

# 1. 实验名称及目的

## 1.1. 实验名称

模型 FaultParamAPI.InitInParams 参数动态修改实验（仅限完整版及以上版本）

## 1.2. 实验目的

本实验通过 Python 动态修改模型中的 FaultParamAPI.InitInParams 参数，以实现仿真中模型状态的更改。以此实验让平台用户熟悉 FaultParamAPI.InitInParams 参数这种 Python 程序动态修改的方法。

## 1.3. 关键知识点

本实验需要电脑中部署 Visual Studio 2022 环境，部署方式见：[\[安装目录\]\RflySimAPIs\1.RflySimIntro\2.AdvExps\6\\_VisualStudioInstall](#)

[\\\*\\*\4.RflySimModel\3.CustExps\0\\_AdvApiExps\5.ParamAPI\Intro.pdf](#)

# 2. 实验效果

将 FaultParamAPI.InitInParams 参数的不同维度配置在飞机模型中的初始位置、速度等位置后，之后启动没有拷贝 csv 文件的启动脚本时初始状态为 InitInParams 的默认值。随后运行 InitParamModDemo.py 程序，实现载具初始状态（初始位置的更改）。

# 3. 文件目录

例程目录：[\[安装目录\]\RflySimAPIs\4.RflySimModel\3.CustExps\0\\_AdvApiExps\5.ParamAPI\1.initParams\2.initParamsAPI\\_py\](#)

文件夹/文件名称	说明
<a href="#">..\Intro.pdf</a>	dll 模型参数实时修改实验原理
InitParamModDemo.slx	四旋翼飞机模型文件。
GenerateModelDLLFile.p	DLL 格式转化文件。
InitParamModDemo_SITL_No.bat	软件在环启动脚本。
MavLinkStruct.mat	MavLink 数据结构体 mat 文件
InitParamModDemo.dll	四旋翼飞机动态链接库
InitParamModDemo_init.m	动力学模型相关参数。
InitParamModDemo.py	动态修改参数脚本。
Python38Run.bat	Python 程序运行脚本

# 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 <sup>①</sup>	1
2	RflySim 工具链	\	\
3	MATLAB 2017B 及以上 <sup>③</sup>	\	\
4	Python	\	\

① : 推荐配置请见: <https://rflysim.com/doc/en>

## 5. 实验步骤

### 5.1 必做实验：模型参数动态修改

#### Step 1: 初始化参数接口

在 InitParamModDemo\_Init.m 文件里面，声明参数 FaultParamAPI.InitInParams 为 32 维的浮点数向量。注意：这里使用结构体是为了自动代码生成时，不会被优化为静态常数，使之具备可修改性。

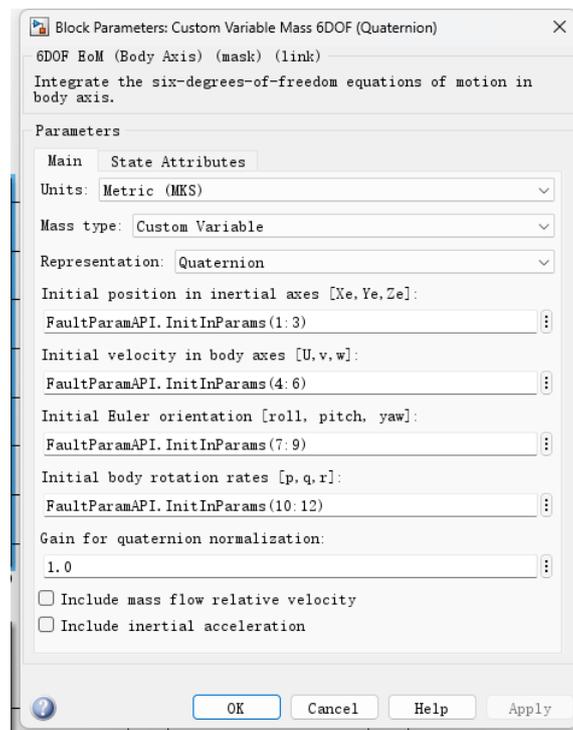
```

34
35
36 %% 初始化参数接口
37 % Define the 32-D InitInParams vector for external modification
38 - FaultParamAPI.InitInParams = zeros(32,1);
39

