

## 1. 实验名称及目的

平台建模模板之最小模板使用介绍：该例程对如何使用平台最小模板进行软/硬件在环仿真进行介绍，其中最小模版为平台满足仿真所需的最简化模型。

## 2. 实验原理

Exp1\_MinModelTemp.slx 是最小系统的模版，包含了最低要求的输入和输出接口

### 2.1. 模型参数介绍（参考 [API.pdf 中 DLL/SO 模型与通信接口的重要参数部分](#)）

#### 1) 重要参数

Exp1\_MinModelTemp\_init.m 中定义了模型的各种参数，针对最小模板需要配置以下几个关键数据。

飞机的三维显示样式

```
ModelParam_uavType = int16(3); %这个参数决定了飞机的三维显示样式，需要和 RflySim3D 的 XML 文件中的 ClassID 相匹配
```

飞机的初始位姿参数

```
ModelInit_PosE=[0,0,0]; %用于设置飞机的初始位置，对应了 CopterSim 上的 X 和 Y 初始值。最小模板的 Z 初始值会由 terrainZ 输入接口限制为 0
```

```
ModelInit_AngEuler=[0,0,0]; %用于设定飞机的初始姿态。飞机姿态角的前两位（俯仰和滚转角）可以通过 ModelInit_AngEuler 参数来配置，但是偏航角需要在 CopterSim 中配置。
```

QGC 中显示的地图坐标和高度原点

```
ModelParam_GPSLatLong = [40.1540302 116.2593683];%飞机初始的纬度和精度，单位度。
```

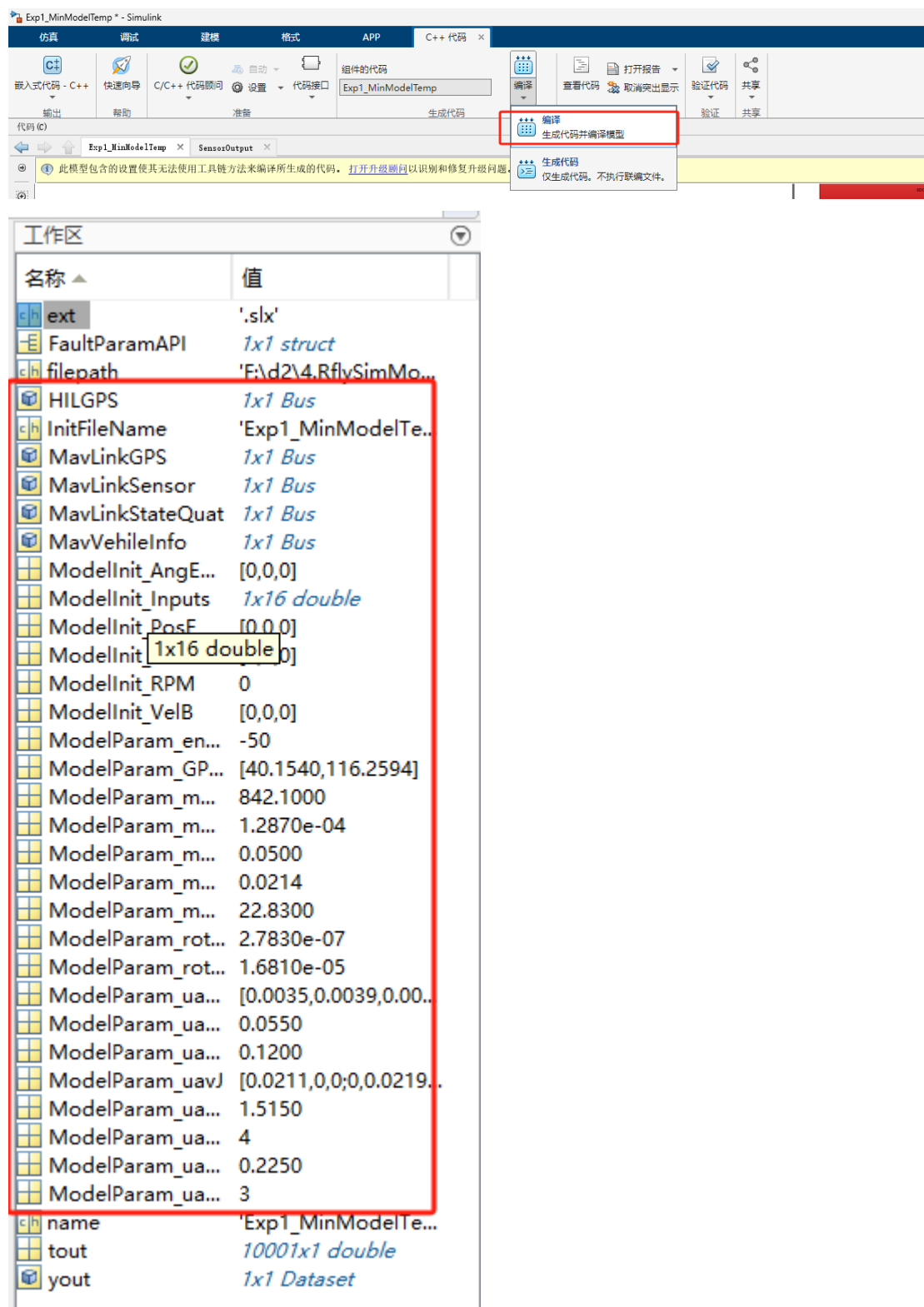
```
ModelParam_envAltitude = -50;%原点的海拔高度，竖直向下为正，高于海平面填负值，单位米。
```

