

## 4、机器人状态估计

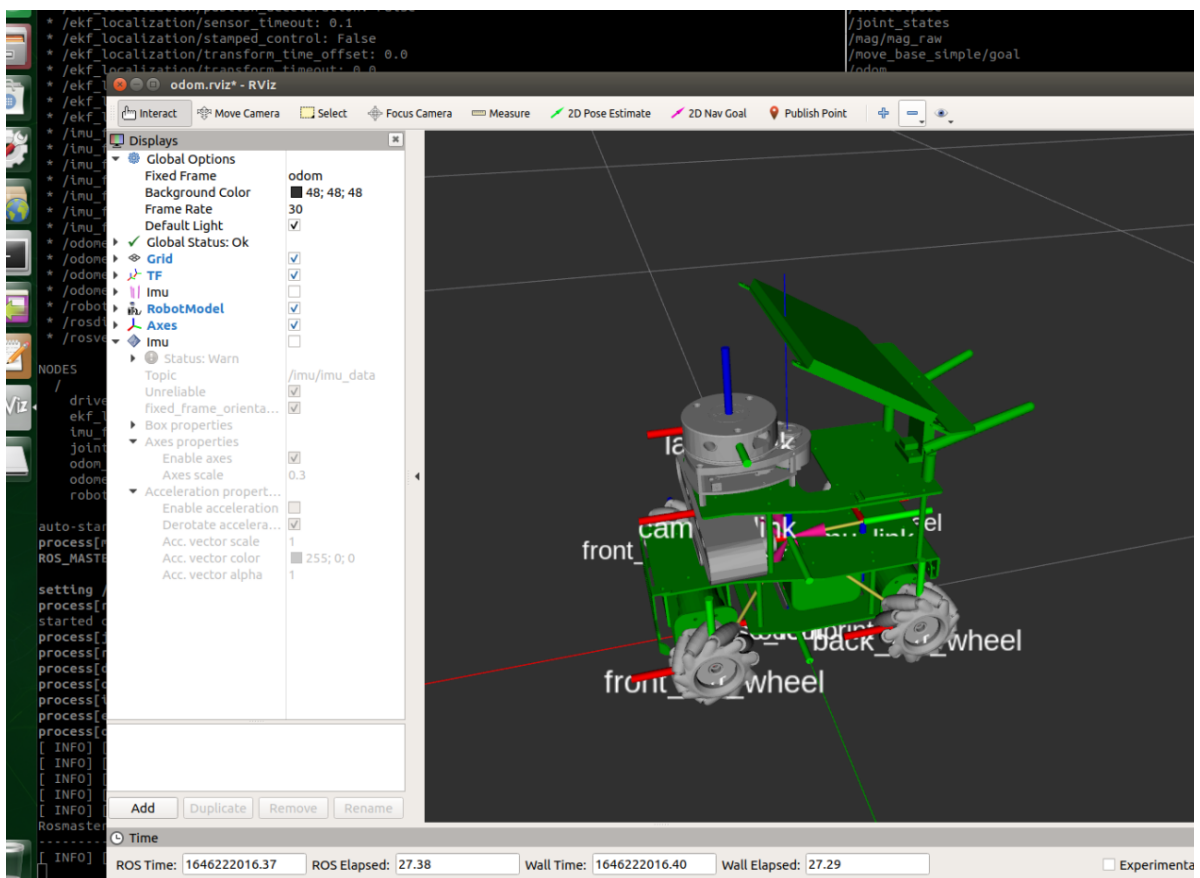
### 4.1、启动

#### 4.1.1、代码参考路径

```
~/driver_ws/src/yahboomcar_bringup/launch/bringup.launch
```

#### 4.1.1、启动

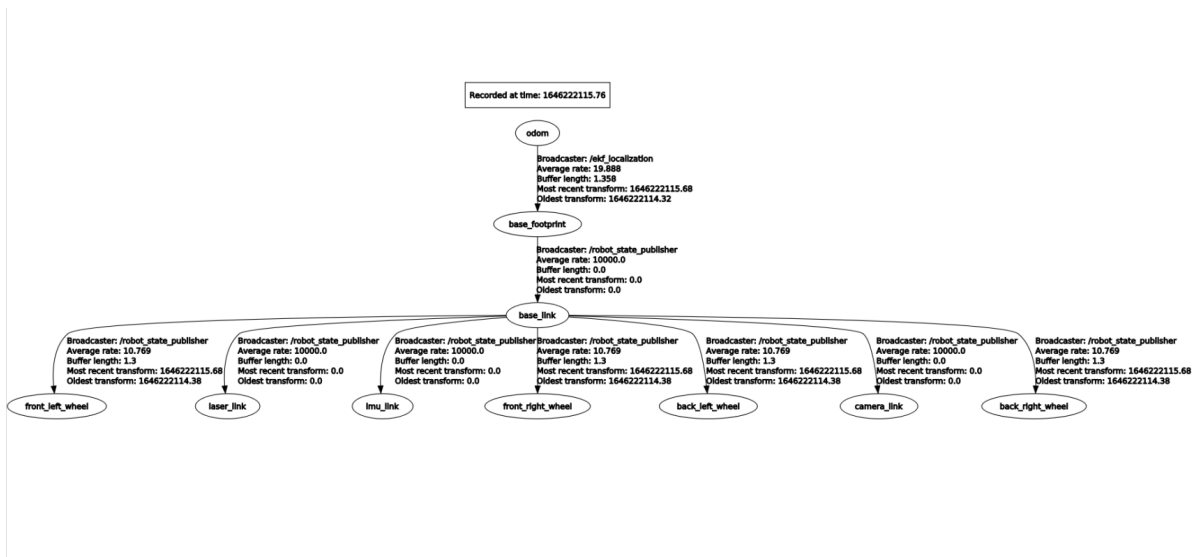
