

1. 实验名称及目的

1.1 实验名称

基于最大模板的螺旋桨模块故障注入。

1.2 实验目的

了解基于最大模板的螺旋桨模块故障注入。

1.3 关键知识点

原理：FaultInjectAPITest.py通过

PX4MavCtrlV4库的sendInDoubCtrls/sendSILIntDouble/sendSILIntFloat 系列接口，将故障信号inSIL28d（或inSILInts/inSILFloats）在UDP端口中传输到simulink最大模块中的螺旋桨故障注入模块，以此实现螺旋桨故障注入功能。编译simulink最大模块生成DLL文件，将DLL文件导入CopterSim后，成功注入螺旋桨故障并完成仿真。

故障ID及对应参数见<../../RflySimSDK/html/md_phm_2md_2Faultinject.html>

螺旋桨故障在故障信号inSIL28d的识别ID为：123451

，输入参数按照顺序为#1~#x号螺旋桨执行效率比（0~1）。若未注入故障，螺旋桨执行效率默认为100%。

注释：螺旋桨执行效率：

螺旋桨执行效率通常定义为螺旋桨实际产生的推力或升力功率与发动机输入功率的比值。

$$\text{螺旋桨效率}(\eta_p) = \frac{\text{推力功率}}{\text{发动机功率}} \times 100\%$$

其中，推力功率是推力（力）与速度（距离/时间）的乘积。

注入故障所用到的simulink模块PropFault结构图 及其 输入输出参数解释：

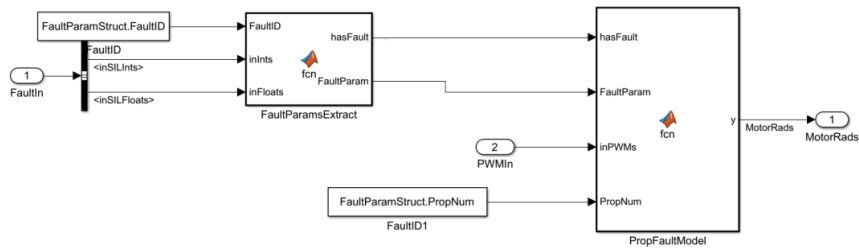
该模块用于提取故障参数。模块输入FaultIn，即inSIL28d，为包含8维的inSILInts，20维的inSILFloats的故障参数。

功能	包含FaultParamsExtract模块， PropFaultModel模块。 FaultParamsExtract模块用于选择螺旋桨故障类型，并输出注入故障后螺旋桨的转速； PropFaultModel模块用于表示螺旋桨的数字建模。
	123451为螺旋桨故障ID
输入	
FaultIn	故障输入，包括故障类型和故障参数。
FaultID	故障的唯一标识密钥
PWMin	注入故障后电机输出的转速
PropNum	一个整数，表示需要考虑的电机数量。
输出	
MotorRads	含注入故障后的电机转速。

inSILInts是8维的输入，inSILFloats是20维的输入，FaultID是本故障的唯一标识密钥
inSILInts的每一位，对应了inSILFloats的两位数字，用于将故障参数传进来。

