

1. 实验名称及目的

1.1 实验名称

dll综合模型Python外部多机控制实验

1.2 实验目的

无需启动CopterSim，直接通过外部程序分别加载多个DLL综合模型，并通过外部控制程序执行多机飞行任务

1.3 关键知识点

本实验使用DllSimCtrlAPI.ModelLoad接口类实例化4个综合模型（控制器+模型一体）DLL模型文件MulticopterNOpx4.dll，调用该实例的CreateVehicle接口初始化载具，StartAutUpdate接口开启实时仿真，并使用其中StartExtCtrl接口启动外部控制功能，会开启UDP端口来模拟CopterSim通过20100端口和30100端口发送UDP消息，并接收控制指令。

使用PX4MavCtrl.PX4MavCtrl接口类创建Mavlink通信实例并通过20100端口收发数据并进行外部控制

注：Python

DLL的外部控制模式只支持UdpMode=1，也就是UDP_Simple模式，因此需要使用InitMavLoop(UdpMode=1)来初始化通信模式

```
for i in range(VehilceNum):
```

```
MavList[i].InitMavLoop(UdpMode=1)
```

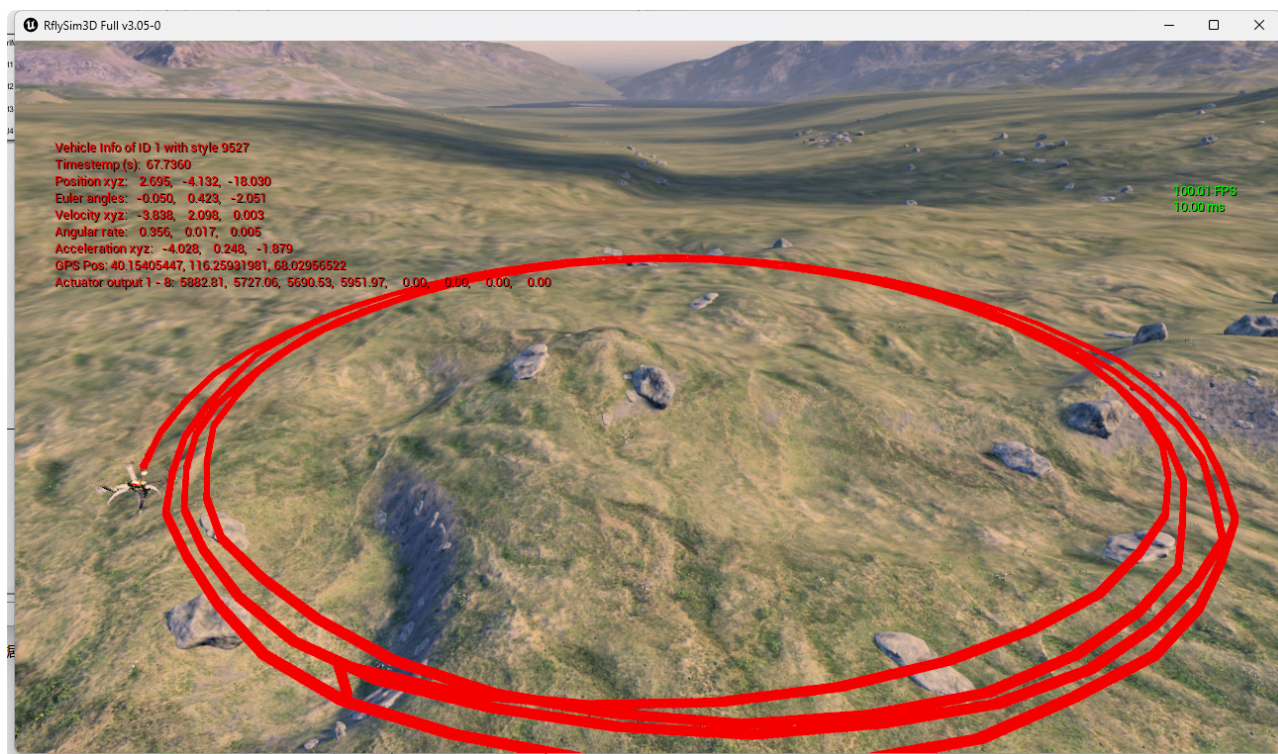
```
MavList[i].initOffboard()
```

注：DllSimCtrlAPI在创建大于1的CopterID时，会复制dll文件，为DLL名字_CopterID.dll的形式。这是因为一个进程中，只能共享加载一个DLL模型文件，因此需要复制为不同文件名，分别加载才能避免冲突。

运行完例程，会创建MulticopterNOpx4_2.dll、MulticopterNOpx4_3.dll、MulticopterNOpx4_4.dll三个新的dll文件。

2. 实验效果

4个飞机成功起飞并按圆形轨迹运行



3. 文件目录

例程目录：

[安装目录]\RflySimAPIs\4.RflySimModel\0.ApiExps\12.DllModelImport\7.ModelLoadExtCtrlpython

文件夹/文件名称	说明
ModelLoadApiTest.py	实验主程序（加载dll模型并开启外部控制功能）
