

1. 实验名称及目的

1.1 实验名称

基于系统模板的不同机架多旋翼模型适配教程

1.2 实验目的

在四旋翼模型的基础上，通过修改相关参数配置，适配六旋翼、八旋翼等其他多旋翼，并进行软硬件在环仿真；本实验的目的在于通过四旋翼模型适配其他旋翼模型，了解各种旋翼之间的异同之处，加深对于模型的理解。

1.3 关键知识点

多旋翼飞行器（如四旋翼、六旋翼、八旋翼等），其动力学模型通常具有对称（旋翼的布局和旋转）和相似（主要是推力和反应特性）特性。因此可以构建统一的模型，通过修改部分参数来确定具体的多旋翼类型。

以平台四旋翼模型为基础，进行适当修改来仿真其他多旋翼模型，那么必须弄清楚平台四旋翼模型中哪些量是四旋翼模型相比于其他多旋翼模型所特有的，剩下的部分即是通用的。

首先是与四旋翼结构相关的：

- 1) 电机数量：四旋翼电机数量为4，若需要适配六旋翼则需要将电机数量修改为6，适配八旋翼和四轴八旋翼则需要将电机数量修改为8
- 2) 模型质量：。四旋翼模型质量为1.515kg，若需要适配为六旋翼和八旋翼则分别修改为 $1.515*1.5\text{kg}$ 和 $1.515*2\text{kg}$ ，若需要适配四轴八旋翼则需要修改为 2.7kg 。
- 3) 四旋翼转动惯量：四旋翼的转动惯量为 $[0.0211,0,0;0,0.0219,0;0,0,0.0366]$ ，需要适配为六旋翼和八旋翼则分别修改为 $[0.0211*1.5,0,0;0,0.0219*1.5,0;0,0,0.0366*1.5]$ 和 $[0.0211*2,0,0;0,0.219*2,0;0,0,0.0366*2]$ ，若需要适配四轴八旋翼则需要修改为 $[0.411,0,0;0,0.419,0;0,0,0.766]$ 。
- 4) 对应力和力矩分配：

多旋翼的结构改变后，根据AirFrames文档每个旋翼之间的夹角改变，旋翼的旋转方向也会改变，不同多旋翼之间的力和力矩计算规则也会不同，通过修改uavType可以将多旋翼适配到相应的力和力矩的计算规则（详细规则可见模型中的Force and Moment Model/Propeller Model old Version模块）。

其次是三维显示相关：

要在UE中将四旋翼的显示模型修改为其他旋翼，要修改显示样式3DType，这里需要和RflySim3D的XML文件中的ClassID相匹配。

2. 实验效果

在四旋翼模型基础上通过修改参数适配其他旋翼模型，并可正常的进行软硬件在环仿真。

3. 文件目录

例程目录：

[\[安装目录\]\RflySimAPIs\4.RflySimModel\2.AdvExps\e2_MultiModelCtrl\8.AdaptationTutorial](#)

文件夹/文件名称	说明
MulticopterNoCtrl.slx	四旋翼飞机模型文件。
MulticopterNoCtrl_HITLRun.bat	硬件在环仿真批处理文件。
MulticopterNoCtrl_SITLRun.bat	软件在环仿真批处理文件。
GenerateModelDLLFile.p	DLL格式转化文件。
MulticopterNoCtrl_init.m	动力学模型相关参数。
MavLinkStruct.mat	MavLink数据结构体mat文件

4. 运行环境

4.1 软件要求

Windows 10及以上版本；RflySim工具链；MATLAB 2017B及以上③。

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；Pixhawk 6X或其它飞控② 1台；数据线 1台。

