Gmapping mapping

Note: The ROS_DOMAIN_ID of the Raspberry Pi and the microROS control board need to be consistent. You can check **[MicroROS Control Board Parameter Configuration]** to set the microROS control board ROS_DOMAIN_ID. Check the tutorial **[Connect MicroROS Agent]** to determine whether the IDs are consistent.

1、Introduction

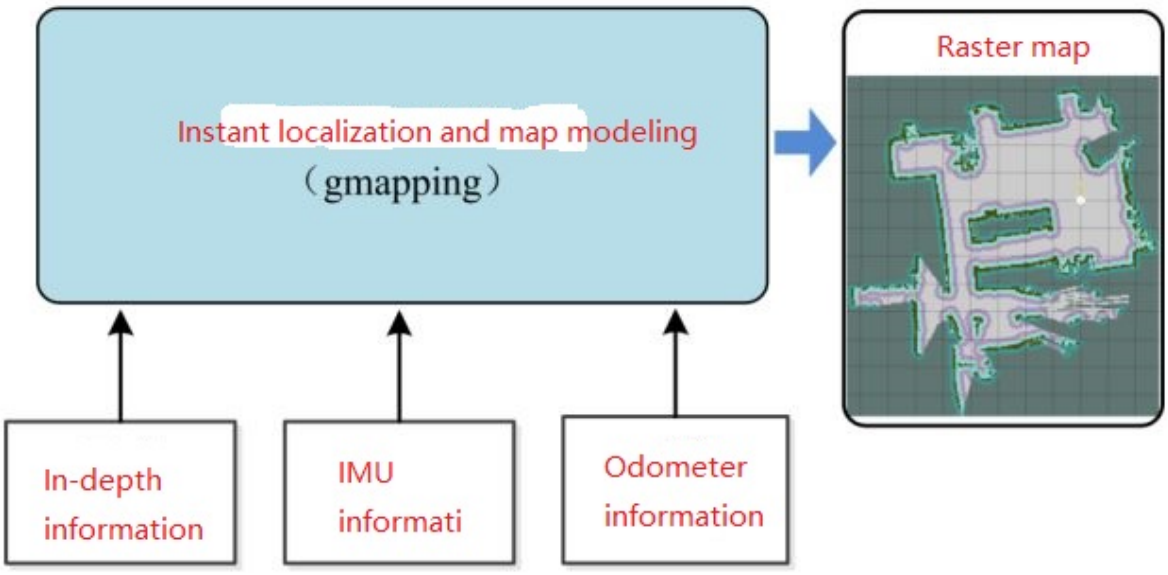
- gmapping is only suitable for points with a single frame two-dimensional laser point count less than 1440, if the number of laser points in a single frame is greater than 1440, then there will be such a problem as [mapping-4] process has died.
- Gmapping is a commonly used open source SLAM algorithm based on filtering SLAM frameworks.
- Based on the RBpf particle filter algorithm, Gmapping separates the real-time localization and mapping process, and locates before mapping.
- Gmapping makes two major improvements to the RBpf algorithm: improved proposed distribution and selective resampling.

Advantages: Gmapping can build indoor maps in real time, which requires less computation and high accuracy in building small scene maps.

Cons: The number of particles required increases as the scene grows, as each particle carries a map, hence the memory required to build a large map

The amount of computation increases. Therefore, it is not suitable for building large scene maps. And there is no loopback detection, so it may cause a map when the loopback is closed

Dislocation, although increasing the number of particles can close the map, but at the cost of increased computation and memory.



!

2、Program function description

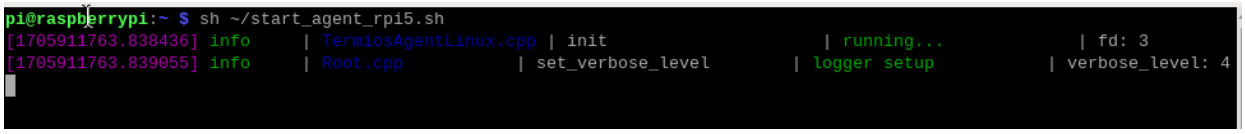
Connect the car to the agent and run the program. The mapping interface will be displayed in rviz. Use the keyboard or handle to control the movement of the car until the map is completed. Then run the save map command to save the map.

