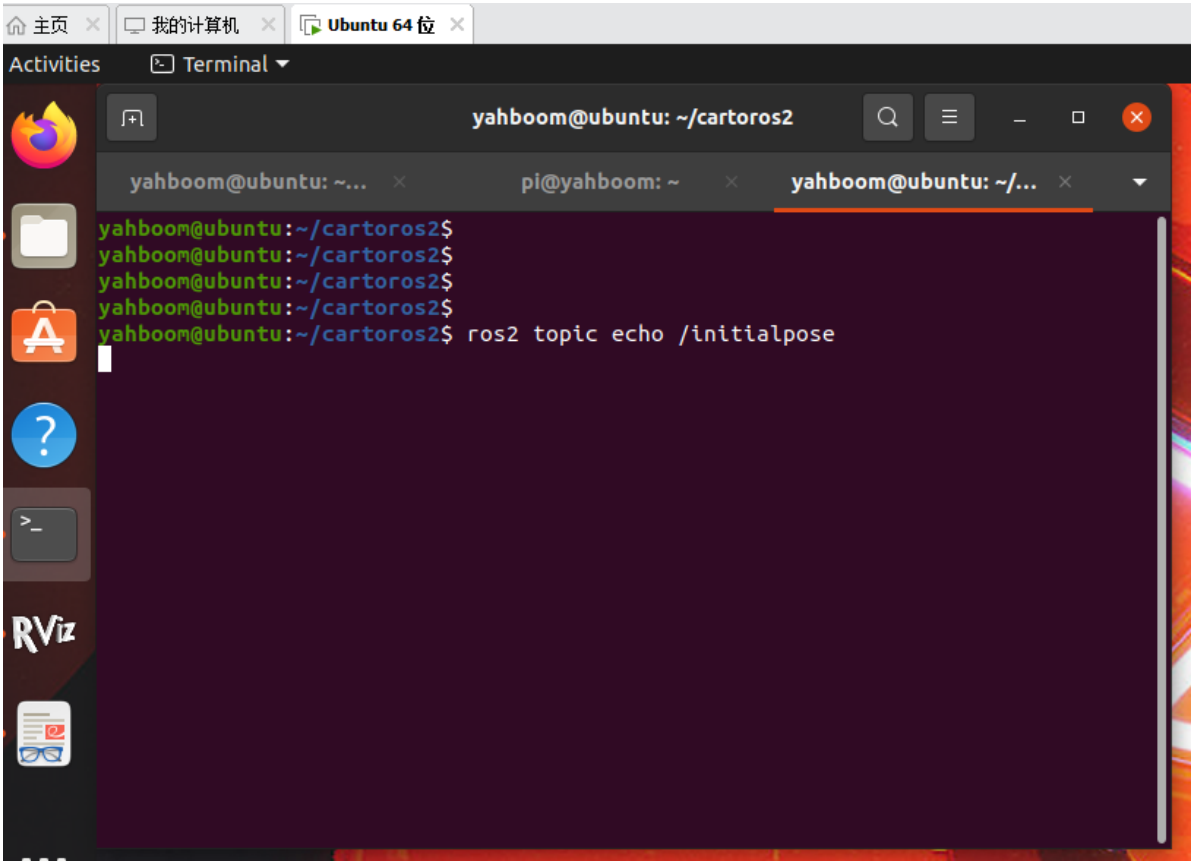


5. Obtaining the navigation point position

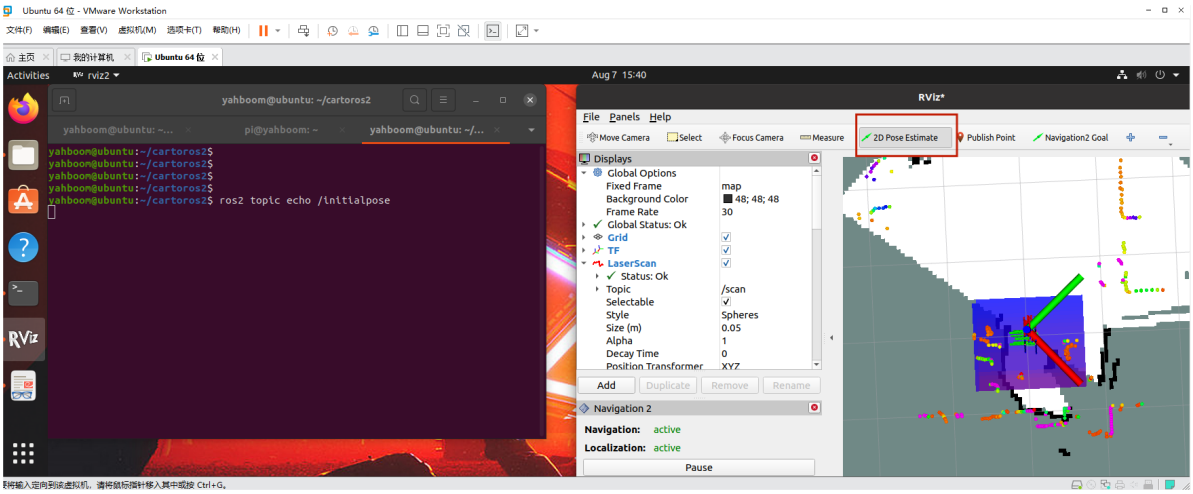
The steps are the same for PI4 and PI5 versions:

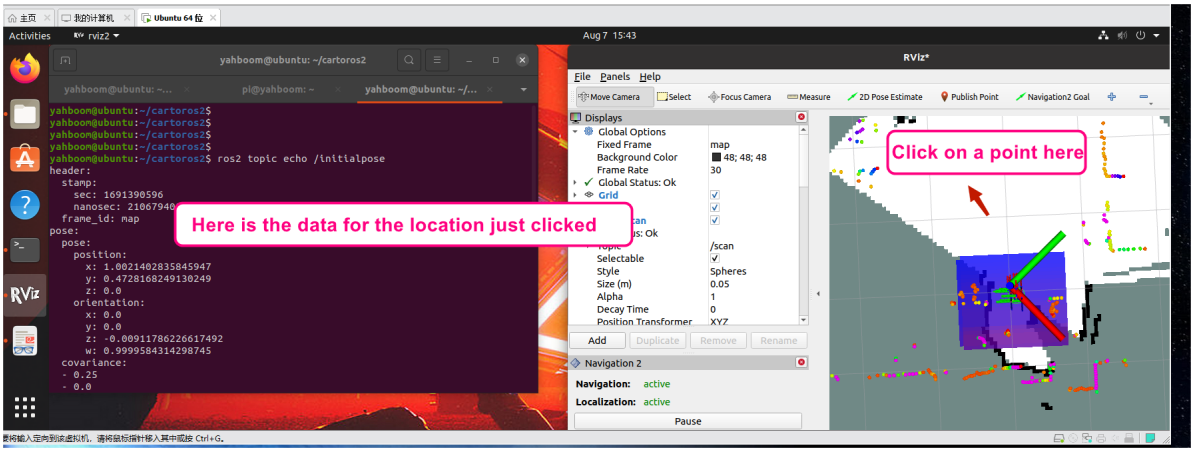
In the virtual machine terminal, press the shortcut key: ctrl + alt + T to reopen a terminal and subscribe to the /initialpose topic in the terminal to obtain the points we set in rviz. The command is as follows:

```
ros2 topic echo /initialpose
```



Click the button "2D Pose Estimate" in rviz and click a point on the map where you can walk. At this time, the location of the point just clicked will be printed in the terminal.