MulticopterNOpx4.dll	一个四旋翼的综合模型动态链接库dll文件
Python38Run.bat	Python环境的一键启用脚本
RunRflySim3D.bat	RflySim3D三维可视化软件的一键启动脚本
NoPX4SITL4Swarm.py	Python外部多机控制程序

| 4.运行环境

| 4.1 软件要求

Windows 10及以上版本；RflySim工具链；MATLAB 2017B及以上版本。

①：若使用Pixhawk 6X飞控，平台安装时的编译命令为：px4_fmu-v6x_default，推荐PX4固件版本为：1.12.3。其他配套飞控及编译命令请见：

<https://rflysim.com/doc/zh/1/Hardware.html>

| 4.2 硬件要求

笔记本/台式电脑① 1台。

①：推荐配置请见：<https://rflysim.com/>

| 5.实验步骤

| 5.1 必做实验：外部加载dll模型

| Step 1：启动RflySim3D

双击运行[RunRflySim3D.bat](#)，启动1个RflySim3D窗口。



Step 2: 运行Python程序启动仿真

在文件夹下，双击 [Python38Run.bat](#)，打开集成好的python环境，在该环境下运行 [ModelLoadApiTest.py](#) 文件，输入 `python ModelLoadApiTest.py`

```
C:\Windows\system32\cmd.exe X + v
Python3.8 environment has been set with openCV+pymavlink+numpy+pyulog etc.
You can use pip or pip3 command to install other libraries
Put Python38Run.bat into your code folder
Use the command: 'python XXX.py' to run the script with Python
F:\d2\4.RflySimModel\0.ApiExps\12.DLLModelImport\7.ModelLoadExtCtrlpython>python ModelLoadApiTest.py
```