```
cd ~/driver_ws/src/yahboomcar_bringup/launch  
roslaunch bringup.launch
```



#### 4.1.1、查看tf树和节点图

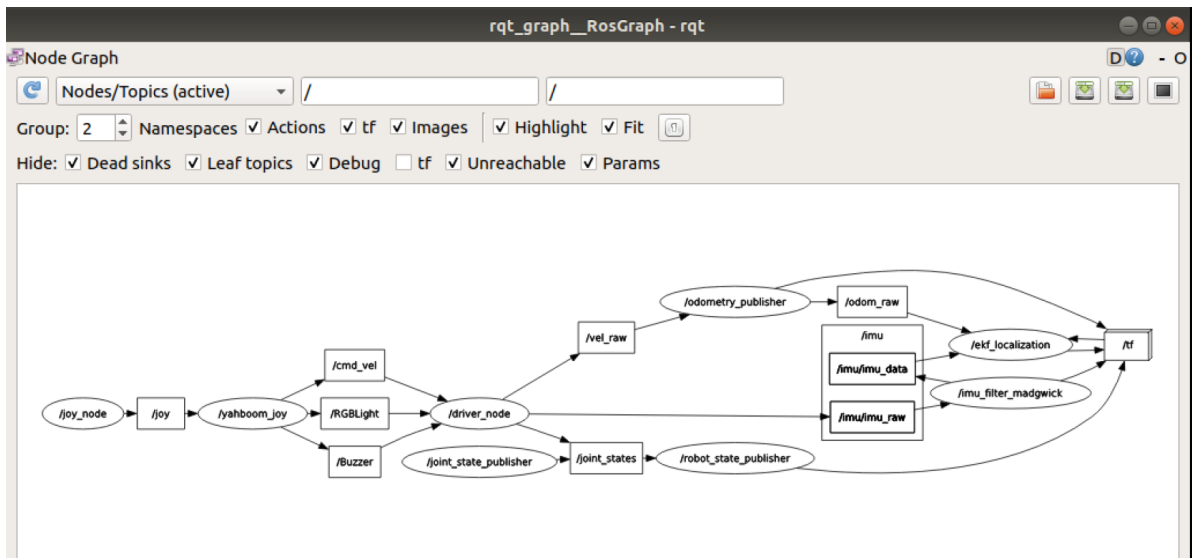
##### 1)、查看TF树

```
roslaunch rqt_tf_tree rqt_tf_tree
```



