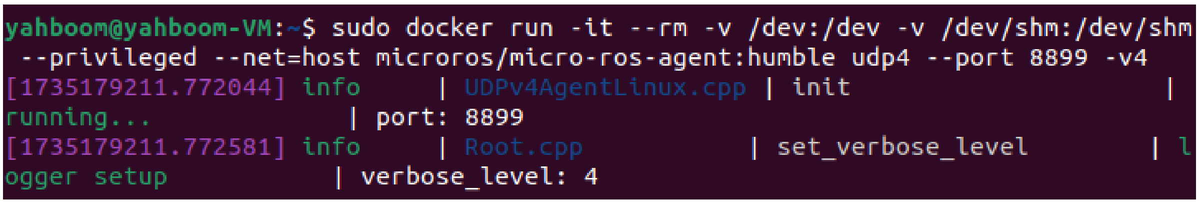
Robot keyboard control

Note: The virtual machine needs to be in the same LAN as the robot, 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. Start and connect the agent

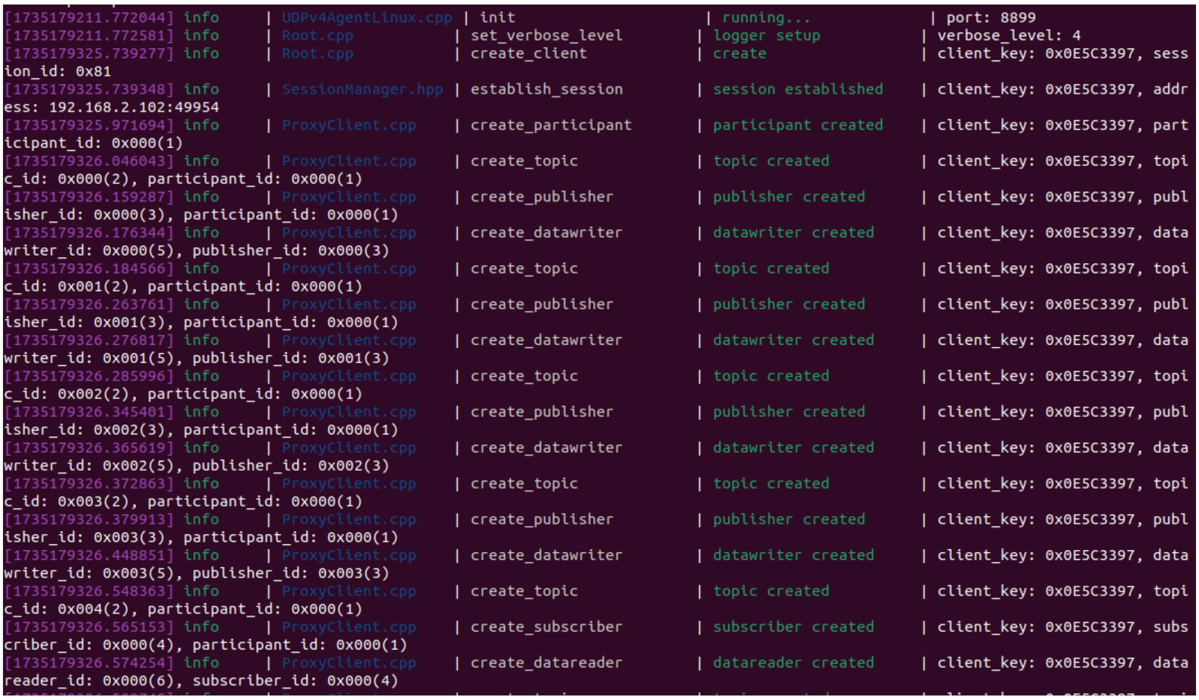
Take the supporting virtual machine as an example, enter the following command to start the agent,

```
sudo docker run -it --rm -v /dev:/dev -v /dev/shm:/dev/shm --privileged --net=host microros/micro-ros-agent:humble udp4 --port 8899 -v4
```



```
yahboom@yahboom-VM:~$ sudo docker run -it --rm -v /dev:/dev -v /dev/shm:/dev/shm
 --privileged --net=host microros/micro-ros-agent:humble udp4 --port 8899 -v4
[1735179211.772044] info     | UDPv4AgentLinux.cpp | init                     |
running...              | port: 8899
[1735179211.772581] info     | Root.cpp            | set_verbose_level        | l
ogger setup             | verbose_level: 4
```