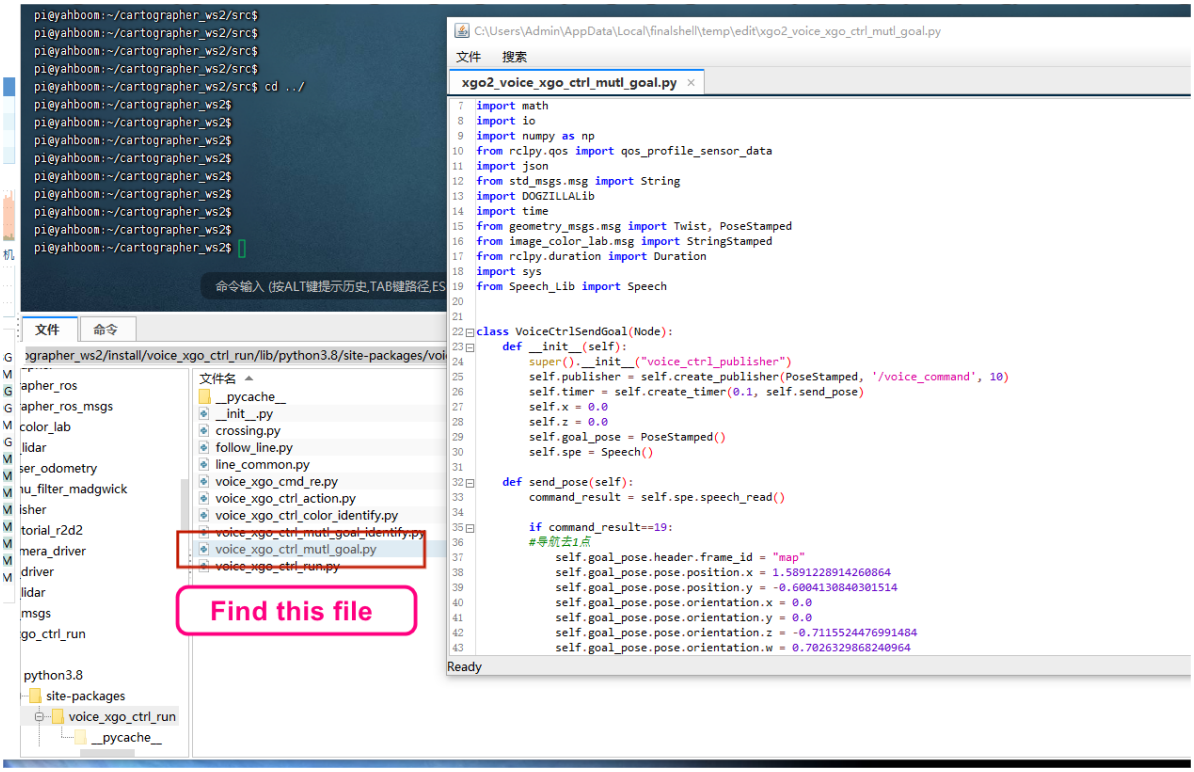


6. Voice navigation point modification

PI4 version steps:

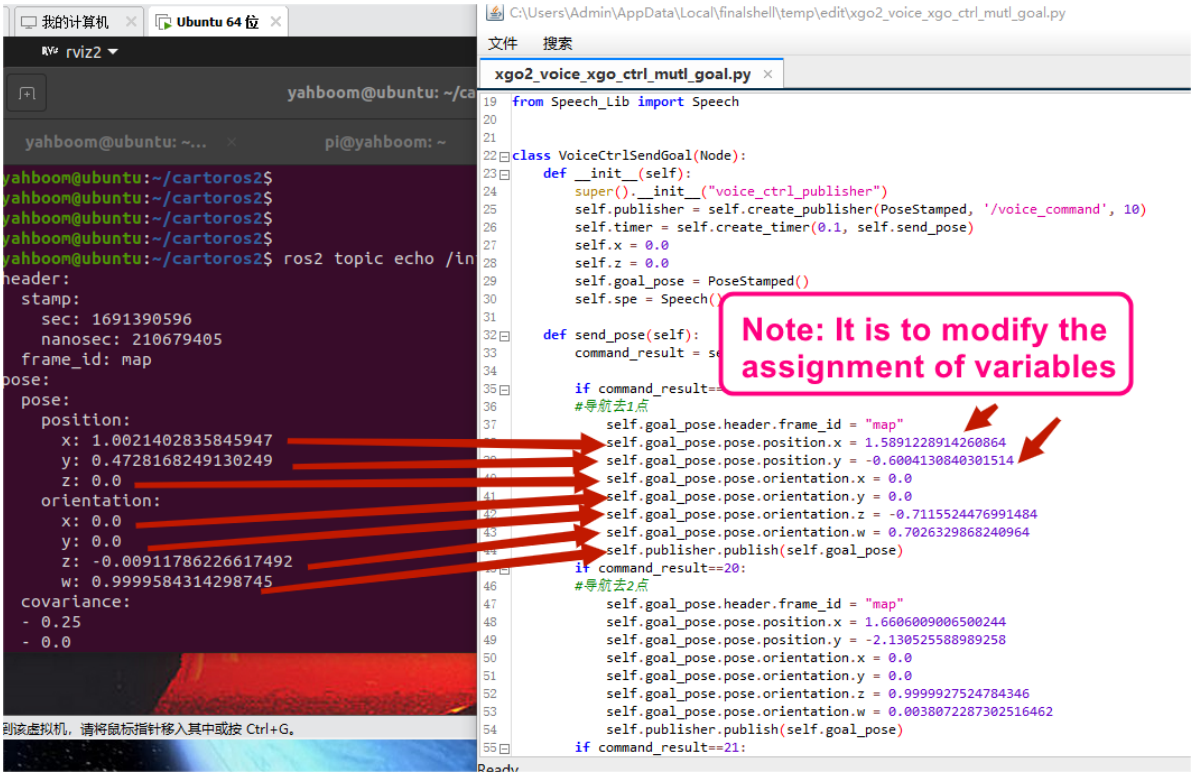
According to the second step, open a new shell terminal, connect to the mechanical dog, and open the file in the path below.

```
#pi4
/home/pi/cartographer_ws2/install/voice_xgo_ctrl_run/lib/python3.8/site-packages/voice_xgo_ctrl_run
```



Then we modify the annotated part and modify the corresponding position.x, position.y, position.z, orientation.x, orientation.y, orientation.z.orientation.w values according to the data subscribed to rviz. As shown below:

```
self.goal_pose.pose.position.x = 1.5891228914260864
self.goal_pose.pose.position.y = -0.6004130840301514
self.goal_pose.pose.orientation.x = 0.0
self.goal_pose.pose.orientation.y = 0.0
self.goal_pose.pose.orientation.z = -0.7115524476991484
self.goal_pose.pose.orientation.w = 0.7026329868240964
```