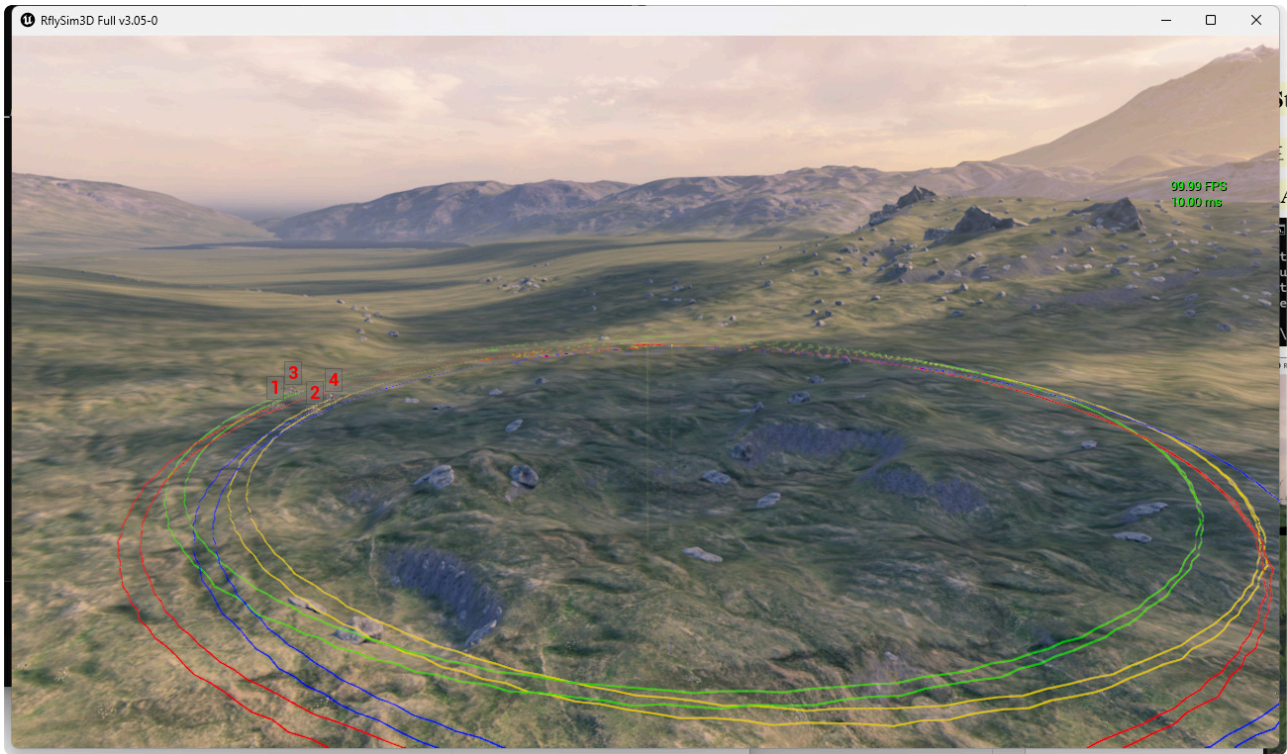
Step 3: 运行Python多机控制程序

再次双击 [Python38Run.bat](#)，打开集成好的python环境，在该环境下运行 [NoPX4SITL4Swarm.py](#) 文件，输入 `python NoPX4SITL4Swarm.py`

```
C:\Windows\system32\cmd.exe x + v
Python3.8 environment has been set with openCV+pymavlink+numpy+pyulog etc.
You can use pip or pip3 command to install other libraries
Put Python38Run.bat into your code folder
Use the command: 'python XXX.py' to run the script with Python
F:\d2\4.RflySimModel\0.ApiExps\12.DllModelImport\7.ModelLoadExtCtrlpython>python NoPX4SITL4Swarm.py
```

Step 4: 观察结果

在RflySim3D中可见，飞机起飞到一定高度后飞圆轨迹



5.2 选做实验（VS Code调试运行）

准备工作

- 先确保已经按 [RflySimAPIs\1.RflySimIntro\2.AdvExps\e3.PythonConfig\Readme.pdf](#) 步骤，正确配置VS Code环境。或者配置了自己的Pycharm等自定义Python环境。
- 其他步骤与上文相同，运行 [ModelLoadApiTest.py](#) 时，可使用VS Code（或Pycharm等工具）来打开 [ModelLoadApiTest.py](#) 文件，并阅读代码，修改代码，调试执行等。

