

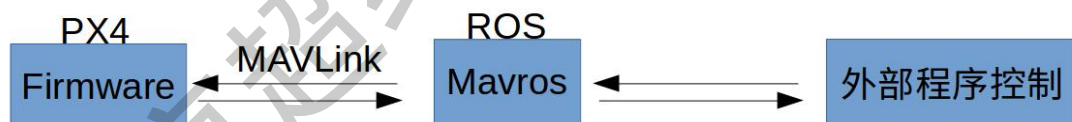
MAVROS 与机载电脑 Jetson nano 通信

简介：MAVROS 和 Jetson nano 通信，设置飞控参数和 MAVROS 文件，实现飞控和机载电脑的通信。MAVROS 相当于 PX4 飞控中的 MAVLINK 模块。MAVROS 将话题转为 MAVLINK 格式消息发送给飞控，飞控中的 MAVLINK 模块将 MAVLINK 格式消息转化为 uORB 消息给 PX4 各模块间传递消息使用，也可以反过来从 uORB 消息一直到 ROS 话题。

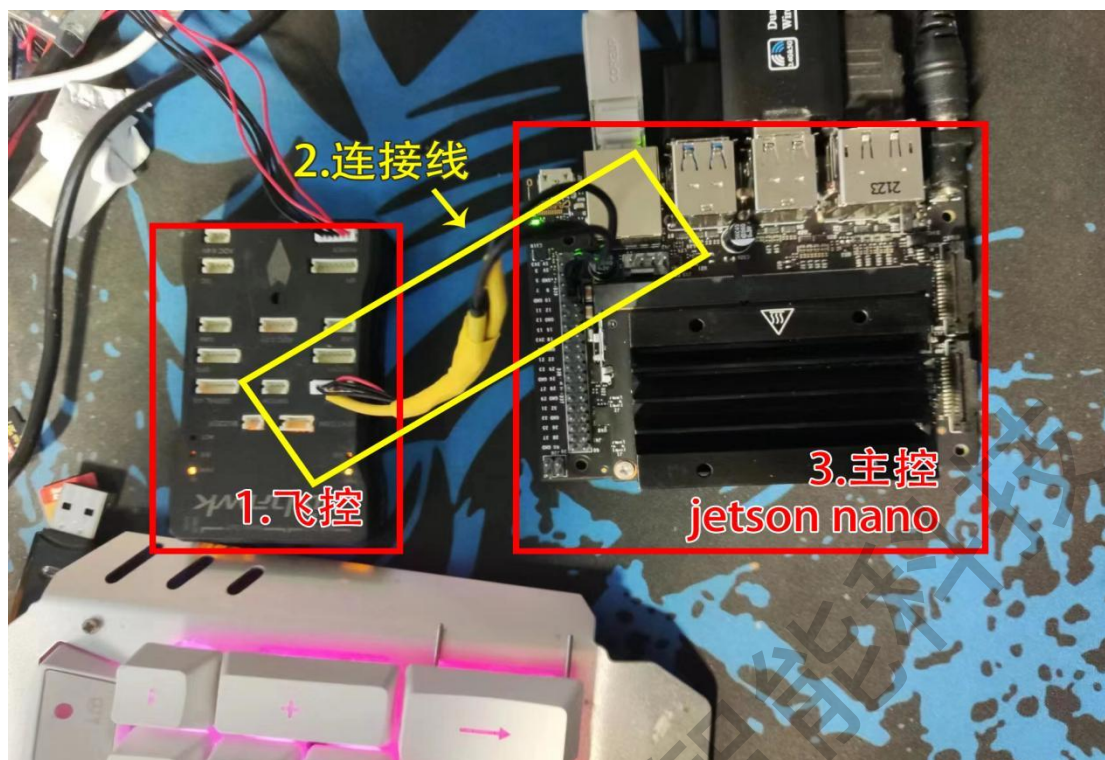
MAVROS 简介

ROS 结合 PX4 的或者 APM 的开发方式,关键在于实现 ROS 和他们之间的通信,这里我们使用 ROS 官方提供的 MAVROS 功能包实现飞控 Pixhawk 和 ROS 主控间的通信。

