Multi-machine handle control

Note: The virtual machine needs to be in the same LAN as the car, and the ROS_DOMAIN_ID needs to be consistent. You can check [Must read before use] to set the IP and ROS_DOMAIN_ID on the board.

1、Program function description

After the program is started, the two cars can be controlled to move synchronously through the handle. The controller program here is based on the [ROS Robot USB Wireless Controller] sold by Yahboom Intelligent Technology as an example. Other controller programs may not be suitable, and the program needs to be modified according to the actual remote control key values.

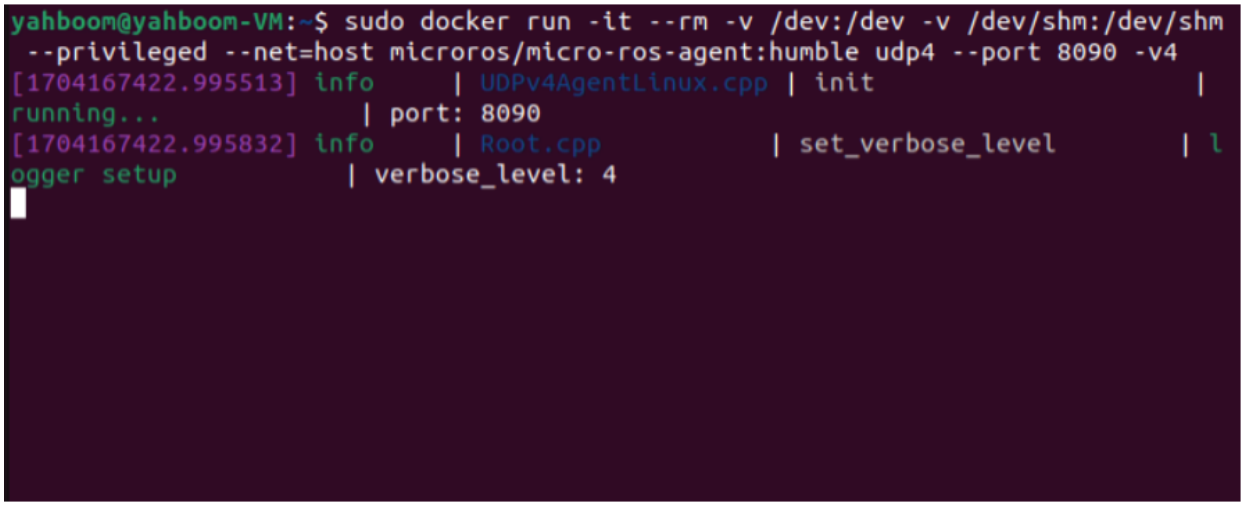
2、Multi-machine function basic settings

Taking two cars as an example, it is recommended to use two computers with matching virtual machines, change the config_robot.py files, and set robot.set_ros_namespace() to robot1 and robot2 respectively. And **the ROS_DOMAIN_ID of the two cars and the ROS_DOMAIN_ID of the virtual machine need to be set to the same**. Then open the terminal in the /home/yahboom directory and enter `sudo python3 config_robot.py` to run this program (you need to change it back and re-run this program to run other programs except multi-car).