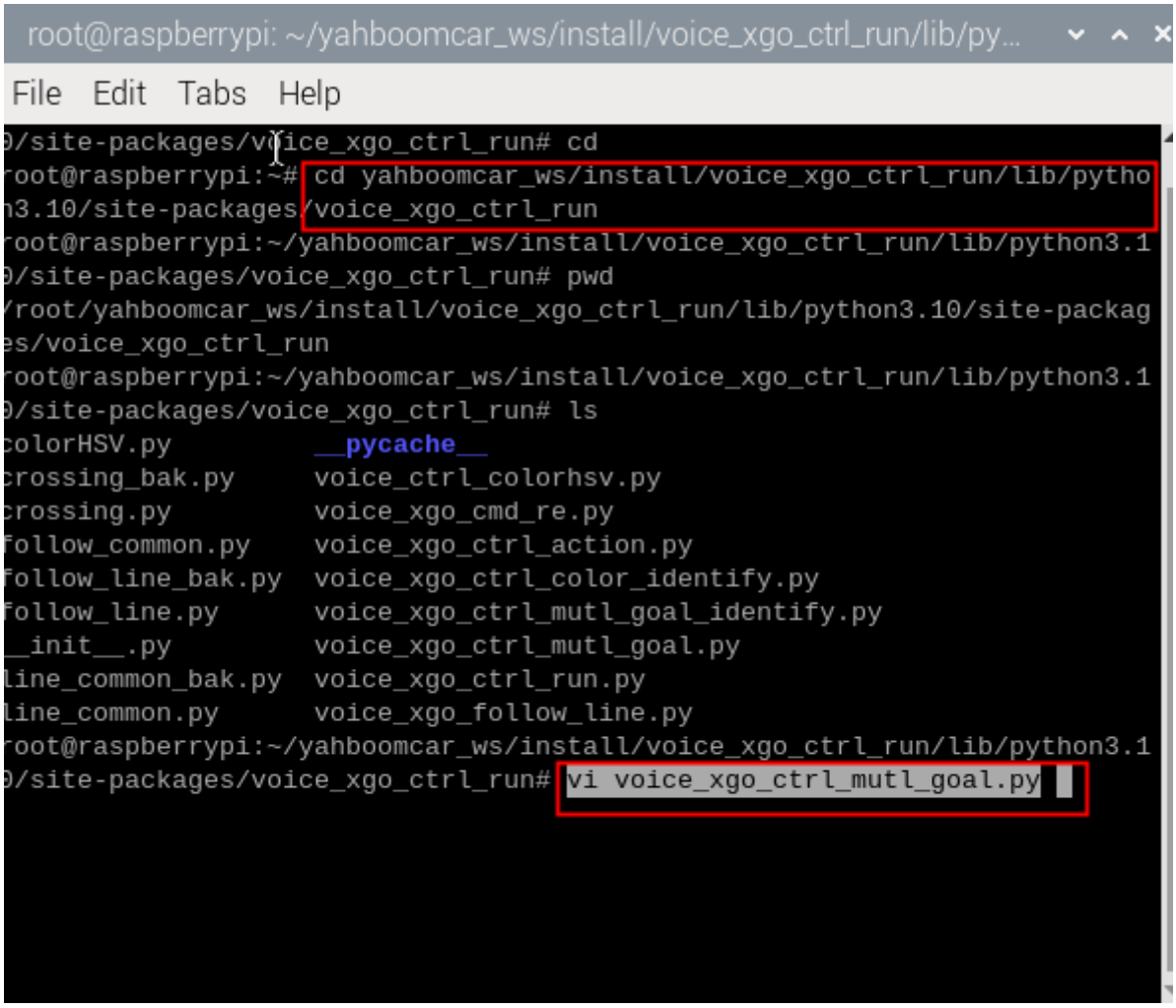


We can see that there are 5 points, but we only need to modify 4 according to the above method, because the fifth point is the origin position.