简单的来说，mavros 的作用是实现 ROS 和 PX4 间的通信，mavros 功能包起到了实现上层 ROS 和底层 PX4 固件的双向通信。基于此，在一般的开发应用中，我们通常在 ROS 层面进行功能开发，并发布相关控制指令给底层即可。同时底层会实时的把无人机的相关状态通过 mavros 上传给 ROS。



我们的无人机 ROS 主控 Jetson nano 通过串口 ttyTHS1 和飞控的 telem2 口进行通信。需要的硬件包括安装了系统镜像的 Jetson nano 和下载了固件的飞控 pixhawk2.4.8，另外需要一根连接线。如图中所示标注所示：



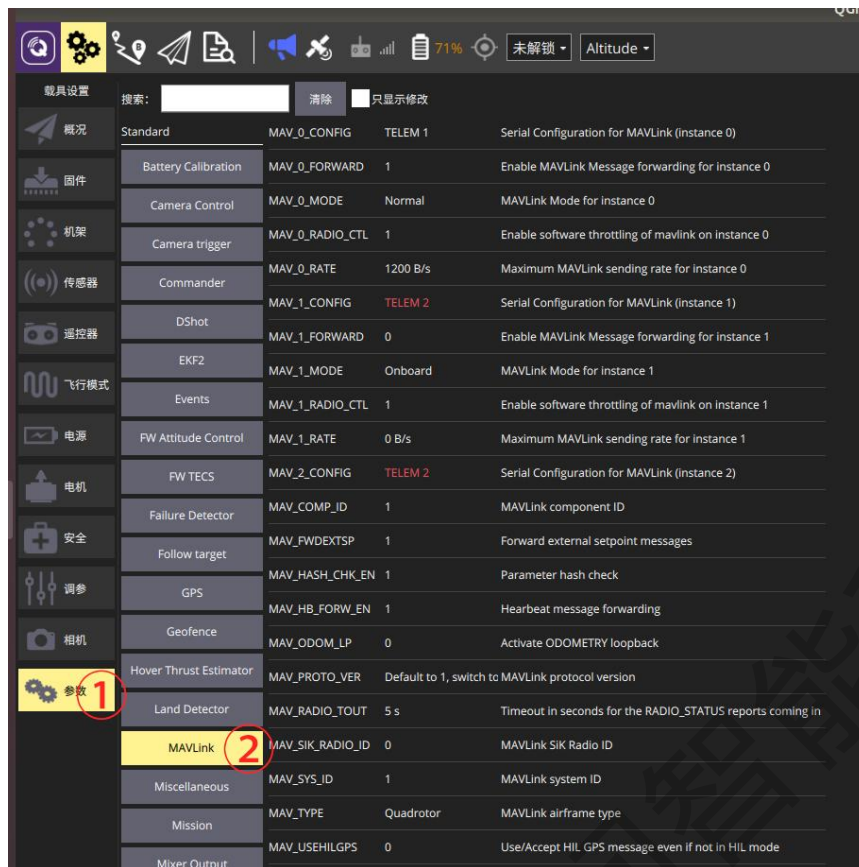
Jetson nano 引脚按序号排列如下：上述图片则是通过 telem2 连接 6，8，10 一脚。其中 telem2 中的 GND 连接 6-GND，telem2 中的 TX 连接 10-RX，RX 连接 8-TX。Telem2 中只需要此三根线连接即可，其余不用管。