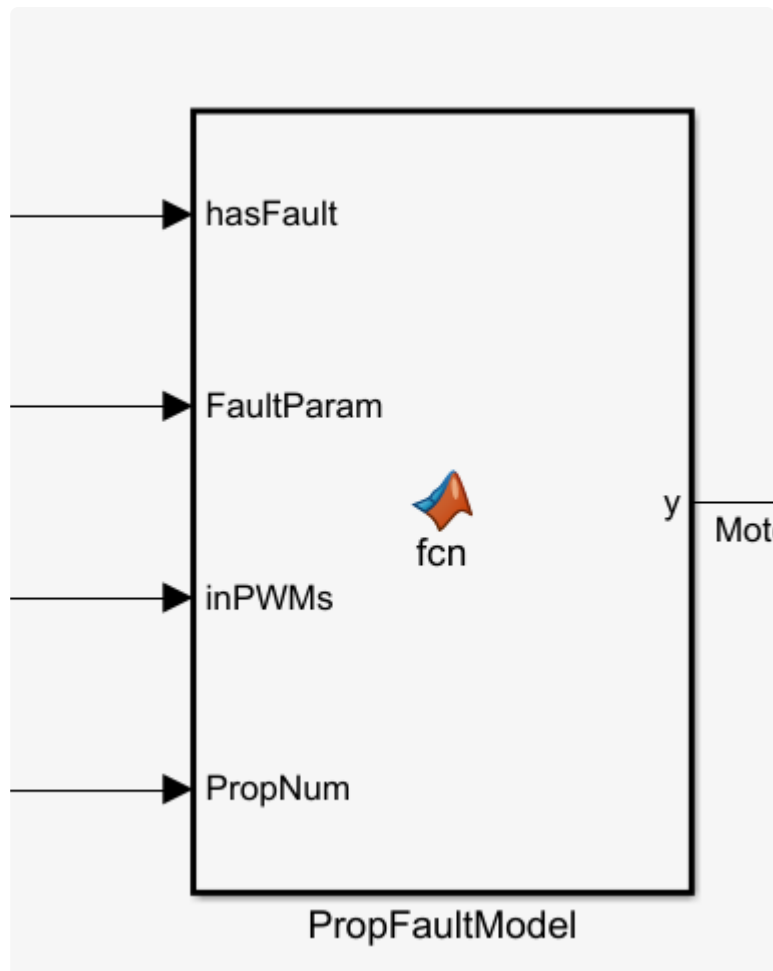
hasFault为是否有故障注入bool变量，true或false
FaultParam和inSILFloats维度相同，都是20维浮点数

123451为螺旋桨故障ID
注：如果没有新的故障注入进来，FaultParamsExtract会保持上一时刻的值。



内部模块结构图：

功能	本模块需要用到8维FaultParam的输入，因此故障ID的格式应该是： inSILInts=[FaultID FaultID FaultID FaultID ...] inSILInts=[param1 param2 param3 param4 param5 param6 param7 param8 ...] -> FaultParam param的每一位表示本电机的健康系数（0到1），实际上就是pwm_out = pwm_in * param 如果param1=0，则电机1直接坏掉，始终输出0
输入	
hasFault	一个布尔值，如果为 true，则表示在 PWM 信号中注入故障。
FaultParam	一个长度为 8 的数组，每个元素代表一个电机的健康系数，范围从 0 到 1。这个系数用于调整电机的 PWM 输出信号。
inPWMs	一个数组，包含电机的 PWM 输入信号。
PropNum	一个整数，表示需要考虑的电机数量。这个值用于表示故障注入的电机数量。
输出	
y	一个数组，包含注入故障后的 PWM 输出信号。



```

PropFaultModel x +
1  function y = fcn(hasFault, FaultParam, inPWMs, PropNum)
2  % 本模块需要用到8维FaultParam的输入
3  %因此故障ID的格式应该是:
4  %inSILInts=[FaultID FaultID FaultID FaultID ...]
5  %inSILInts=[param1 param2 param3 param4 param5 param6 param7 param8 ...] -> FaultParam
6  % param的每一位表示本电机的健康系数(0到1), 实际上就是pwm_out = pwm_in * param
7  % 如果param1=0, 则电机1直接坏掉, 始终输出0
8
9  y=inPWMs; %默认情况下不注入故障, 直接将数据输出
10
11  if PropNum<1
12      PropNum=int32(1);
13  end
14
15  if PropNum>8
16      PropNum=int32(8);
17  end
18
19  if hasFault %如果注入了故障
20      for i=1:PropNum
21          y(i) = inPWMs(i) * FaultParam(i);
22      end
23
24  end

```

主要要求掌握:

螺旋桨故障注入模块的功能及其实现。

说明文档	说明文档链接以及地址
simulink中PropFault模块	PX4PSP/RflySimSDK/html/md_phm_2md_2Prop_01FaultModel.html
FaultInjectAPITest.py	PX4PSP\RflySimAPIs\7.RflySimPHM\0.ApiExps\xe4_FaultInjectAPITest_py\readme.pdf
PX4MavCtrlV4库	PX4PSP/RflySimSDK/html/PX4MavCtrlV4_8py.html