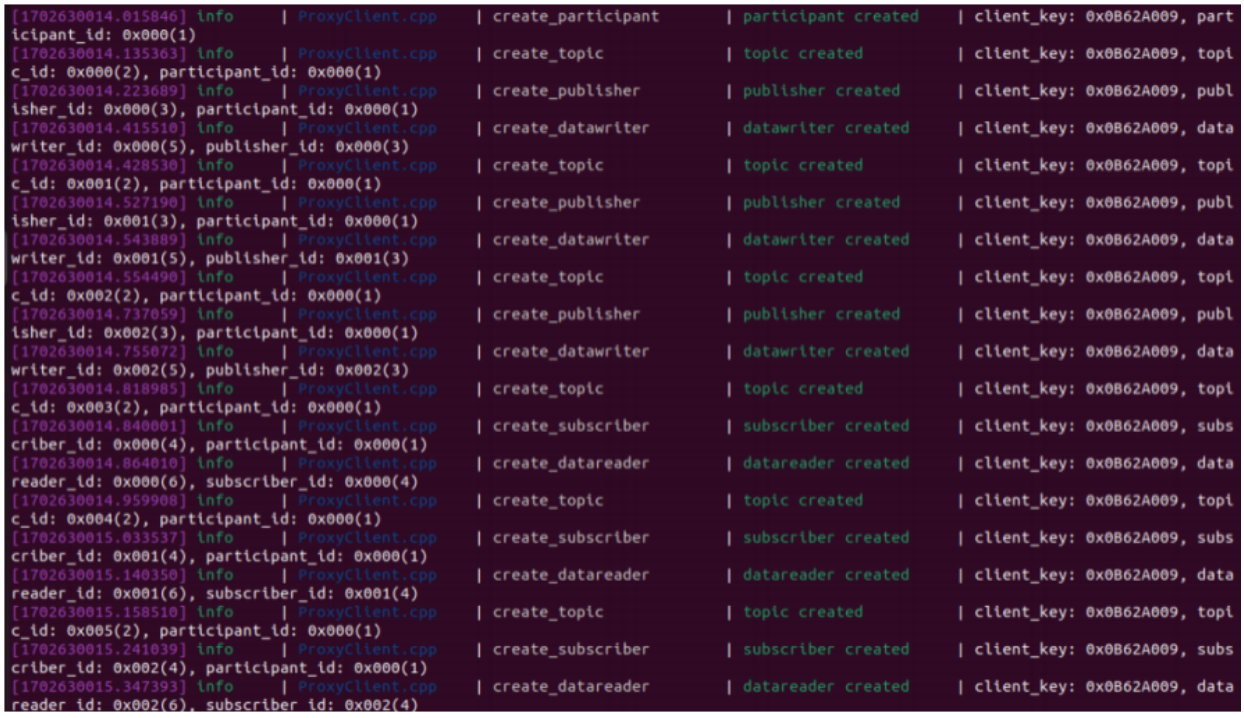
3、Start and connect to the agent

Taking the supporting virtual machine as an example, under the two virtual machines, enter the following commands to start the agents of the respective cars:

```
sudo docker run -it --rm -v /dev:/dev -v /dev/shm:/dev/shm --privileged --net=host
microros/micro-ros-agent:humble udp4 --port 8090 -v4
```



然后，打开两部小车开关，等待两部小车连接上各自的代理，连接成功，终端显示如下图所示，



Check the currently started node. Select one of the two virtual machines, open the terminal and enter the following command:

```
ros2 node list
```

```
yahboom@yahboom-VM:~$ ros2 node list
/robot1/YB_Car_Node
/robot2/YB_Car_Node
yahboom@yahboom-VM:~$
```

As shown in the picture above, the nodes of both cars have been started. To query the current topic information, enter the following command in the terminal

```
ros2 topic list
```

```
yahboom@yahboom-VM:~$ ros2 topic list
/parameter_events
/robot1/beep
/robot1/cmd_vel
/robot1/imu
/robot1/odom_raw
/robot1/scan
/robot1/servo_s1
/robot1/servo_s2
/robot2/beep
/robot2/cmd_vel
/robot2/imu
/robot2/odom_raw
/robot2/scan
/robot2/servo_s1
/robot2/servo_s2
/rosout
```

3、Start the handle control program

You can connect the controller receiver to any virtual machine. You need to ensure that the virtual machine can recognize the controller receiver. As shown in the figure below, the connection is successful.

```
yahboom@yahboom-VM:~$ lsusb
Bus 002 Device 001: ID 1d6b:0002 Linux Foundation 2.0 root hub
Bus 001 Device 004: ID 0079:181c DragonRise Inc. Controller
Bus 001 Device 003: ID 0e0f:0002 VMware, Inc. Virtual USB Hub
Bus 001 Device 002: ID 0e0f:0003 VMware, Inc. Virtual Mouse
Bus 001 Device 001: ID 1d6b:0001 Linux Foundation 1.1 root hub
```

If it is not connected, check **[Virtual Machine]**->**[Removable Devices]** in the menu bar above the virtual machine toolbar and check whether **[DragonRise Controller]** is checked.

In the virtual machine of the connected controller receiver, open the terminal and enter the following command,

```
ros2 run joy joy_node
ros2 run yahboomcar_multi multi_joy_ctrl
```

