1. 实验名称及目的

1.1. 实验名称

基于数字孪生与深度学习的无人机故障诊断。（仅限完整版及以上版本）

1.2. 实验目的

本实验旨在探讨数字孪生技术与深度学习方法在无人机故障诊断中的应用。

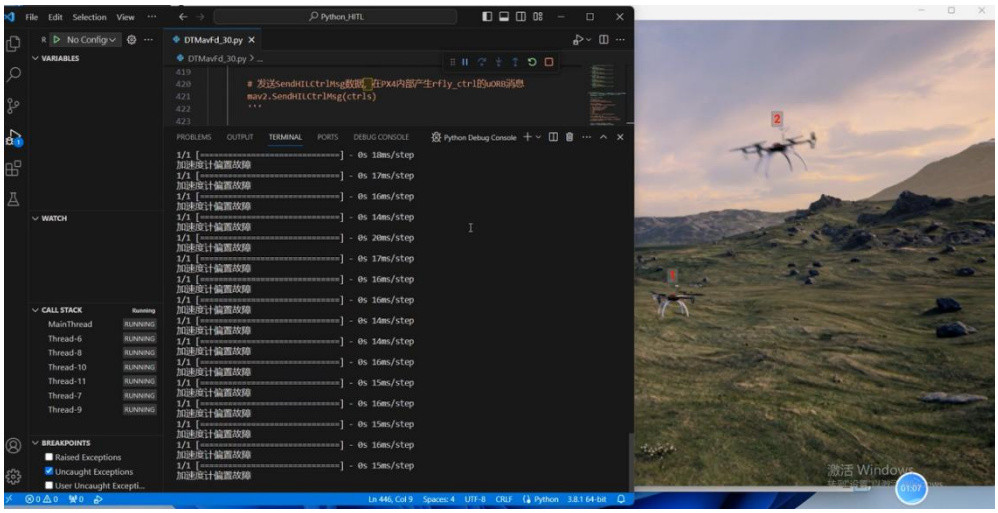
1.3. 关键知识点

故障注入方式：将故障建模的模型导出为 DLL 文件，再通过 CopterSim 加载 DLL 文件，最后通过 udp 模式（python/matlab 形式）注入故障码进行故障注入仿真。

实验原理：物理无人机与孪生机处于同步运行状态，此时对物理无人机进行故障注入，将监测到的数字孪生模型输出的孪生输出与无人机输出的实际输出，通过数据预处理得到残差数据，再输入到深度学习算法构建的故障诊断模型中，实现对无人机故障的诊断与定位。

2. 实验效果

CopterSim 导入 DLL 文件后，注入故障，故障诊断模型实时诊断，并打印诊断结果。



3. 文件目录

例程目录：***\RflySimAPIs\7.RflySimPHM\3.CustExps\e1_DigitalTwinExp\

文件夹/文件名称	说明
data 文件夹	仿真数据存放处
src 文件	源码存放处

4. 运行环境

序号	软件要求	硬件要求	
		名称	数量(个)
1	Windows 10 及以上版本	笔记本/台式电脑①	1
2	Visual Studio Code		

3	RflySim 工具链		

① ：推荐配置请见：HowToInstall.pdf (rflysim.com)