J41引脚列表

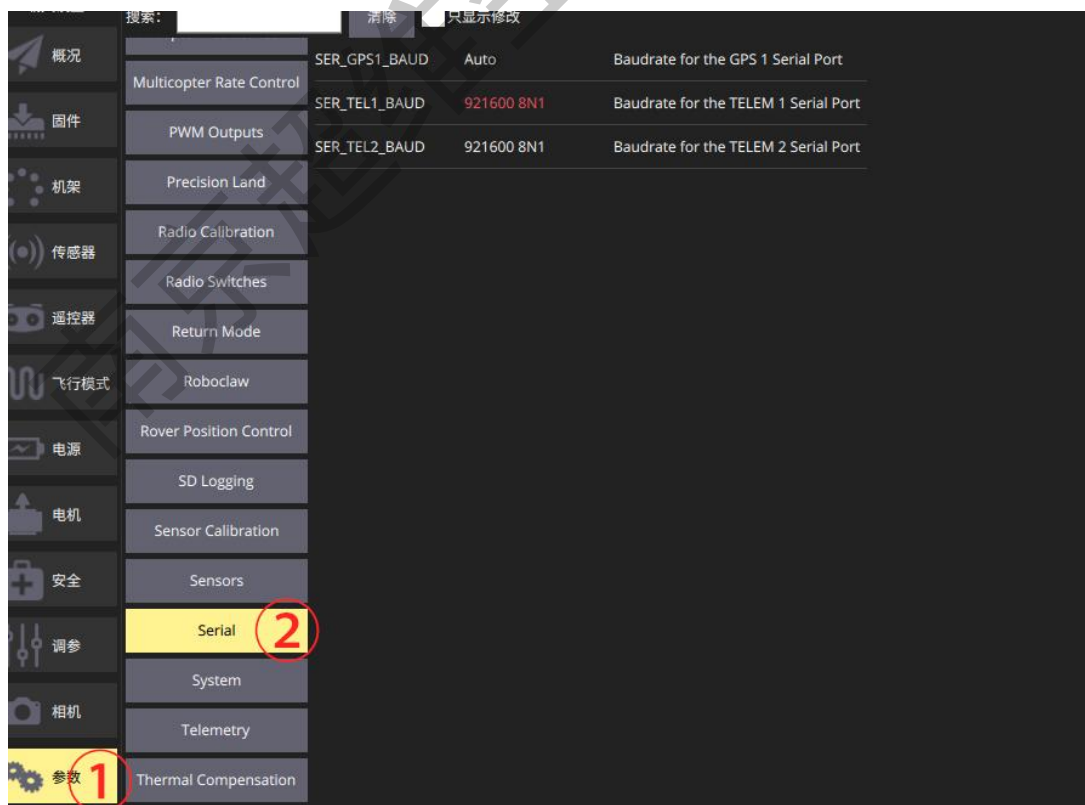
GPIO (BCM)	名称	引脚	引脚	名称	GPIO (BCM)
	3.3V输出	1	2	5.0V 输出	
	I2C_2_SDA I2C Bus 1	3	4	5.0V 输出	
	I2C_2_SCL I2C Bus 1	5	6	GND	
GPIO216 (D4)	AUDIO_MCLK	7	8	UART_TX2 /dev/ttyTHS1	
	GND	9	10	UART_RX2 /dev/ttyTHS1	
GPIO50 (D17)	UART_2_RTS	11	12	I2S_4_SCLK	GPIO79 (D18)
GPIO232 (D23)	SPL_2_SCK	13	14	GND	