2. 实验效果

3. 文件目录

例程目录：[\[安装目录\]](#)\RflySimAPIs\7.RflySimPHM\1.BasicExps\xe7_PropFault

文件夹/文件名称	说明
MulticopterModel.slx	故障注入模块的最大模板模型文件。
MulticopterModelHITL.bat	硬件在环仿真批处理文件。
MulticopterModelSITL.bat	软件在环仿真批处理文件。
GenerateModelDLLFile.p	DLL格式转化文件。
Init.m	动力学模型相关参数。
Python38Run.bat	Python程序执行脚本。
FaultInjectAPITest.py	故障注入程序。

4. 运行环境

4.1 软件要求

Windows 10及以上版本；RflySim工具链；MATLAB 2022B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。

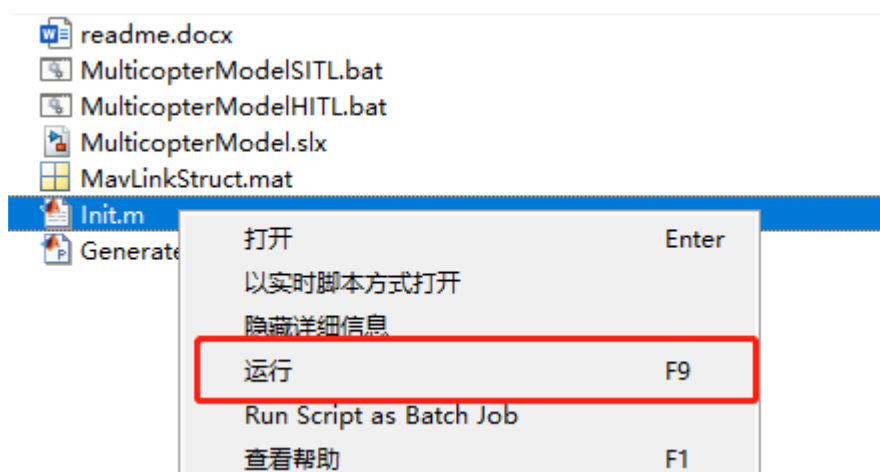
①：推荐配置请见：<https://rflysim.com/>

5. 实验步骤

5.1. 必做实验

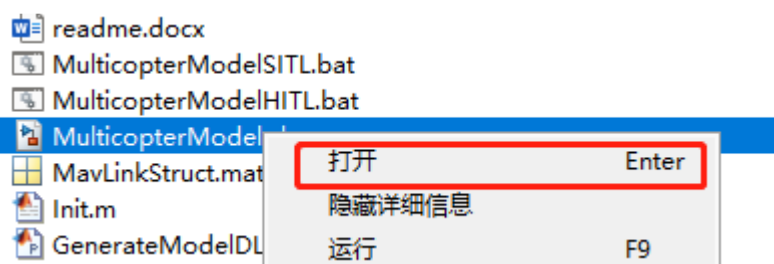