```

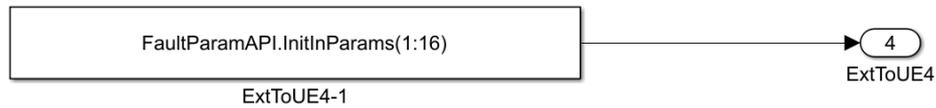
#### Step 2: 修改 6DOF 模块

打开 InitParamModDemo.slx 将 6DOF 模块中能够配置参数的地方，用 FaultParamAPI.InitInParams 的特定维数来填充。



### Step 3: 设置 ExtToUE4 输出接口

将前 16 维输出到 ExtToUE4，便于观察结果。



### Step 4: 编译模型

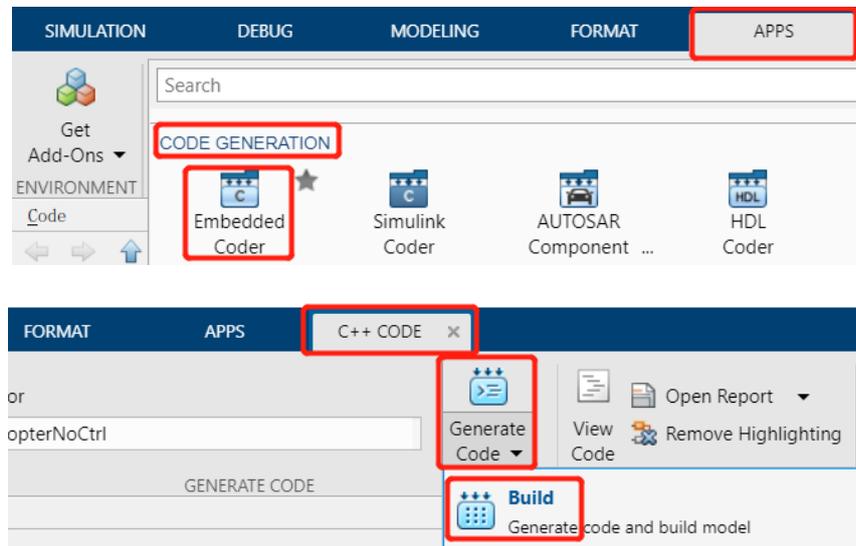
修改模型完成后，将模型编译为 C++ 代码。

编译配置可参考 [4.RflySimModel\0.ApiExps\2.UserDefinedC++\2.GenC++\Readme.pdf](#)

对于 MATLAB 2019a 及之前版本，工具栏样式见下图，直接点击它的编译按钮“Build”即可。

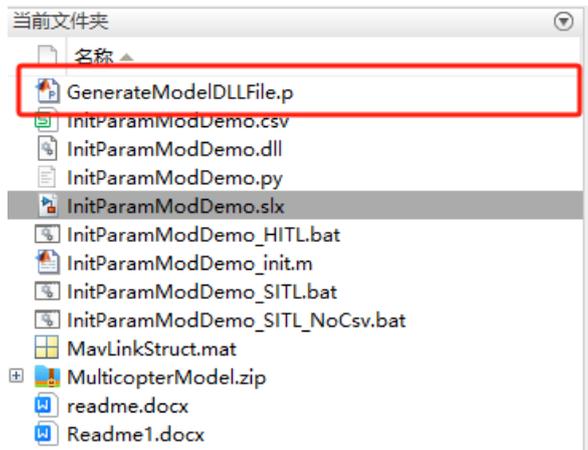


对于 2019b 及之后版本，点击 APPS - CODE GENERATION - Embedded Coder 才能弹出代码生成工具栏，在其中如下图所示点击“C++CODE” -“Generate Code”-“Build”按钮就能编译生成代码。