5. 实验步骤

5.1. 必做实验

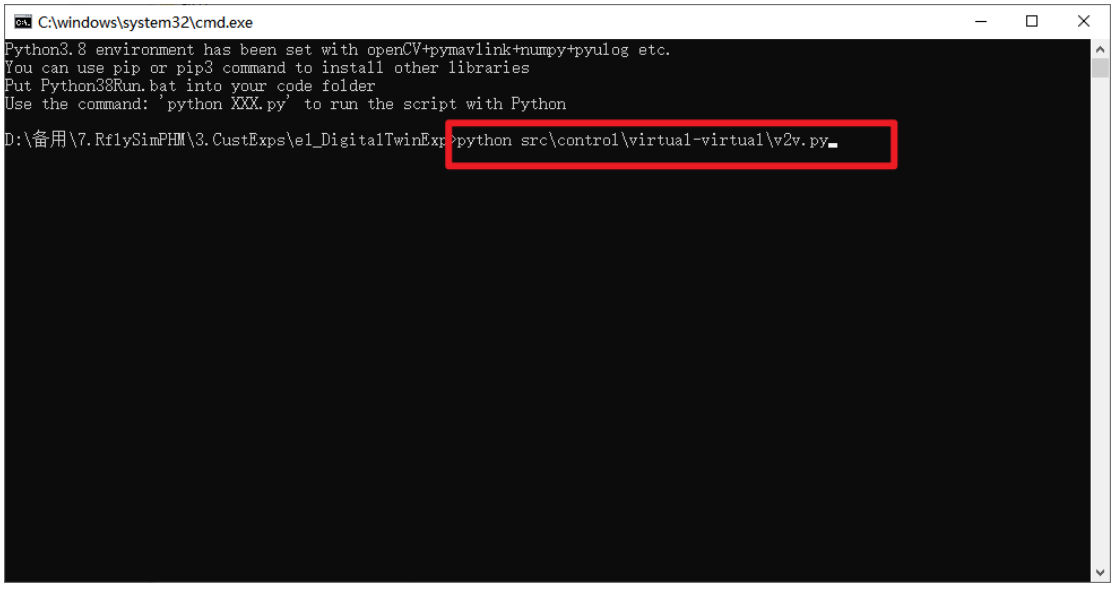
Step 1：修改程序路径

根据安装路径修改"e1_DigitalTwinExp\src\control\utils\RflyDP.py"第 189 行。

```
184         if logName!= '':
185             shutil.copyfile(qgc.QGCPath+'\\'+logName, self
186             print('Download log ' + logName+ f' for mav{se
187         '''
188         DataREG.lock.acquire()
189         self.PlatFormpath = 'C:/PX4PSP'
190         # 1、List the directories under the file
191         log_path = self.PlatFormpath + f'/Firmware/build/px4_s
192 
193         PlatForm_log_dirs = os.listdir(log_path)
```

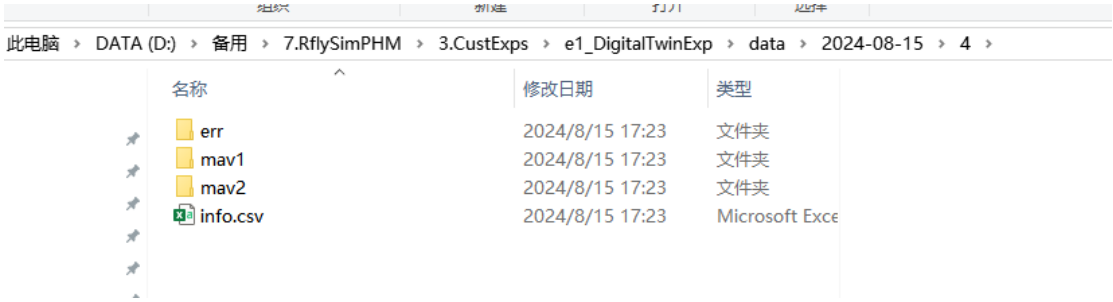
Step 2：运行 python 程序

在文件夹下，双击 Python38Run.bat，打开集成好的 python 环境，打开\src\control\virtual-virtual，在该环境下运行 v2v.py 文件，输入 python v2v.py，运行 v2v.py。



Step 3：查看数据

仿真结束后，在 data 文件夹下存放了此次仿真的飞行数据



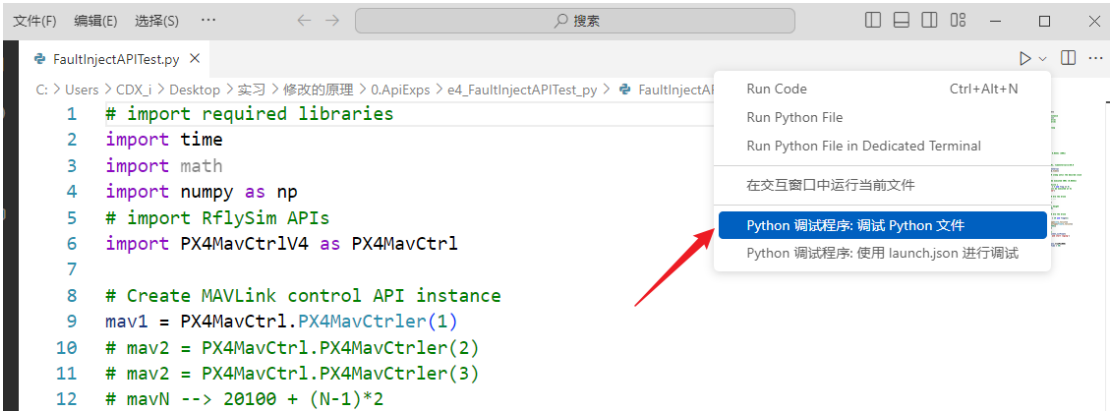
5.2. 选作实验（VS Code 调试运行）

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 v2v.py 时，可使用 VS Code（或 Pycharm 等工具）来打开 v2v.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 v2v.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]. 无

7. 常见问题

Q1：无

A1：无