

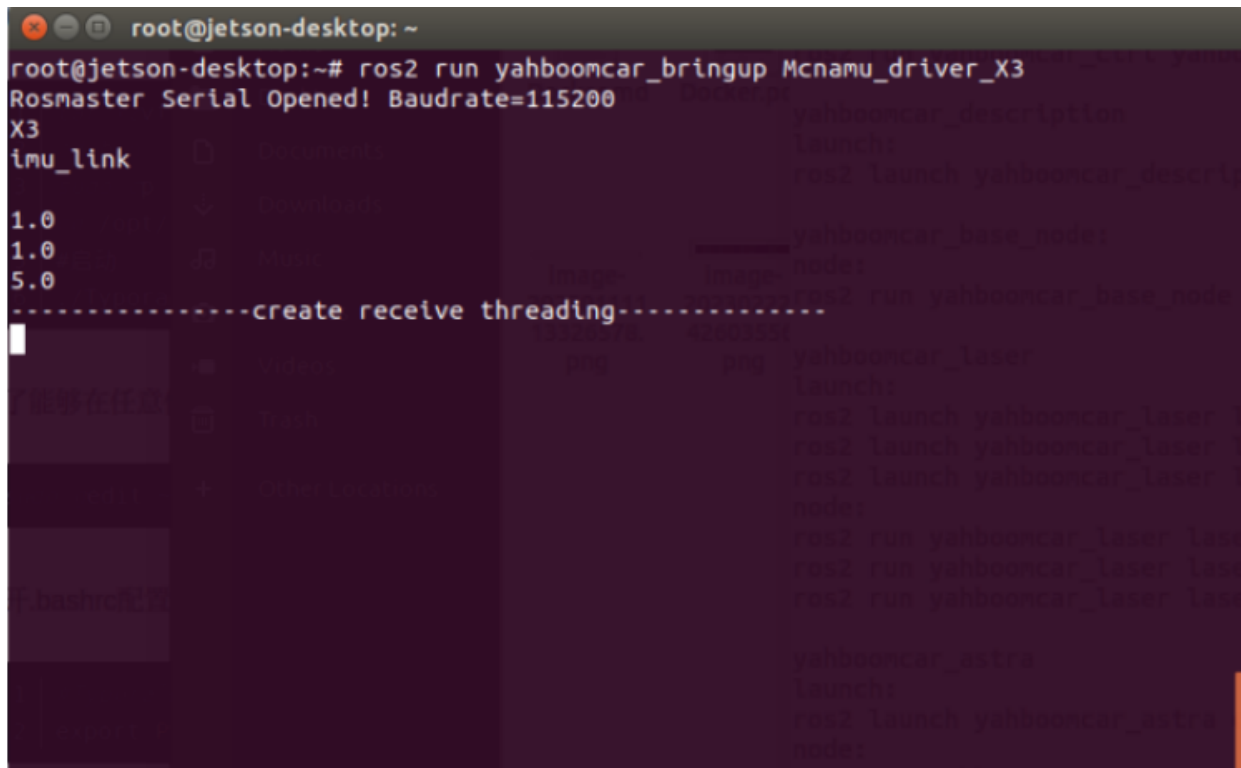
2、 Robot information release

After the program runs, the ROS expansion board will return real-time data, including voltage and IMU data. Connecting the car encoder can also obtain the speed of the car.

1、 start mode

Taking our company's Rosmaster-X3 as an example, the terminal enters the following command to start,

```
ros2 run yahboomcar_bringup Mcnamu_driver_X3
```



```
root@jetson-desktop: ~
root@jetson-desktop:~# ros2 run yahboomcar_bringup Mcnamu_driver_X3
Rosmaster Serial Opened! Baudrate=115200
X3
imu_link
1.0
1.0
5.0
-----create receive threading-----
yahboomcar_description
launch:
ros2 launch yahboomcar_descri
yahboomcar_base_node:
ros2 run yahboomcar_base node
yahboomcar_laser
launch:
ros2 launch yahboomcar_laser
ros2 launch yahboomcar_laser
ros2 launch yahboomcar_laser
node:
ros2 run yahboomcar_laser las
ros2 run yahboomcar_laser las
ros2 run yahboomcar_laser las
yahboomcar_astra
launch:
ros2 launch yahboomcar_astra
node:
```

2、 View topic nodes

Enter the command at the terminal,

```
ros2 topic list
```

```
root@jetson-desktop:/# ros2 topic list
/Buzzer
/RGBLight
/cmd_vel
/edition
/imu/data_raw
/imu/mag
/joint_states
/parameter_events
/rosout
/vel_raw
/voltage
```