打开 QGC 地面站，设置如下：

- ①、选中 MAVlink 设置参数如下，设置 telem2 通信



②、紧接着设置 Serial1，波特率设置为 921600。



③、找到启动 mavros 的 px4.launch 文件，设置通信的串口为 /dev/ttyTHS1:921600 即可



```
<?xml version='1.0' encoding='utf-8'>
<launch>
  <!-- vim: set ft=xml noet : -->
  <!-- example launch script for PX4 based FCU's -->

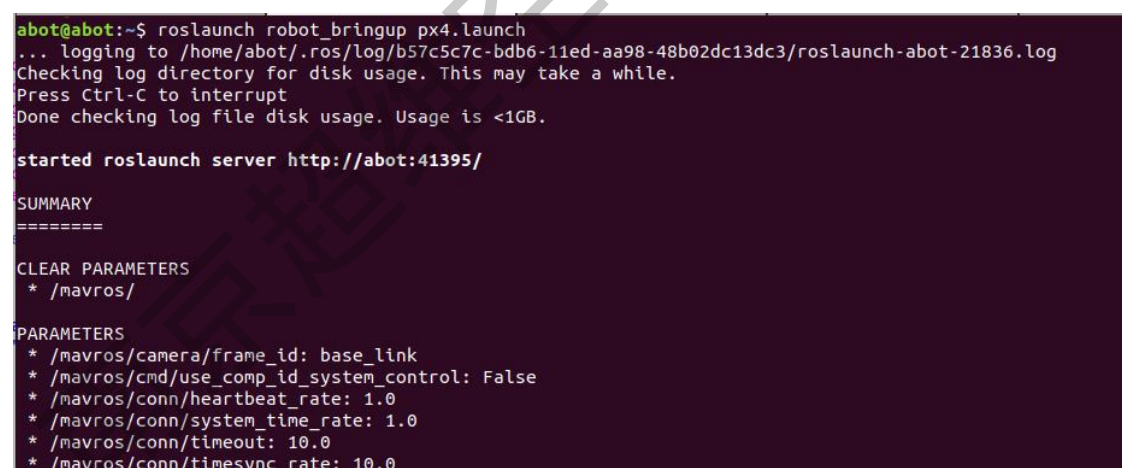
  <arg name="fcu_url" default="/dev/ttyTHS1:921600" />
  <arg name="gcs_url" default="" />
  <arg name="tgt_system" default="1" />
  <arg name="tgt_component" default="1" />
  <arg name="log_output" default="screen" />
  <arg name="fcu_protocol" default="v2.0" />
  <arg name="respawn_mavros" default="false" />

  <include file="$(find mavros)/launch/node.launch">
    <arg name="pluginlists_yaml" value="$(find mavros)/launch/px4_pluginlists.yaml" />

    <arg name="config_yaml" value="$(find mavros)/launch/px4_config.yaml" />
    <!--arg name="config_yaml" value="$(find robot_bringup)/launch/
px4_config_noTF.yaml" /-->

    <arg name="fcu_url" value="$(arg fcu_url)" />
    <arg name="gcs_url" value="$(arg gcs_url)" />
    <arg name="tgt_system" value="$(arg tgt_system)" />
    <arg name="tgt_component" value="$(arg tgt_component)" />
    <arg name="log_output" value="$(arg log_output)" />
    <arg name="fcu_protocol" value="$(arg fcu_protocol)" />
    <arg name="respawn_mavros" default="$(arg respawn_mavros)" />
  </include>
</launch>
```

④、确保硬件和参数设置完成后，启动 mavros



```
abot@abot:~$ roslaunch robot_bringup px4.launch
... logging to /home/abot/.ros/log/b57c5c7c-bdb6-11ed-aa98-48b02dc13dc3/roslaunch-abot-21836.log
Checking log directory for disk usage. This may take a while.
Press Ctrl-C to interrupt
Done checking log file disk usage. Usage is <1GB.

started roslaunch server http://abot:41395/

SUMMARY
=====

CLEAR PARAMETERS
* /mavros/

PARAMETERS
* /mavros/camera/frame_id: base_link
* /mavros/cmd/use_comp_id_system_control: False
* /mavros/conn/heartbeat_rate: 1.0
* /mavros/conn/system_time_rate: 1.0
* /mavros/conn/timeout: 10.0
* /mavros/conn/timesync_rate: 10.0
```

⑤、出现以下内容表示 MAVROS 实现双向通信


```

[INFO] [1678282799.450313344]: Known MAVLink dialects: common ardupilotmega ASLUAV AVSSUAS all de
xpilot paparazzi standard storm32 uAvionix ualberta
[INFO] [1678282799.450376991]: MAVROS started. MY ID 1.240, TARGET ID 1.1
[INFO] [1678282799.450717258]: IMU: High resolution IMU detected!
[INFO] [1678282799.451242267]: IMU: Attitude quaternion IMU detected!
[INFO] [1678282799.454552746]: CON: Got HEARTBEAT, connected. FCU: PX4 Autopilot
[INFO] [1678282799.460450147]: IMU: High resolution IMU detected!
[INFO] [1678282799.460647338]: IMU: Attitude quaternion IMU detected!
[INFO] [1678282800.469318432]: GF: Using MISSION_ITEM_INT
[INFO] [1678282800.469544530]: RP: Using MISSION_ITEM_INT
[INFO] [1678282800.469771513]: WP: Using MISSION_ITEM_INT
[INFO] [1678282800.469958444]: VER: 1.1: Capabilities          0x0000000000000e4ff
[INFO] [1678282800.470124853]: VER: 1.1: Flight software:      010d0200 (46a12a09bf000000)
[INFO] [1678282800.470340066]: VER: 1.1: Middleware software: 010d0200 (46a12a09bf000000)
[INFO] [1678282800.470576893]: VER: 1.1: OS software:         0b0000ff (91bece51afbe7da9)
[INFO] [1678282800.470750438]: VER: 1.1: Board hardware:      00000011
[INFO] [1678282800.470913566]: VER: 1.1: VID/PID:             26ac:0011
[INFO] [1678282800.471167164]: VER: 1.1: UID:                 3037511631363732
[WARN] [1678282800.471612746]: CMD: Unexpected command 520, result 0
[INFO] [1678282809.456810653]: HP: requesting home position
[INFO] [1678282814.458571214]: GF: mission received
[INFO] [1678282814.460270725]: RP: mission received
[INFO] [1678282814.460542709]: WP: mission received
[INFO] [1678282819.456776684]: HP: requesting home position
[INFO] [1678282829.456781947]: HP: requesting home position

```

总结：MAVROS 在 ROS+PX4/APM 的开发方式中属于重点中的重点，一定要深刻理解 MAVROS 的功能和作用，在此基础上进行开发，可以极大的提高开发效率。基于此，我们推荐采用这种开发方式的朋友静下心来研究一番 MAVROS，可以参考网上相关博客资料等，也可以直接查看官网介绍。推荐 MAVROS 官网地址如下：

<http://wiki.ros.org/mavros>

步骤整理如下：

一、配置 Pixhawk 上的 Telem2 作为 MAVLINK 端口

MAV_1_CONFIG = TELEM 2

MAV_1_MODE = Onboard

SER_TEL2_BAUD = 921600 8N1

注：一开始参数里可能只有 MAV_1_CONFIG，搜不到其他的参数，只需要先把 MAV_1_CONFIG 设置为 TELEM 2，然后把飞控重启后就有了。

二、在机载电脑上启动 MAVROS

这里用的是 Jetson nano 的串口 2，也就是 `dev/ttyTHS1`。最后的 921600 是波特率，就是 1.1 中设置的 `SER_TEL2_BAUD`，改成设置的值就行了。

```
roslaunch mavros px4.launch fcu_url:=serial:///dev/ttyTHS1:921600  
gcs_url:=udp://@192.168.0.0
```

`gcs_url` 为运行 QGC 的主机的 IP

三、设置为以下参数表示自动寻址，直到连上 QGC

```
roslaunch mavros px4.launch fcu_url:=serial:///dev/ttyTHS1:921600  
gcs_url:=udp-b://@
```

可能会报错

```
FCU: DeviceError:serial:open: Permission denied
```

四、解决方法是给对应的串口权限

```
sudo chmod 777 /dev/ttyTHS1
```

五、添加永久生效的权限

每次要在后面加一堆参数很烦，所以直接修改 `launch` 文件，使用 `sudo` 权限打开 `/opt/ros/melodic/share/mavros/launch/px4.launch`，将

```
<arg name="fcu_url" default="/dev/ttyACM0:57600" />
```

修改为自己的端口和波特率

```
<arg name="fcu_url" default="/dev/ttyTHS1:921600" />
```

南京超维空间智能科技有限公司