Then, turn on the car switch and wait for the car to connect to the agent. The connection is successful as shown in the figure below,



```
[1735179211.772044] info     | UDPv4AgentLinux.cpp | init                   | running...             | port: 8899
[1735179211.772581] info     | Root.cpp            | set_verbose_level      | logger setup           | verbose_level: 4
[1735179325.739277] info     | Root.cpp            | create_client          | create                 | client_key: 0x0E5C3397, sess
ion_id: 0x81
[1735179325.739348] info     | SessionManager.hpp | establish_session       | session established    | client_key: 0x0E5C3397, addr
ess: 192.168.2.102:49954
[1735179325.971694] info     | ProxyClient.cpp     | create_participant     | participant created    | client_key: 0x0E5C3397, part
icipant_id: 0x000(1)
[1735179326.046043] info     | ProxyClient.cpp     | create_topic           | topic created          | client_key: 0x0E5C3397, topi
c_id: 0x000(2), participant_id: 0x000(1)
[1735179326.159287] info     | ProxyClient.cpp     | create_publisher       | publisher created      | client_key: 0x0E5C3397, publ
isher_id: 0x000(3), participant_id: 0x000(1)
[1735179326.176344] info     | ProxyClient.cpp     | create_datawriter      | datawriter created     | client_key: 0x0E5C3397, data
writer_id: 0x000(5), publisher_id: 0x000(3)
[1735179326.184566] info     | ProxyClient.cpp     | create_topic           | topic created          | client_key: 0x0E5C3397, topi
c_id: 0x001(2), participant_id: 0x000(1)
[1735179326.263761] info     | ProxyClient.cpp     | create_publisher       | publisher created      | client_key: 0x0E5C3397, publ
isher_id: 0x001(3), participant_id: 0x000(1)
[1735179326.276817] info     | ProxyClient.cpp     | create_datawriter      | datawriter created     | client_key: 0x0E5C3397, data
writer_id: 0x001(5), publisher_id: 0x001(3)
[1735179326.285996] info     | ProxyClient.cpp     | create_topic           | topic created          | client_key: 0x0E5C3397, topi
c_id: 0x002(2), participant_id: 0x000(1)
[1735179326.345401] info     | ProxyClient.cpp     | create_publisher       | publisher created      | client_key: 0x0E5C3397, publ
isher_id: 0x002(3), participant_id: 0x000(1)
[1735179326.365619] info     | ProxyClient.cpp     | create_datawriter      | datawriter created     | client_key: 0x0E5C3397, data
writer_id: 0x002(5), publisher_id: 0x002(3)
[1735179326.372863] info     | ProxyClient.cpp     | create_topic           | topic created          | client_key: 0x0E5C3397, topi
c_id: 0x003(2), participant_id: 0x000(1)
[1735179326.379913] info     | ProxyClient.cpp     | create_publisher       | publisher created      | client_key: 0x0E5C3397, publ
isher_id: 0x003(3), participant_id: 0x000(1)
[1735179326.448851] info     | ProxyClient.cpp     | create_datawriter      | datawriter created     | client_key: 0x0E5C3397, data
writer_id: 0x003(5), publisher_id: 0x003(3)
[1735179326.548363] info     | ProxyClient.cpp     | create_topic           | topic created          | client_key: 0x0E5C3397, topi
c_id: 0x004(2), participant_id: 0x000(1)
[1735179326.565153] info     | ProxyClient.cpp     | create_subscriber      | subscriber created     | client_key: 0x0E5C3397, subs
criber_id: 0x000(4), participant_id: 0x000(1)
[1735179326.574254] info     | ProxyClient.cpp     | create_datareader      | datareader created     | client_key: 0x0E5C3397, data
reader_id: 0x000(6), subscriber_id: 0x000(4)
```

