**实验一：电机故障建模与注入（多旋翼为例）**

**实验目的：掌握电机故障建模与注入的使用**。

**实验步骤：**

1. 本实验例程在文件src/examples/ 3.MotorFaultModel内，文件结构如下所示：

|  |  |
| --- | --- |
| File | Meaning |
| Init.m | Model initialization file |
| MavLinkStruct.mat | Mavlink structure file data |
| MulticopterModelLib.slx | Multi-rotor model library file |
| NoFaultModelMaxTemplate.slx | Maximum fault-free template model |
| GenerateModelDLLFile.p | Dll model automatically generates files |
| NoFaultModelMaxTemplateHITL.bat | Maximum fault-free template model hardware-in-the-loop one-click script |
| NoFaultModelMaxTemplateSITL.bat | Maximum fault-free template model software-in-the-loop one-click script |

2） 在src\examples\ 3.MotorFaultModel\Init.m中设计电机故障ID和故障参数结构体，在模型文件src\examples\ 3.MotorFaultModel\ MulticopterModel.slx订阅故障消息，并发布为bus结构体。



3） 编写电机故障模块，设计故障算法



4） 封装故障模块并形成故障模块库

全选步骤2的所有框图，右击🡪基于所选内容创建子系统，创建完毕后右击所创建的子系统🡪封装🡪创建封装，将创建的库在相同路径下另存为一个\*\*\*Lib.slx文件



5） 在MulticopterModel.slx的电机模块调用编写好的故障库



6） 编译模型，自动生成代码，生成MulticopterModel.zip包



7） 右键运行GenerateModelDLLFile.p，生成动态链接库MulticopterModel.dll文件

8） 将生成的MulticopterModel.dll拷贝到PX4PSP\CopterSim\external\model文件夹中

9） 打开src\examples\3.MotorFaultModel\MulticopterModelSITL.bat，并等待其连接正常

10） VSCode打开src\examples\ 3.MotorFaultModel \PythonSender文件夹运行FaultInjectAPI.py，观察故障注入效果