Step 1: 初始化数据

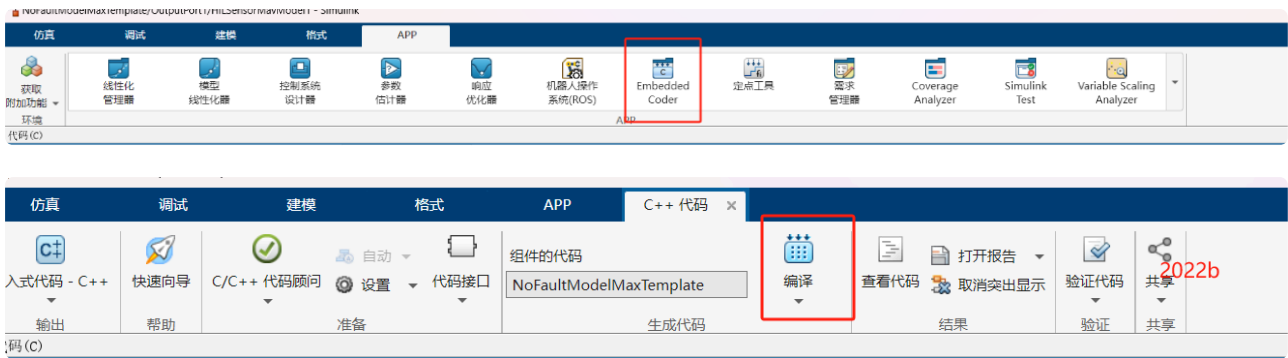
打开“Init.m”文件并运行。



Step 2: 编译文件

打开“MulticopterModel.slx” Simulink 文件，点击Build Model 按钮生成代码。

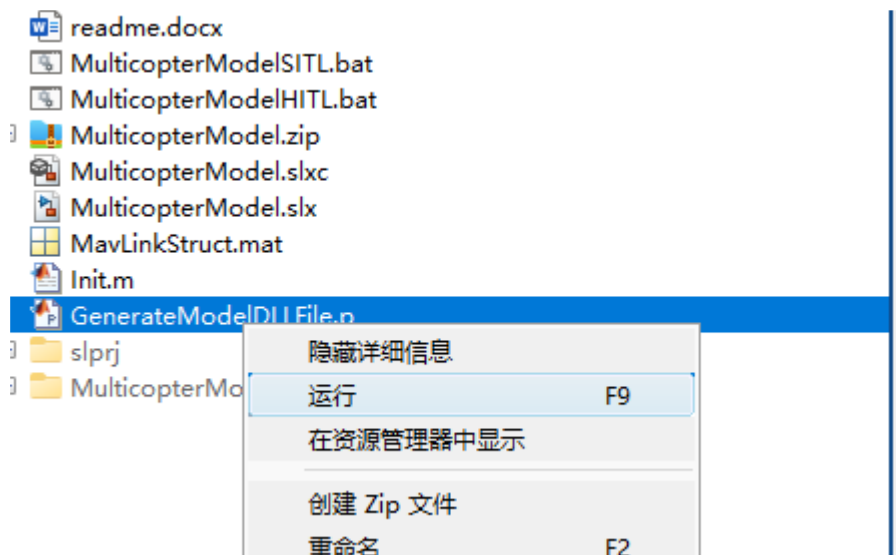




注：如果故障模块版本错误，无法编译，需要从故障模块库中选择对应的模块进行替换。

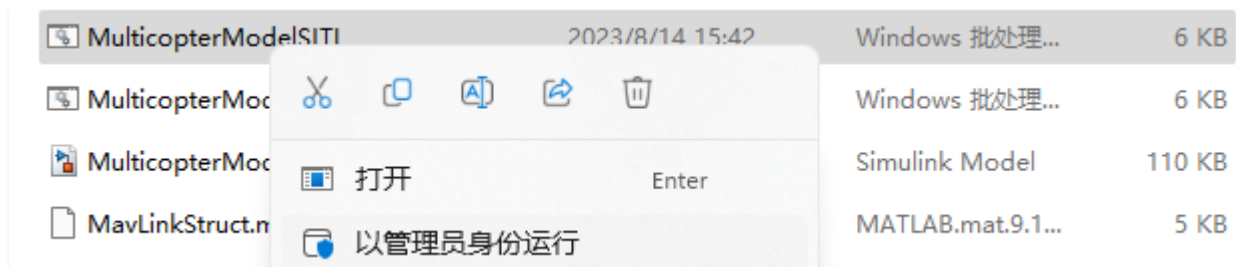
Step 3: 生成dll文件

代码生成完毕后，在 matlab 中右键“GenerateModelDLLFile.p”文件，点击运行，生成 DLL 文件。



Step 4: 运行软件在环仿真

以管理员身份运行软件在环脚本。



Step 5: 打开python文件，修改故障参数

打开Visual Studio

Code，选择打开文件夹，打开文件夹

RflySimAPIs\7.RflySimPHM\0.ApiExps\e4_FaultInjectAPITest_py。



对FaultInjectAPITest.py其中的故障注入代码按照

