

ROS2 environment obtains the real joint data of the robot dog

Quick use

1. Log in to the robot dog desktop through vnc viewer

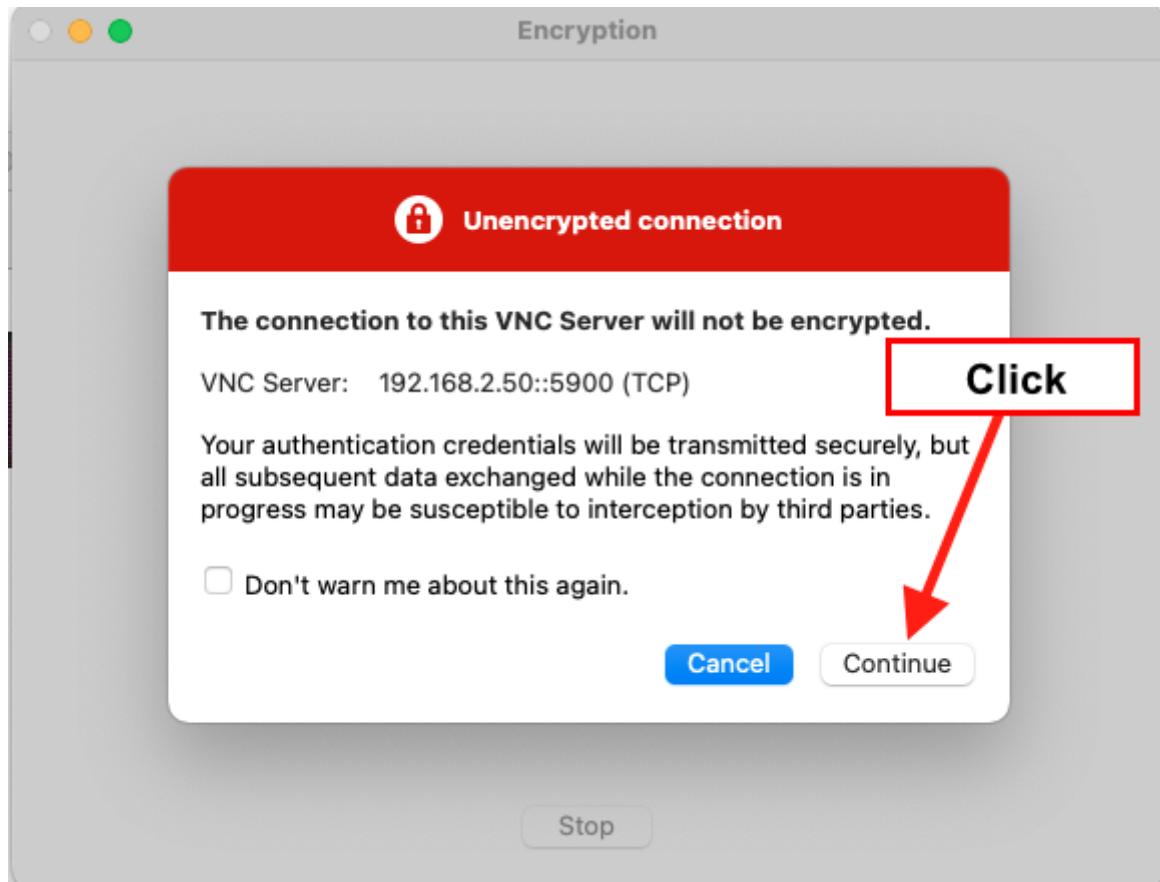
The premise is that the computer has the vnc viewer software installed.

Open vnc viewer and enter the IP address of the mechanical dog.