### Step 5: 生成 DLL 文件

运行 GenerateModelDLLFile.p，生成 DLL 文件

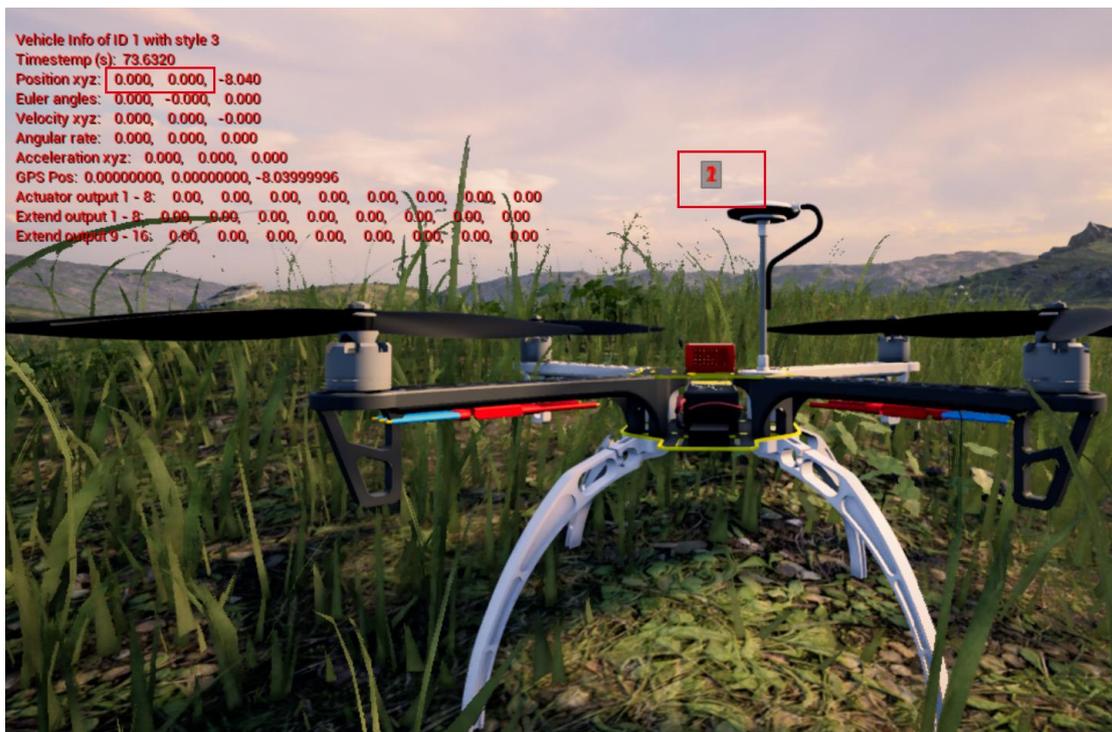


## Step 6: 启动仿真

运行 `InitParamModDemo_SITL_NoCsv.bat` (没有自动拷贝 csv 文件的版本), 输入 2, 创建两个飞机。(为了保证不会有 csv 文件存在, 干扰实验, 这个 bat 还增加了删除 csv 文件的代码)

```
7 REM Copy the latest dll file to CopterSim folder
8 copy /Y "%~dp0"%DLLModel%.dll %PSP_PATH%\CopterSim\externa
9 del %PSP_PATH%\CopterSim\external\model\%DLLModel%.csv
0
1
```

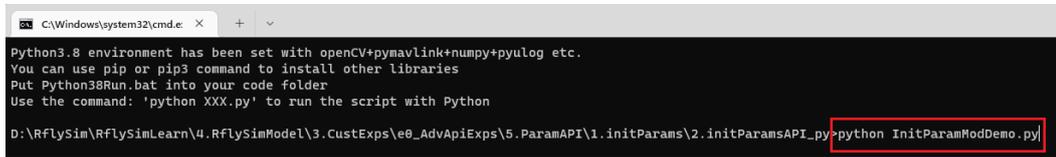
可以看到, 创建了两个飞机, 没有响应 bat 脚本的位置布局控制 (取决于 dll 的 `ModelInit_PosE` 参数, 这里并没有用到), 初始位置都是 0,0,0, 两架飞机重叠在一起, 有一个慢慢从地下上升到地表的过程 (地面模型的作用), 这是因为现在使用的是 `InitInParams` 的默认值, 全为 0, `RflySim3D` 界面按下键盘 D 可以看到飞机参数。



```
Vehicle Info of ID 1 with style 3
Timestamp (s): 73.6320
Position xyz: 0.000, 0.000, -8.040
Euler angles: 0.000, -0.000, 0.000
Velocity xyz: 0.000, 0.000, -0.000
Angular rate: 0.000, 0.000, 0.000
Acceleration xyz: 0.000, 0.000, 0.000
GPS Pos: 0.00000000, 0.00000000, -8.03999996
Actuator output 1 - 8: 0.00, 0.00, 0.00, 0.00, 0.00, 0.00, 0.00, 0.00
Extend output 1 - 8: 0.00, 0.00, 0.00, 0.00, 0.00, 0.00, 0.00, 0.00
Extend output 9 - 16: 0.00, 0.00, 0.00, 0.00, 0.00, 0.00, 0.00, 0.00
```