#### 2) 参数调用过程

Exp1\_MinModelTemp.slx 是 DLL 模型生成的最小文件，模型启动运行（编译）时会调用 Exp1\_MinModelTemp\_init.m



Exp1\_MinModelTemp\_init.m 中包含了模型的参数信息，本脚本会在 Exp1\_MinModelTemp.slx 编译（编译所需环境配置参考 [API.pdf 中的环境配置](#)）时被调用将参数载入 MATLAB 工作空间。



GenerateModelDLLFile.p 是将 slx 模型转化为 DLL 模型文件的脚本，使用 RflySim 平台进行载具软硬件在环仿真时，需要将 DLL(windows 下)/SO(Linux 下)模型导入到 CopterSim，形成运动仿真模型，因此，在 Simulink 模型编译完成后，需要将模型对应的 C++文件打包成 DLL/SO 模型。

---

## 2.2. 输入信号（参考 [API.pdf](#) 中 [DLL/SO 模型与通信接口的数据协议部分](#)）

最小模板必须的两个输入数据包括电机控制量和地形数据

### 1) 电机数据 inPwms

输入接口 inPWMs, 16 维执行器控制量输入, 已归一化到-1 到 1 尺度(通常电机是 0-1, 舵机是 -1~1), 它的数据来自飞控回传的电机控制 MAVLink 消息 mavlink\_hil\_actuator\_controls\_t 的 controls, 具体定义如下:

```
typedef struct __mavlink_hil_actuator_controls_t {  
    uint64_t time_usec; //时间戳, 从开机后的时间, 单位 ms  
    uint64_t flags; //标志位, 用于显示当前的飞行状态  
    float controls[16]; //控制量, 16 维电机的控制量, 发送到模型中, 驱动飞机飞行  
    uint8_t mode; //模型, 用于显示飞机当前的飞行模式和是否上锁等信息 }  
mavlink_hil_actuator_controls_t;
```

软件在环仿真时, 电机控制指令从 PX4 SITL 控制器通过 TCP4561 系列端口以 MAVLink 协议发送到运动仿真模型的 inPWMs 接口, 而硬件在环仿真时, 该指令是从飞控通过串口以 MAVLink 协议发送到运动仿真模型的 inPWMs 接口。

### 2) 地形高度 terrainZ

最小模板默认 terrainz 值为 0, 故只能使用平坦地形仿真

## 2.3. 模型模块（参考 [API.pdf](#) 中的 [Simulink 建模模板介绍](#)）

推荐不改变本模型模版, 而是在其中修改模型参数(质量、转动惯量等)和电机模块(转动动态响应)与力和力矩模块(驱动力、气动力、地面支撑和阻力等)两个自定义模块, 来适配不同的载具的模型。

### 1) Motor Model 电机模块

在该模块中输入为 PWM 值(通过 inPWMs 接口获取), 经过各电机的非线性动力学模型后得到各电机转速, 该模块的输出分别为输入给力和力矩模型的电机转速(弧度每秒); 输入给 UE 的电机转速(转每分)

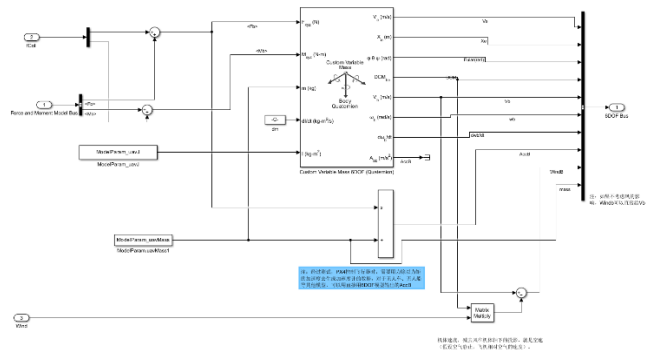
### 2) Force and Moment Model 力和力矩模块

该模块输入为电机转速 MotorRads、飞机运动学姿态 6DOF 和地形高度输入 TerrainZ, 输出为多旋翼合力、合力矩 Force and Moment Model Bus。

### 3) 6DOF 六自由度刚体运动学模块

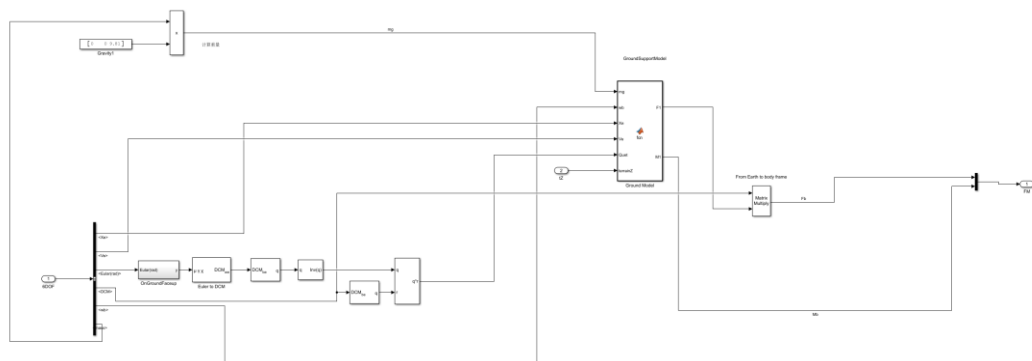
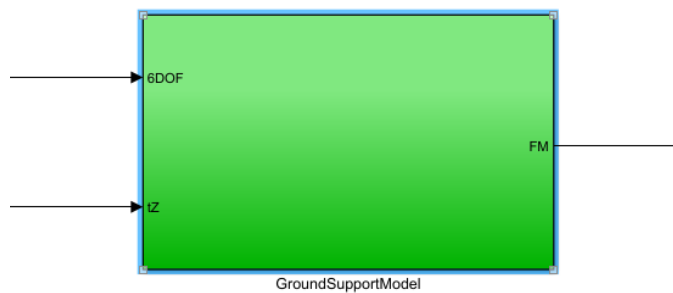
用于描述无人机在空中运动时的姿态和位置变化。考虑了无人机在三个坐标轴上的旋转运动(俯仰、横滚和偏航)以及机体与地球坐标系上的平移运动(前后、左右和上下)。

