# 2、Basic communication



## 2.1、Node settings

Function pack path: ~/transbot_ws/src/transbot_bringup

Transbot needs to realize the functions: speed control, speed information feedback, robotic arm control, robotic arm status feedback, battery voltage feedback, buzzer control, water lamp control, searchlight control, servo platform control.

The content of setting the Transbot bottom driver node according to the requirements is as follows:

- Topic

  Publish odometer message【/transbot/get_vel】

  Publish imu news【/transbot/imu】

  Publish battery voltage message【/voltage】

  Subscribe to car sports news [/cmd_vel]

  Subscribe to robotic arm control messages [/TargetAngle]

  Subscribe to gimbal servo control message [/PWMServo]

- Service (client)

  Receive buzzer control message【/Buzzer】

  Receive searchlight control message [/Headlight]

  Receive and feedback the current angle message of the robotic arm [/CurrentAngle]

  Receive flow light control message [/RGBLight]

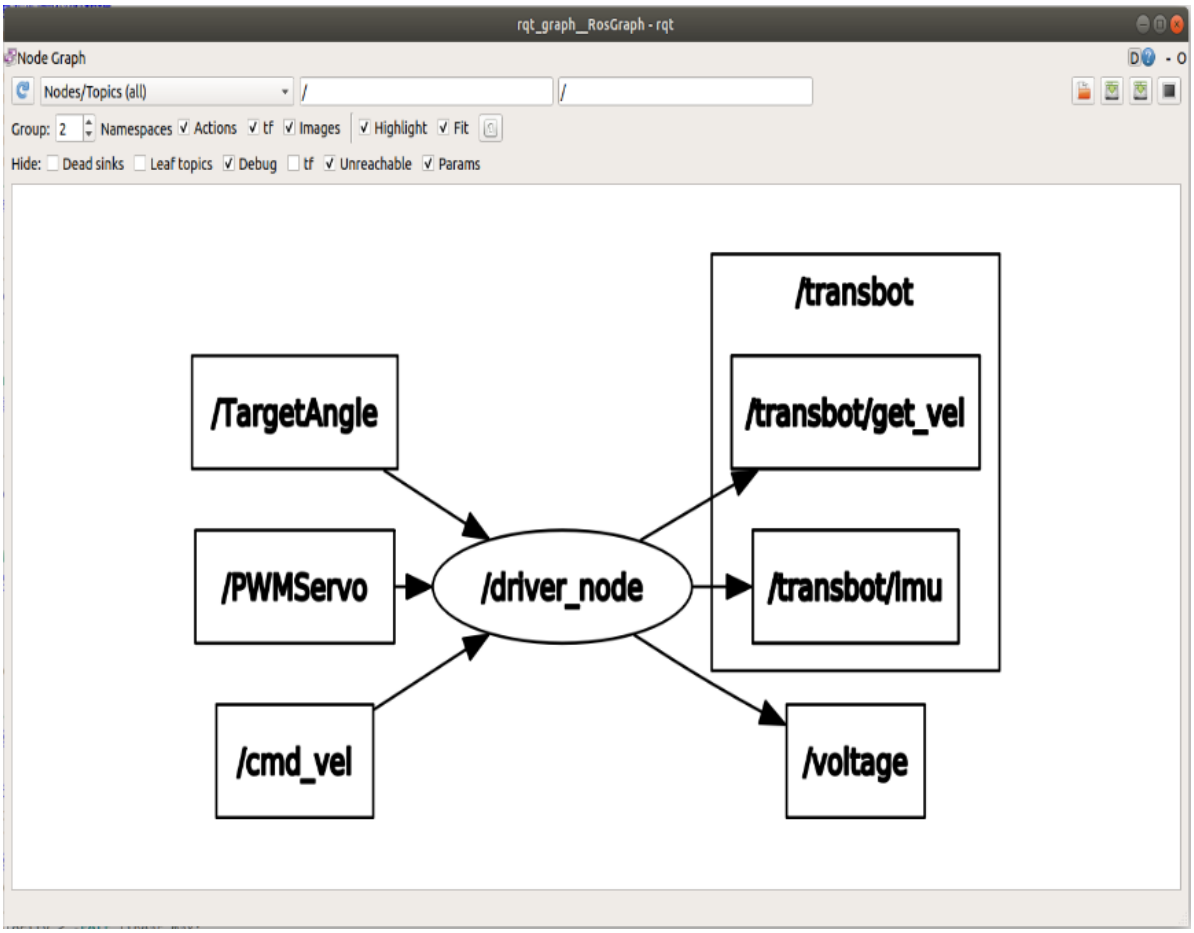
## 2.2、Node data view

### 1）Start up

```
roscore
rosrun transbot_bringup transbot_driver.py
```

## 2) View node graph

```
rqt_graph
```




## 3) View service

```
rosservice list
```

The system will print the following content:

```
/Buzzer
/CurrentAngle
/Headlight
/RGBLight
... ...
```

```
# Buzzer
rosservice call /Buzzer "buzzer: 1"  # open
rosservice call /Buzzer "buzzer: 0"  # close
# running water light
rosservice call /RGBLight "effect: 1
speed: 0"
# searchlight
rosservice call /Headlight "Headlight: 1"
# Get the current angle of the robotic arm
rosservice call /CurrentAngle "apply: 'GetJoint'"
```

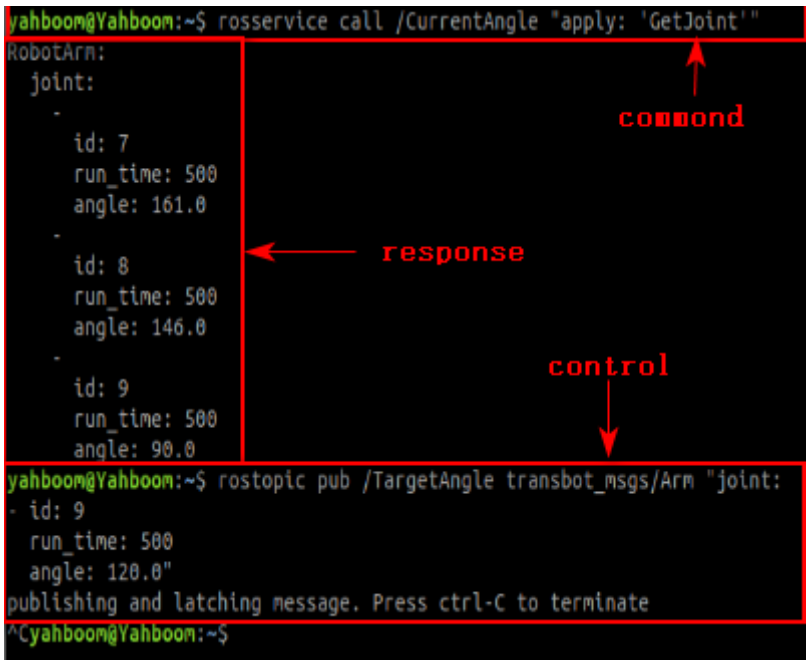
- Running water light

【effect】：[0: stop light effect, 1: running water light, 2: marquee light, 3: breathing light, 4: gradient light, 5: starlight, 6: battery display]
【speed】：[1, 10]，The smaller the value, the faster the change.

- Searchlight

【Headlight】：[0, 100]，The larger the value, the greater the brightness.

- Robotic arm

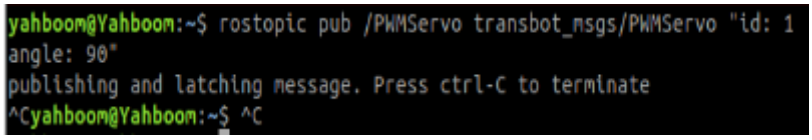



| id | servo | angle | running time |
| --- | --- | --- | --- |
| 7 | servo connected to the body | 【0，225】 | 【10，2000】 |
| 8 | Second section servo | 【30，270】 | 【10，2000】 |
| 9 | Servo on the clip | 【30，180】 | 【10，2000】 |

- Control servo platform **

**Note: If you are using the Astra camera, please do not use the command line to control the servo, it will damage the camera. The neutral angle of all servos is 90.**