## Step 7: 运行 Python 程序

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 InitParamModDemo.py 文件，输入 python InitParamModDemo.py



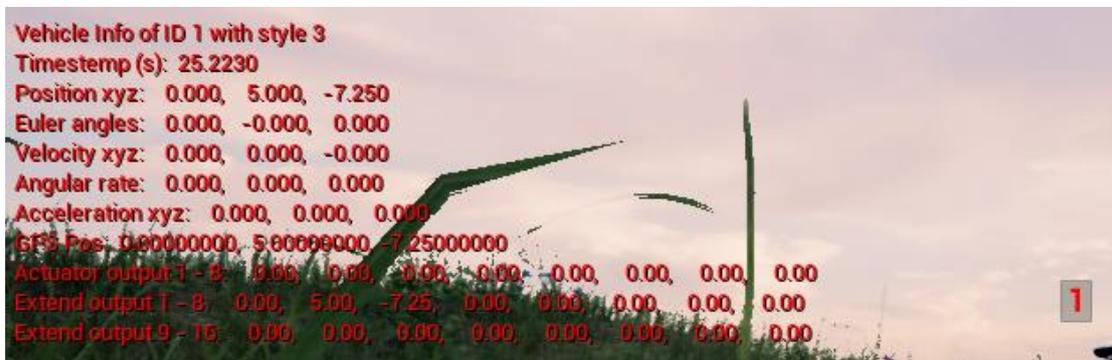
```
C:\Windows\system32\cmd.e  x  +  v
Python3.8 environment has been set with openCV+pymavlink+numpy+pyulog etc.
You can use pip or pip3 command to install other libraries
Put Python38Run.bat into your code folder
Use the command: 'python XXX.py' to run the script with Python
D:\RflySim\RflySimLearn\4.RflySimModel\3.CustExps\e0_AdvApiExps\5.ParamAPI\1.initParams\2.initParamsAPI_py>python InitParamModDemo.py
```

## Step 8: 观察结果

通过 Python 向模型发送了改变参数的消息（使用 DllSimCtrlAPI 的 sendInitInParams 接口），可以看到飞机位置发生了变化。



```
1 import DllSimCtrlAPI
2
3 dll = DllSimCtrlAPI.DllSimCtrlAPI()
4
5 inParam=[0]*32
6 inParam[0:3]=[0,5,-7.25]
7 # 设置1号飞机, bitmask=-1所有参数都设置, inParam设置飞机位置为0,5,-7.25 (和2号飞机对调)
8 dll.sendInitInParams(-1,inParam,1)
9
10 inParam[0:3]=[1,0,-8.08]
11 # 设置2号飞机, bitmask=-1所有参数都设置, inParam设置飞机位置为1,0,-8.08 (和1号飞机对调)
12 dll.sendInitInParams(-1,inParam,2)
```



## 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\3\\_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 InitParamModDemo.py 时，可使用 VS Code（或 Pycharm 等工具）来打开 InitParamModDemo.py 文件，并阅读代码，修改代码，调试执行等。

### 扩展实验：

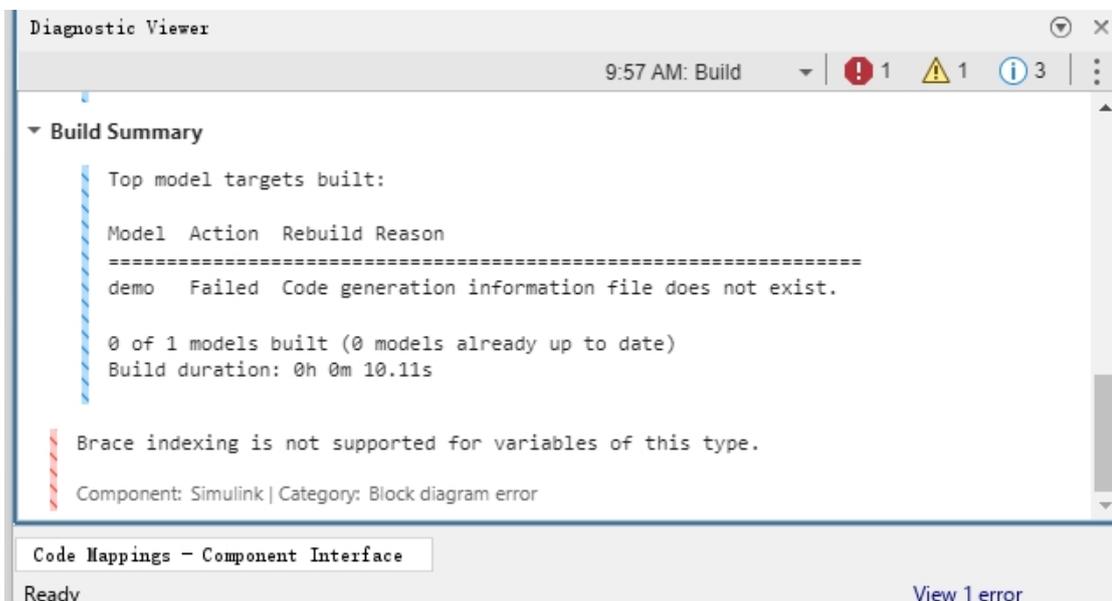
- 请自行使用 VS Code 阅读 InitParamModDemo.py 源码，通过程序跳转，了解每条代码的执行原理；再通过调试工具，验证每条指令的执行效果。