## 2)、查看节点图

```
roslaunch rqt_graph rqt_graph
```



## 4.1.2、launch文件解析

在bringup.launch这个文件里边，有几个重要的节点

### 1)、/drive\_node

这个节点主要是发布了/imu/imu\_raw和/vel\_raw数据，终端输入

```
roslaunch bringup bringup.launch
```

```
rojetson@yahboom:~$ rosnode info /driver_node
-----
Node [/driver_node]
Publications:
  * /driver_node/parameter_descriptions [dynamic_reconfigure/ConfigDescription]
  * /driver_node/parameter_updates [dynamic_reconfigure/Config]
  * /redition [std_msgs/Float32]
  * /imu/imu_raw [sensor_msgs/Imu]
  * /joint_states [sensor_msgs/JointState]
  * /mag/mag_raw [sensor_msgs/MagneticField]
  * /rosout [roscpp_msgs/Log]
  * /vel_raw [geometry_msgs/Twist]
  * /voltage [std_msgs/Float32]
Subscriptions:
  * / buzzer [unknown type]
  * /rcbLight [unknown type]
  * /cmd_vel [unknown type]
Services:
  * /driver_node/get_loggers
  * /driver_node/set_logger_level
  * /driver_node/set_parameters
contacting node http://192.168.2.98:40137/ ...
Pid: 972
Connections:
  * topic: /joint_states
    * to: /robot_state_publisher
      * direction: outbound (36531 - 192.168.2.98:49708) [27]
      * transport: TCPROS
  * topic: /vel_raw
    * to: /odometry_publisher
      * direction: outbound (36531 - 192.168.2.98:49712) [24]
      * transport: TCPROS
  * topic: /rosout
    * to: /rosout
      * direction: outbound (36531 - 192.168.2.98:49706) [9]
      * transport: TCPROS
  * topic: /imu/imu_raw
    * to: /imu_filter_madgwick
      * direction: outbound (36531 - 192.168.2.98:49710) [23]
      * transport: TCPROS
```

可以看出，

- /driver\_node节点发布/imu/imu\_raw话题数据到/imu\_filter\_madgwick这个节点，后者做过滤和融合imu数据；
- /driver\_node节点发布/vel\_raw话题数据到/odometry\_publisher这个节点，后者整合后，再发布/odom\_raw数据；

## 2)、/odometry\_publisher

这个节点主要是接收/driver\_node节点发送过来的vel\_raw话题数据、发布/odom\_raw话题数据到/ekf\_localization以及把/tf话题数据发布到其他节点，终端输入，

```
rosnode info /odometry_publisher
```

```
rojetson@yahboom:~$ rosnode info /odometry_publisher
-----
Node [/odometry_publisher]
Publications:
  * /odom_raw [nav_msgs/Odometry]
  * /rosout [roscpp_msgs/Log]
  * /tf [tf2_msgs/TFMessage]
Subscriptions:
  * /vel_raw [geometry_msgs/Twist]
Services:
  * /odometry_publisher/get_loggers
  * /odometry_publisher/set_logger_level
contacting node http://192.168.2.98:43625/ ...
Pid: 974
Connections:
  * topic: /rosout
    * to: /rosout
      * direction: outbound (36909 - 192.168.2.98:58714) [10]
      * transport: TCPROS
  * topic: /tf
    * to: /ekf_localization
      * direction: outbound (36909 - 192.168.2.98:58902) [12]
      * transport: TCPROS
  * topic: /tf
    * to: /odom_rviz
      * direction: outbound (36909 - 192.168.2.98:59218) [14]
      * transport: TCPROS
  * topic: /tf
    * to: /rqt_gui_py_node_5177
      * direction: outbound (36909 - 192.168.2.98:59390) [15]
      * transport: TCPROS
  * topic: /odom_raw
    * to: /ekf_localization
      * direction: outbound (36909 - 192.168.2.98:59100) [13]
      * transport: TCPROS
  * topic: /vel_raw
    * to: /driver_node (http://192.168.2.98:40137/)
      * direction: inbound (49712 - 192.168.2.98:36531) [16]
      * transport: TCPROS
```

## 3)、/imu\_filter\_madgwick

这个节点主要是过滤和融合imu数据，然后再把处理过后的/imu/imu\_data话题数据发布出去给/ekf\_localization这个节点接收以及发布/tf数据到其他节点，终端输入，

```
rosnode info /imu_filter_madgwick
```

```
transport: TCPROS
* topic: /tf
* to: /ekf_localization
* direction: outbound (57775 - 192.168.2.98:40314) [12]
* transport: TCPROS
* topic: /tf
* to: /odom_rviz
* direction: outbound (57775 - 192.168.2.98:40630) [14]
* transport: TCPROS
* topic: /tf
* to: /rqt_gui_py_node_5177
* direction: outbound (57775 - 192.168.2.98:40802) [15]
* transport: TCPROS
* topic: /imu/imu_data
* to: /ekf_localization
* direction: outbound (57775 - 192.168.2.98:40512) [13]
* transport: TCPROS
* topic: /imu/imu_raw
* to: /driver_node (http://192.168.2.98:40137/)
* direction: inbound (40710 - 192.168.2.98:36531) [16]
* transport: TCPROS
```

