

1、实验名称及目的

飞控源码故障注入实验：本实验不在依靠自动生成代码进行故障注入实验，而是直接对源码进行修改，从而实现故障注入的效果。

2、实验效果

本实验通过对源码进行修改，导入到飞控中后，通过硬件在环来显示故障注入的效果。

3、文件目录

文件夹/文件名称	说明
PythonSender udp 版本.zip	故障注入信息发送 udp 版本文件。
PythonSender 串口版本.zip	故障注入信息发送串口版本文件。
飞控源码故障注入	修改过后的飞控源码

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	H7 飞控	1
3	Visual Studio Code		

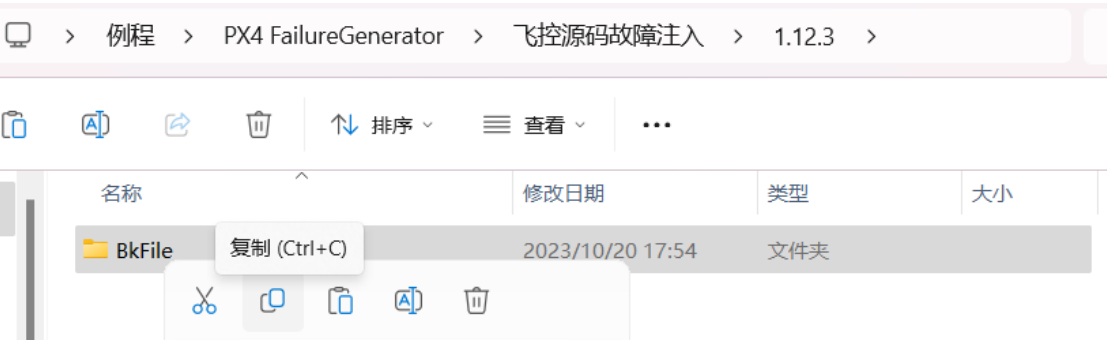
① ：推荐配置请见：<https://doc.rflysim.com/1.1InstallMethod.html>

5、实验步骤

Step 1:

将例程文件下 PX4 FailureGenerator\飞控源码故障注入\1.12.3\BkFile 文件夹整体复制到此路径下 C:\PX4PSP\Firmware，将 BkFile 文件夹进行覆盖替换。

此例程所使用的飞控为卓翼 H7 飞控 1.12.3 版本。





Step 2:

执行./EnvFault.sh 脚本会自动将 Fault 文件中的故障代码进行替换，之后，根据编译命令对源码进行编译。

打开平台中的 Win10WSL 软件。



通过命令进入到 root@DESKTOP-PDPDT4V:/mnt/c/PX4PSP/Firmware/BkFile# 下去

```
root@DESKTOP-PDPDT4V:/mnt/c/PX4PSP/Firmware# ls
CMakeLists.txt  Firmware.sublime-project  README.md  eclipse.cproject  package.xml  test_data
CODE_OF_CONDUCT.md  LICENSE  README.md  eclipse.project  platform  validation
CONTRIBUTING.md  Makefile  appveyor.yml  integrationtest  basic-config  src
CTestConfig.cmake  PULL_REQUEST_TEMPLATE.md  board  launch  test
Documentation  PX4VerRfly.mat  build  mavlink  test

root@DESKTOP-PDPDT4V:/mnt/c/PX4PSP/Firmware# cd BkFile/
root@DESKTOP-PDPDT4V:/mnt/c/PX4PSP/Firmware/BkFile# ls
CMakeLists.txt  EnvCode.sh  EnvCur.sh  EnvFault.sh  EnvOri.sh  Fault  FlightGear
root@DESKTOP-PDPDT4V:/mnt/c/PX4PSP/Firmware/BkFile# ./EnvFault.sh
Copy File ./Fault/Firmware/src/drivers/gps/gps.cpp
Copy File ./Fault/Firmware/src/drivers/pwm_out/PWMOut.cpp
Copy File ./Fault/Firmware/src/drivers/pwm_out/PWMOut.hpp
Copy File ./Fault/Firmware/src/drivers/rc_input/RCInput.cpp
Copy File ./Fault/Firmware/src/drivers/rc_input/RCInput.hpp
Copy File ./Fault/Firmware/src/lib/drivers/accelerometer/PX4Accelerometer.cpp
Copy File ./Fault/Firmware/src/lib/drivers/accelerometer/PX4Accelerometer.hpp
Copy File ./Fault/Firmware/src/lib/drivers/magnetometer/PX4Magnetometer.cpp
Copy File ./Fault/Firmware/src/lib/drivers/magnetometer/PX4Magnetometer.hpp
Copy File ./Fault/Firmware/src/lib/mixer_module/mixer_module.cpp
Copy File ./Fault/Firmware/src/lib/mixer_module/mixer_module.hpp
Copy File ./Fault/Firmware/src/modules/mavlink/mavlink_receiver.cpp
Copy File ./Fault/Firmware/src/modules/mavlink/mavlink_receiver.h
```