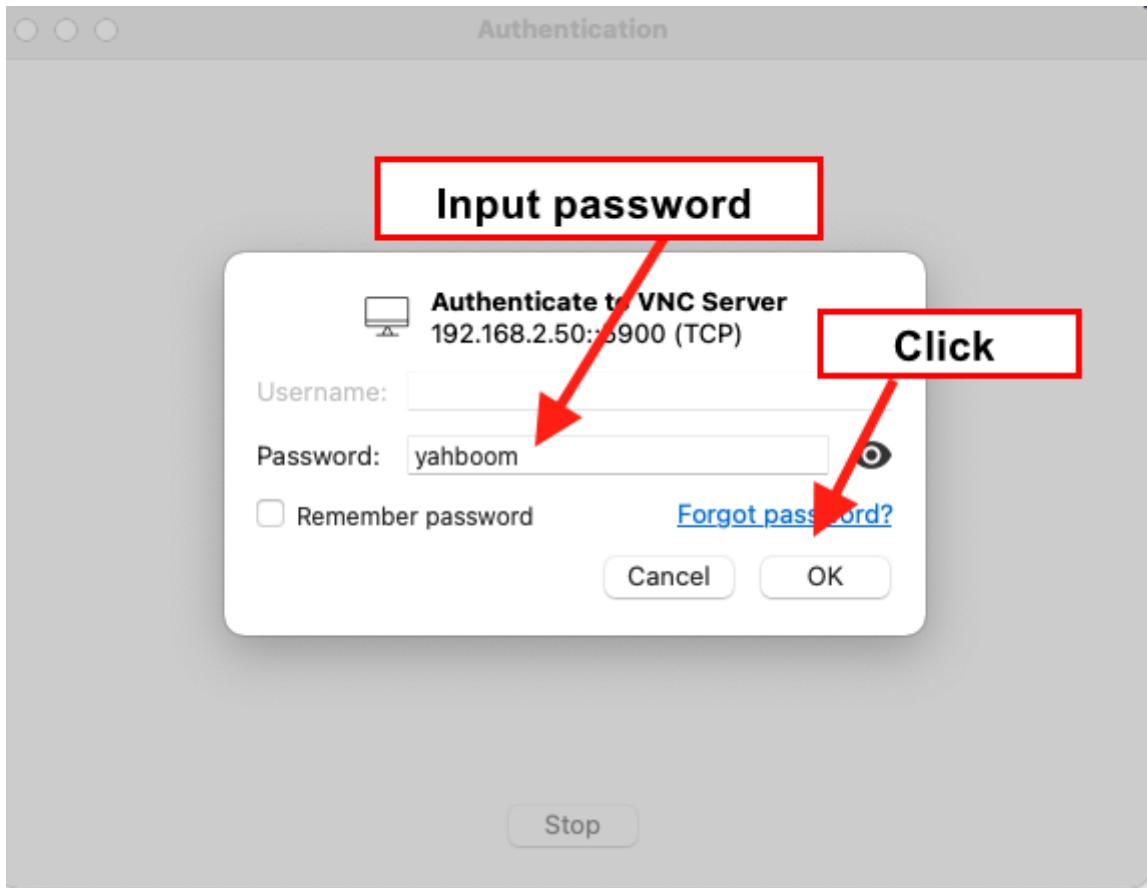


The IP address can be viewed on the LCD screen of the robot dog.

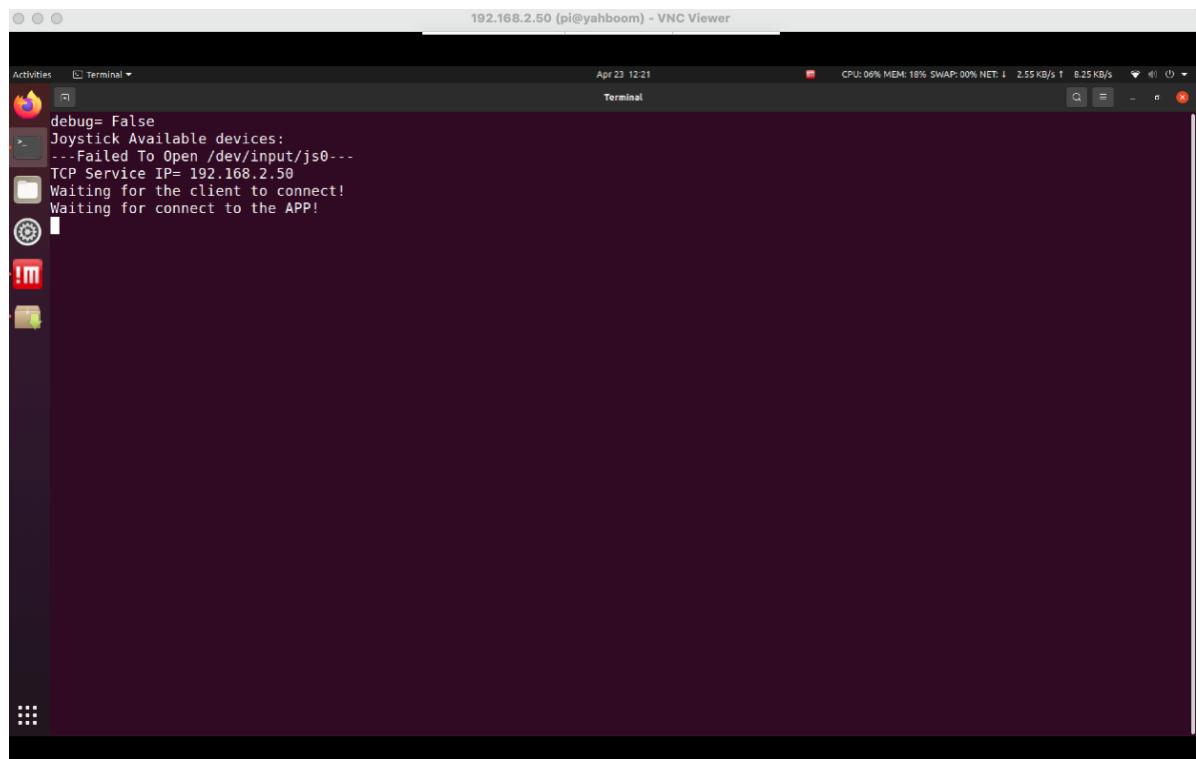
Then press the Enter key and click the Continue button



Enter the login password of the robot dog: yahboom and click OK.