3、Query car information

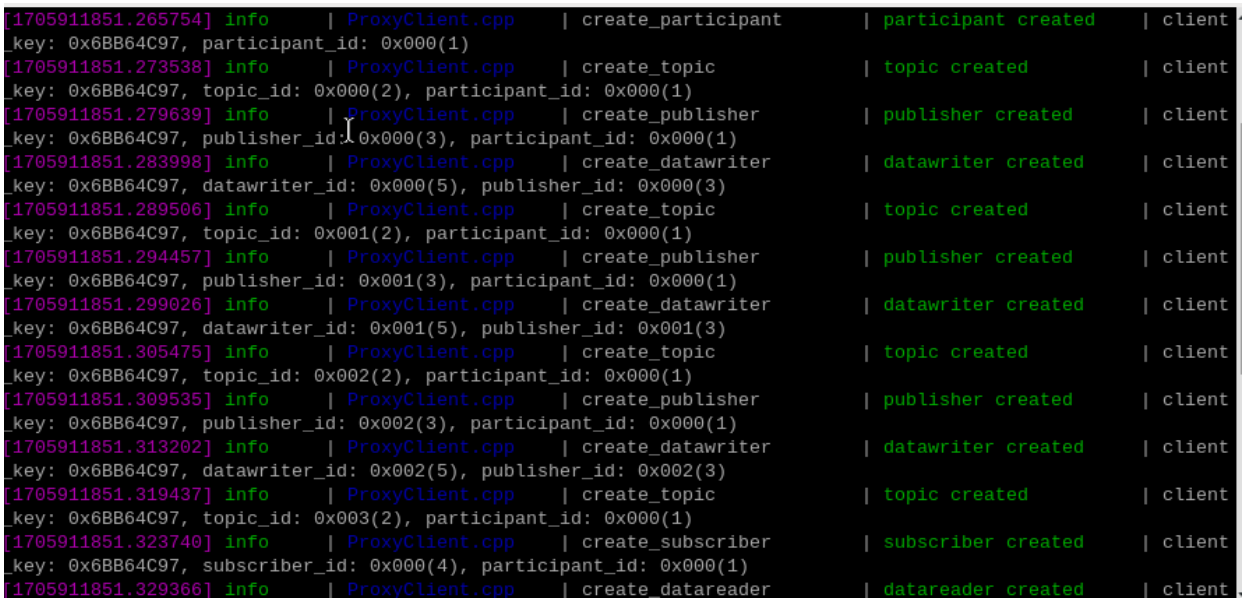
3.1、Start and connect to the agent

After the Raspberry Pi is successfully powered on, open the terminal and enter the following command to open the agent.

```
sh ~/start_agent_rpi5.sh
```



Press the reset button on the microROS control board and wait for the car to connect to the agent. The connection is successful as shown in the figure below.

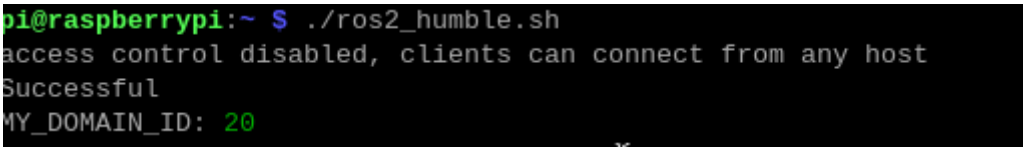


3.2、Enter the car docker

Open another terminal and enter the following command to enter docker.

```
sh ros2_humble.sh
```

When the following interface appears, you have successfully entered docker. Now you can control the car through commands.