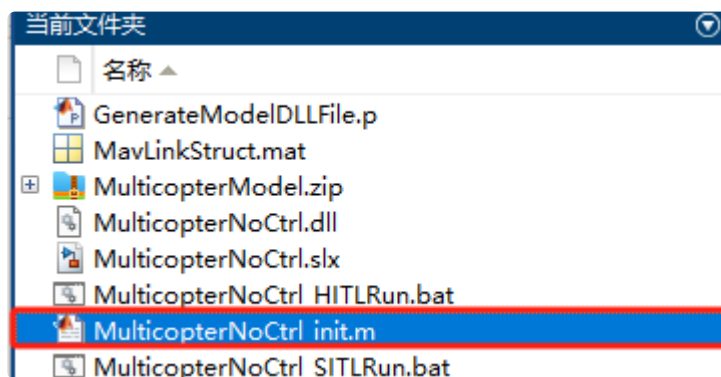
①：推荐配置请见：<https://rflysim.com/>

5. 实验步骤

5.1. 必做实验：适配六旋翼模型

Step 1: 修改.m文件

在Matlab中打开“MulticopterNoCtrl_init.m”文件。



修改模型参数，根据AirFrames文档将3Dtype修改为int16(5)，该参数将决定在UE中模型显示的类型；将uavtype修改为int16(5)，该参数决定了力和力矩的计算。

```

%% 关键参数
% 下面这个参数决定了飞机的显示样式
ModelParam_3DType = int16(5); %这里是四旋翼x型，具体定义见文档“机型定义文档.docx”
% 注意，这里需要和RfFlySim3D的XML文件中的ClassID相匹配。

% 这里和力和力矩的计算有关
ModelParam_uavType = int16(5);

% 初始位姿参数，这两个变量名必须存在，CopterSim才能修改飞机的初始位置
ModelInit_PosE=[0,0,0]; % 飞机初始位置，单位米
ModelInit_AngEuler=[0,0,0];% 飞机初始姿态，单位弧度
% 注意1：当ModelInit_PosE参数存在时，CopterSim会根据UI上的x和y坐标来修改ModelInit_PosE[0]和ModelInit_PosE[1]的值，来配置飞
% ModelInit_PosE[2]初始高度的值，会被CopterSim中读取的地图txt文件中的高度修改，不可以手动配置。
% ModelInit_AngEuler[2]会被CopterSim会根据UI上的yaw输入填充，从而改变飞机初始偏航角
% 初始滚转角和俯仰角ModelInit_AngEuler[0]和ModelInit_AngEuler[1]由这里设置的值决定，CopterSim无法修改。

% 注意2：你也可以不使用ModelInit_PosE和ModelInit_AngEuler变量名，这样CopterSim就没法修改你的初始位置，在集群仿真时，所有飞

% 下面的参数，决定了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。
% 注意1：由于Simulink使用的地球大气模型不支持海平面以下的输入，ModelParam_envAltitude取值必须为负。
% 注意2：你也可以不使用ModelParam_GPSLatLong和ModelParam_envAltitude变量名，这样CopterSim无法从外部（地图txt中读取）修改GP

```

修改模型的质量参数与转动惯量参数，将以下参数在四旋翼的基础上都乘以1.5倍，四旋翼适配六旋翼时旋翼数量增加了1.5倍，以下参数相应修改。

```

%% 6DOF模块相关参数
% 飞机质量：
ModelParam_uavMass=1.515*1.5;
% 转动惯量
ModelParam_uavJ= [0.0211*1.5,0,0;0,0.0219*1.5,0;0,0,0.0366*1.5];
ModelInit_VelB=[0,0,0];
ModelInit_RateB=[0,0,0];

```

修改启用的电机个数，四旋翼适配六旋翼后启用的电机数量也由四个增加为六个，故将uavMotNumbs由int8(4)修改为int8（6）。

```

%% 电机模型参数
ModelParam_uavMotNumbs = int8(6);
%ModelParam_ControlMode = int8(1); %整型 1表示Auto模式，0表示Manual模式
ModelParam_motorMinThr=0.05;
ModelParam_motorCr=842.1;
ModelParam_motorWb=22.83;
ModelParam_motorT= 0.0214;%0.0261;
ModelParam_motorJm =0.0001287;
ModelParam_rotorCm=2.783e-07;
ModelParam_rotorCt=1.681e-05;
ModelInit_RPM = 0; %Initial motor speed (rad/s)
ModelInit_Inputs = [0 0 0 0 0 0 0 0 0 0 0 0 0 0];