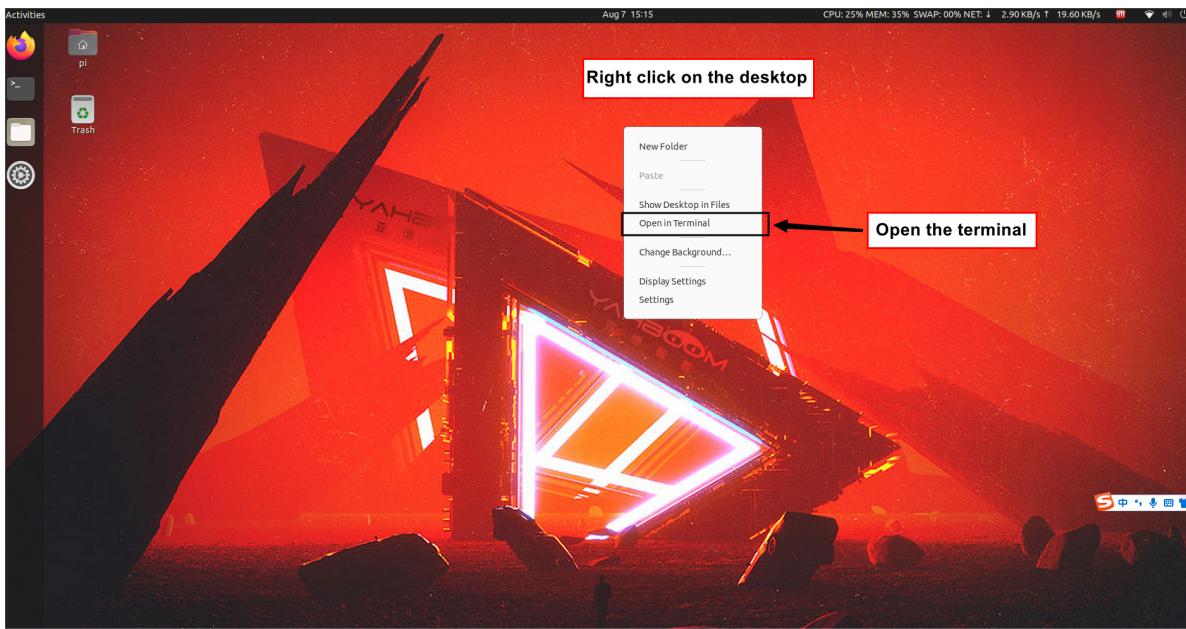
Enter the robot dog operating system.



2. Activate the virtual machine ros2 workspace

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

Right-click on the desktop and select 'Open in Terminal' to open the terminal.



Enter the command in the terminal:

```
cd ~/cartographer_ws
```

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

At this point we have activated the workspace.

3. Start the robot dog status node

PI4 version steps:

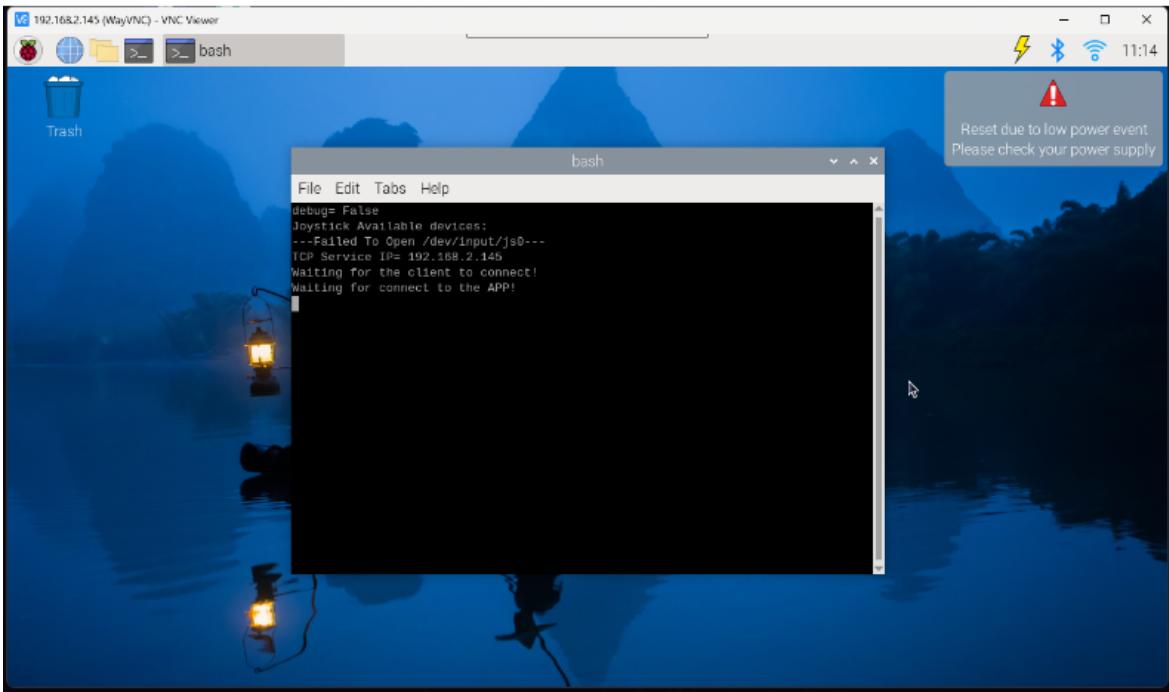
You can restart the chassis node. In normal circumstances, the chassis node starts automatically. The command to restart the chassis node is:

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

PI5 version steps:

You can restart the chassis node. Under normal circumstances, the chassis node starts automatically. The command to restart the chassis node is:

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
```

A terminal window showing the execution of the script. The output includes:

```
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:/#
```

The terminal prompt changes from "pi@raspberrypi:~ \$" to "root@raspberrypi:/#".

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

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

The screenshot shows a terminal window with the title bar "root@raspberrypi: /" and a language selection dropdown set to "代码语言" (Code Language). The window contains several lines of ROS message output from a node named "yahboomcar_joint_state-3". The messages are timestamped at 0x7fff363522f0 and show joint state data for a robot dog. The data includes various parameters such as position, velocity, and time.

```
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] #####
```

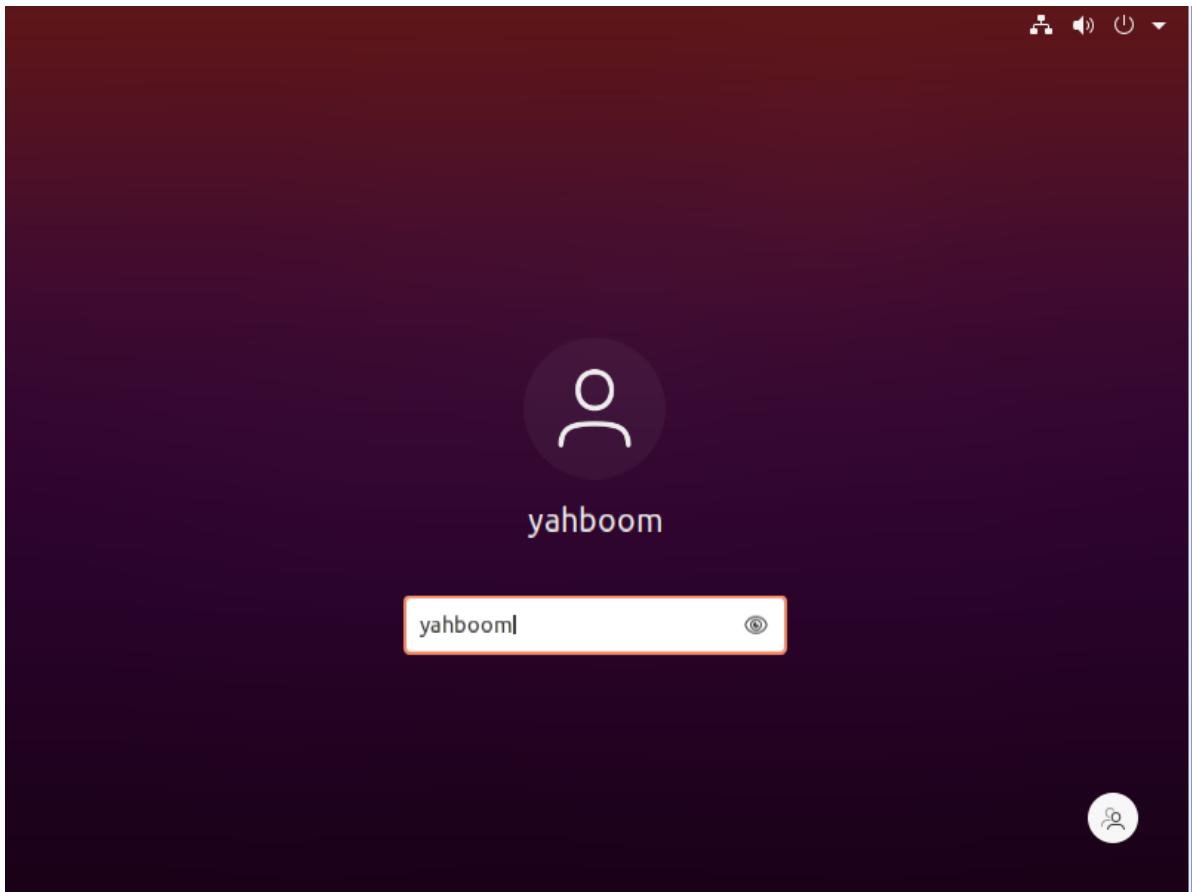
4. Check the status of the robot dog in the virtual machine rviz2

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

It should be noted here that the LAN communication ID enabled by the robot dog image is 16 by default. At this time, you need to check whether the ROS_DOMAIN_id of the virtual machine is equal to 16. View method:

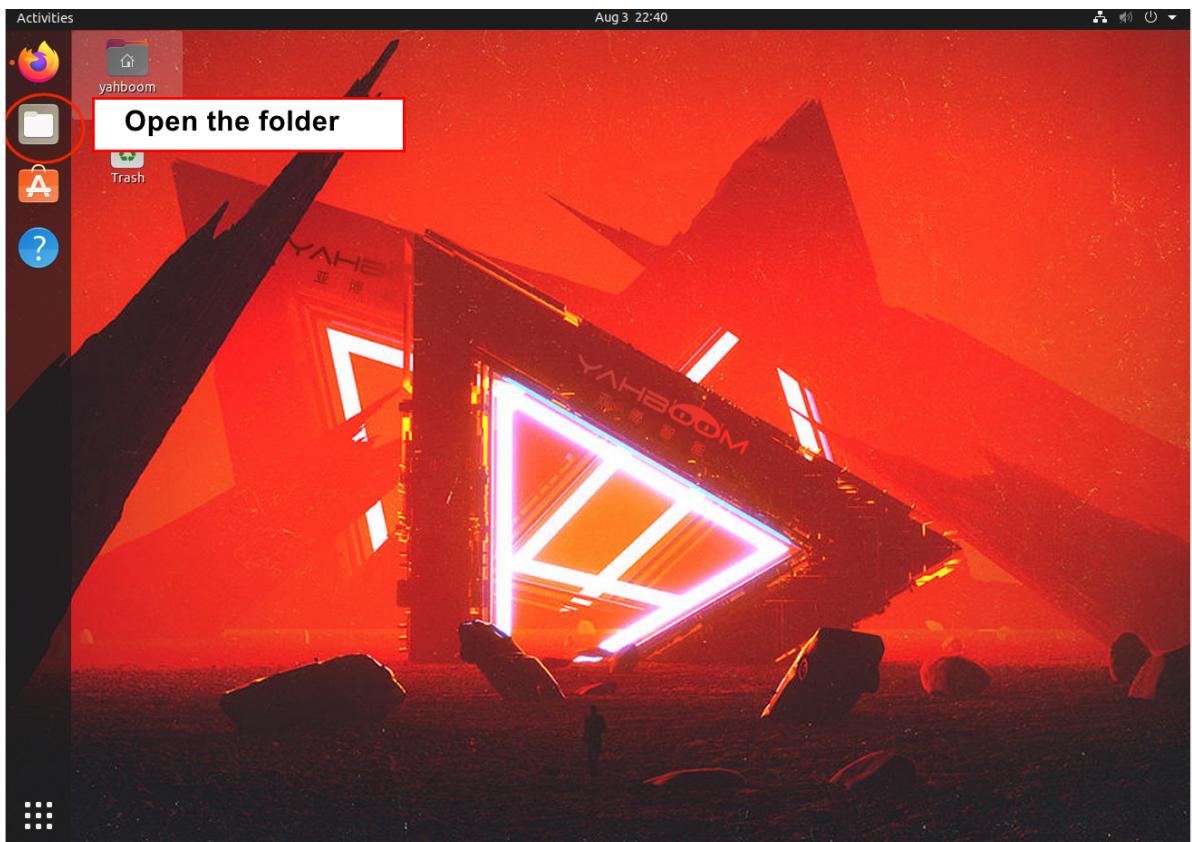
Note: The default virtual machine has been installed here.

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