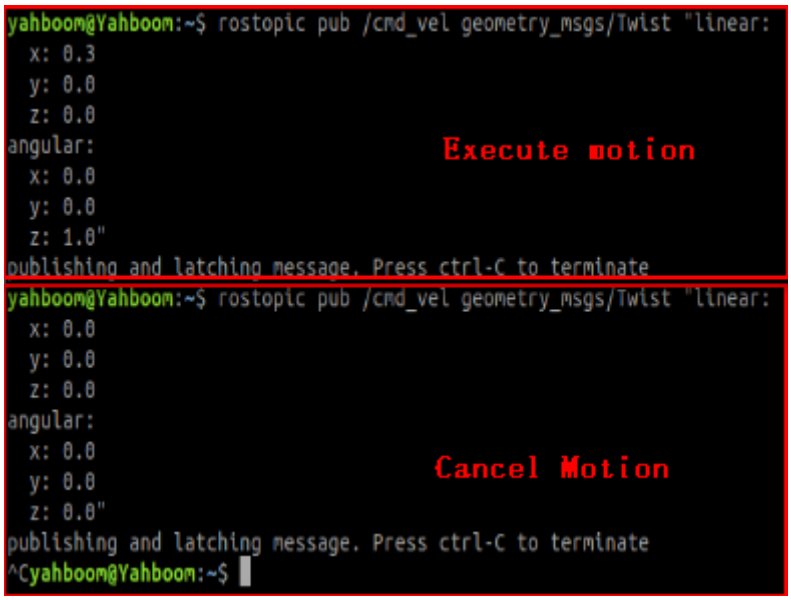
Input command in the terminal, as shown below.




| id | servo | angle | camera |
| --- | --- | --- | --- |
| 1 | Move left and right（X） | 【0，180】 | High frame rate |
| 2 | Move up and down（Y） | 【0，180】 | High frame rate |
| 1 | Move left and right（X） | 【50，130】 | Astra |