还可以根据实际需求对模型进行扩展, 考虑更多的因素, 如飞行器的非线性特性、气动力和惯性矩等。



#### 4) GroundSupportModel 地面支撑模块

GroundSupportModel 地面支撑模块实际上是 [PhysicalCollisionModel 碰撞检测模块](#) 的一个子模块，这里将所有物体简化为较为简单的基本几何体（例如圆柱体或者长方体）来计算其与地面之间的物理接触受力。



#### 5) SensorOutput 传感器输出模块

该模块中包括了环境模型、传感器模型和 GPS 模型

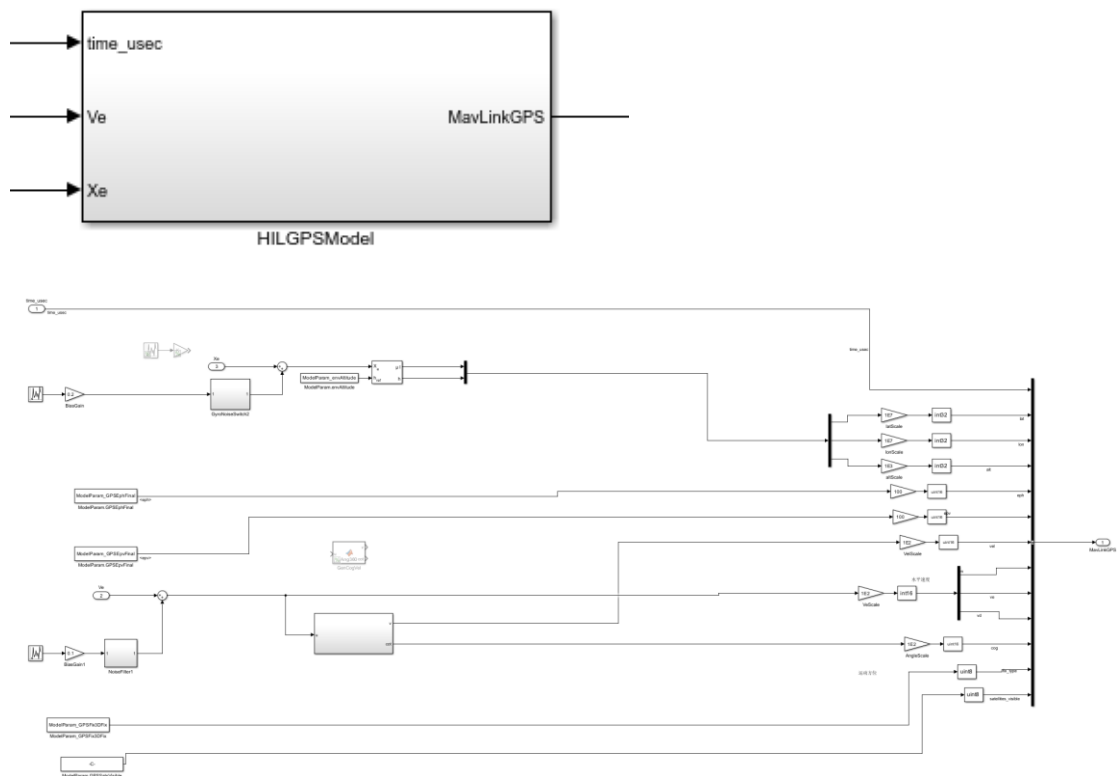
##### 环境模型

环境模型对重力和大气压强对无人系统飞行产生的影响进行了模拟



## GPS 模型

GPS 模型用于计算 GPS 数据，在仿真时反馈回 PX4 控制器



## 6) 3DOutput 三维显示模块

该模块会将 `***_init.m` 中的 `ModelParam_uavType`（三维显示 ID）、来自电机模型的 `ActuatorToUE` 以及来自 6DOF 模型的 6DOF Bus 的位置、速度、姿态和加速度等输出为 `MavVehile3DInfo`，并按协议对输入信息进行数据打包后通过该接口将数据发送至三维引擎

## 2.4. 输出信号（参考 [API.pdf 中的数据协议](#)）

`***_init.m` 会调用 `MavLinkStruct.mat` 导入四个输出结构体 `bus`（`MavLinkGPS`、`MavLinkSensor`、`MavLinkStateQuat` 以及 `MavVehileInfo`）的定义。

```
load MavLinkStruct;
%load path;
```

最小模型模版包含了三个输出信号，分别是 `MavHILSensor`、`MavHILGPS`、`MavVehile3DInfo`。

### 1) MavHILSensor（传感器接口集合）

模型发送给 `RflySim3D` 的真实仿真数据，是平滑的理想值，这些数据可用于 `Simulink` 下的飞控与模型进行软件仿真测试。对应了 MAVLink 的 `mavlink_hil_sensor_t` 消息，本结构体包含了，加速度传感器的加速度值、陀螺仪传感器的角速度值、磁罗盘传感器的磁场值，气压和空速传感器的气压值等。这些传感器的值在仿真时由我们的模型提供，在真机飞行时