RflySimAPIs\7.RflySimPHM\0.ApiExps\e4_FaultInjectAPITest_py中的

FaultInjectAPITest_py中的故障注入代码更改为螺旋桨模块故障（螺旋桨模块故障注入代码可以查看参考文献），并对故障参数进行修改。（具体修改方法可以参考e4_FaultInjectAPITest_py文件夹中的readme）

```
62 silInt=np.zeros(8).astype(int).tolist()
63 silFloat=np.zeros(20).astype(float).tolist()
64 silInt[0:2]=[123451,123451]
65 silFloat[0:4]=[0.5,0,0.4,0]
66 # silInt[0:1]=[123542]
67 # silFloat[0:2]=[15,0]
68 mav1.sendSTIntFloat(silInt,silFloat)
```

注：螺旋桨模块故障只有螺旋桨执行效率故障这一种，但是需要设置四个故障参数。

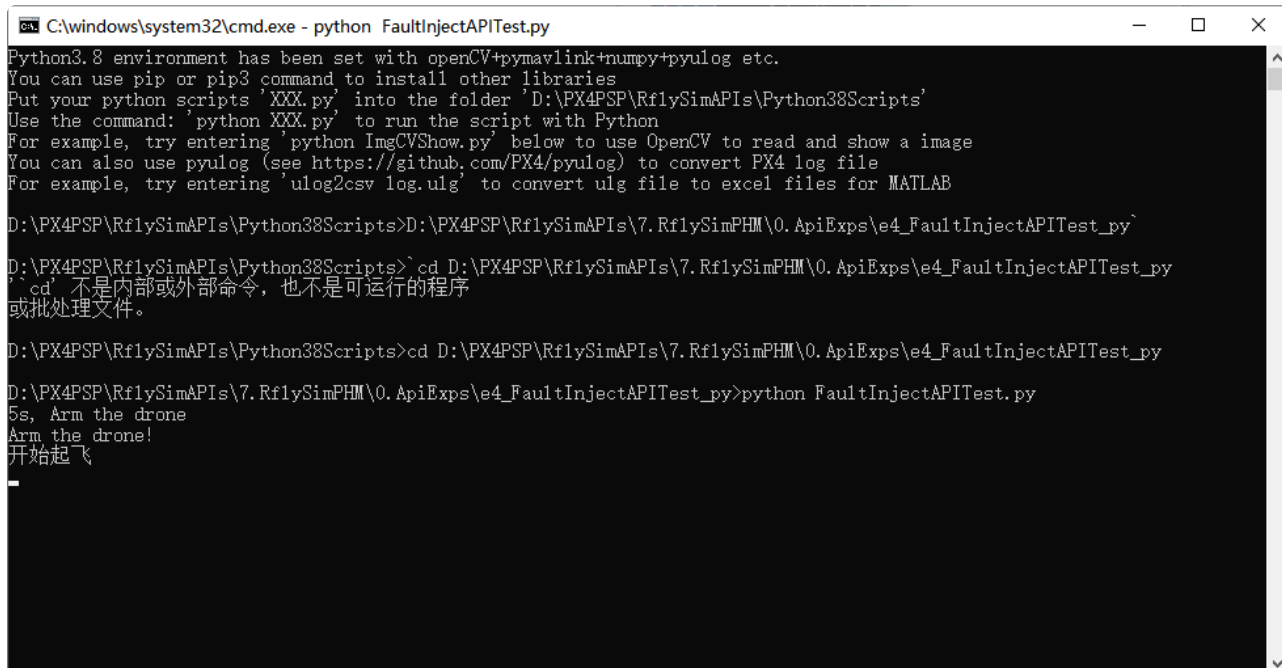
螺旋桨模块故障的设置方法如上图所示，故障ID不能进行更改，但是其故障参数可以自己进行设置。文件中的silFloat已经修改好，可以直接运行注入例程对应故障。