#### 4) 、/ekf\_localization

这个节点主要是融合imu数据和odom数据以及发布tf数据，终端输入，

```
rostopic info /ekf_localization
```

```
contacting node http://192.168.2.98:42883/ ...
Pid: 993
Connections:
* topic: /rosout
* to: /rosout
* direction: outbound (40483 - 192.168.2.98:49346) [11]
* transport: TCPROS
* topic: /tf
* to: /ekf_localization
* direction: outbound
* transport: INTRAPROCESS
* topic: /tf
* to: /odom_rviz
* direction: outbound (40483 - 192.168.2.98:49800) [12]
* transport: TCPROS
* topic: /tf
* to: /rqt_gui_py_node_5177
* direction: outbound (40483 - 192.168.2.98:49906) [14]
* transport: TCPROS
* topic: /tf
* to: /ekf_localization (http://192.168.2.98:42883/)
* direction: inbound
* transport: INTRAPROCESS
* topic: /tf
* to: /robot_state_publisher (http://192.168.2.98:45697/)
* direction: inbound (55732 - 192.168.2.98:52579) [17]
* transport: TCPROS
* topic: /tf
* to: /odometry_publisher (http://192.168.2.98:43625/)
* direction: inbound (58902 - 192.168.2.98:36909) [18]
* transport: TCPROS
* topic: /tf
* to: /imu_filter_madgwick (http://192.168.2.98:40889/)
* direction: inbound (40314 - 192.168.2.98:57775) [19]
* transport: TCPROS
* topic: /tf_static
* to: /robot_state_publisher (http://192.168.2.98:45697/)
* direction: inbound (55730 - 192.168.2.98:52579) [16]
* transport: TCPROS
* topic: /odom_raw
* to: /odometry_publisher (http://192.168.2.98:43625/)
* direction: inbound (59100 - 192.168.2.98:36909) [13]
* transport: TCPROS
* topic: /imu/imu_data
* to: /imu_filter_madgwick (http://192.168.2.98:40889/)
* direction: inbound (40512 - 192.168.2.98:57775) [10]
* transport: TCPROS
```

接下来，我们对这几个节点做详细说明。

### 4.1.3、imu\_filter\_madgwick

#### 1) 、简介

IMU 是指六轴传感器，包含陀螺仪和加速度计。MARG 是指九轴传感器，在 IMU 的基础上添加了磁力计。

```
IMU = gyroscope + accelerometer
MARG(Magnetic, Angular Rate, and Gravity) = gyroscope + accelerometer + magnetometer
```

Madgwick 是一个 Orientation Filter，用于过滤和融合来自IMU设备的原始数据。它将来自通用IMU设备的角速度、加速度和（可选）磁力计读数融合到方向四元数中，并在IMU话题上发布融合数据，并不考虑整个 IMU 的积分过程。

#### 2) 、话题

/	话题名	类型	解析
Subscribed	/imu/data_raw	sensor_msgs/Imu	校准后的IMU数据的消息，包括角速度和线加速度
Subscribed	/imu/mag	sensor_msgs/MagneticField	[可选]磁力计，会受磁场影响
Published	/imu/data	sensor_msgs/Imu	融合的Imu信息。

3) 、参数

参数名	类型	默认值	解析
~gain	double	0.1	滤波器的增益。 值越高，收敛速度越快，但噪声越大。 值越低，收敛速度越慢，但信号越平滑。范围：0.0至1.0
~zeta	double	0.0	陀螺漂移增益（约rad/s）。范围：-1.0至1.0
~mag_bias_x	double	0.0	磁强计偏压（硬铁校正），x分量。范围：-10.0至10.0
~mag_bias_y	double	0.0	磁强计偏压（硬铁校正），y分量。范围：-10.0至10.0
~mag_bias_z	double	0.0	磁强计偏压（硬铁校正），z分量。范围：-10.0至10.0
~orientation_stddev	double	0.0	方向估计的标准偏差。范围：0.0至1.0
~world_frame	string	"nwu"	指示方向的世界框架（见REP-145）。 旧的默认值为"nwu"（西北向上）。 新部署应使用"enu"。有效值："nwu"、 "enu"、"ned"。
~use_mag	bool	true	在数据融合中是否使用磁场数据。
~use_magnetic_field_msg	bool	false	如果设置为true， 则订阅/imu、/mag话题作为 sensor_msgs/MagneticField； 如果设置为false（不推荐使用）， 则使用geometry_msgs/Vector3Stamped
~fixed_frame	string	odom	要在发布中使用的父坐标系
~publish_tf	bool	false	是否发布表示IMU方向的TF变换，作为IMU的位姿； 使用固定坐标系作为父坐标系， 输入的imu信息作为子坐标系
~reverse_tf	bool	false	如果设置为true，则发布从imu坐标系到固定坐标系的变换，而不是相反的方式。
~constant_dt	double	0.0	要使用的dt；如果为0.0（默认值），则从消息起始位置计算dt动态值。
~publish_debug_topics	bool	false	如果设置为true，则发布两个调试话题。
~stateless	bool	false	如果设置为true，则不发布过滤后的方向。 相反，仅根据最新的加速计（以及可选的磁强计）读数发布方位的无状态估计。用于调试。
~remove_gravity_vector	bool	false	如果设置为true，则从发布的IMU消息中的加速度场中减去重力矢量。