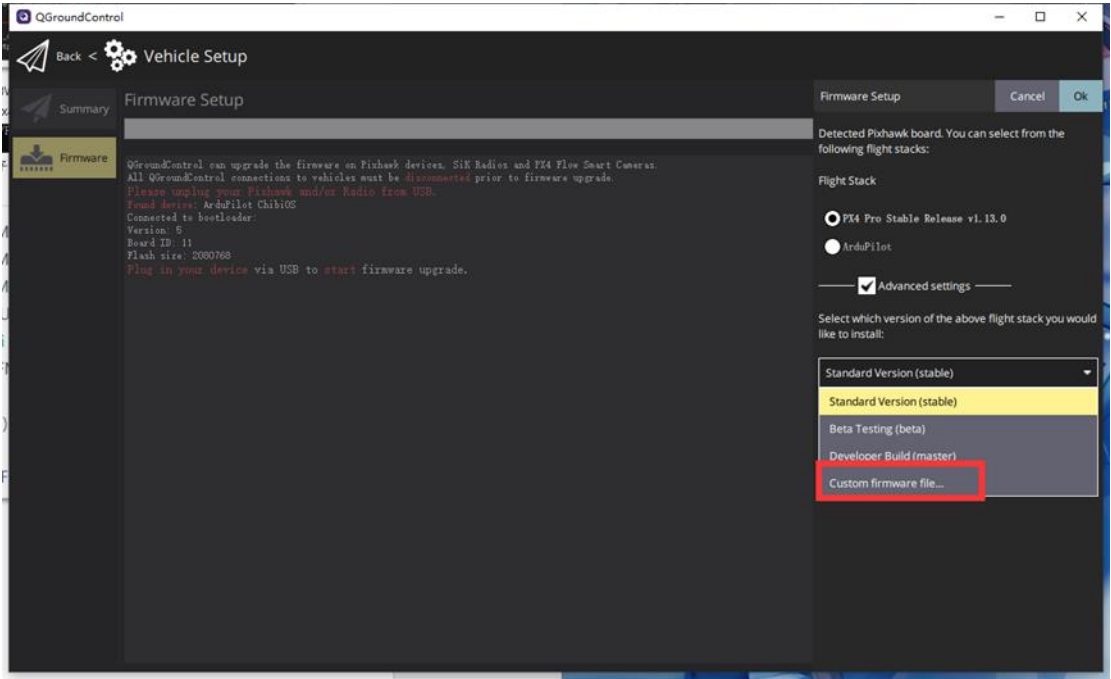
出现 Copy 字样之后，便说明已经复制完成，之后重新打开该软件，输入下图中的编译命令，进行编译。

```
root@DESKTOP-PDPDT4V:/mnt/c/PX4PSP/Firmware# make droneyee_zyfc-h7_default
-- PX4 version: v1.12.3
-- PX4 config file: /mnt/c/PX4PSP/Firmware/boards/droneyee/zyfc-h7/default.conf
```

Step 3:

在编译完成之后，打开 QGC，插入飞控，将编译完成后的固件进行烧录。

生成的固件位于 `PX4PSP\Firmware\build\droneyee_zyfc-h7_default` 中，找到对应的 `px4` 文件。



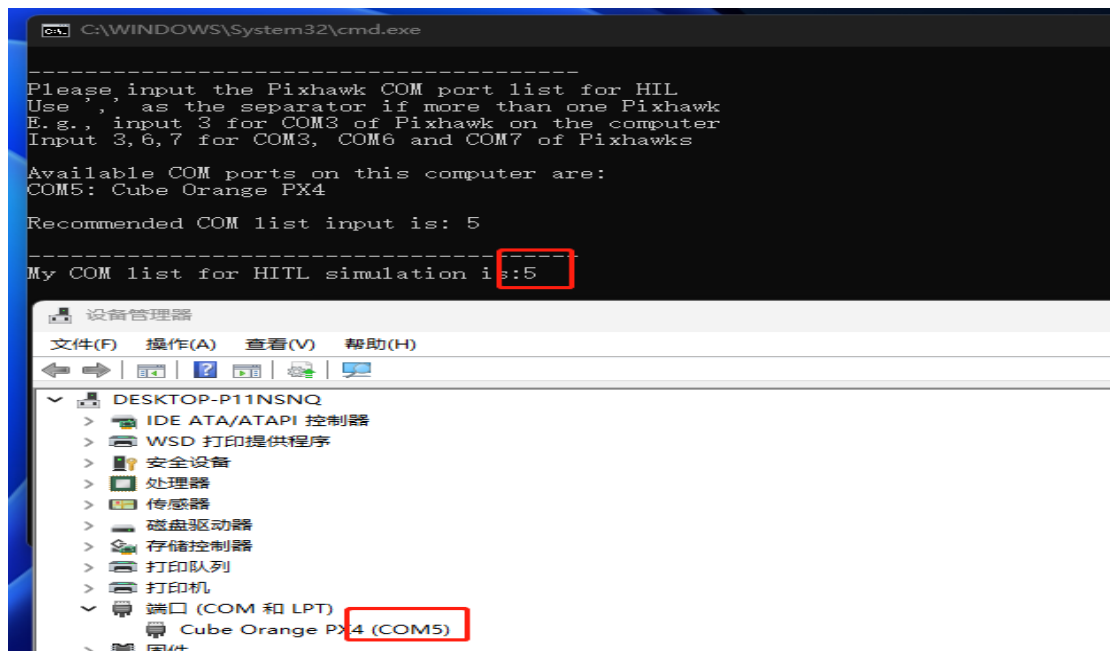
Step 4:

本次演示是通过 `udp` 进行连接飞控。（串口模式下后续操作步骤相同）

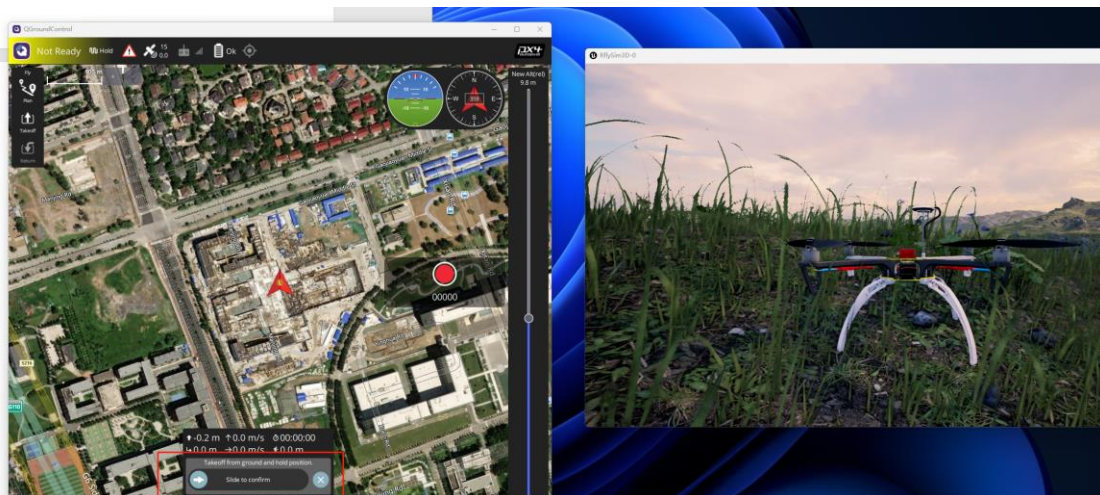
用管理员权限打开 `PythonSender` 文件中的 `bat` 文件。

__pycache__	2023/11/1 14:13	文件夹	
PX4MavCtrlV4.py	2022/8/1 14:09	Python File	99 KB
PythonSender.bat	2022/8/9 20:24	Windows 批处理...	6 KB
PythonSender.py	2022/8/9 20:23	Python File	2 KB
readme.txt	2022/8/9 20:25	文本文档	1 KB

输入连接飞控板的端口数字



在 QGC 中点击起飞飞机。



Step 5:

打开 PythonSender 文件中的 PythonSender.py 文件。

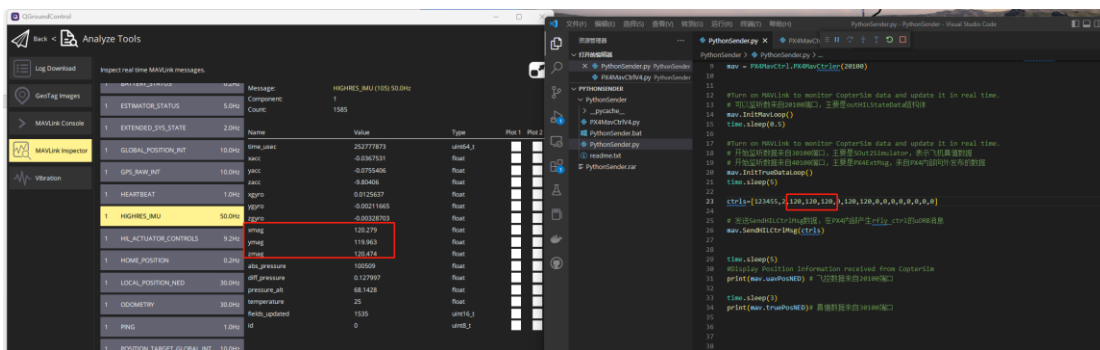
```
(G) 运行(R) 终端(T) 帮助(H) PythonSender.py - PythonSender - Visual Studio Code
PythonSender.py x PX4MavCtrlV4.py
PythonSender > PythonSender.py > ...
9 mav = PX4MavCtrl.PX4MavCtrl(20100)
10
11
12 #Turn on MAVLink to monitor CopterSim data and update it in real time.
13 # 可以监听数据来自20100端口, 主要是outHILStateData结构体
14 mav.InitMavLoop()
15 time.sleep(0.5)
16
17 #Turn on MAVLink to monitor CopterSim data and update it in real time.
18 # 开始监听数据来自30100端口, 主要是SOut2Simulator, 表示飞机真值数据
19 # 开始监听数据来自40100端口, 主要是PX4ExtMsg, 来自PX4内部向外发布的数据
20 mav.InitTrueDataLoop()
21 time.sleep(5)
22
23 ctrls=[123450,2,120,120,120,0,120,120,0,0,0,0,0,0,0]
24
25 # 发送SendHILCtrlMsg数据, 在PX4内部产生rfly_ctrl的uORB消息
26 mav.SendHILCtrlMsg(ctrls)
27
28
29 time.sleep(5)
30 #Display Position information received from CopterSim
31 print(mav.uavPosNED) # 飞控数据来自20100端口
32
33 time.sleep(3)
34 print(mav.truePosNED)# 真值数据来自30100端口
35
```

进行故障注入的参数 (重点):

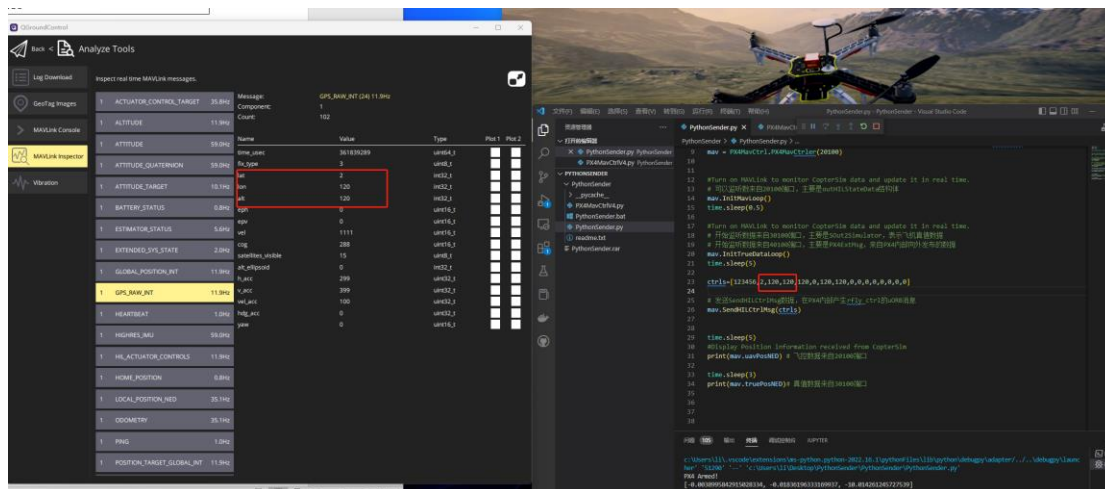
红色框为我们在 msg 文件中设置的 16 位控制数, 其中蓝色划线为 ID 端口

类型	ID 号	备注
地磁	123455	第 0 位是 ID, 第一位是选择位 (若为 1, 则是赋值方式, 若为 2, 则是叠加方式), 2, 3, 4 位是参数
GPS	123456	第 0 位是 ID, 第一位是选择位, 2, 3, 4 位是参数
遥控器	123457	第 0 位是 ID, 第一位是选择位, 后面是参数, 根据遥控输出个数定
电机	123450	第 0 位是 ID, 第一位是选择位, 后面是参数, 参数个数根据电机输出个数定
加速度计	123542	第 0 位是 ID, 第一位是选择位, 后面是参数, 参数设定是 X Y Z

地磁故障注入



GPS 故障



6、参考文献

【1】 无