扩展实验

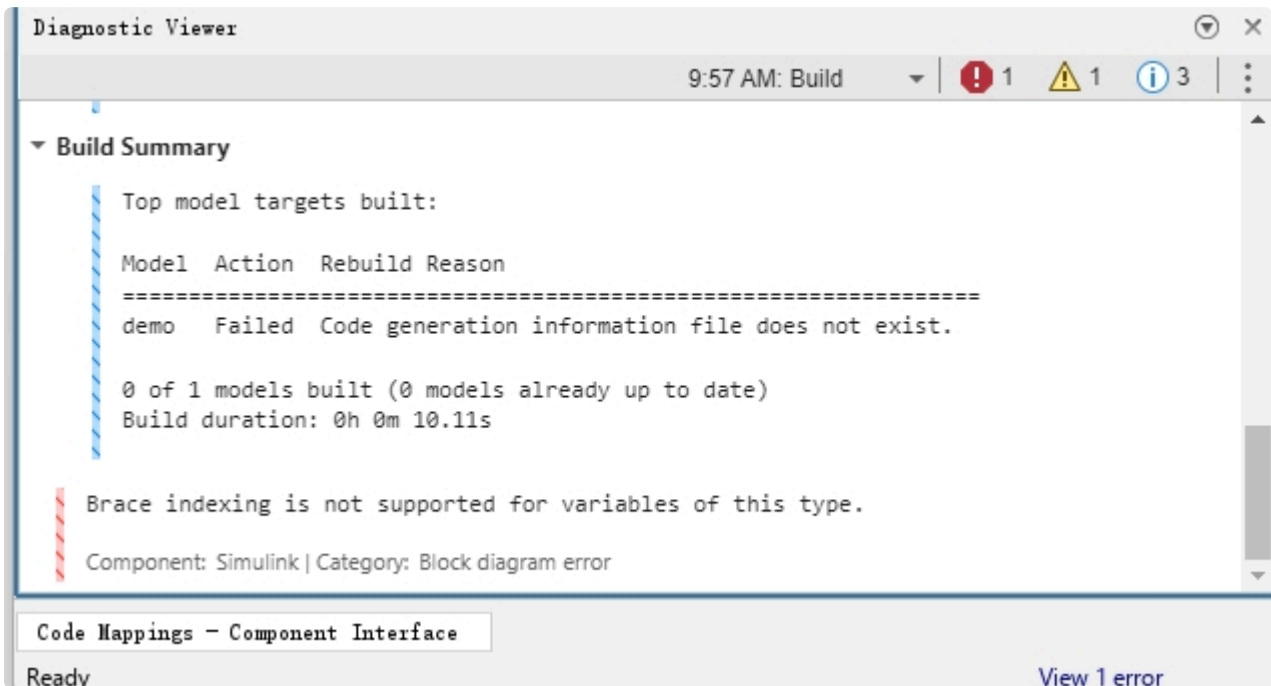
- 请自行使用VS Code阅读 [ModelLoadApiTest.py](#) 源码，通过程序跳转，了解每条代码的执行原理；再通过调试工具，验证每条指令的执行效果。

6.参考资料

1. outCopterData接口 [..\..\8.OutCopterData\Readme.pdf](#)
2. DLL/SO模型与通信接口 [..\..\PX4PSP\RflySimAPIs\4.RflySimModel\API.pdf](#)

7. 常见问题

Q1: 未正确安装visual studio c++编译环境并配置mex，导致Simulink文件编译失败



A1: 首先将低于当前MATLAB版本的Visual Studio C++编译环境安装到VS默认安装目录，然后在MATLAB的命令行窗口中输入指令“mex -setup”，一般来说会自动识别并安装上支持的编译器（例如Visual C++ 2017），命令行显示“MEX 配置使用 ‘Microsoft Visual C++ 2017’ 以进行编译”的字样说明安装正确。详细环境配置参考” [RflySim平台安装目录]\RflySimAPIs\4.RflySimModel\API.pdf “中的环境配置



Q2: 编译报错，无法加载库文件



A2: 这可能是由于安装平台时PX4PSP工具箱未更新到最新版，更新RflySim安装包后按照如下配置重新安装平台即可