4、starting program

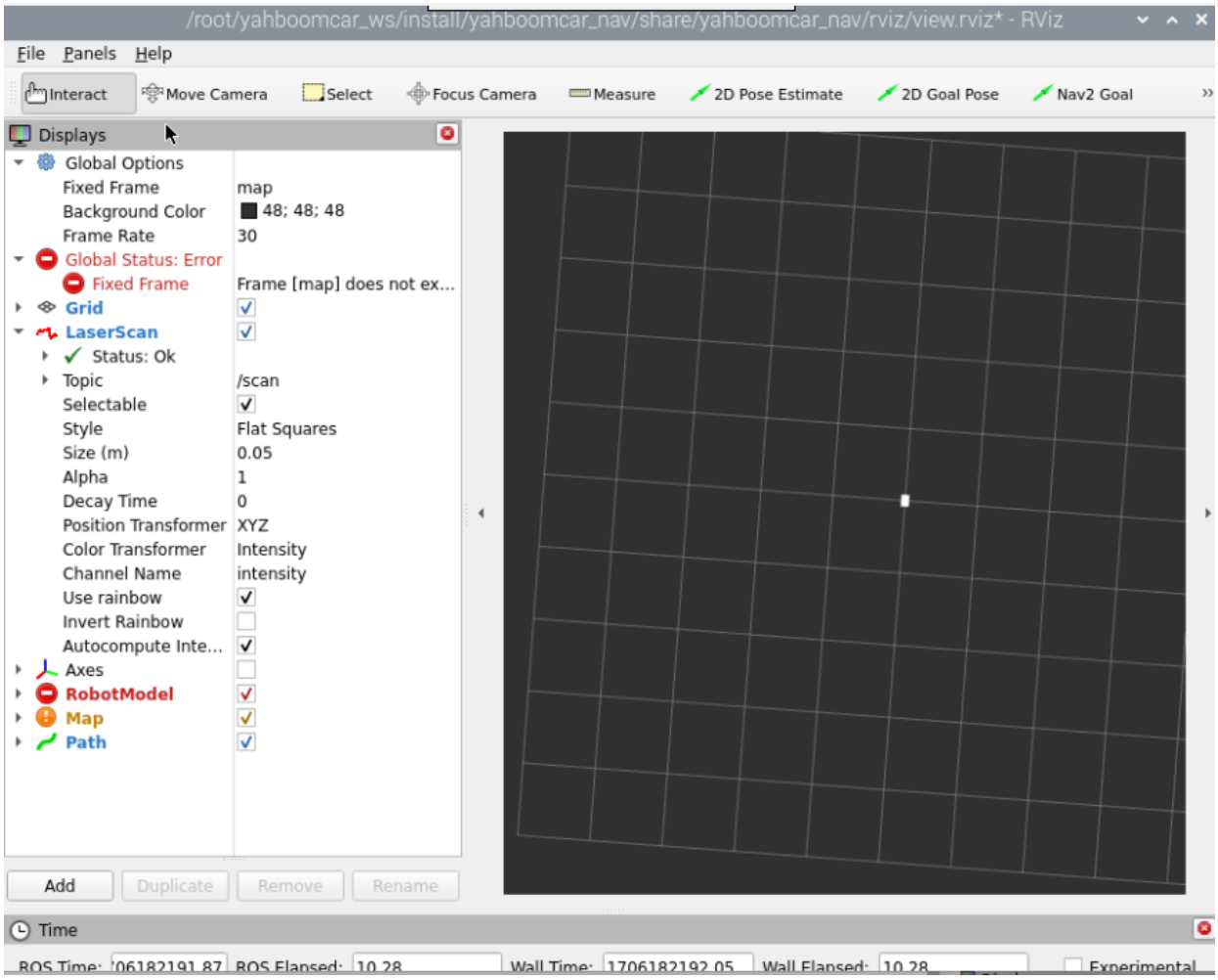
First, start the car to process the underlying data program, and enter the following command in the terminal.

```
ros2 launch yahboomcar_bringup yahboomcar_bringup_launch.py
```

```
[INFO] [imu_filter_madgwick_node-1]: process started with pid [6263]
[INFO] [ekf_node-2]: process started with pid [6265]
[INFO] [static_transform_publisher-3]: process started with pid [6267]
[INFO] [joint_state_publisher-4]: process started with pid [6269]
[INFO] [robot_state_publisher-5]: process started with pid [6271]
[INFO] [static_transform_publisher-6]: process started with pid [6286]
[static_transform_publisher-3] [WARN] [1706181650.342105372] []: Old-style arguments are deprecated; see --help for new-style arguments
[static_transform_publisher-3] [INFO] [1706181650.459314055] [base_link_to_base_imu]: Spinning until stopped - publishing transform
[static_transform_publisher-3] translation: ('-0.002999', '-0.003000', '0.031701')
[static_transform_publisher-3] rotation: ('0.000000', '0.000000', '0.000000', '1.000000')
[static_transform_publisher-3] from 'base_link' to 'imu_frame'
[imu_filter_madgwick_node-1] [INFO] [1706181650.478942143] [imu_filter]: Starting ImuFilter
[imu_filter_madgwick_node-1] [INFO] [1706181650.480114862] [imu_filter]: Using dt computed from message headers
[imu_filter_madgwick_node-1] [INFO] [1706181650.480197121] [imu_filter]: The gravity vector is kept in the IMU message.
[imu_filter_madgwick_node-1] [INFO] [1706181650.480631749] [imu_filter]: Imu filter gain set to 0.100000
[imu_filter_madgwick_node-1] [INFO] [1706181650.480707508] [imu_filter]: Gyro drift bias set to 0.000000
[imu_filter_madgwick_node-1] [INFO] [1706181650.480720249] [imu_filter]: Magnetometer bias values: 0.000000 0.000000 0.000000
[static_transform_publisher-6] [WARN] [1706181650.493533858] []: Old-style arguments are deprecated; see --help for new-style arguments
[robot_state_publisher-5] [WARN] [1706181650.639387954] [kdl_parser]: The root link base_link has an inertia specified in the URDF, but KDL does not support a root link with an inertia.  As a workaround, you can add an extra dummy link to your URDF.
[robot_state_publisher-5] [INFO] [1706181650.640061915] [robot_state_publisher]: got segment base_link
[robot_state_publisher-5] [INFO] [1706181650.640174267] [robot_state_publisher]: got segment imu_Link
[robot_state_publisher-5] [INFO] [1706181650.640191229] [robot_state_publisher]: got segment jq1_Link
[robot_state_publisher-5] [INFO] [1706181650.640201822] [robot_state_publisher]: got segment jq2_Link
[robot_state_publisher-5] [INFO] [1706181650.640211952] [robot_state_publisher]: got segment radar_Link
[robot_state_publisher-5] [INFO] [1706181650.640221211] [robot_state_publisher]: got segment yh_Link
[robot_state_publisher-5] [INFO] [1706181650.640229452] [robot_state_publisher]: got segment yq_Link
[robot_state_publisher-5] [INFO] [1706181650.640238470] [robot_state_publisher]: got segment zh_Link
[robot_state_publisher-5] [INFO] [1706181650.640246655] [robot_state_publisher]: got segment zq_Link
[imu_filter_madgwick_node-1] [INFO] [1706181650.655098332] [imu_filter]: First IMU message received.
[static_transform_publisher-6] [INFO] [1706181650.681177050] [static_transform_publisher_JarNTEa1OrW2k0Zb]: Spinning until stopped - publishing transform
[static_transform_publisher-6] translation: ('0.000000', '0.000000', '0.050000')
[static_transform_publisher-6] rotation: ('0.000000', '0.000000', '0.000000', '1.000000')
[static_transform_publisher-6] from 'base_footprint' to 'base_link'
[joint_state_publisher-4] [INFO] [1706181650.989117137] [joint_state_publisher]: Waiting for robot_description to be published on the robot_description topic...
```

