

# 1. 实验名称及目的

## 1.1 实验名称

PX4与ROS2直连环境配置与基本使用教程

## 1.2 实验目的

- 掌握PX4与ROS2的基本通信配置方法。
- 学会使用MicroXRCEAgent实现UDP通信。
- 熟悉ROS2话题的发布与订阅操作。
- 能够通过ROS2控制PX4 SITL仿真飞机。

## 1.3 关键知识点

- **PX4与ROS2通信**：通过MicroXRCEAgent实现UDP消息传递。
- **ROS2话题操作**：发布、订阅和监听ROS2话题。
- **环境配置**：配置WSL、PX4固件和ROS2环境。

# 2. 实验效果

通过终端输出和ROS2话题监听验证通信效果。

# 3. 文件目录

例程目录：[\[安装目录\]](#)\RflySimAPIs\2.RflySimUsage\0.ApiExps\e12\_PX4ROS2

文件夹/文件名称	说明
<code>InstallPX4Msg.sh</code>	安装PX4消息到ROS2的脚本
<code>InstallXRCE-DDS.sh</code>	安装MicroXRCEAgent的脚本

## 4. 实验环境

## 5. 实验步骤

### 5.1 安装PX4 ROS2(选做)

下面步骤RflySim工具链已完成部署，只有真机或虚拟机实验时需要使用

运行\RflySimAPIs\2.RflySimUsage\0.ApiExps\e12\_PX4ROS2\InstallXRCE-DDS.sh 来安装MicroXRCEAgent

先运行~/ros\_switch.sh 2 来切换到ROS2

运行\RflySimAPIs\2.RflySimUsage\0.ApiExps\e12\_PX4ROS2\InstallPX4Msg.sh 来安装px4\_msgs到ROS2

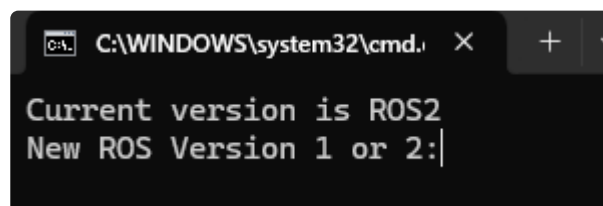
本脚本会将px4\_msgs和px4\_ros\_com安装到/opt/ros/humble/overlays/px4\_msgs/目录

在bashrc中添加 `source /opt/ros/humble/overlays/px4_msgs/setup.bash` 来注册消息

详细原理见：[https://docs.px4.io/main/en/ros2/user\\_guide#install-px4](https://docs.px4.io/main/en/ros2/user_guide#install-px4)

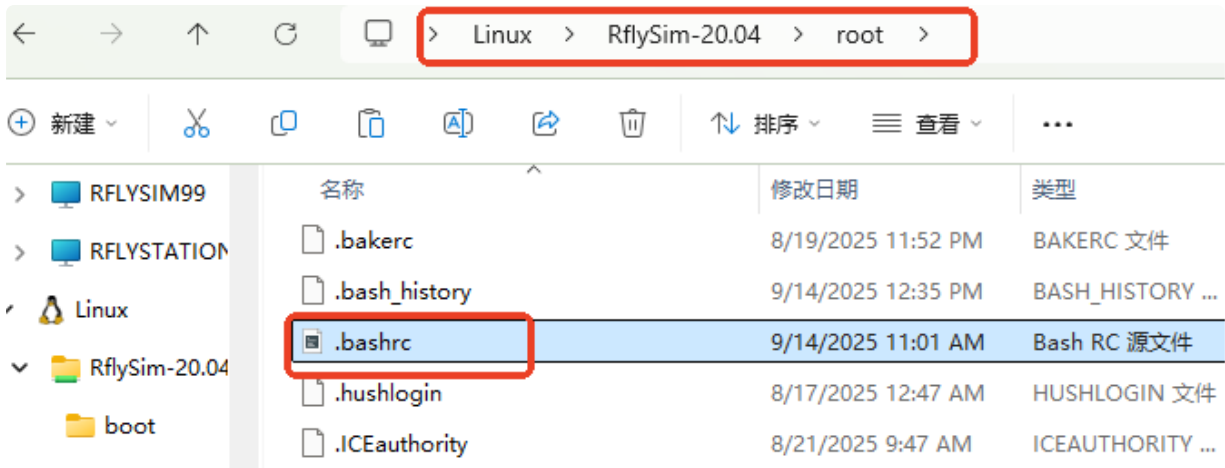
### 5.2 步骤1：配置ROS2环境

1. 在\Desktop\RflyTools中找到双击运行“RosSwitch.bat”，切换到ROS2



```
C:\WINDOWS\system32\cmd. x + v
Current version is ROS2
New ROS Version 1 or 2:|
```

2. 修改 `\\wsl.localhost\RflySim-20.04\root\.\bashrc` :

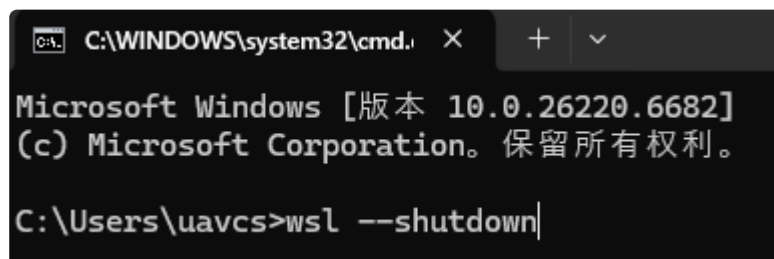


- 将 `export ROS_LOCALHOST_ONLY=1` 改为 `export ROS_LOCALHOST_ONLY=0`，启用广播支持
  - 设置 `export RMW_IMPLEMENTATION=rmw_fastrtps_cpp`，启用fast-dds，来和MicroXRCEAgent匹配。注意注释掉cyclonedds的内容。

```
163     # # 使用cyclonedds
164     # export ROS_LOCALHOST_ONLY=1 # 防止局域网干扰
165     # export RMW_IMPLEMENTATION=rmw_cyclonedds_cpp
166
167     # 使用默认fastrtps
168     export ROS_LOCALHOST_ONLY=0 # 开启局域网访问
169     export RMW_IMPLEMENTATION=rmw_fastrtps_cpp
170
171     source /opt/ros/humble/setup.bash
172
```

MicroXRCEAgent似乎必须这样才能发送消息。ROS\_LOCALHOST\_ONLY=1是为了防止局域网内电脑相互干扰，在真机情况下ROS\_LOCALHOST\_ONLY=0是更可靠的方法。

2. 打开一个CMD窗口，输入 `wsl --shutdown` 关闭所有WSL，确保下次启动生效



## 5.3 步骤2：启动软件在环并启动Micro-XRCE-DDS-Agent

1. 然后从 `\Desktop\RflyTools` 打开SITL\_Run输入1，创建1个飞机，确保平台PX4版本为 1.14
2. 打开一个WinWSL，输入 `MicroXRCEAgent udp4 -p 8888` 来创建连接

```
root@DaiPC: /mnt/c/PX4PSP/ x + v
root@DaiPC:/mnt/c/PX4PSP/Firmware# MicroXRCEAgent udp4 -p 8888
[1756395000.484286] info | UDPv4AgentLinux.cpp | init | running... | port: 8888
[1756395000.484492] info | Root.cpp | set_verbose_level | logger setup | verbose_level: 4
[1756395001.091985] info | Root.cpp | create_client | create | client_key: 0x00
000001, session_id: 0x81
[1756395001.092541] info | SessionManager.hpp | establish_session | session established | client_key: 0x00
000001, address: 127.0.0.1:35766
[1756395001.200972] info | ProxyClient.cpp | create_participant | participant created | client_key: 0x00
000001, participant_id: 0x001(1)
[1756395001.201634] info | ProxyClient.cpp | create_topic | topic created | client_key: 0x00
000001, topic_id: 0x3E8(2), participant_id: 0x001(1)
[1756395001.201697] info | ProxyClient.cpp | create_subscriber | subscriber created | client_key: 0x00
000001, subscriber_id: 0x3E8(4), participant_id: 0x001(1)
[1756395001.202435] info | ProxyClient.cpp | create_datareader | datareader created | client_key: 0x00
000001, datareader_id: 0x3E8(6), subscriber_id: 0x3E8(4)
[1756395001.202763] info | ProxyClient.cpp | create_topic | topic created | client_key: 0x00
000001, topic_id: 0x3E9(2), participant_id: 0x001(1)
[1756395001.202796] info | ProxyClient.cpp | create_subscriber | subscriber created | client_key: 0x00
```

## 5.4 步骤3：ROS2订阅PX4消息

新打开一个WinWSL，输入 `ros2 topic list`，能看到fmu打头的消息

```
root@DaiPC: /mnt/c/PX4PSP/Firmware# ros2 topic list
/fmu/in/obstacle_distance
/fmu/in/offboard_control_mode
/fmu/in/onboard_computer_status
/fmu/in/sensor_optical_flow
/fmu/in/telemetry_status
/fmu/in/trajectory_setpoint
/fmu/in/vehicle_attitude_setpoint
/fmu/in/vehicle_command
/fmu/in/vehicle_mocap_odometry
/fmu/in/vehicle_rates_setpoint
/fmu/in/vehicle_trajectory_bezier
/fmu/in/vehicle_trajectory_waypoint
/fmu/in/vehicle_visual_odometry
/fmu/out/failsafe_flags
/fmu/out/position_setpoint_triplet
/fmu/out/sensor_combined
/fmu/out/timesync_status
/fmu/out/vehicle_attitude
/fmu/out/vehicle_control_mode
/fmu/out/vehicle_global_position
/fmu/out/vehicle_gps_position
/fmu/out/vehicle_local_position
/fmu/out/vehicle_odometry
/fmu/out/vehicle_status
/parameter_events
/rosout
root@DaiPC: /mnt/c/PX4PSP/Firmware# |
```

6. 继续输入 `ros2 topic echo /fmu/out/vehicle_local_position`，可以看到

```
hagl_min: .inf
hagl_max: .inf
---
timestamp: 1756395728872012
timestamp_sample: 1756395728856012
xy_valid: true
z_valid: true
v_xy_valid: true
v_z_valid: true
x: -0.06013060361146927
y: 0.03828397020697594
z: -0.01609179936349392
delta_xy:
- -0.022562652826309204
- 0.07645218819379807
xy_reset_counter: 2
delta_z: 0.02296033874154091
z_reset_counter: 2
vx: -0.0001836838200688362
vy: 0.0026768590323626995
vz: -0.005816010758280754
z_deriv: -0.009399570524692535
delta_vxy:
- -0.0007439343607984483
- 0.0005394339677877724
vxy_reset_counter: 2
delta_vz: -0.0031241104006767273
```

7. CTRL-C结束上述程序，再运行如下指

令 `ros2 launch px4_ros_com sensor_combined_listener.launch.py`，可以看到消息如下：

```
sensor_combined_listener-1]
sensor_combined_listener-1]
sensor_combined_listener-1] RECEIVED SENSOR COMBINED DATA
sensor_combined_listener-1] =====
sensor_combined_listener-1] ts: 1756395880640113
sensor_combined_listener-1] gyro_rad[0]: -0.000532632
sensor_combined_listener-1] gyro_rad[1]: 0.000266316
sensor_combined_listener-1] gyro_rad[2]: -0.000266316
sensor_combined_listener-1] gyro_integral_dt: 4000
sensor_combined_listener-1] accelerometer_timestamp_relative: 0
sensor_combined_listener-1] accelerometer_m_s2[0]: 0.0107739
sensor_combined_listener-1] accelerometer_m_s2[1]: 1.71096e-07
sensor_combined_listener-1] accelerometer_m_s2[2]: -9.80306
sensor_combined_listener-1] accelerometer_integral_dt: 4000
```

8. CTRL-C结束上述程序，再运行`ros2 run px4_ros_com offboard_control`，可以控制飞机进入offboard并飞行。

## 7. 常见问题

1. 问题：运行 `MicroXRCEAgent` 时提示端口占用。

- 原因：可能有其他程序占用了8888端口。
- 解决方案：使用 `netstat -ano | findstr 8888` 查找占用程序并结束。

2. 问题： `ros2 topic list` 未显示任何话题。

- 原因：MicroXRCEAgent未正确启动或PX4未连接。
- 解决方案：检查MicroXRCEAgent日志，确保PX4 SITL已运行。

3. 问题： `ros2 launch` 命令无法启动。

- 原因：缺少必要的ROS2包。
- 解决方案：运行 `rosdep install` 安装依赖。