2. Start the keyboard control program

2.1 Code path

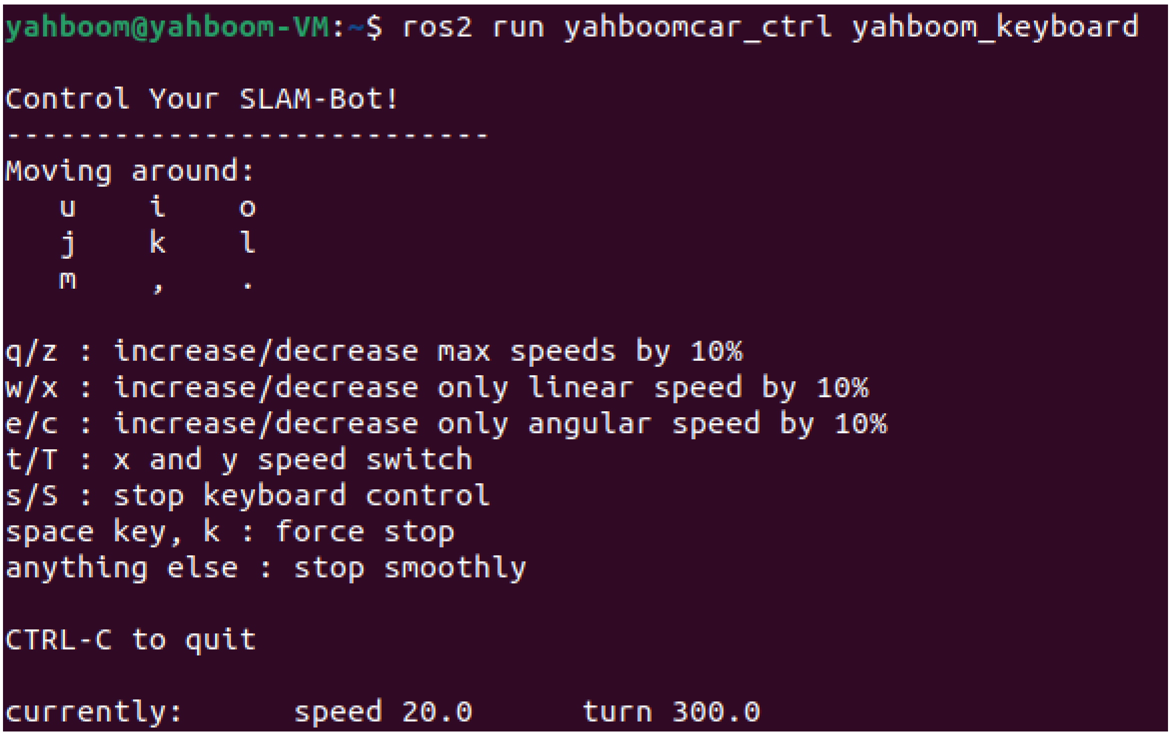
Code path:

```
/root/yahboomcar_ws/src/yahboomcar_ctrl/yahboomcar_ctrl
```

2.2 Run command

Take the matching virtual machine as an example, terminal input,

```
ros2 run yahboomcar_ctrl yahboom_keyboard
```



```
yahboom@yahboom-VM:~$ ros2 run yahboomcar_ctrl yahboom_keyboard

Control Your SLAM-Bot!
---------------------------
Moving around:
   u    i    o
   j    k    l
   m    ,    .

q/z : increase/decrease max speeds by 10%
w/x : increase/decrease only linear speed by 10%
e/c : increase/decrease only angular speed by 10%
t/T : x and y speed switch
s/S : stop keyboard control
space key, k : force stop
anything else : stop smoothly

CTRL-C to quit

currently:      speed 20.0      turn 300.0
```

The keyboard key description is as follows,

2.3 Direction control

Key	Linear angular velocity	Direction	Key	Linear angular velocity	Direction
【i】 or 【I】	【linear，0】	Go forward	【u】 or 【U】	【linear，angular】	Turn left
【,】	【-linear，0】	Go backward	【o】 or 【O】	【linear，-angular】	Turn right
【j】 or 【J】	【0，angular】	Rotate left	【m】 or 【M】	【-linear，-angular】	Reverse left
【l】 or 【L】	【0，-angular】	Rotate right	【.】	【-linear，angular】	Reverse right

2.4 Speed control

Button	Speed change	Button	Speed change
【q】	Increase linear and angular speeds by 10%	【z】	Both linear speed and angular speed decrease by 10%

Button	Speed change	Button	Speed change
【w】	Only linear speed increases by 10%	【x】	Only linear speed decreases by 10%
【e】	Only angular speed increases by 10%	【c】	Only angular speed decreases by 10%
【t】	Linear speed X-axis/Y-axis direction switch	【s】	Stop keyboard control

Note: Since the car has a common tire differential wheel structure and cannot move sideways, the 【t】 key is meaningless. Before using keyboard control each time, you need to click the terminal that starts the program, otherwise the key event cannot be detected.