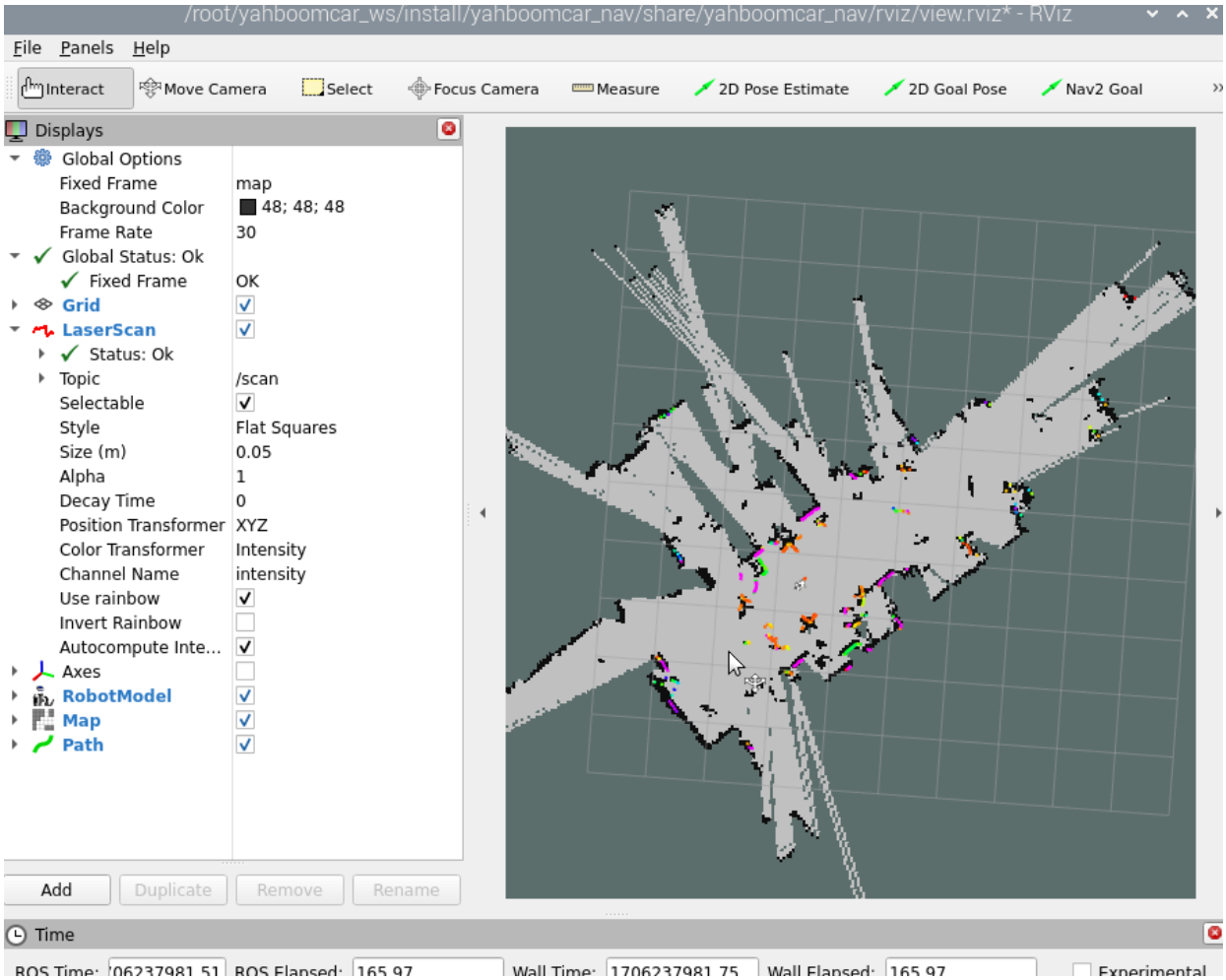
Then enter the following command in the terminal to start rviz for visual mapping.

```
ros2 launch yahboomcar_nav display_launch.py
```