由真实传感器芯片提供。

```
typedef struct __mavlink_hil_sensor_t {
    uint64_t time_usec; /*时间戳, 单位毫秒 ms*/
    float xacc; /*机体坐标系 x 方向加速度, 单位 m/s^2 */
    float yacc; /*机体坐标系 y 方向加速度, 单位 m/s^2 */
    float zacc; /*机体坐标系 z 方向加速度, 单位 m/s^2 */
    float xgyro; /*机体坐标系 x 方向角加速度, 单位 rad/s */
    float ygyro; /*机体坐标系 y 方向角加速度, 单位 rad/s */
    float zgyro; /*机体坐标系 z 方向角加速度, 单位 rad/s */
    float xmag; /*机体坐标系 x 方向磁通量, 单位 Gauss =T/10000*/
    float ymag; /*机体坐标系 y 方向磁通量, 单位 Gauss =T/10000*/
    float zmag; /*机体坐标系 z 方向磁通量, 单位 Gauss =T/10000*/
    float abs_pressure; /*绝对气压值, 单位 millibar=100Pa*/
    float diff_pressure; /*相对气压值, 单位 millibar=100Pa*/
    float pressure_alt; /*气压解算高度值, 单位 m*/
    float temperature; /*温度, 单位摄氏度*/
    uint32_t fields_updated; /*传感器参数初始化标志位, bit 0 = xacc, bit 12: temperature, bit 31:
全部重新初始化 */
} mavlink_hil_sensor_t;
```

## 2) MavHILGPS (GPS 接口)

模型发送给飞控的 GPS 数据值, 它对应了 MAVLink 消息的 mavlink\_hil\_gps\_t 结构体。输出信号中包含了经纬高、水平竖直精度、地速、北东地的速度、偏航角、定位状态和卫星数量等数据。这些传感器的值在仿真时由我们的模型提供, 在真机飞行时由真实 GPS 模块提供。

```
typedef struct __mavlink_hil_gps_t {
    uint64_t time_usec; /*时间戳, 单位毫秒 ms*/
    int32_t lat; /*纬度(WGS84 地球模型), 单位度, 再乘以 1E7*/
    int32_t lon; /*经度(WGS84 地球模型), 单位度, 再乘以 1E7*/
    int32_t alt; /*高度 (AMSL 地球模型, 而不是 WGS84), 单位 m, 再乘以 1000 (向上为正)*/
    uint16_t eph; /*GPS 水平方向定位精度, 单位 cm, 如果不知道设为 65535*/
    uint16_t epv; /*GPS 垂直方向定位精度, 单位 cm, 如果不知道设为 65535*/
    uint16_t vel; /*GPS 地速, 单位 cm/s, 如果不知道设为 65535*/
    int16_t vn; /*GPS 地速朝北方向分量, 单位 cm/s */
    int16_t ve; /*GPS 地速朝东方向分量, 单位 cm/s */
    int16_t vd; /*GPS 地速朝下方向分量, 单位 cm/s */
    uint16_t cog; /*运动方向, 单位和范围 0~359.99 度, 再乘以 100 degrees * 100, 如果不知道设为 65535*/
    uint8_t fix_type; /*定位类型 0-1: no fix, 2: 2D fix, 3: 3D fix. */
    uint8_t satellites_visible; /*可见卫星数, 如果不知道设为 255*/
} mavlink_hil_gps_t;
```

注: GPS 数据的发送频率与真实传感器硬件基本相同为 10Hz, 因此飞控的实时位置并不能靠 GPS 直接提供, 需要与 IMU 等传感器进行融合滤波估计得到。

## 3) MavVehicle3Dinfo (真实仿真数据输出)

模型发送给飞控的各种传感器数据的集合, 对应了 MAVLink 的 mavlink\_hil\_sensor\_t 消息。输出信号中包括了加速度传感器的加速度值、陀螺仪传感器的角速度值、磁罗盘传感器的磁场值, 气压和空速传感器的气压值等。

```
struct SOut2Simulator {
    int copterID; //飞机 ID, 用于区分局域网内不同飞机
```

```
int vehicleType; //飞机样式，区分同种飞机（如四旋翼）下的不同样式（例如，大疆、AR.Drone）
double runnedTime; //时间戳，当前时刻的时间，单位毫秒
float VelE[3]; //速度向量，地球坐标系的xyz速度（z向下为正），单位m/s
float PosE[3]; //位置向量，地球坐标系下的xyz方向（z向下为正，单位m，以起飞点为坐标原点
float AngEuler[3]; //姿态角，飞机的欧拉角，定义于机体坐标系，单位弧度
float AngQuatern[4]; //四元数，飞机姿态的四元数，定义于机体坐标系
float MotorRPMS[8]; //电机转速，飞机的各个旋翼转速，单位转每分
float AccB[3]; //加速度，飞机的运动加速度，单位m/s^2
float RateB[3]; //角速度，飞机的转动角速度，单位rad/s
double PosGPS[3]; //GPS坐标，飞机的经纬高坐标，单位度、度、米
};
```

### 3. 实验效果

实现最小模型 DLL 模型文件生成，以及完成最小模型无人机软/硬件在环仿真。



### 4. 文件目录

文件夹/文件名称	说明
Exp1_MinModelTemp.slx	四旋翼机模型文件。
MavLinkStruct.mat	包含数据结构体。
Exp1_MinModelTemp_SITL.bat	软件在环仿真批处理文件。
Exp1_MinModelTemp_HITL.bat	硬件在环仿真批处理文件。
GenerateModelDLLFile.p	DLL 格式转化文件。
Exp1_MinModelTemp_init.m	四旋翼动力模型相关参数。