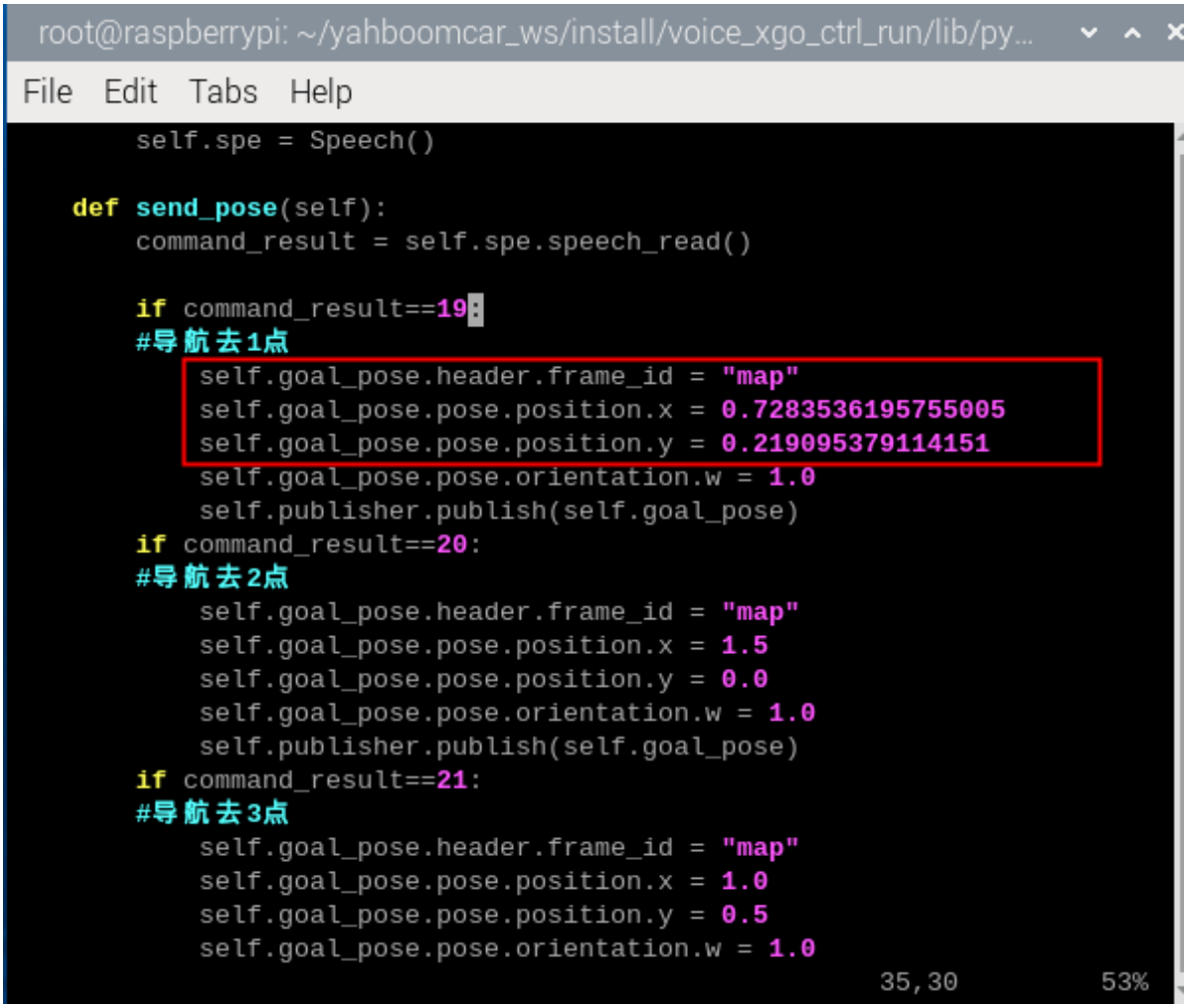
PI5 version steps:

According to the method in the second step, enter the same docker terminal** (you can see the previous steps to enter the same docker step)**, enter the following command to open,

```
#path
cd yahboomcar_ws/install/voice_xgo_ctrl_run/lib/python3.10/site-
packages/voice_xgo_ctrl_run
#Open the modified file
vi voice_xgo_ctrl_mutl_goal.py
```



After opening, you can modify it according to the values of position.x, position.y and orientation.w of the point just obtained. The values here are examples and should be subject to the values in your own environment. You can modify five points, save and exit after modification.



vi editor operation: After opening the file, enter i to start editing. After editing is completed, press ESC, then enter: wq to save and exit the file!

5. Start voice control multi-point navigation node

The steps are the same for PI4 and PI5 versions:

Open another shell or docker terminal and enter the following command in the terminal:

Notice：This terminal is the terminal that opens the remote connection to the mechanical dog.

```
#pi4
cd ~/cartographer_ws2/
source install/setup.bash
ros2 run voice_xgo_ctrl_run voice_xgo_ctrl_mult_goal

#pi5
cd yahbomcar_ws/
source install/setup.bash
ros2 run voice_xgo_ctrl_run voice_xgo_ctrl_mult_goal
```