The mapping node has not been run yet, so there is no data. Next, run the mapping node and enter the following command in the terminal

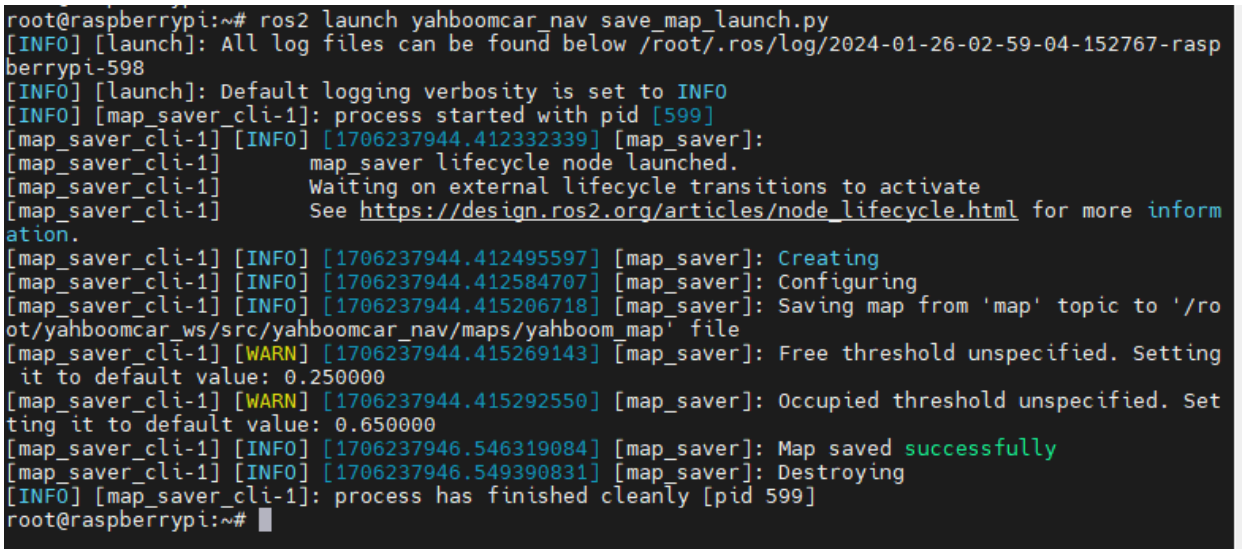
```
ros2 launch yahboomcar_nav map_gmapping_launch.py
```



Then run handle control or keyboard control, choose one of the two, and enter the following command in the terminal.

```
#keyboard
ros2 run yahboomcar_ctrl yahboom_keyboard
#handle
ros2 run yahboomcar_ctrl yahboom_joy
ros2 run joy joy_node
```

Then control the car and slowly walk through the area that needs to be mapped. After the map is completed, enter the following command to save the map, and enter the following command on the terminal.




```
ros2 launch yahboomcar_nav save_map_launch.py
```