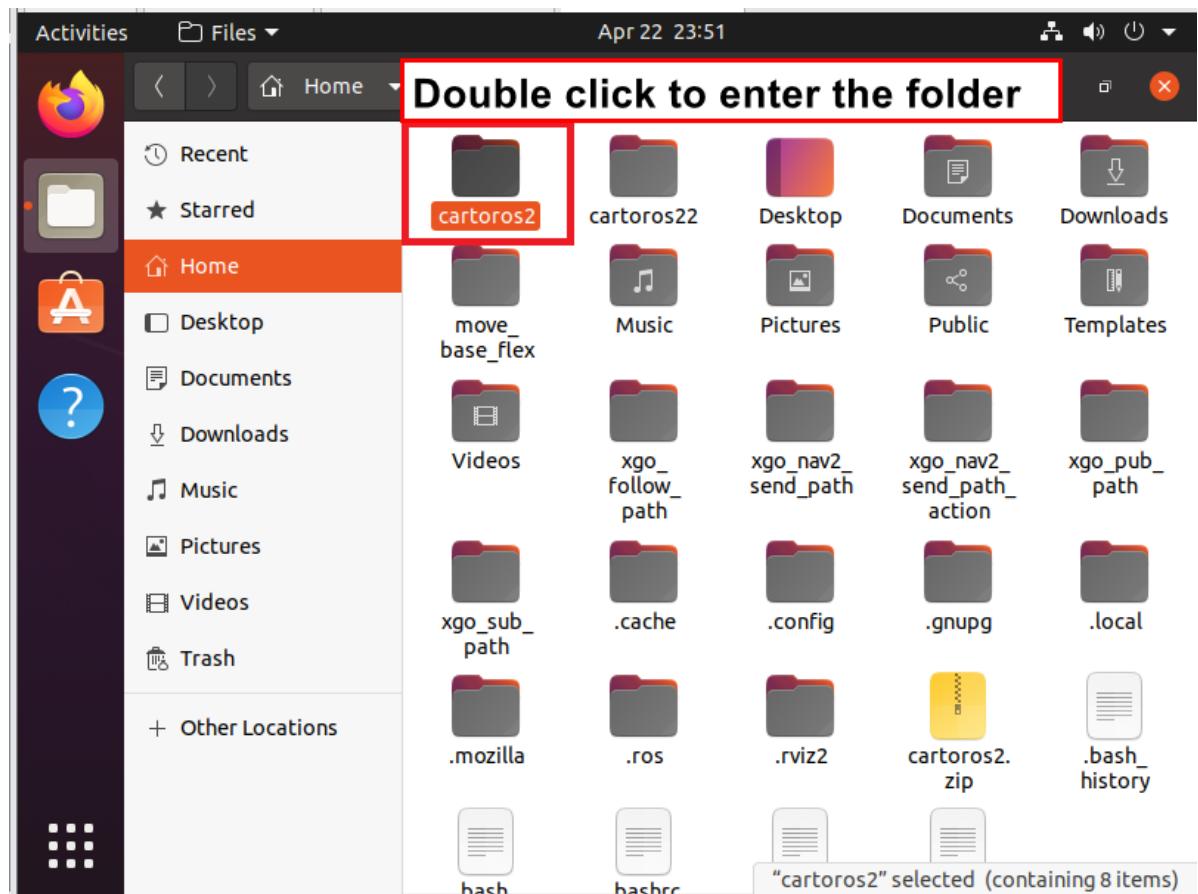


3. Start the robot dog mapping node

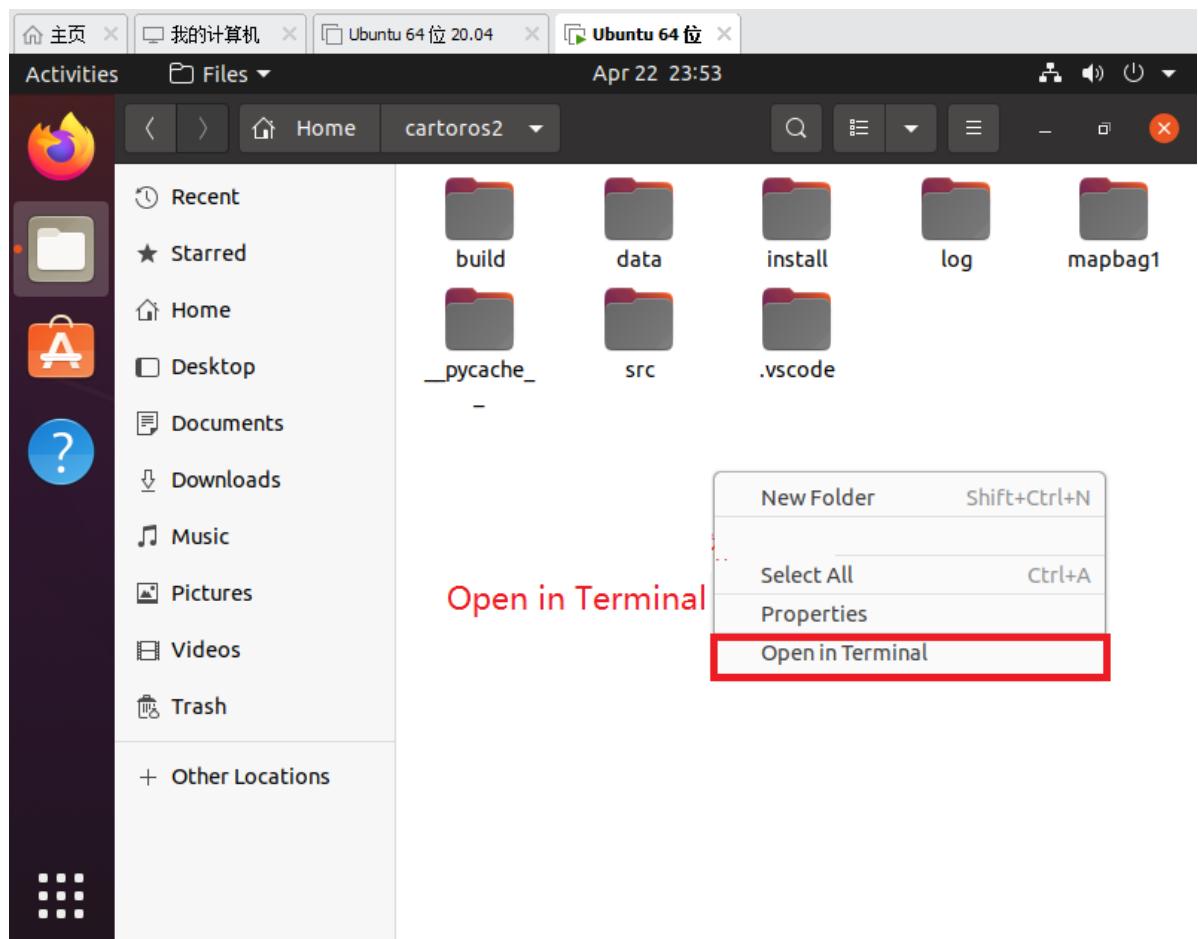
Enter the desktop system and open the folder.



Then double click on the cartoros2 folder



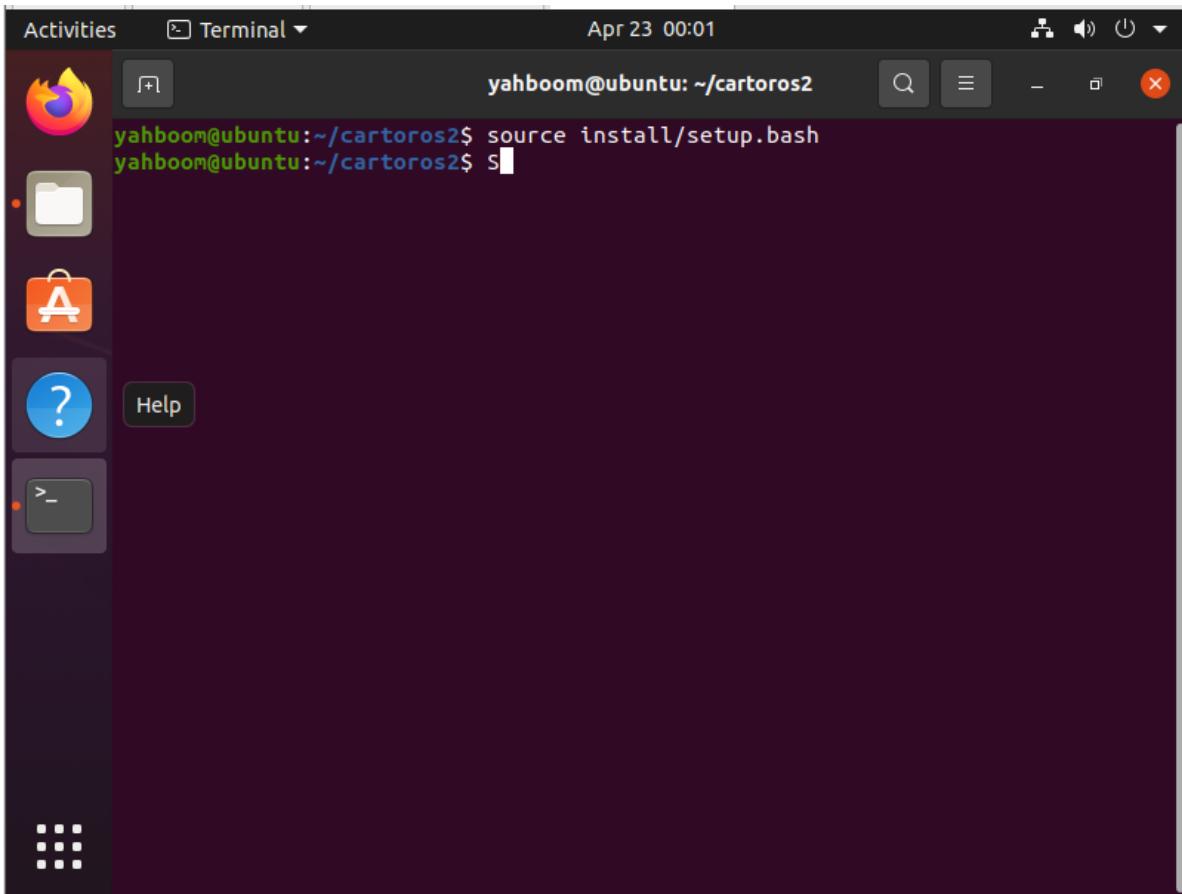
Then right-click in an empty space of the folder and select Open in Terminal



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

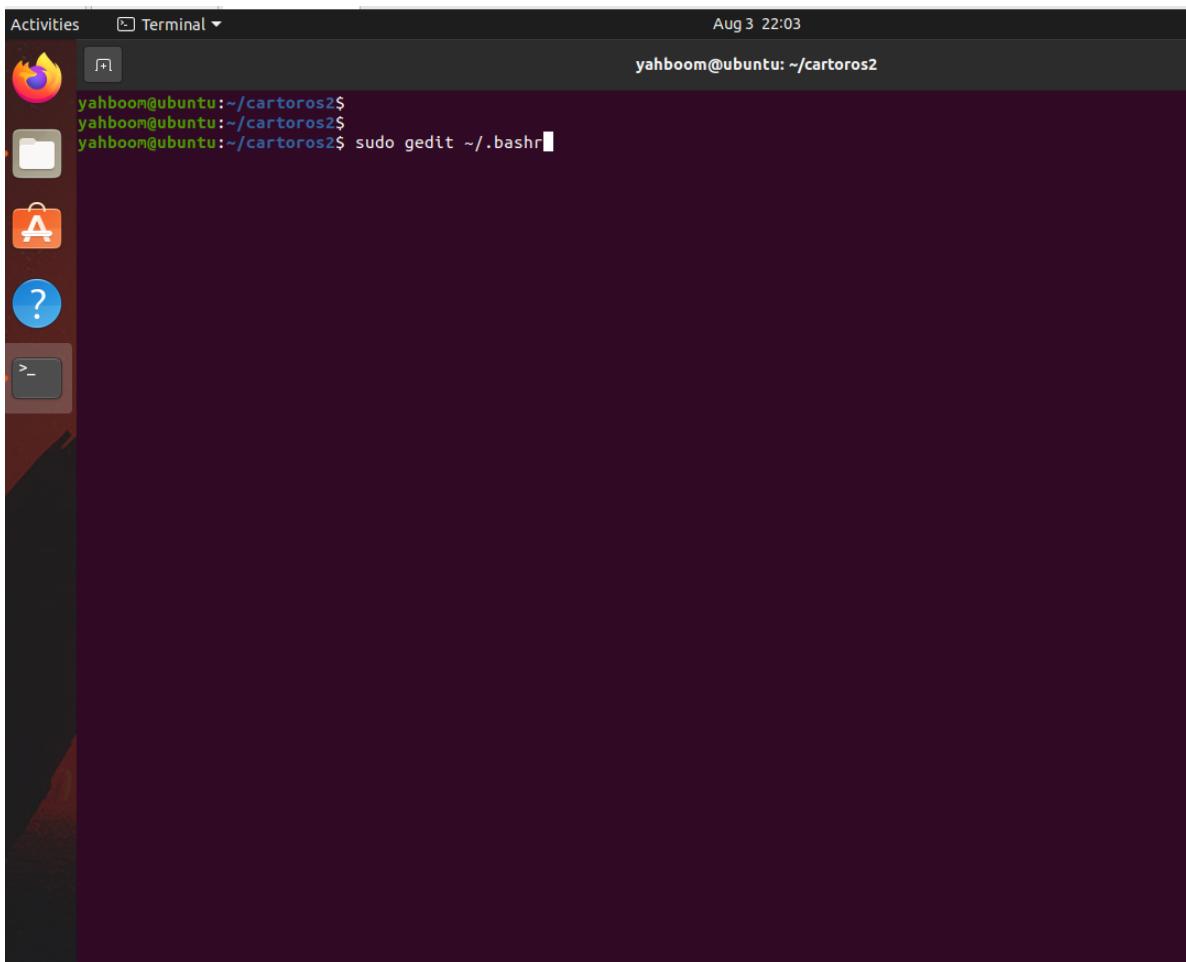
```
source install/setup.bash
```

After completing the input, press the Enter key.



Then enter the command,