### 5. 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 平台免费版	Pixhawk 6C <sup>②</sup>	1
3	MATLAB 2017B 及以上 <sup>③</sup>	数据线	1

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

② 平台安装时的推荐编译命令为：px4\_fmu-v6c\_default.px4，PX4 固件版本为：1.13.3。其他配套飞控请见：<http://doc.rflysim.com/hardware.html>。

## 6. 实验步骤

### 6.1. 软件在环仿真

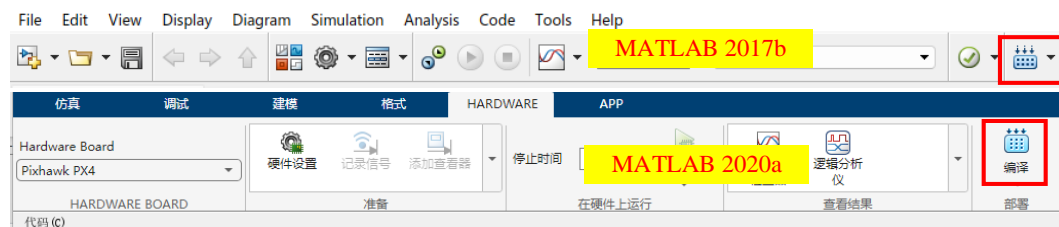
#### Step 1:

打开“Exp1\_MinModelTemp.slx”文件，点击编译（“Build Model”）按钮。

名称	修改日期	类型	大小
Exp1_MinModelTemp.slx	2023/10/25 16:26	Simulink Model	66 KB
Exp1_MinModelTemp_HITL.bat	2023/10/24 15:33	Windows 批处理...	6 KB
Exp1_MinModelTemp_init.m	2023/10/24 15:33	Objective C 源文件	3 KB
Exp1_MinModelTemp_SITL.bat	2023/10/24 15:33	Windows 批处理...	6 KB
GenerateModelDLLFile.p	2023/10/17 15:11	MATLAB.p.9.14.0	6 KB
MavLinkStruct.mat	2023/10/17 15:11	MATLAB Data	5 KB
Readme.docx	2023/10/17 15:11	Microsoft Word ...	3,848 KB
Readme.pdf	2023/10/24 15:33	Foxit PhantomP...	528 KB
MulticopterModel.zip	2023/10/25 16:26	压缩(zipped)文件...	100 KB
Exp1_MinModelTemp.dll	2023/10/25 16:26	应用程序扩展	226 KB