A map named yahboom_map will be saved. This map is saved in .

```
/root/yahboomcar_ws/src/yahboomcar_nav/maps
```

Two files will be generated, one is yahboom_map.pgm and the other is yahboom_map.yaml. Take a look at the content of yaml.

```
image: yahboom_map.pgm
mode: trinary
resolution: 0.05
origin: [-10, -10, 0]
negate: 0
occupied_thresh: 0.65
free_thresh: 0.25
```

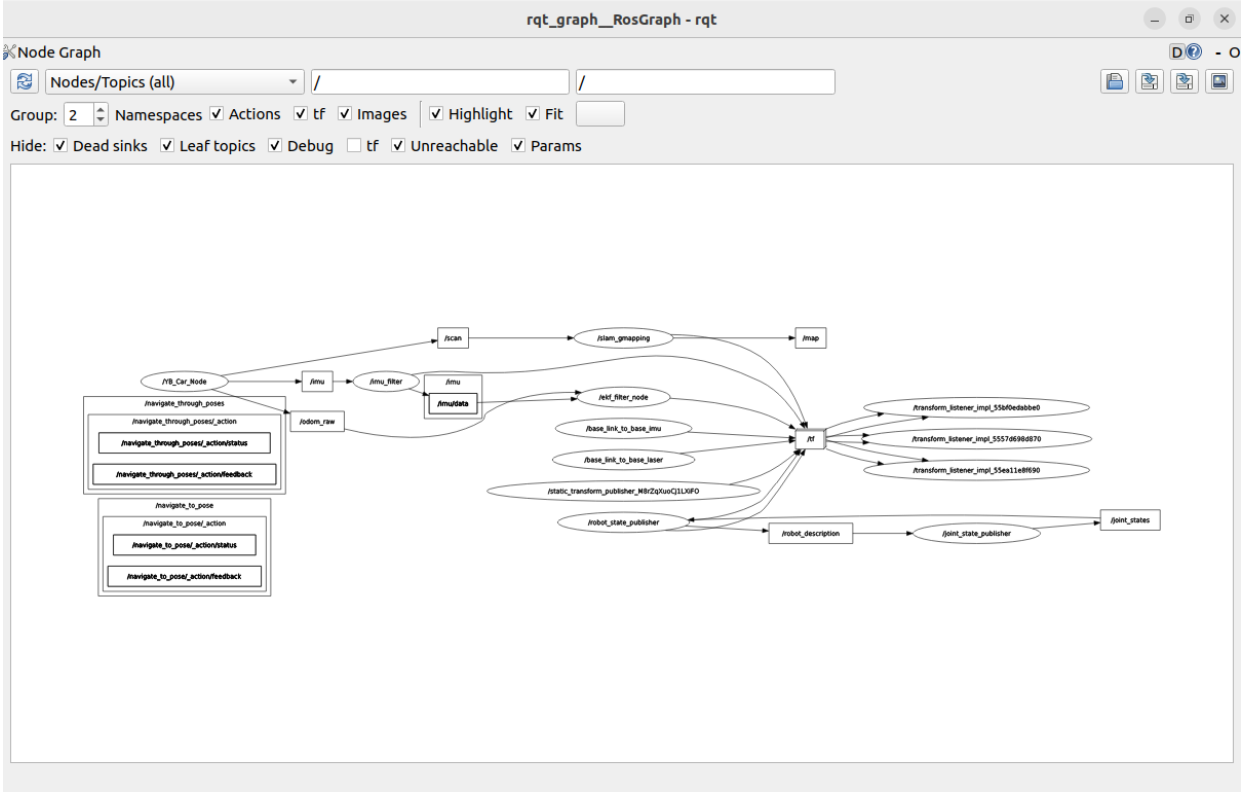
- image：The picture representing the map, that is, yahboom_map.pgm
- mode：
- This attribute can be one of trinary, scale, or raw, depending on the mode selected, trinary mode is the default mode
- Resolution: The resolution of the map, meters per pixel
- resolution：The resolution of the map, meters per pixel
- origin：2D pose (x,y,yaw) in the lower left corner of the map, where yaw is rotated counterclockwise (yaw=0 means no rotation). Many parts of the system today ignore the YAW value.
- negate：whether to reverse the meaning of white/black, free/occupy (the interpretation of the threshold is not affected)
- occupied_thresh：Pixels with a probability of occupancy greater than this threshold are considered fully occupied.

- free_thresh：Pixels with a probability of occupancy less than this threshold are considered completely free.