Step 6: 运行python文件实现故障注入

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FaultInjectAPITest.py文件，输入

python FaultInjectAPITest.py，运行FaultInjectAPITest.py。

```
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:\PX4PSP\RflySimAPIs\7.RflySimPHM\0.ApiExps\e4_FaultInjectAPITest.py>python FaultInjectAPITest.py
```



```
C:\windows\system32\cmd.exe - python FaultInjectAPITest.py
Python3.8 environment has been set with openCV+pymavlink+numpy+pyulog etc.
You can use pip or pip3 command to install other libraries
Put your python scripts 'XXX.py' into the folder 'D:\PX4PSP\RflySimAPIs\Python38Scripts'
Use the command: 'python XXX.py' to run the script with Python
For example, try entering 'python ImgCVShow.py' below to use OpenCV to read and show a image
You can also use pyulog (see https://github.com/PX4/pyulog) to convert PX4 log file
For example, try entering 'ulog2csv log.ulg' to convert ulg file to excel files for MATLAB
D:\PX4PSP\RflySimAPIs\Python38Scripts>D:\PX4PSP\RflySimAPIs\7.RflySimPHM\0.ApiExps\e4_FaultInjectAPITest_py
D:\PX4PSP\RflySimAPIs\Python38Scripts> cd D:\PX4PSP\RflySimAPIs\7.RflySimPHM\0.ApiExps\e4_FaultInjectAPITest_py
'cd' 不是内部或外部命令，也不是可运行的程序
或批处理文件。
D:\PX4PSP\RflySimAPIs\Python38Scripts>cd D:\PX4PSP\RflySimAPIs\7.RflySimPHM\0.ApiExps\e4_FaultInjectAPITest_py
D:\PX4PSP\RflySimAPIs\7.RflySimPHM\0.ApiExps\e4_FaultInjectAPITest_py>python FaultInjectAPITest.py
5s, Arm the drone
Arm the drone!
开始起飞
```

然后可以看到无人机已经注入故障，可以看到两个螺旋桨失效，飞机直接坠毁



5.2. 选作实验 (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

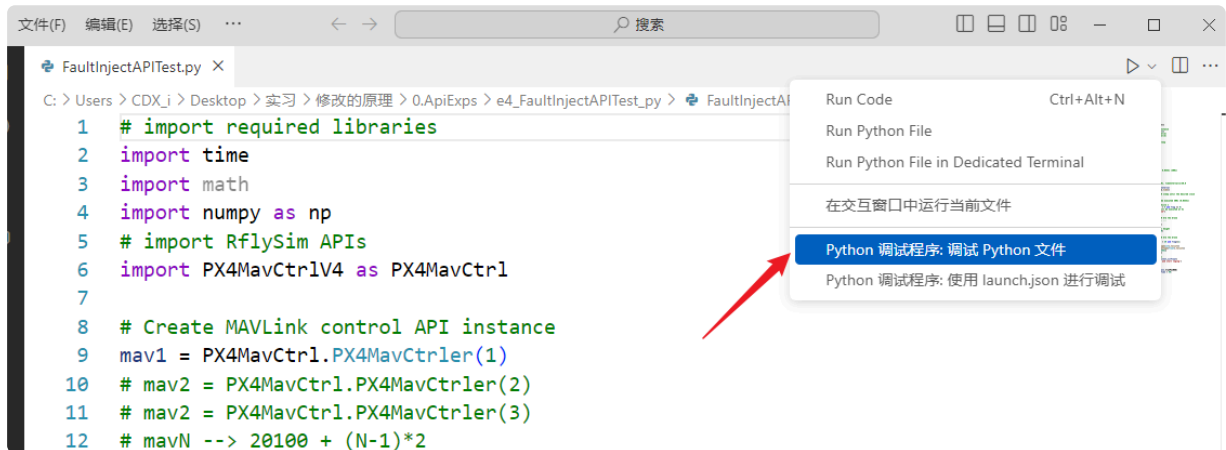
配置VS

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，在运行FaultInjectAPITest.py时，可使用VS Code（或Pycharm等工具）来打开FaultInjectAPITest.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FaultInjectAPITest.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

故障ID及对应参数见<../././RflySimSDK/html/md_phm_2md_2Faultinject.html>

7.常见问题

Q1: ***

A1: ***