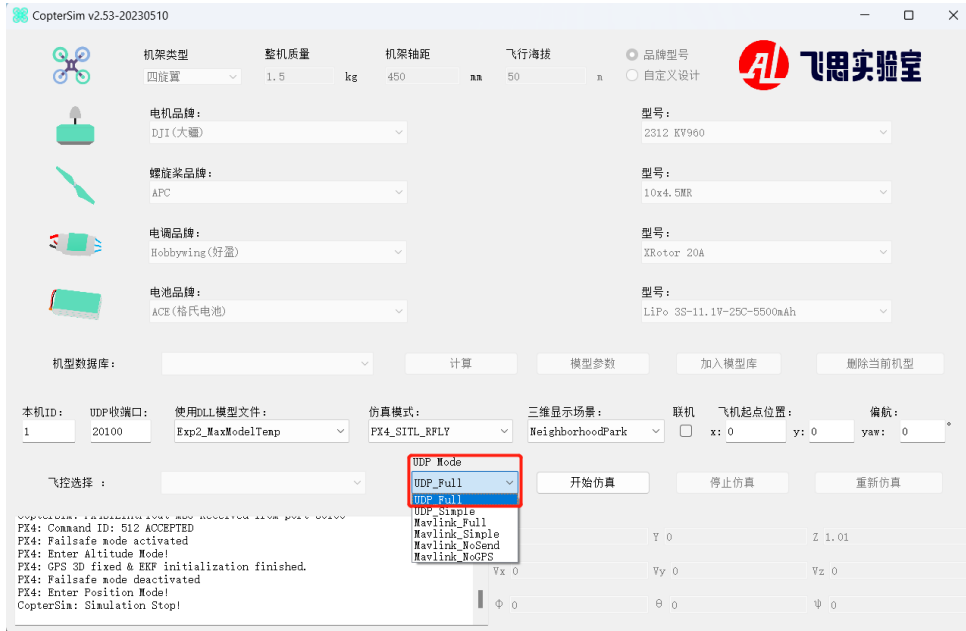


## 1. 实验名称及目的

**外部通信实验之读取状态估计值：**在使用 RflySim 平台以 UDP\_Full 模式进行软/硬件在环仿真时，可以通过监听 UDP20101 系列端口接收 PX4 内部状态估计值。



## 2. 实验原理

### 2.1. 软/硬件在环仿真（SIL/HIL）的实现[1][2]

从实现机制的角度分析，可将 RflySim 平台分为运动仿真模型、底层控制器、三维引擎、外部控制四部分。

- **运动仿真模型：**这是模拟飞行器运动的核心部分。在 RflySim 平台中，运动仿真模型是通过 MATLAB/Simulink 开发的，然后通过自动生成的 C++代码转化成 DLL（动态链接库）文件。在使用 RflySim 平台进行软硬件在环仿真时，会将 DLL 模型导入到 CopterSim，形成运动仿真模型。这个模型在仿真中负责生成飞行器的运动响应，它拥有多个输入输出接口与底层控制器、三维引擎、地面控制站和外部控制进行数据交互，具体数据链路、通信协议及通信端口号见 [API.pdf 中的通信接口部分](#)。
- **底层控制器：**在软/硬件在环仿真（SIL/HIL）中，真实的飞行控制硬件（如 PX4 飞行控制器）被集成到一个虚拟的飞行环境中。在软件在环仿真（SIL）中，底层控制器（通过 wsl 上的 PX4 仿真环境运行）通过网络通信与运动仿真模型交互数据。在硬件在环仿真（HIL）中，它（将 PX4 固件在真实的飞行控制器（即飞控）硬件上运行）则通过串口通信与运动仿真模型进行数据交互。底层控制器是实际控制飞行器硬件（如电机和传感器）的部分。
- **三维引擎：**这部分负责生成和处理仿真的视觉效果，提供仿真环境的三维视图，使

用户能够视觉上跟踪和分析飞行器的运动。

- 外部控制 (offboard): 从仿真系统外部对飞行器进行的控制, 包括自动飞行路径规划、远程控制指令等。在平台例程中主要通过地面控制站 (QGC)、MATLAB 和 Python 调用对应接口实现。

在本例程中, 主要验证将飞机位姿仿真数据 (通过运动仿真模型解算出的位姿数据) 通过 UDP 20101++2 端口发送到三维引擎 RflySim3D, 以及验证在仿真过程中利用 PX4ExtMsgSender.slx 模块通过监听 UDP 20101++2 端口接收飞控状态数据 (PX4 内部状态估计值)。即飞机位姿仿真数据来自顶层模型, 而飞控状态数据来自底层控制器。

## 2.2. UDP 通信模式

UDP (用户数据报协议) 是一种无连接的网络协议, 允许快速的数据传输, 但不保证数据包的顺序或完整性。在仿真中, UDP 通信通常用于实时数据传输, 如飞行器的状态信息 (这里包括顶层模型解算出的位姿信息 (真值) 和底层控制器给出的状态估计值)。

RflySim 平台的不同通信模式主要分为如下几种 (可在 CopterSim 的 UI 中进行配置)。

<b>UDP_Full:</b> Python 传输完整的 UDP 数据给 CopterSim, 传输数据量小; CopterSim 收到数据后, 再转换为 Mavlink 后传输给 PX4 飞控; 适合中小规模集群 (数量小于 10) 仿真。
<b>UDP_Simple:</b> 数据包大小与发送频率比 UDP_Full 模式小; 适合大规模集群仿真, 无人机数量小于 100。
<b>Mavlink_Full:</b> Python 直接发送 MAVLink 消息给 CopterSim, 再转发给 PX4, 数据量较大适合单机控制; 适合单机或少量飞机仿真, 无人机数量小于 4;
<b>Mavlink_Simple:</b> 屏蔽部分 MAVLink 消息包, 并降低数据频率, 发送数据量比 MAVLink_Full 小很多, 适合多机集群控制; 适合小规模集群仿真, 无人机数量小于 8。
<b>Mavlink_NoSend:</b> 模式下 CopterSim 不会向外发送 MAVLink 数据, 此模式需要配合硬件在环仿真+数传串口通信, 通过有线方式传输 MAVLink, 此模式局域网内数据量最小, 适合分布式视觉硬件在环仿真, 无人机数量不限制
<b>Mavlink_NoGPS:</b> 模式下 CopterSim 不会向外发送 MAVLink 数据和 GPS 数据。

## 2.3. 读取 PX4 状态估计值

PX4 是一种流行的开源飞行控制平台, 它能提供实时的飞行器状态数据, 如位置、速度、姿态等, 因为这些值是经过滤波平滑后的结果, 不是直接由传感器输出的值, 所以称为状态估计值。(注意, 由于飞控提供的位姿数据是实际飞行环境中状态数据的唯一来源, 所以有时也把这个状态估计值称为真值)

在 RflySim 平台上, 这些数据通过特定的 UDP 端口 (本例程是 UDP20101 系列) 传输。通过监听这些端口, 可以从仿真系统外部获取这些关键数据, 从而使得开发者能分析飞行器的性能和行为。

## 3. 实验效果

以 UDP\_Full 模式启动软/硬件在环仿真, 待仿真初始化完成 (GPS 3D fixed) 后, 监听 UDP20101 系列端口可以接收到 PX4 内部状态估计值。

## 4. 文件目录

文件夹/文件名称	说明
----------	----

MulticopterNoCtrl.dll	四旋翼动力运动学模型，软硬件在环仿真时调用该动态链接库并配合官方控制器实现仿真过程
MulticopterNoCtrlHITLRun.bat	四旋翼硬件在环仿真启动脚本
MulticopterNoCtrlSITLRun.bat	四旋翼软件在环仿真启动脚本
PX4ExtMsgSender.slx	外部通信接口程序

## 5. 运行环境

序号	软件要求	硬件要求	
		名称	数量
1	Windows 10 及以上版本	笔记本/台式电脑 <sup>①</sup>	1
2	RflySim 平台收费版	Pixhawk6C <sup>②</sup>	1
3	MATLAB 2017B 及以上 <sup>③</sup>	数据线	1

①：推荐配置请见：<https://doc.rflysim.com/1.1InstallMethod.html>

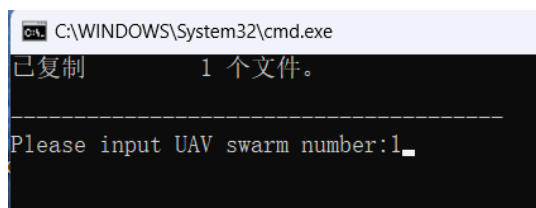
②：平台安装时的推荐编译命令为：px4\_fmu-v6c\_default，PX4 固件版本为：1.13.3。其他配套飞控请见：<http://doc.rflysim.com/hardware.html>。

## 6. 实验步骤

### 6.1. 软件在环

#### Step 1:

右键以管理员身份运行 MulticopterNoCtrlSITLRun.bat，在弹出的页面中输入 1，启动 1 架四旋翼飞机的软件在环仿真。



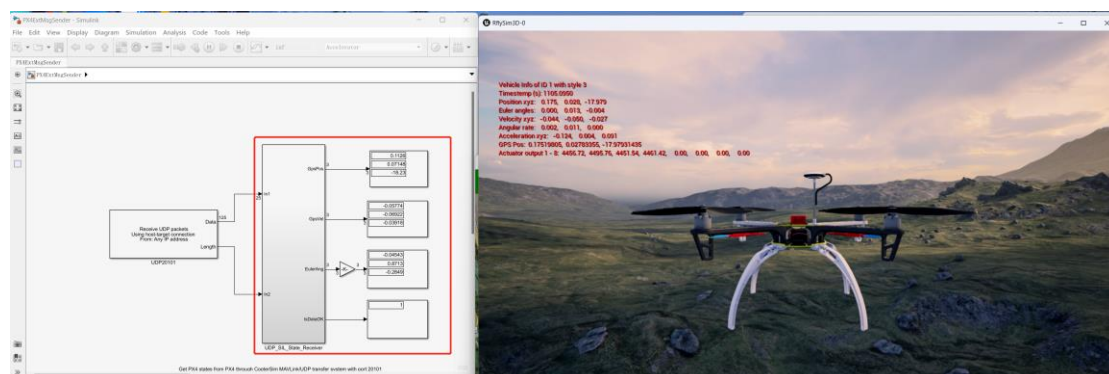
#### Step 2:

打开 PX4ExtMsgSender.slx，待仿真初始化完成后，运行 slx 文件。点击 RflySim3D 并按下 D，可在左上角查看相关数据。



## Step 4:

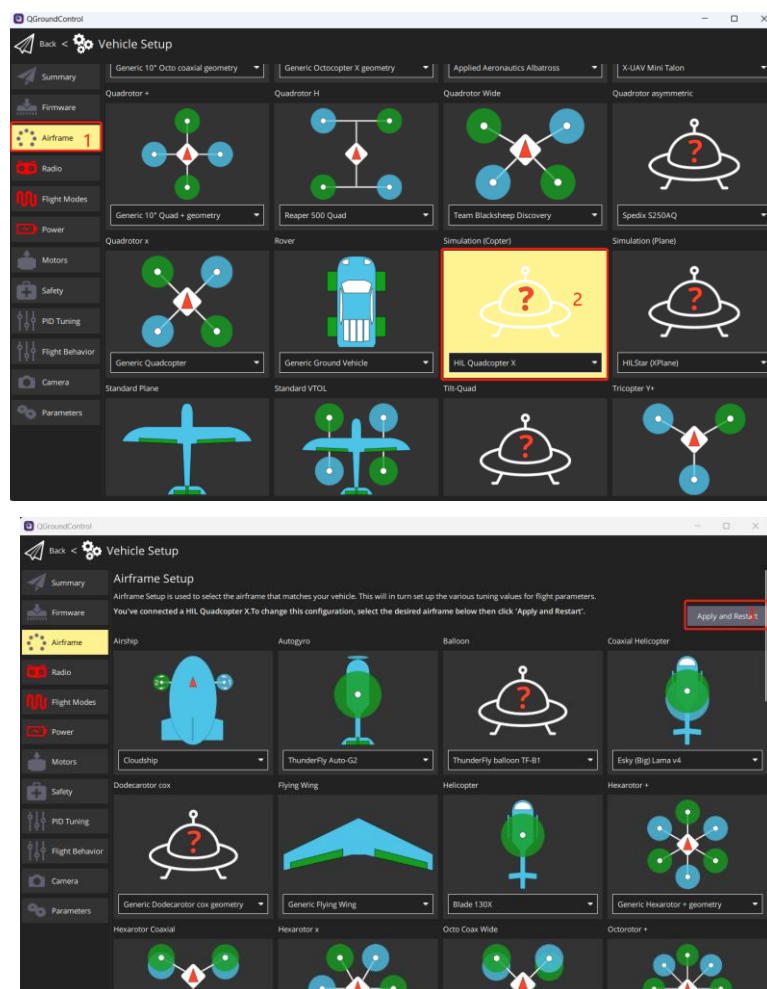
在 PX4ExtMsgSender.slx 的 UDP\_SIL\_State\_Receiver 模块 display 实时查看状态估计值变化。



## 6.2. 硬件在环

### Step 1:

点击 Airframe, 选择 HIL Quadcopter X 机架类型后, 点击应用并且重启。

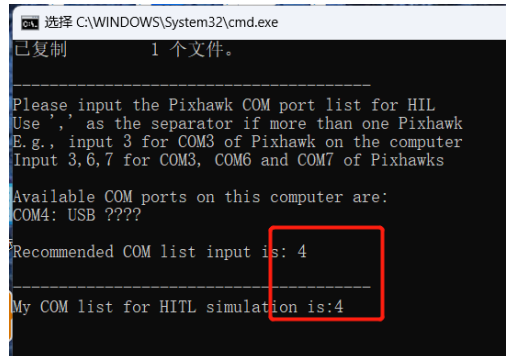




---

## Step 2:

右键以管理员身份运行 `MulticopterNoCtrlHITLRun.bat`，输入飞控所对应的端口号，这里输入 4，启动硬件在环仿真。



```
选择 C:\WINDOWS\System32\cmd.exe
已复制 1 个文件。

-----
Please input the Pixhawk COM port list for HITL
Use ',' as the separator if more than one Pixhawk
E.g., input 3 for COM3 of Pixhawk on the computer
Input 3,6,7 for COM3, COM6 and COM7 of Pixhawks

Available COM ports on this computer are:
COM4: USB ???

Recommended COM list input is: 4

-----
My COM list for HITL simulation is:4
```

## Step 3:

重复 6.1 的 Step2-Step4，效果与软件在环一致。

## 7. 参考资料

- [1]. DLL/SO 模型与通信接口 [..\..\API.pdf](#)
- [2]. 外部控制接口 [..\..\API.pdf](#)

## 8. 常见问题

Q1.

A1.