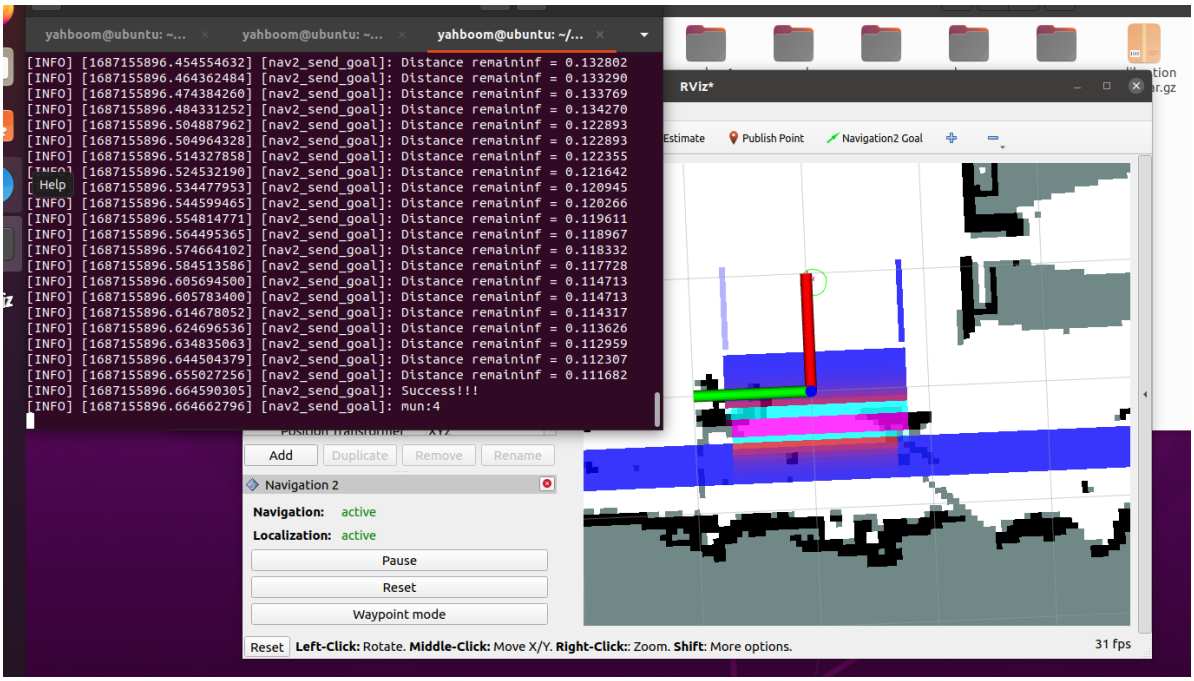
```
pi@yahboom:~/cartographer_ws2$
pi@yahboom:~/cartographer_ws2$
pi@yahboom:~/cartographer_ws2$
pi@yahboom:~/cartographer_ws2$
pi@yahboom:~/cartographer_ws2$ cd ~/cartographer_ws2/
pi@yahboom:~/cartographer_ws2$ source install/setup.bash
pi@yahboom:~/cartographer_ws2$ ros2 run voice_xgo_ctrl_run
--prefix
voice_xgo_ctrl_action
voice_xgo_ctrl_color_identify
voice_xgo_ctrl_color_identify\ =\ voice_xgo_ctrl_run.voice_xgo_ctrl_color_identify:mainvoice_xgo_ctrl_mult_goal
voice_xgo_ctrl_mult_goal
voice_xgo_ctrl_mutl_goal_identify
voice_xgo_ctrl_run
pi@yahboom:~/cartographer_ws2$ ros2 run voice_xgo_ctrl_run voice_xgo_ctrl_mult_goal
Speech Serial Opened! Baudrate=115200
```

Then he said to the mechanical dog: "Hello, Xiaoya"

The mechanical dog replied: "Yes"

Then say to the mechanical dog: "" Navigate to position 1

As shown in the picture below, the mechanical dog will automatically navigate to position 1.



The internal commands of the mechanical dog include: navigate to position 1, navigate to position 2, navigate to position 3, navigate to position 4 and other voice navigation commands.