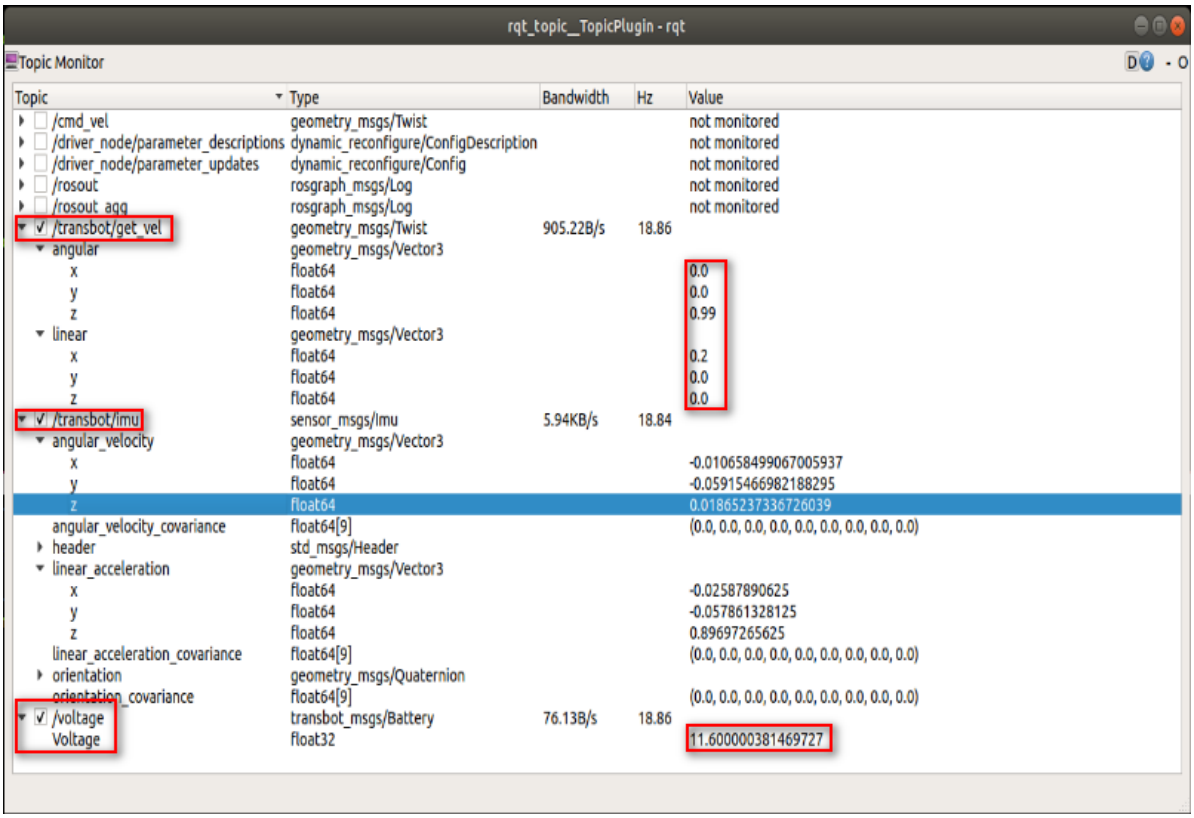
## 4) Check topic

Input command in the terminal, as shown below.




Use rqt_topic tool

```
rosrun rqt_topic rqt_topic
```



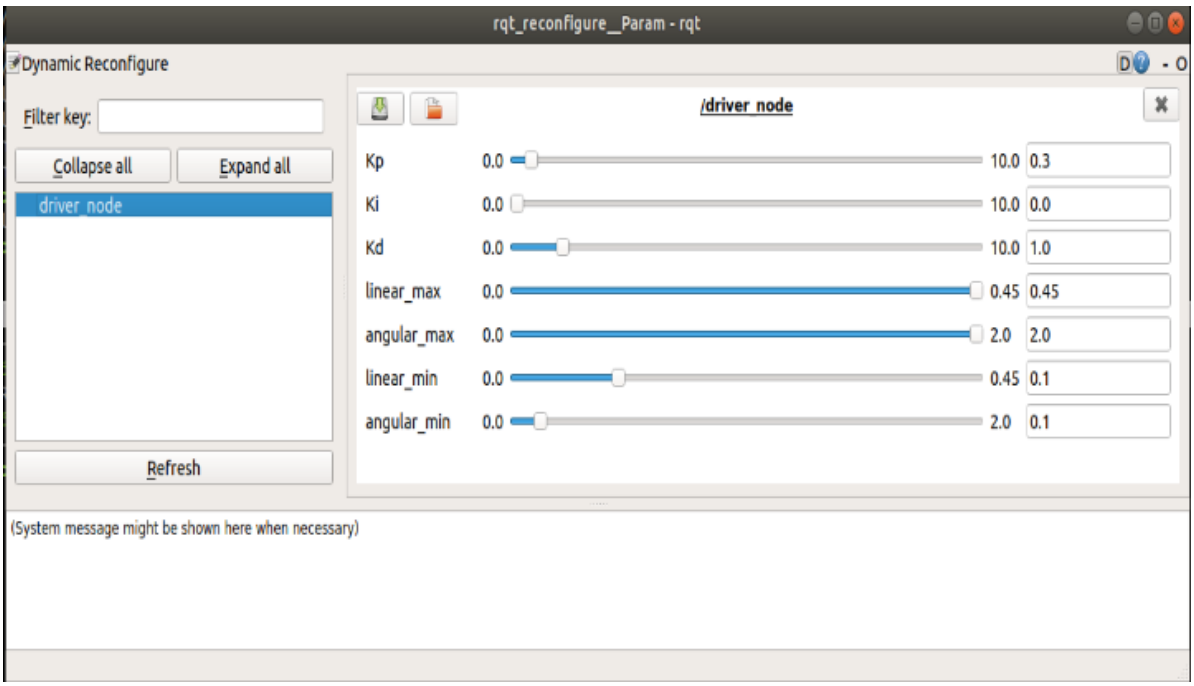

According to the figure above, we need to put a check mark in the front box before we can print the data.

At this time, the battery voltage is 11.6V; the linear velocity and angular velocity of the car are [0.2, 0.99] in decibels; there is also the imu information of the car.

## 5) Dynamic parameter configuration

No need adjust in our image system, this course just for reference.

```
rosrun rqt_reconfigure rqt_reconfigure
```




【linear_max】：line speed max value

【angular_max】：Angle speed max value

【linear_min】：line speed min value

【angular_min】：Angle speed min value