5、View node communication diagram

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

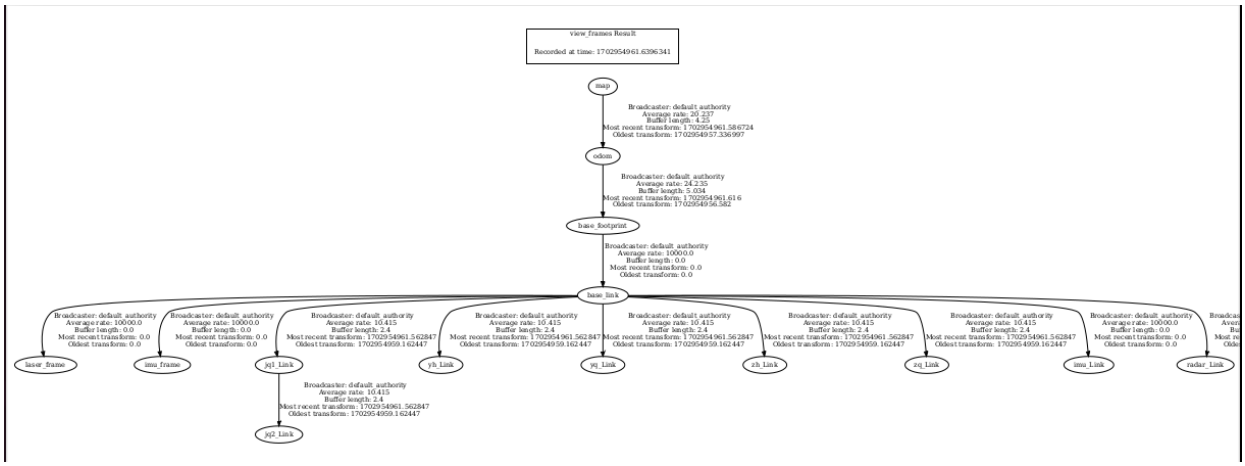
6、View TF tree

Enter the following command in the terminal.

```
ros2 run tf2_tools view_frames
```

```
root@raspberrypi:~# ros2 run tf2_tools view_frames
[INFO] [1706240274.802480879] [view_frames]: Listening to tf data for 5.0 seconds...
[INFO] [1706240279.805539455] [view_frames]: Generating graph in frames.pdf file...
[INFO] [1706240279.811022965] [view_frames]: Result:tf2_msgs.srv.FrameGraph_Response(frame_yaml=
"odom: \n  parent: 'map'\n  broadcaster: 'default_authority'\n  rate: 193.930\n  most_recent_tra
nsform: 1706240279.804413\n  oldest_transform: 1706240274.771664\n  buffer_length: 5.033\nimu_fr
ame: \n  parent: 'base_link'\n  broadcaster: 'default_authority'\n  rate: 10000.000\n  most_rece
nt_transform: 0.000000\n  oldest_transform: 0.000000\n  buffer_length: 0.000\nbase_link: \n  par
ent: 'base_footprint'\n  broadcaster: 'default_authority'\n  rate: 10000.000\n  most_recent_tran
sform: 0.000000\n  oldest_transform: 0.000000\n  buffer_length: 0.000\nbase_footprint: \n  paren
t: 'odom'\n  broadcaster: 'default_authority'\n  rate: 19.545\n  most_recent_transform: 17062402
79.773000\n  oldest_transform: 1706240274.810000\n  buffer_length: 4.963\nlaser_frame: \n  paren
t: 'base_link'\n  broadcaster: 'default_authority'\n  rate: 10000.000\n  most_recent_transform:
0.000000\n  oldest_transform: 0.000000\n  buffer_length: 0.000\njq1_Link: \n  parent: 'base_link
'\n  broadcaster: 'default_authority'\n  rate: 10.204\n  most_recent_transform: 1706240279.71725
1\n  oldest_transform: 1706240274.817256\n  buffer_length: 4.900\njq2_Link: \n  parent: 'jq1_Lin
k'\n  broadcaster: 'default_authority'\n  rate: 10.204\n  most_recent_transform: 1706240279.7172
51\n  oldest_transform: 1706240274.817256\n  buffer_length: 4.900\nyh_Link: \n  parent: 'base_li
nk'\n  broadcaster: 'default_authority'\n  rate: 10.204\n  most_recent_transform: 1706240279.717
251\n  oldest_transform: 1706240274.817256\n  buffer_length: 4.900\nyq_Link: \n  parent: 'base_l
ink'\n  broadcaster: 'default_authority'\n  rate: 10.204\n  most_recent_transform: 1706240279.71
7251\n  oldest_transform: 1706240274.817256\n  buffer_length: 4.900\nzh_Link: \n  parent: 'base_
link'\n  broadcaster: 'default_authority'\n  rate: 10.204\n  most_recent_transform: 1706240279.7
17251\n  oldest_transform: 1706240274.817256\n  buffer_length: 4.900\nzq_Link: \n  parent: 'base
_link'\n  broadcaster: 'default_authority'\n  rate: 10.204\n  most_recent_transform: 1706240279.
717251\n  oldest_transform: 1706240274.817256\n  buffer_length: 4.900\nimu_Link: \n  parent: 'ba
se_link'\n  broadcaster: 'default_authority'\n  rate: 10000.000\n  most_recent_transform: 0.0000
00\n  oldest_transform: 0.000000\n  buffer_length: 0.000\nradar_Link: \n  parent: 'base_link'\n
 broadcaster: 'default_authority'\n  rate: 10000.000\n  most_recent_transform: 0.000000\n  oldes
t_transform: 0.000000\n  buffer_length: 0.000\n")
root@raspberrypi:~#
```