Topic Name	topical contents
/Buzzer	buzzer
/RGBLight	Light strip effect control
/cmd_vel	Speed control
/edition	Version information
/imu/data_raw	IMU sensor data
/imu/mag	IMU Magnetometer Data
/vel_raw	Car speed information
/voltage	Battery voltage information

3、Reading Topic Data

If you want to read the voltage value, you need to enter the following command at the terminal ,

```
ros2 topic echo /voltage
```

```

root@jetson-desktop:/# ros2 topic echo /voltage
data: 10.600000381469727
---
data: 10.600000381469727
---
data: 10.600000381469727
---
data: 10.600000381469727
---
data: 10.600000381469727
---
data: 10.600000381469727
---
data: 10.600000381469727
---
data: 10.600000381469727

```

4、Publish topic data

To publish/cmd_ Taking speed data as an example, terminal input,

```

ros2 topic pub /cmd_vel geometry_msgs/msg/Twist "{linear: {x: 0.5, y: 0.0, z:0.0},
angular: {x: 0.0, y: 0.0, z: 0.2}}"

```

```

root@jetson-desktop:~# ros2 topic pub /cmd_vel geometry_msgs/msg/Twist "{linear:
{x: 0.5, y: 0.0, z: 0.0}, angular: {x: 0.0, y: 0.0, z: 0.0}}"
publisher: beginning loop
publishing #1: geometry_msgs.msg.Twist(linear=geometry_msgs.msg.Vector3(x=0.5, y
=0.0, z=0.0), angular=geometry_msgs.msg.Vector3(x=0.0, y=0.0, z=0.0))

publishing #2: geometry_msgs.msg.Twist(linear=geometry_msgs.msg.Vector3(x=0.5, y
=0.0, z=0.0), angular=geometry_msgs.msg.Vector3(x=0.0, y=0.0, z=0.0))

publishing #3: geometry_msgs.msg.Twist(linear=geometry_msgs.msg.Vector3(x=0.5, y
=0.0, z=0.0), angular=geometry_msgs.msg.Vector3(x=0.0, y=0.0, z=0.0))

publishing #4: geometry_msgs.msg.Twist(linear=geometry_msgs.msg.Vector3(x=0.5, y
=0.0, z=0.0), angular=geometry_msgs.msg.Vector3(x=0.0, y=0.0, z=0.0))

publishing #5: geometry_msgs.msg.Twist(linear=geometry_msgs.msg.Vector3(x=0.5, y
=0.0, z=0.0), angular=geometry_msgs.msg.Vector3(x=0.0, y=0.0, z=0.0))

publishing #6: geometry_msgs.msg.Twist(linear=geometry_msgs.msg.Vector3(x=0.5, y
=0.0, z=0.0), angular=geometry_msgs.msg.Vector3(x=0.0, y=0.0, z=0.0))

```

5、Source code parsing

```

from Rosmaster_Lib import Rosmaster #Import Driver Library
self.car = Rosmaster() #Instantiating Rosmaster objects
#create subscriber 创建订阅者
self.sub_cmd_vel =
self.create_subscription(Twist,"cmd_vel",self.cmd_vel_callback,1)
self.sub_RGBLight =
self.create_subscription(Int32,"RGBLight",self.RGBLightcallback,100)
self.sub_BUZZer =

```

```

self.create_subscription(Bool, "Buzzer", self.Buzzercallback, 100)
#create publisher
self.EdiPublisher = self.create_publisher(Float32, "edition", 100)
self.volPublisher = self.create_publisher(Float32, "voltage", 100)
self.staPublisher = self.create_publisher(JointState, "joint_states", 100)
self.velPublisher = self.create_publisher(Twist, "vel_raw", 50)
self.imuPublisher = self.create_publisher(Imu, "/imu/data_raw", 100)
self.magPublisher = self.create_publisher(MagneticField, "/imu/mag", 100)
#Call the library to read the information of the ROS expansion board
edition.data = self.car.get_version()*1.0
battery.data = self.car.get_battery_voltage()*1.0
ax, ay, az = self.car.get_accelerometer_data()
gx, gy, gz = self.car.get_gyroscope_data()
mx, my, mz = self.car.get_magnetometer_data()
vx, vy, angular = self.car.get_motion_data()
#Publish topic data
self.imuPublisher.publish(imu)
self.magPublisher.publish(mag)
self.volPublisher.publish(battery)
self.EdiPublisher.publish(edition)
self.velPublisher.publish(twist)
#Subscriber callback function
def cmd_vel_callback(self, msg)
def RGBLightcallback(self, msg)
def Buzzercallback(self, msg):

```

Please refer to the following path code for detailed
code:~/driver_ws/src/yahboomcar_bringup/yahboomcar_bringup/Mcnamu_driver_X3.py