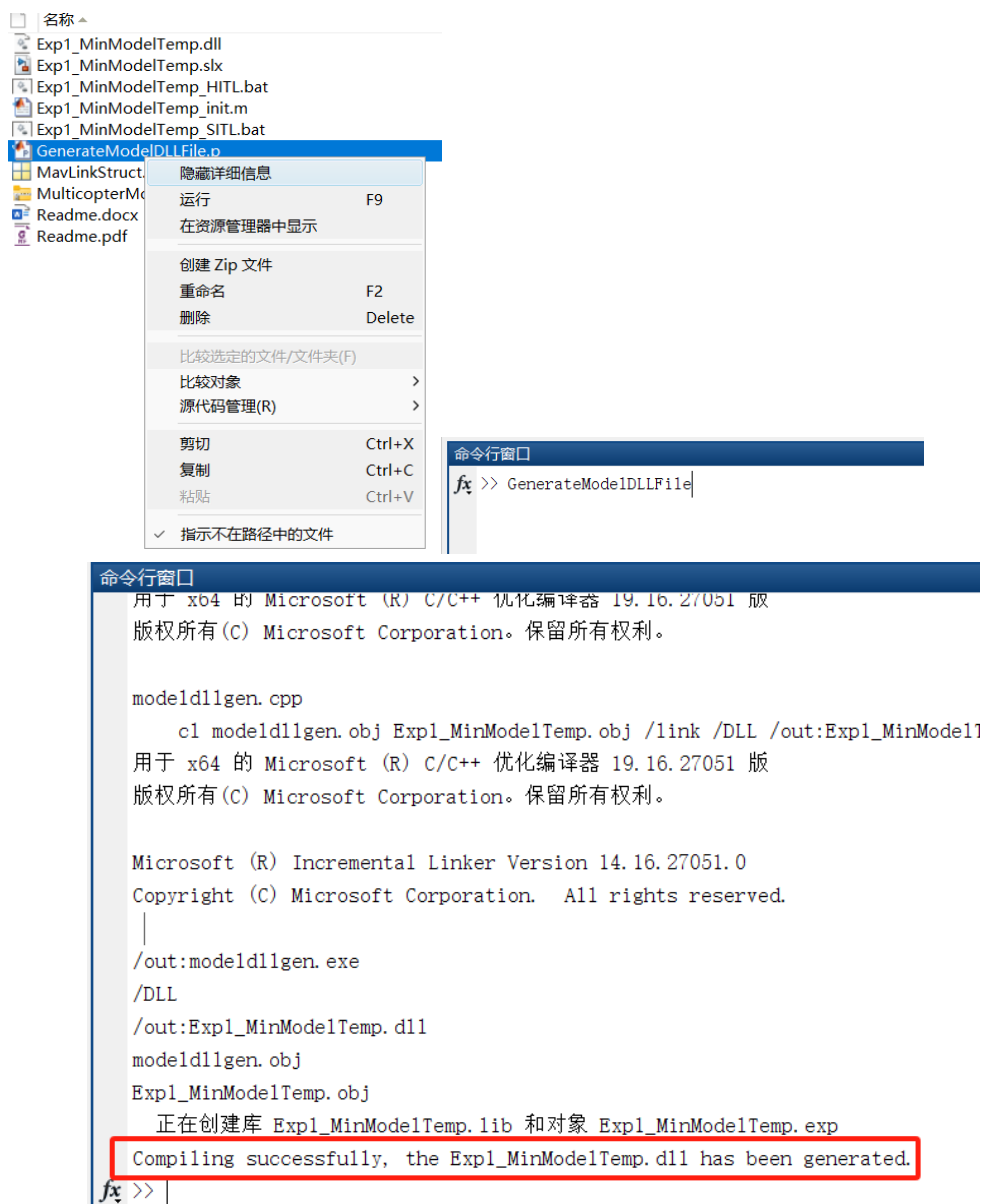
与常规载具模型先比，最小模型特点如下：

在 Exp1\_MinModelTemp.slx 的 Motor Model 中，只有 inPWMs 与 TerrainZ 两个系统输入，MavHILSensor、MavHILGPS 与 MavVehile3DInfo 三个系统输出；相较于其他模型特别是最大模型 Exp2\_MaxModelTemp.slx 可实现功能较少。













#### Step 2:

编译完成后，右键 GenerateModelDLLFile.p 并点击运行（或者在 MATLAB 的命令窗口中输入 GenerateModelDLLFile 后回车），即可以得到“Exp1\_MinModelTemp.dll”的 DLL 模型文件。




### Step 3:

以管理员身份运行“Exp1\_MinModelTemp\_SITL.bat”脚本，在提示框中输入“1”后并回车，系统将自动打开 RflySim 平台，并完成软件在环仿真所有需要配置。

名称	修改日期	类型	大小
 Exp1_MinModelTemp.slx	2023/10/25 16:26	Simulink Model	66 KB
 Exp1_MinModelTemp_HITL.bat	2023/10/24 15:33	Windows 批处理...	6 KB
 Exp1_MinModelTemp_init.m	2023/10/24 15:33	Objective C 源文件	3 KB
 Exp1_MinModelTemp_SITL.bat	2023/10/24 15:33	Windows 批处理...	6 KB
 GenerateModelDLLFile.p	2023/10/17 15:11	MATLAB.p.9.14.0	6 KB
 MavLinkStruct.mat	2023/10/17 15:11	MATLAB Data	5 KB
 Readme.docx	2023/10/17 15:11	Microsoft Word ...	3,848 KB
 Readme.pdf	2023/10/24 15:33	Foxit PhantomP...	528 KB
 MulticopterModel.zip	2023/10/25 16:26	压缩(zipped)文件...	100 KB
 Exp1_MinModelTemp.dll	2023/10/25 16:39	应用程序扩展	226 KB

 选择 C:\WINDOWS\System32\cmd.exe
 

```

已复制      1 个文件。

-----
Please input UAV swarm number:1

```

注：在“Exp1\_MinModelTemp\_SITL.bat”软件在环的脚本文件中，需要设置对应无人车的 DLL 名：

```

REM Set use DLL model name or not, use number index or name string
REM This option is useful for simulation with other types of vehicles instead of multicopters
set DLLModel=Exp1_MinModelTemp

```

在 SimMode 处选择 CopterSim 中对应的软件在环仿真模式：

```

REM Set the simulation mode on CopterSim, use number index or name string
REM e.g., SimMode=2 equals to SimMode=PX4_SITL_RFLY
set SimMode=2

```

在机架设置处设置四旋翼的对应机架，若不设置对应机架则仿真的默认机架则为四旋翼：

```

REM Set the vehicle-model (airframe) of PX4 SITL simulation, the default airframe is a quadcopter: iris
REM Check folder Firmware\ROMFS\px4fmu_common\init.d-posix (or init.d/airframes) for supported airframes (Note: You can also create your a
REM E.g., fixed-wing aircraft: PX4SITLFrame=plane; small cars: PX4SITLFrame=rover
set PX4SITLFrame=iris

```

## Step 4:

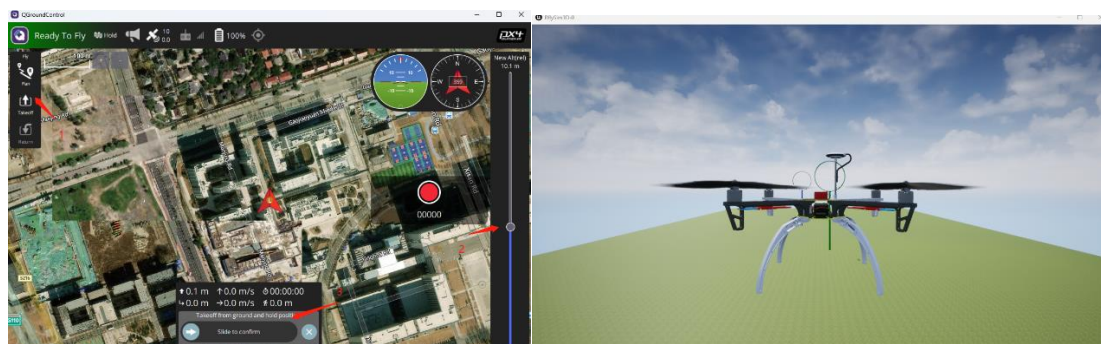
当 RflySim3D 显示 “CopterSim/PX4 EKF 3D Fixed:1/1”，CopterSim 显示 “PX4: GPS 3D fixed & EKF initialization finished.” 时，表明 RflySim 平台已完成初始化，可以进行软件在环仿真。



