1、Environmental construction

This course takes ROS2 and Foxy versions as examples to illustrate how to install and run the required environment, and make modifications based on the installed version of ROS

1、Installation dependencies

Terminal input,

```
#Replace the foxy here with the actual installed version name of ROS
sudo apt install ros-foxy-joy*
```

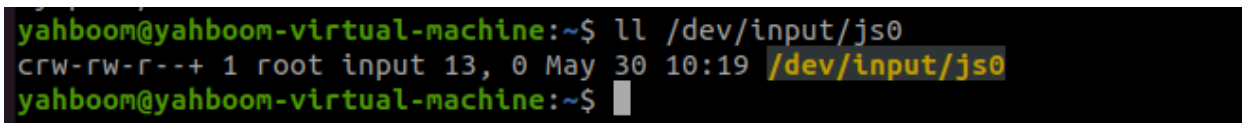
After installation is complete, refresh the environment variables again,

```
source ~/.bashrc
```

2、Test handle

Connect the handle to the motherboard and enter the following command to check if it is detected,

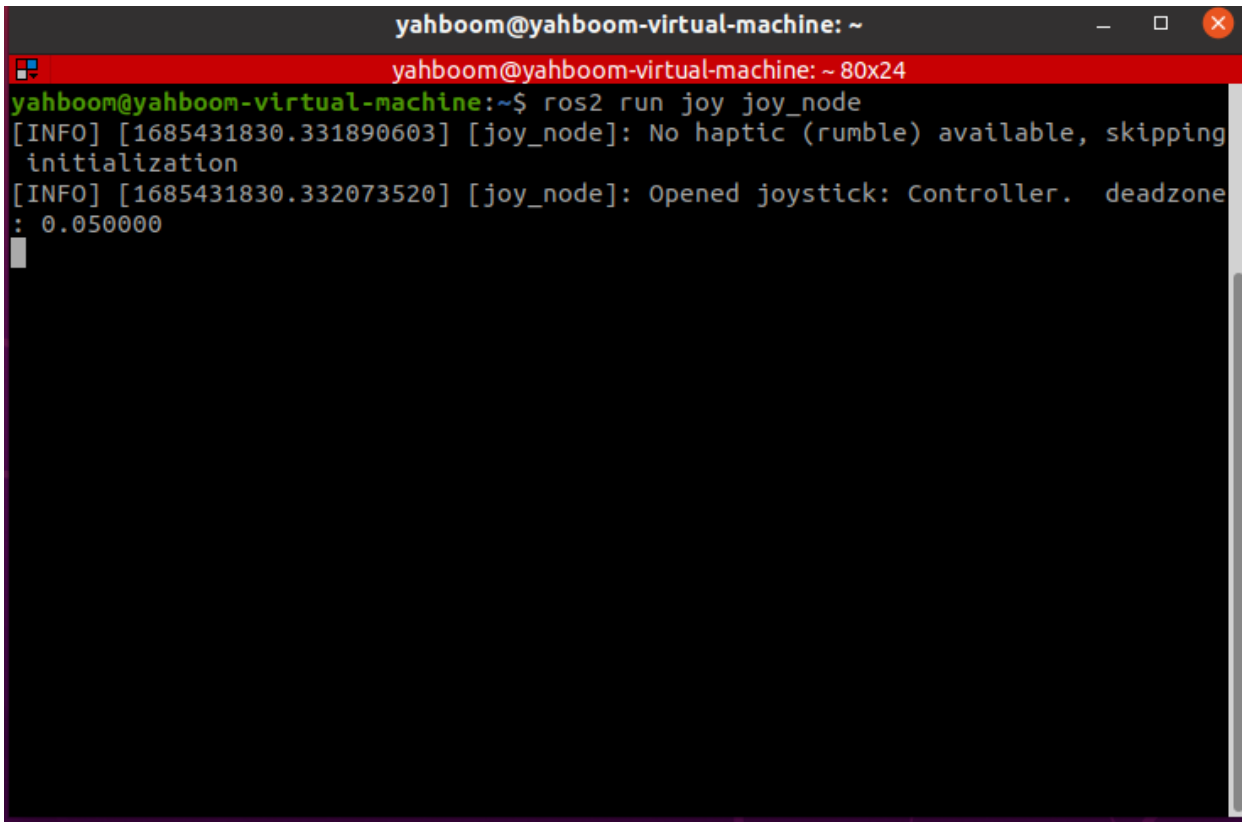
```
ll /dev/input/js0
```



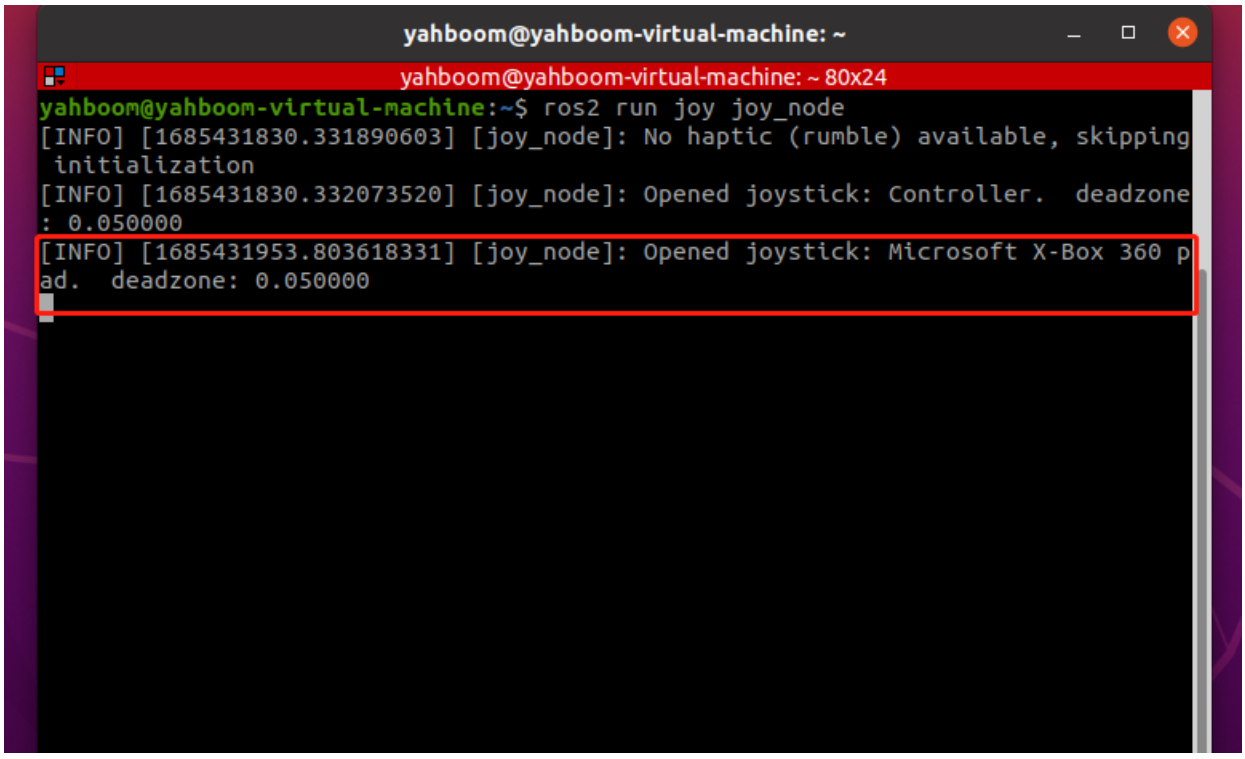
As shown in the figure, it indicates the recognition of the handle receiver. Note that if a mouse or keyboard is connected, it may also recognize the JS device, but the program only recognizes JS0. Therefore, if multiple JS devices appear, first connect the handle receiver and then connect other devices. If the handle is connected, the indicator light of the handle is in a constant state, and flashing indicates disconnection.