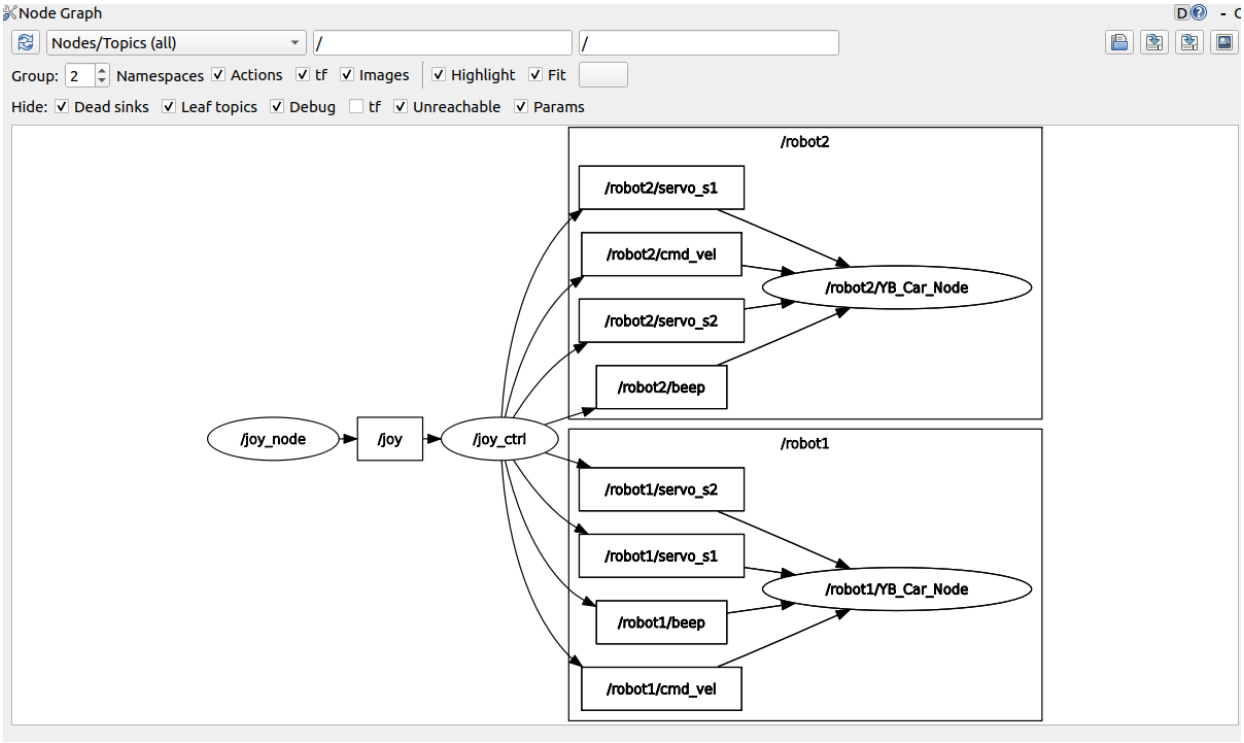
The remote control button description is as follows:

- Left rocker: valid in the front and rear directions, controls the car forward and backward, invalid in the left and right directions
- Right joystick: valid in left and right directions, controls the car to rotate left and right, invalid in forward and backward directions
- START key: buzzer control
- Y key: control the S2 servo upward
- A key: control the S2 servo down
- X key: control the S1 servo to the left
- B key: control the S1 servo to the right
- R1 key: The handle controls the speed switch. Only after pressing it can the speed of the car be controlled by remote control.
- MODE key: Switch modes and use the default mode. After switching modes, if the key value is incorrect, the program will exit with an error.

4、View node communication diagram

Select any one of the two virtual machines and enter the following command in the terminal

```
ros2 run rqt_graph rqt_graph
```



If it is not displayed at first, select [Nodes/Topics(all)], and then click the refresh button in the upper left corner.

5、Source code analysis

Source code reference path (taking the supporting virtual machine as an example):

```
/home/yahboom/yahboomcar_ws/src/yahboomcar_multi/yahboomcar_multi
```

multi_yahboom_joy.py,

```
#Create topic publisher
#create pub
self.pub_goal = self.create_publisher(GoalID,"move_base/cancel",10)
self.pub_JoyState = self.create_publisher(Bool,"JoyState",  10)
#cmd_vel
self.pub_cmdVel_r1 = self.create_publisher(Twist,'/robot1/cmd_vel',10)
self.pub_cmdVel_r2 = self.create_publisher(Twist,'/robot2/cmd_vel',10)
#beep
self.pub_Buzzer_r1 = self.create_publisher(UInt16,"/robot1/beep",  1)
self.pub_Buzzer_r2 = self.create_publisher(UInt16,"/robot2/beep",  1)
#servo1
self.pub_Servo1_r1 = self.create_publisher(Int32,"/robot1/servo_s1" , 10)
self.pub_Servo1_r2 = self.create_publisher(Int32,"robot2/servo_s1" , 10)
#servo2
self.pub_Servo2_r1 = self.create_publisher(Int32,"/robot1/servo_s2" , 10)
self.pub_Servo2_r2= self.create_publisher(Int32,"/robot2/servo_s2" , 10)
#Create topic subscribers and subscribe to /joy node information
self.sub_Joy = self.create_subscription(Joy,'joy', self.buttonCallback,10)
#Callback
def buttonCallback(self,joy_data):
```

```
    if not isinstance(joy_data, Joy): return
    self.user_jetson(joy_data)
#Process the remote control key value. For detailed code, please refer to the
user_jetson function.
#Publishing speed topic
self.pub_cmdVel_r1.publish(twist)
self.pub_cmdVel_r2.publish(twist)
#Publishing buzzer topic
self.pub_Buzzer_r1.publish(b)
self.pub_Buzzer_r2.publish(b)
#Publish gimbal servo data (Raspberry Pi 5 version)
#PTZ 1
self.pub_Servo1_r1.publish(servo1_angle)
self.pub_Servo1_r2.publish(servo1_angle)
#PTZ 2
self.pub_Servo2_r1.publish(servo2_angle)
self.pub_Servo2_r2.publish(servo2_angle)
```