## 6. 参考资料

- [1]. [API.pdf 中 DLL/SO 模型与通信接口的重要参数部分。](#)
- [2]. [API.pdf 中的环境配置](#)
- [3]. [API.pdf 中的 Simulink 建模模板介绍](#)

## 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



```
命令窗口
>> mex -setup
MEX 配置为使用 'Microsoft Visual C++ 2017 (C)' 以进行 C 语言编译。
警告: MATLAB C 和 Fortran API 已更改, 现可支持
包含 2^32-1 个以上元素的 MATLAB 变量。您需要
更新代码以利用新的 API。
您可以在以下网址找到更多的相关信息:
http://www.mathworks.com/help/matlab/matlab\_external/upgrading-mex-files-to-use-64-bit-

要选择不同的 C 编译器, 请从以下选项中选择一种命令:
Microsoft Visual C++ 2013 (C) mex -setup:D:\MATLAB\R2017b\bin\win64\mexopts\msvc2013.xml C
Microsoft Visual C++ 2015 (C) mex -setup:D:\MATLAB\R2017b\bin\win64\mexopts\msvc2015.xml C
Microsoft Visual C++ 2017 (C) mex -setup:C:\Users\dream\AppData\Roaming\MathWorks\MATLAB\R2

要选择不同的语言, 请从以下选项中选择一种命令:
mex -setup C++
mex -setup FORTRAN
fx >>
```

Q2: 编译报错, 无法加载库文件



```
诊断台名称
下午4:48:编译
-----
Exp1_MInModelTemp 信息保存文件完成工作失败。 无法编译。有关详细信息, 请参阅编译日志。 ed
编译了 0 个模型, 共 1 个模型(0 个模型已经是最新的)
编译持续时间: 0h 0m 3.7699s
-----
无法加载 'pixhawk_slib_adv\cortexforcermodel' 引用的库 'pixhawk_slib_adv1'。
附件: Simulink|类型: block diagram 错误
代码映射 - 应用程序
```

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

Toolbox one-key installation script: RflySimA... — □ ×

(1) Software package installation directory  
C:\PX4PSP

(2) PX4 firmware compiling command: firmware versions <= PX4-1.8 use format px4fmu-v3\_default; >= PX4-1.9 use format px4\_fmu-v3\_default  
px4\_fmu-v6c\_default

(3) PX4 firmware version (1: PX4-1.7.3, ... , 6: PX4-1.12.3, 7: PX4-1.13.2, 8: PX4-1.14.4, 9: PX4-1.15.0)  
9

(4) PX4 firmware compiling toolchain (1: WinWSL[suitable for all versions], 2: Msys2[suitable for <= PX4-1.8], 3: Cygwin[for >=PX4-1.8])  
1

(5) Whether to reinstall PSP toolbox (yes to reinstall and no to remain current installation)  
yes

(6) Whether to reinstall the dependent software packages (CopterSim, QGroundControl, CopterSim, etc. About 5 minites)  
no

(7) Whether to reinstall the selected compiling toolchain (yes to reinstall and no to remain unchanged, about 5 minites)  
no

(8) Whether to reinstall the selected PX4 firmware source code (yes to reinstall and no to remain unchanged, about 5 minites)  
no

(9) Whether to pre-compile the selected firmware with the selected command (yes to compile and no to remain unchanged, about 5 minites)  
no

(10) Whether to block the actuator outputs in the PX4 firmware code ("yes" to use Simulink controller, "no" to use PX4 official controller)  
no

OK Cancel