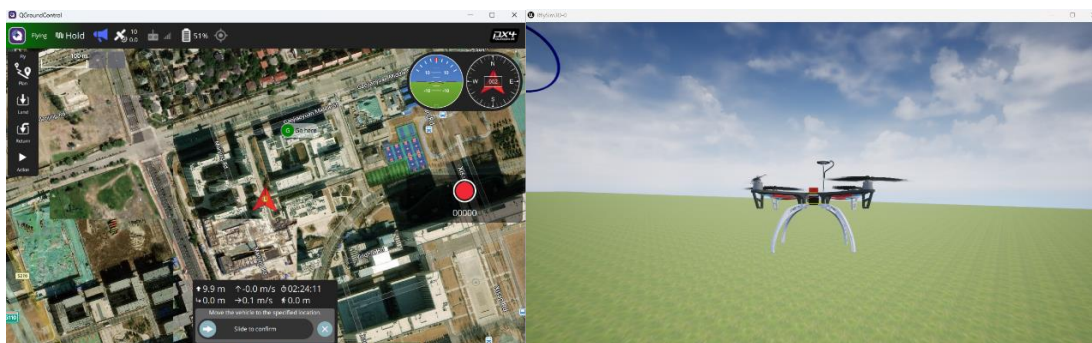
## Step 5:

在 QGC 中观察是否能正常控制飞机的起飞、降落、航路飞行等行为。

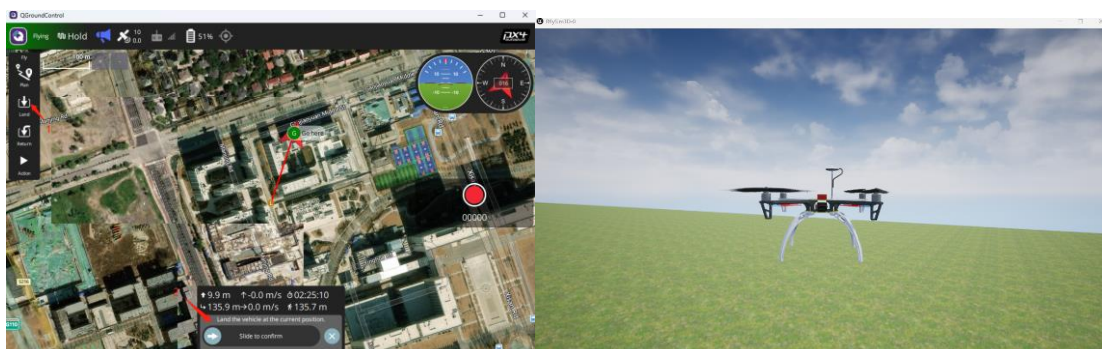
### 1) 起飞



### 2) 飞行



### 3) 着陆

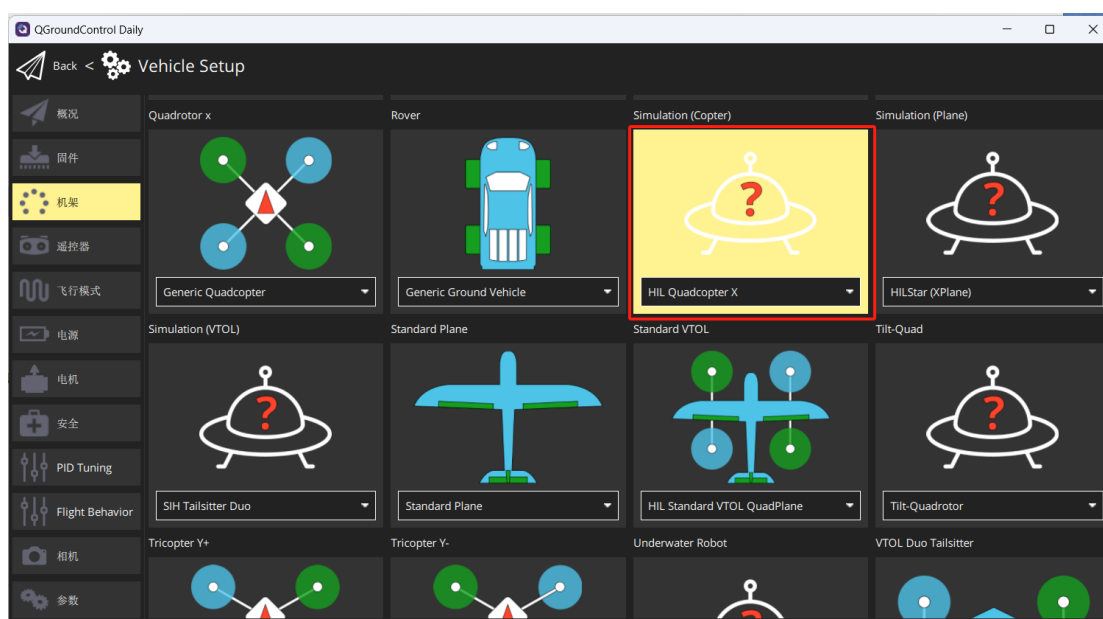


注：由于本接口没有使用地形模块，地形高度始终为 0，因此只能适用于平地场景不能使用 Grassland 等带地形的地图。

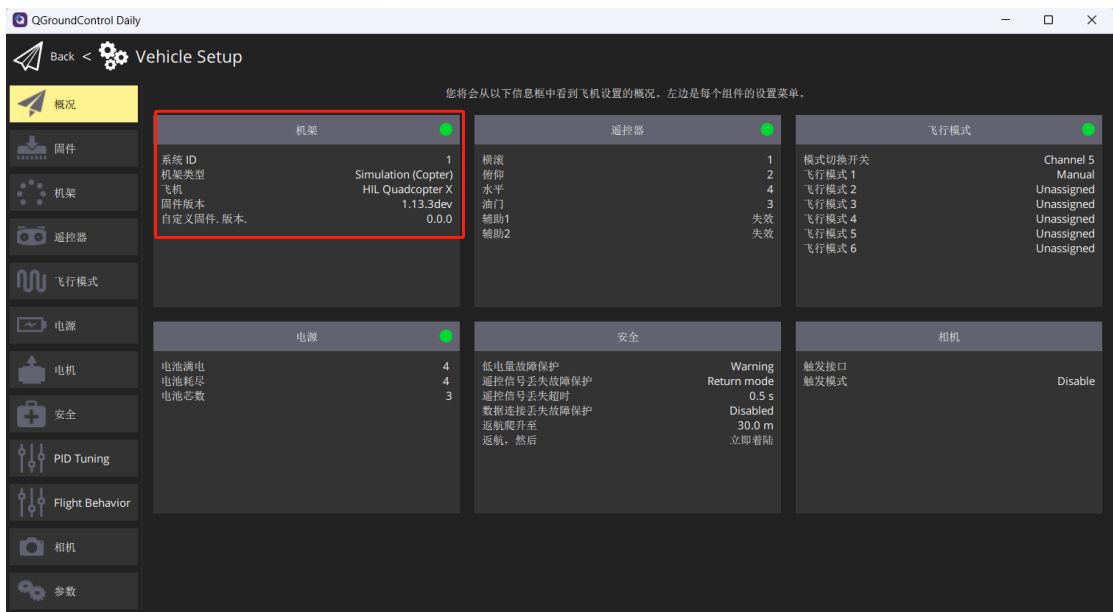
## 6.2. 硬件在环仿真

### Step 1:

推荐使用 Pixhawk 6C 飞控进行硬件在环仿真，1.13.3 固件版本，机架设置为“HIL Quadcopter X”。

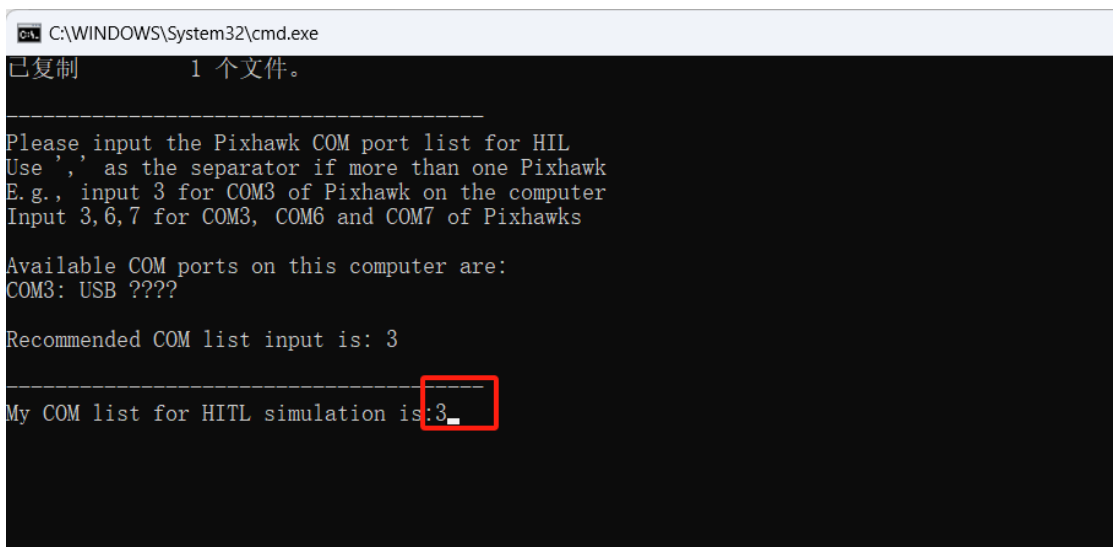


下图为完成仿真参数设置的示意图。



## Step 2:

完成配置后，以管理员身份运行 Exp1\_MinModelTemp\_HITL.bat，在光标处输入飞控和电脑连接的端口号后回车即可启动平台最小模型的硬件在环仿真。



注：在“Exp1\_MinModelTemp\_HITL.bat”硬件在环的脚本文件中，同样需要设置对应无人车的 DLL 名：

```
REM Set use DLL model name or not, use number index or name string
REM This option is useful for simulation with other types of vehicles instead of multicopters
set DLLModel=Exp1_MinModelTemp
```

在 SimMode 处选择 CopterSim 中对应的硬件在环仿真模式：

```
REM Set the simulation mode on CopterSim, use number index or name string
REM e.g., SimMode=0 equals to SimMode=PX4_HITL
set SimMode=0
```

与软件在环仿真不同的是，在之前的配置准备环节中已经在 QGC 中设置了对应机架，

---

所以在该脚本文件中不用设置机架。

### **Step 3:**

重复软件在环 Step4-Step5, 仿真效果和软件在环一致。支持航点、航线、遥控器、Offboard 等控制方式。

## **7. 参考文献**

[1]. 无。

[2].

## **8. 常见问题**

Q1: \*\*\*\*

A1: \*\*\*\*