After the operation is completed, two files will be generated in the terminal directory, namely .gv and .pdf files. The pdf file is the TF tree.



7、Code analysis

Here we only describe map_gmapping_launch.py for mapping. The path to this file is as follows.

```
/root/yahboomcar_ws/src/yahboomcar_nav/launch
```

map_gmapping_launch.py

```
from launch import LaunchDescription
from launch_ros.actions import Node
import os
from launch.actions import IncludeLaunchDescription
from launch.launch_description_sources import PythonLaunchDescriptionSource
```

```
from ament_index_python.packages import get_package_share_directory


def generate_launch_description():
    slam_gmapping_launch = IncludeLaunchDescription(
        PythonLaunchDescriptionSource([os.path.join(
        get_package_share_directory('slam_gmapping'), 'launch'),
         '/slam_gmapping.launch.py'])
    )

    base_link_to_laser_tf_node = Node(
     package='tf2_ros',
     executable='static_transform_publisher',
     name='base_link_to_base_laser',
     arguments=['-0.0046412', '0' ,
'0.094079','0','0','0','base_link','laser_frame']
    )

    return LaunchDescription([slam_gmapping_launch,base_link_to_laser_tf_node])
```

A launch file-slam_gmapping_launch and a node for publishing static transformation-base_link_to_laser_tf_node are started here. The slam_gmapping_launch file is as follows, you can go into the system to view the code

```
/root/gmapping_ws/src/slam_gmapping/launch
```

slam_gmapping.launch.py,

```
from launch import LaunchDescription
from launch.substitutions import EnvironmentVariable
import launch.actions
import launch_ros.actions
import os
from ament_index_python.packages import get_package_share_directory

def generate_launch_description():
    return LaunchDescription([
        launch_ros.actions.Node(
            package='slam_gmapping',
            executable='slam_gmapping',
            output='screen',
            parameters=[os.path.join(get_package_share_directory("slam_gmapping"),
"params", "slam_gmapping.yaml")]),
    ])
```

The slam_gmapping node is started here, and the slam_gmapping.yaml parameter file is loaded. The file is located (taking the supporting virtual machine as an example).

```
/root/gmapping_ws/src/slam_gmapping/params
```

slam_gmapping.yaml

```yaml
/slam_gmapping:
  ros__parameters:
    angularUpdate: 0.5
    astep: 0.05
    base_frame: base_footprint
    map_frame: map
    odom_frame: odom
    delta: 0.05
    iterations: 5
    kernelSize: 1
    lasamplerange: 0.005
    lasamplestep: 0.005
    linearUpdate: 1.0
    llsamplerange: 0.01
    llsamplestep: 0.01
    lsigma: 0.075
    lskip: 0
    lstep: 0.05
    map_update_interval: 5.0
    maxRange: 6.0
    maxUrange: 4.0
    minimum_score: 0.0
    occ_thresh: 0.25
    ogain: 3.0
    particles: 30
    qos_overrides:
      /parameter_events:
        publisher:
          depth: 1000
          durability: volatile
          history: keep_all
          reliability: reliable
      /tf:
        publisher:
          depth: 1000
          durability: volatile
          history: keep_last
          reliability: reliable
    resampleThreshold: 0.5
    sigma: 0.05
    srr: 0.1
    srt: 0.2
    str: 0.1
    stt: 0.2
    temporalUpdate: 1.0
    transform_publish_period: 0.05
    use_sim_time: false
    xmax: 10.0
    xmin: -10.0
    ymax: 10.0
```

```
    ymin: -10.0
```