```
sudo gedit ~/.bashrc
```



Press the Enter key and look at the configuration below in the document.

A screenshot of a Gedit text editor window. The title bar says "Activities Gedit". The main area shows the contents of the ".bashrc" file. The file contains several alias definitions and a check for a bash aliases file. It also includes code for enabling programmable completion and sourcing setup files for ROS and PKG_CONFIG_PATH. The code is color-coded with syntax highlighting. The status bar at the bottom shows "sh Tab Width: 8 Ln 1, Col 1 INS".

```
90 # some more ls aliases
^C
91 alias ll='ls -alF'
92 alias la='ls -A'
93 alias l='ls -CF'
94
95 # Add an "alert" alias for long running commands.  Use like so:
96 #   sleep 10; alert
97 alias alert='notify-send --urgency=low -i "$(([ $? = 0 ] && echo terminal || echo error)" "$-
98 (history|tail -n1|sed -e '\''$s/^s*[0-9]\+\s*//;s/[;&]\s*alert$/'\''"
99
100 # Alias definitions.
101 # You may want to put all your additions into a separate file like
102 # ~/._bash_aliases, instead of adding them here directly.
103 # See /usr/share/doc/bash-doc/examples in the bash-doc package.
104 if [ -f ~/._bash_aliases ]; then
105     . ~/._bash_aliases
106 fi
107
108 # enable programmable completion features (you don't need to enable
109 # this, if it's already enabled in /etc/bash.bashrc and /etc/profile
110 # sources /etc/bash.bashrc).
111 if ! shopt -oq posix; then
112     if [ -f /usr/share/bash-completion/bash_completion ]; then
113         . /usr/share/bash-completion/bash_completion
114     elif [ -f /etc/bash_completion ]; then
115         . /etc/bash_completion
116     fi
117 fi
118
119 source /opt/ros/foxy/setup.bash
120
121 export ROS_DOMAIN_ID=16
122
123
124
125 PKG_CONFIG_PATH=$PKG_CONFIG_PATH:/usr/local/lib/pkgconfig
```

If correct close the document. Enter in the terminal:

```
rviz2
```

Press the Enter key to see the current status of the mechanical dog. As shown below:

