

1、实验名称及目的

BAT 脚本启动组件实验：熟悉 BAT 配置。

2、实验效果

BAT 脚本一键启动 QGroundControl、RflySim3D、CopterSim。

3、文件目录

文件夹/文件名称	说明
OneStartup.bat	一键启动脚本。

4、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 ^①	1
2	RflySim 平台免费版		

①：推荐配置请见：<https://doc.rflysim.com>

5、实验步骤

Step 1:

双击运行 “OneStartup.bat”，即可看到打开了 QGroundControl、RflySim3D、CopterSim 三款软件。

注：RflySim 免费体验版无法最多只能创建 4 个飞机，超过数量 CopterSim 会报错。

Step 2:

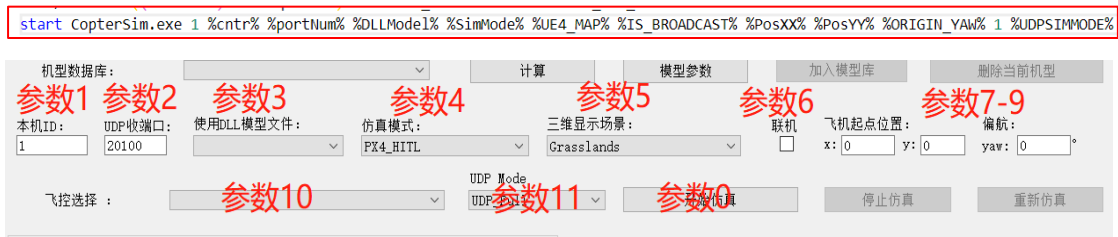
RflySimUE5 与 RflySim3D 的使用方法完全相同，区别在于需要修改 bat 脚本中字符串 “%PSP_PATH%\RflySim3D” 为 “%PSP_PATH%\RflySimUE5” 以启用 UE5 版本引擎。其他 bat 脚本修改相同。

```
38 REM Set the map, use index of
39 REM e.g., UE4 MAP=1 equals to
40 SET UE4_MAP=Quarry
41
```

```
113 REM UE4Path
114 tasklist|find /i "RflySim3D.exe" || start %PSP_PATH%\RflySimUE5\RflySim3D.exe
115 choice /t 5 /d y /n >nul
```

Step 3:

用文本编辑器打开“RflySimAPIs\SITLRun.bat”（对应桌面 SITLRun 快捷方式）



SET /a START_INDEX=1 #对应“参数1”，飞机起始ID

SET /a UDP_START_PORT=20100 #对应“参数2”，UDP起始接收端口号

SET DLLModel=0 #对应“参数3”，是否用DLL模型，及选项序号

SET SimMode=2 #对应“参数4”，仿真模式，SITL选2，HITL选0

SET UE4_MAP=Grasslands #对应“参数5”，三维场景地图，这里是Grassland

SET IS_BROADCAST=0 #对应“参数6”，是否局域网广播数据

SET /a ORIGIN_POS_X=0 #对应“参数7”，初始X位置，整数，单位m

SET /a ORIGIN_POS_Y=0 #对应“参数8”，初始Y位置，整数，单位m

SET /a ORIGIN_YAW=0 #对应“参数9”，初始yaw角度，整数单位度

SET /a VEHICLE_INTERVAL=2 #如果开启多机，两飞机的间隔m（整数），单位m

SET UDPSIMMODE=0 #对应“参数11”，UDP数据模式，选项序号

SET PX4SidlFrame=iris #软件在环时设置机架样式，这里是四旋翼

6、参考文献

[1].其他一键启动脚本简介:

HITLRun.bat

常规硬件在环仿真脚本，支持输入串口序列（英文逗号","分隔），来启动多机的硬件在环仿真

注：REM打头的行是注释语句，不会被执行。其他的bat脚本语法规则可自行搜索学习。

注：本脚本的飞机位置是由脚本按矩形队列自动生成的，控制变量包括：

SET /a START_INDEX=1（初始飞机序号，本脚本生成的飞机的CopterID，以此START_INDEX为初始值，依次递增1）

SET /a TOTOAL_COPTER=8（总飞机数量，仅在多机联机仿真才需要幅值，告诉本脚本实际的飞机总数，以此来确定矩形队列的边长）

SET UE4_MAP=Grasslands（设置地图名字）

SET /a ORIGIN_POS_X=0（矩形编队的原点X位置，单位米，只支持整数输入）

SET /a ORIGIN_POS_Y=0（矩形编队的原点Y位置，单位米，只支持整数输入）

SET /a ORIGIN_YAW=0 （矩形编队的原点 yaw 角度，单位度，只支持整数输入）
SET /a VEHICLE_INTERVAL=2 （矩形编队的飞机间隔，单位米，只支持整数输入）
SET /a UDP_START_PORT=20100 （接收外部控制数据的 UDP 通信接口，与 CopterID 对应会自动加 2，这里通常不需要修改，仅在电脑端口被占用时才修改）
SET /A DLLModelVal=DLLModel （是否使用 DLL 模型，DLL 模型的名字。这里支持将 Simulink 的飞行器模型生成 DLL 来导入平台，此模式支持固定翼、无人车等模型）
SimMode （仿真模式，这里设置为 0 或 PX4_HITL，表示硬件在环仿真）
SET IS_BROADCAST=0 （是否联机仿真，这里可输入目标 IP 地址序列）
SET UDPSIMMODE=0 （UDP_START_PORT 端口接收的数据协议，UDP 模式传输的是平台私有结构体，支持 Simulink 控制；MAVLink 模式传输的是 MAVLink 协议，支持 Python 和 mavros 等控制模式）

SITLRun.bat

常规软件在环脚本，支持输入飞机数量，自动开启多机软件在环仿真
与 HITLRun.bat 相比，关键代码如下

set SimMode=2 （这里设置为软件在环模式，对应 CopterSimUI 的值）
set PX4SidlFrame=iris （这里对应了 PX4 飞控的机架模式，这里是一个四旋翼）

HITLPos.bat

开启硬件在环仿真，支持输入 PosX、PosY 和 Yaw 的值来初始化飞机位置偏航角

注：输入的位置字符串，以英文逗号“,”分隔

关键代码如下：

SET /P PosXStr=Please enter the PosX (m) list: （输入 X 的初始位置序列，单位米，可以为浮点数）

SET /P PosYStr=Please enter the PosY (m) list: （输入 Y 的初始位置序列，单位米，可以为浮点数）

SET /P YawStr=Please enter the Yaw (degree) list: （输入偏航的初始值序列，单位度，可以为浮点数）

SITLPos.bat

开启软件在环仿真，支持输入 PosX、PosY 和 Yaw 的值来初始化飞机位置偏航角

注：其余注意事项同上

HITLPosStr.bat

开启硬件在环仿真，PosX、PosY 和 Yaw 的值以字符串的形式写在 bat 文件的 PosXStr、PosYStr、YawStr 变量中，不需要手动输入

注：使用时，需要手动去修改 PosXStr、PosYStr、YawStr 字符串的值，来设置生成飞机的位置和偏航

注：输入的位置字符串，以英文逗号“,”分隔

注：位置序列的数量可以大于插入飞控数量，会依次取前面的位置进行幅值，例如本例子位置字符串包含 10 组飞机初始位姿，因此支持仿真 1 到 10 个飞机的硬件在环仿真

关键代码如下：

SET PosXStr=1,2,3,4,5,6,7,8,9,10（X 的初始位置序列，单位米，可以为浮点数）

SET PosYStr=0,0,0,0,0,0,0,0,0,0（Y 的初始位置序列，单位米，可以为浮点数）

SET YawStr=0,0,0,0,0,0,0,0,0,0（偏航的初始值序列，单位度，可以为浮点数）

SITLPosStr.bat

开启软件在环仿真，PosX、PosY 和 Yaw 的值以字符串的形式写在 bat 文件的 PosXStr、PosYStr、YawStr 变量中，不需要手动输入

注：其余注意事项同上

HITLRunSysID.bat

开启硬件在环仿真，但是 CopterID 不是自动增 1 排序，而是根据 SysID 的值（在 QGroundControl 中可配置）来自动确定

注：如果使用常规模式，CopterID 是和电脑串口序列先后顺序绑定的，每次插拔飞控后串口号会变化，因此无法通过 CopterID 定位到确定的飞控硬件

注：使用本方法，CopterID 可以直接和飞控硬件绑定，便于故障时快速确定飞控序号。

注：如果要再使用常规模式的 HITLRun.bat 脚本，建议将 SysID 改回默认值 1

关键代码如下：

SET /a IsSysID=1（开启通过 SysID 自动计算 CopterID 的功能）

HITLPosSysIDStr.bat

开启硬件在环仿真，通过 SysID 自动确定 CopterID 取值，且支持配置初始位置序列

注：本脚本会以 SysID - START_INDEX 的值，去索引 PosXStr、PosYStr、YawStr 列表来确定最终位置

注：例如 SysID 为 15，START_INDEX 为 11，PosXStr=1,2,3,4,5,6,7,8,9,10，则飞机最终的 x 取值应该是 PosXStr 的第 4 位（从 0 开始数），也就是 PosX=5

关键代码如下：

SET /a IsSysID=1 （开启通过 SysID 自动计算 CopterID 的功能）

SET /a START_INDEX=1 (位置列表的初始飞机的值)

SET PosXStr=1,2,3,4,5,6,7,8,9,10 （X 的初始位置序列，单位米，可以为浮点数）

SET PosYStr=0,0,0,0,0,0,0,0,0,0 （Y 的初始位置序列，单位米，可以为浮点数）

SET YawStr=0,0,0,0,0,0,0,0,0,0 （偏航的初始值序列，单位度，可以为浮点数）

HITLPosSysID.bat

开启硬件在环仿真，通过 SysID 自动确定 CopterID 取值（同 HITLPosSysIDStr.bat），但初始位置序列是手动输入的

HITLRunLowGPU.bat

开启常规硬件在环仿真，不使用 RflySim3D 但是使用 3DDisplay，来保证低配电脑能运行平台

注：核心修改是 HITLRun.bat 将 RflySim3D 字符替换为 3DDisplay

注：3DDisplay 只支持单机显示，虽然打开速度较慢，但是显卡需求小

关键代码如下：

```
cd %PSP_PATH%\3DDisplay
```

```
tasklist|find /i "3DDisplay.exe" || start %PSP_PATH%\3DDisplay\3DDisplay.exe
```

SITLRunLowGPU.bat

开启常规软件在环仿真，不使用 RflySim3D 但是使用 3DDisplay，来保证低配电脑能运行平台

注：其余注意事项同上

HITLRunMAVLink.bat

开启常规硬件在环仿真，使用 MAVLink_Full 的 UDPSIMMODE 取值，支持 Python 视觉控制和真机飞行控制

注：这里仅修改了 UDPSIMMODE 的值

注：UDPSIMMODE 可以取数字或字符串，对应了 CopterSim 的 UI 界面

关键代码如下：

```
SET UDPSIMMODE=Mavlink_Full （使用 Mavlink_Full 的通信模式于 20100 系列端口）
```

SITLRunMAVLink.bat

开启常规软件在环仿真，使用 MAVLink_Full 的 UDPSIMMODE 取值，支持 Python 视觉控制和真机飞行控制

注：其余注意事项同上

HITLRunNoUI.bat

开启硬件在环仿真，使用无 UI 版 CopterSimNoUI.exe，可以减小计算量，适合大规模集群使用

注：核心修改是将 CopterSim.exe 的字符串替换为 CopterSimNoUI.exe

注：平台的集群例程，超过数量 4 时，会自动使用 NoUI 版 CopterSim 以提高飞行稳定性。

SITLRunNoUI.bat

开启软件在环仿真，使用无 UI 版 CopterSimNoUI.exe，可以减小计算量，适合大规模集群使用

注：其余注意事项同上

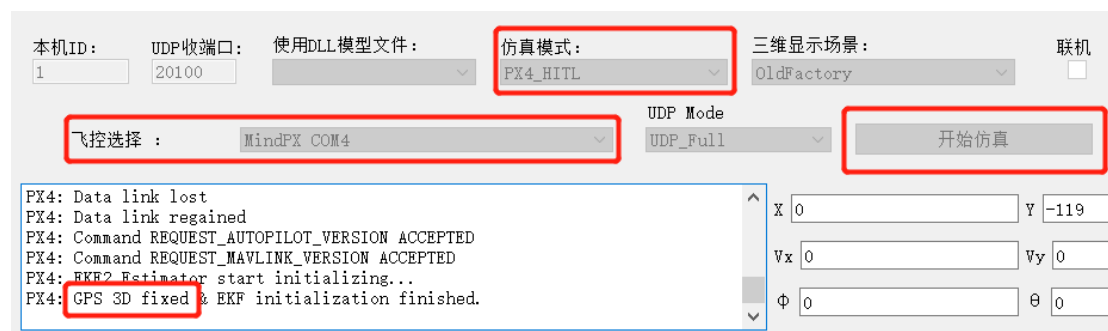
[2].开启单机硬件在环仿真（HITL）步骤：

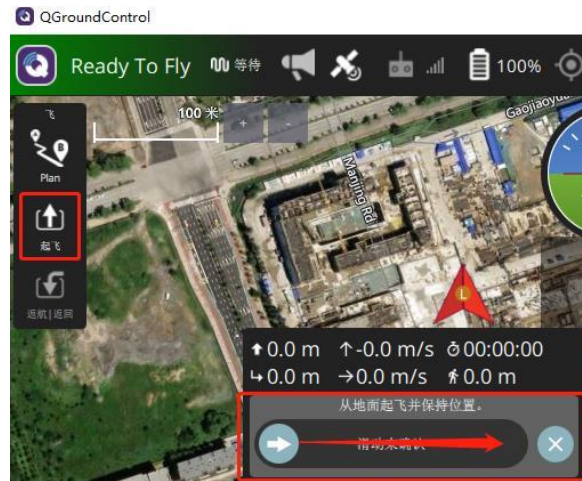
将 Pixhawk 自驾仪（已经按上一节配置好，并连接遥控器和接收机）通过 USB 插入电脑

打开 CopterSim 软件，并在其“仿真模式”选项选择标签“PX4_HITL”，同时检查其他配置，确保模型参数、地图、初始位置和偏航满足需求，再打开 RflySim3D 配置视景。

在“飞控选择”中选择 Pixhawk 对应串口，然后点击 CopterSim 界面的“开始仿真”按钮，即可开始硬件在环仿真（注：也可以双击 HITLRun 快捷方式，并输出飞控串口号数字直接开始硬件在环仿真），等待左下角消息框提示“GPS 3D Fixed”完成初始化。

打开 QGroundControl（QGC）地面站软件，在 FLY 页面（纸飞机图标）点击“起飞”按钮，向右滑动滑块，即可控制飞机自动起飞；也可用遥控器手动控制飞机起飞。

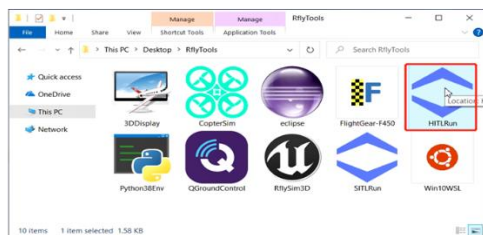




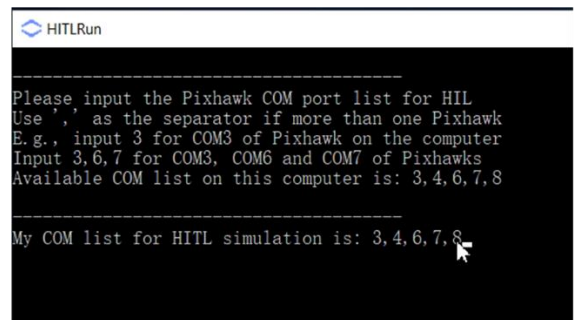
3.7 同时连多个Pixhawk进行仿真



1. 插入任意个Pixhawk飞控



2. 双击HITLRun一键启动脚本



3. 输入Pixhawk串口号序列



4. QGC控制起飞