## 4.1.4、robot\_localization

### 1)、简介

`robot_localization` 是状态估计节点的集合，每个节点都是非线性状态估计器的一种实现，用于在3D空间中移动的机器人。它包括两个状态估计节点 `ekf_localization_node` 和 `ukf_localization_node`。另外，`robot_localization` 提供 `navsat_transform_node`，它有助于集成GPS数据。

`ekf_localization_node` 是[扩展卡尔曼滤波器](#)的实现。它使用全向运动模型来及时预测状态，并使用感知到的传感器数据校正预测的估计值。

`ukf_localization_node` 是[无迹卡尔曼过滤器](#)的实现。它使用一组精心选择的sigma点，以通过EKF中使用的相同运动模型来投影状态，然后使用这些投影的sigma点来恢复状态估计和协方差。这消除了雅可比矩阵的使用，并使滤波器更稳定。但是，与 `ekf_localization_node` 相比，它在计算上也更加繁重。

### 2)、话题

/	话题名	类型	解析
Subscribed	/imu/data	sensor_msgs/Imu	过滤后的imu信息
Subscribed	/odom_raw	nav_msgs/Odometry	里程计信息
Published	/odom	nav_msgs/Odometry	融合的里程计信息
Published	/tf	tf2_msgs/TFMessage	坐标系信息

### 3)、参数

- frequency: 滤波器产生状态估计值的真实频率（单位为Hz）。**注意：滤波器只有从输入之一接收到至少一条消息后才会开始计算。**
- [sensor]: 对于每个传感器，用户需要根据消息类型定义此参数。每个参数名称的索引都是从0开始的（例如odom0, odom1等），并且必须按顺序定义（例如，如果尚未定义pose1，则不要使用pose0和pose2）。每个参数的值是该传感器的话题名称。

```
odom0: /odom_raw
imu0: /imu/data
```

- [sensor]\_differential: 对于上面定义的每个包含位姿信息的传感器消息，用户可以指定是否应差分集成位姿变量。如果给定值设置为true，则对于从相关传感器在时间t进行的测量，我们首先将减去在时间t-1处的测量值，然后将所得值转换为速度。

```
~odomN_differential
~imuN_differential
~poseN_differential
```

- [sensor]\_relative: 如果将此参数设置为true，则来自该传感器的任何测量值都将相对于从该传感器接收到的第一个测量值进行融合。这在以下情况下很有用，例如，如果您希望状态估计值始终从(0,0,0)开始，并且侧倾，俯仰和偏航角值为(0,0,0)。

```
~odomN_relative
~imuN_relative
~poseN_relative
```

- `two_d_mode`: 如果您的机器人在平面环境中运行，并且可以忽略地面的细微变化（如IMU所报告），则将其设置为true。它将所有3D变量（Z，侧倾，俯仰以及它们各自的速度和加速度）融合成值0。这样可以确保这些值的协方差不会爆炸，同时确保您的机器人的状态估算值仍固定在X-Y平面上。
- `odom0_config`: [ false, false, false, false, false, false, true, true, false, false, false, true, false, false, false ]

布尔值的顺序为: [ 【X】 , 【Y】 , 【Z】 , 【roll】 , 【pitch】 , 【yaw】 , 【X'】 , 【Y'】 , 【Z'】 , 【roll'】 , 【pitch'】 , 【yaw'】 , 【X''】 , 【Y''】 , 【Z''】 ]。用户必须指定应将这些消息的哪些变量融合到最终状态估计中。

#### 4)、发布的变换

如果用户的 `world_frame` 参数设置为 `odom_frame` 的值，则将转换从 `odom_frame` 参数给出的坐标系发布到 `base_link_frame` 参数给出的坐标系。如果用户的 `world_frame` 参数设置为 `map_frame` 的值，则将转换从 `map_frame` 参数给出的坐标系发布到 `odom_frame` 参数给出的坐标系。

例如，我们设置将转换从【odom\_frame】参数给出的坐标系发布到【base\_link\_frame】参数给出的坐标系。

```
odom_frame: odom
base_link_frame: base_footprint
world_frame: odom
```