The terminal enters the following command to run the handle node program,

```
ros2 run joy joy_node
```



The controller displayed on the terminal is the current mode controlled by the handle. Press and hold the mode for 15 seconds to switch modes, and the terminal will print the replaced mode，



The above image shows switching to X-Box mode. The code values of the two modes are different。

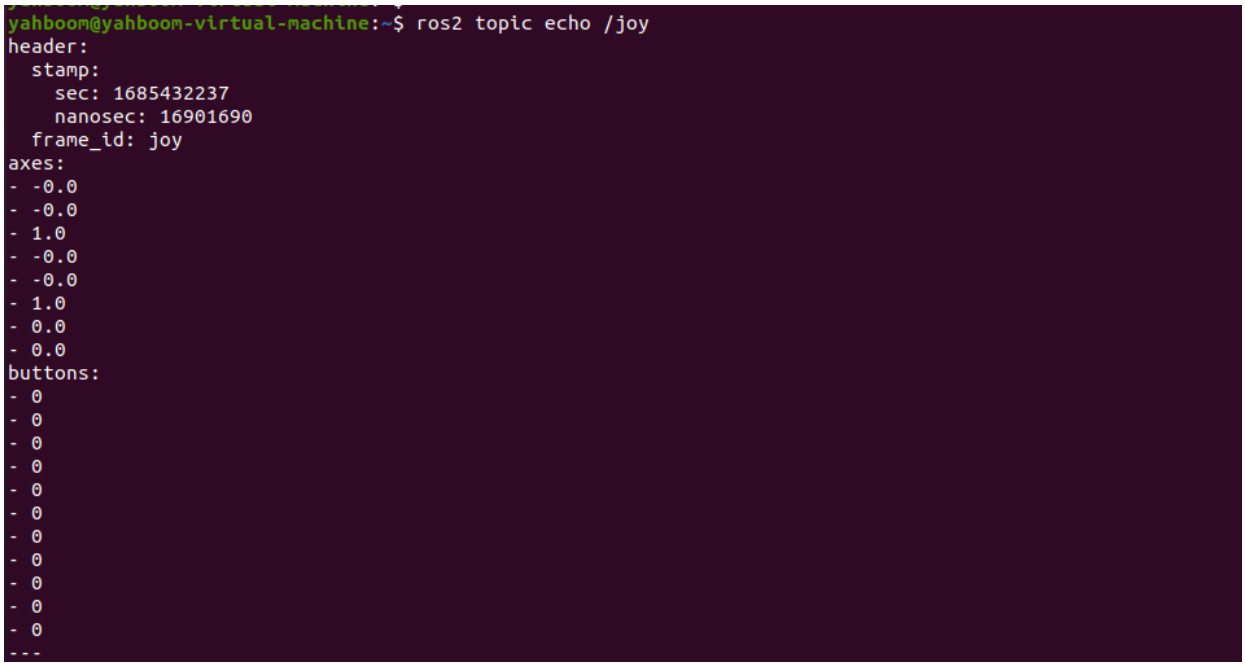
We can first check what topics have been posted and input them through the terminal，

```
ros2 topic list
```

```
yahboom@yahboom-virtual-machine:~$ ros2 topic list
/joy
/joy/set_feedback
/parameter_events
/rosout
yahboom@yahboom-virtual-machine:~$
```

The '/joy' here is the terminal we focus on, which will publish the current status of various code values of the controller. You can view the content of the topic data through the following command ,

```
ros2 topic echo /joy
```



As shown in the above figure, the data content of the topic is represented by [axes], which represents the two joysticks, and [buttons], which represents the values of the buttons.