```

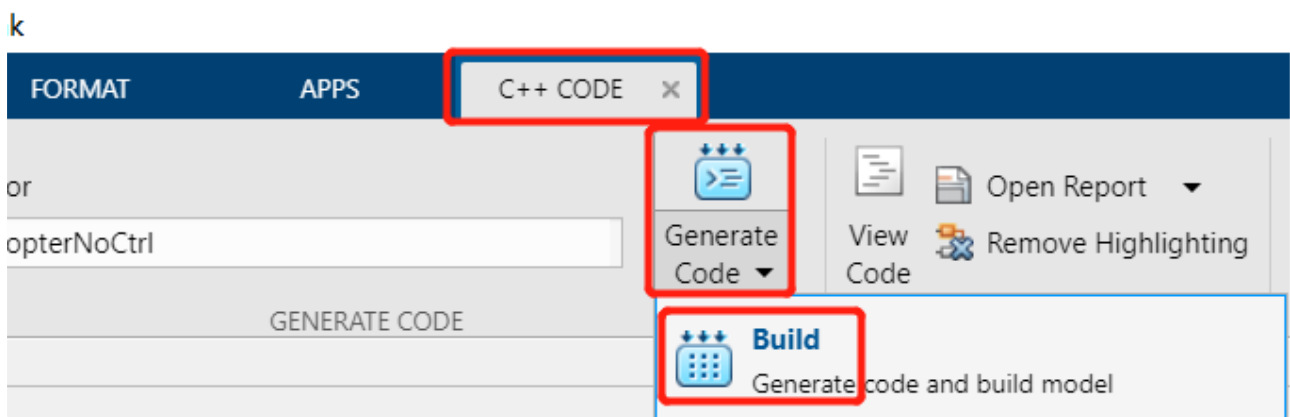
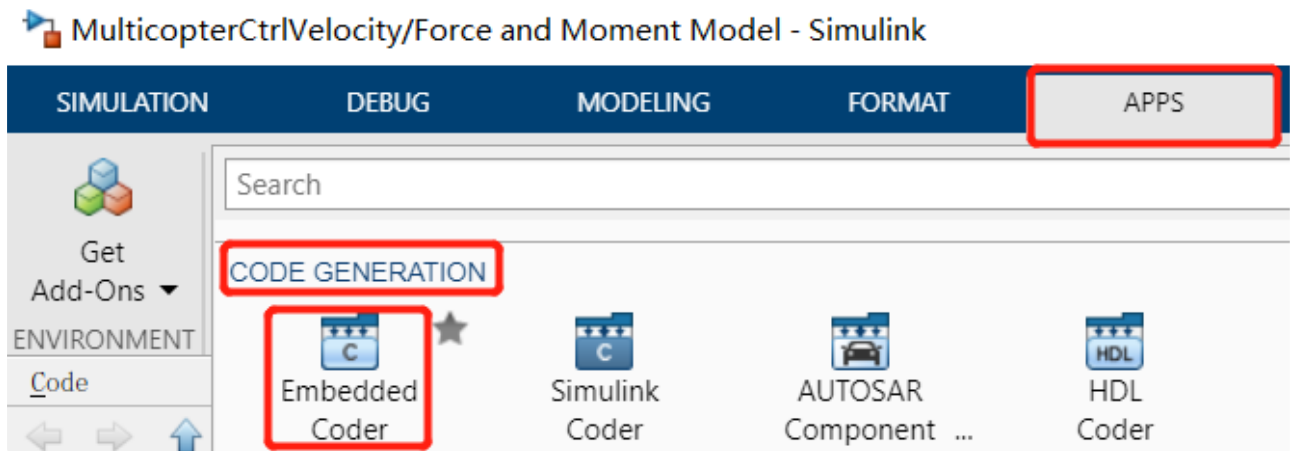
Step 2: 编译模型

将.m文件修改保存后打开slx模型文件后编译。

对于MATLAB 2019a及之前版本，工具栏样式见下图，直接点击它的编译按钮“Build”即可。

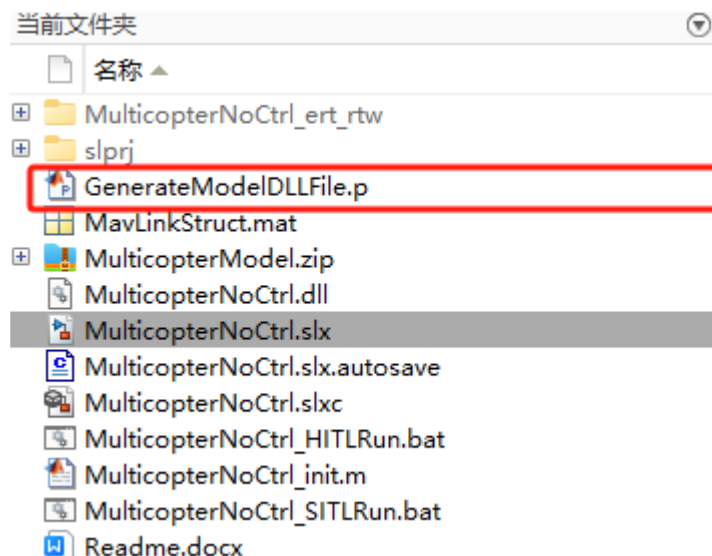


对于2019b及之后版本，点击APPS - CODE GENERATION - Embedded Coder才能弹出代码生成工具栏，在其中如下图所示点击“C++CODE” - “Generate Code” - “Build”按钮就能编译生成代码。



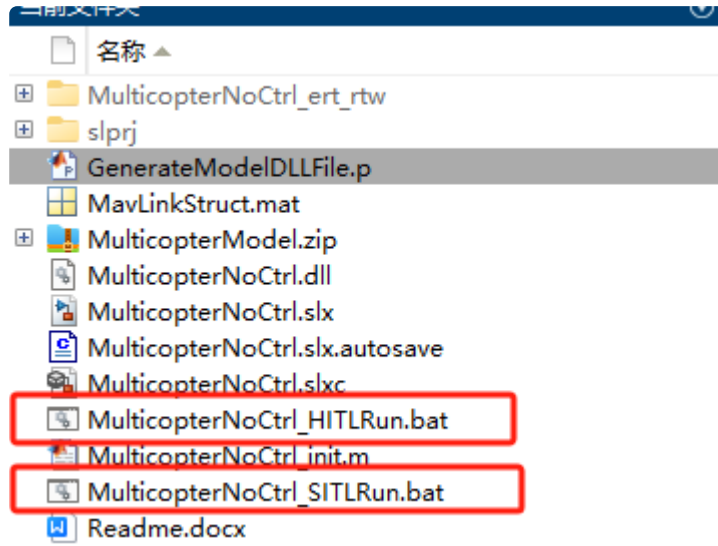
Step 3: 生成DLL文件

运行GenerateModelDLLFile.p文件将刚刚编译的模型生成动态链接库。



Step 4: 仿真过程

按照..\4.HexModelCtrl\Readme.pdf中的软硬件在环的操作分别运行软件在环脚本“[MulticopterNoCtrl_SITLRun.bat](#)”和硬件在环脚本“[MulticopterNoCtrl_HITLRun.bat](#)”。



软硬件在环启动后均可以在UE中看到六旋翼模型，并可以响应QGC发送的运动指令。





注意：

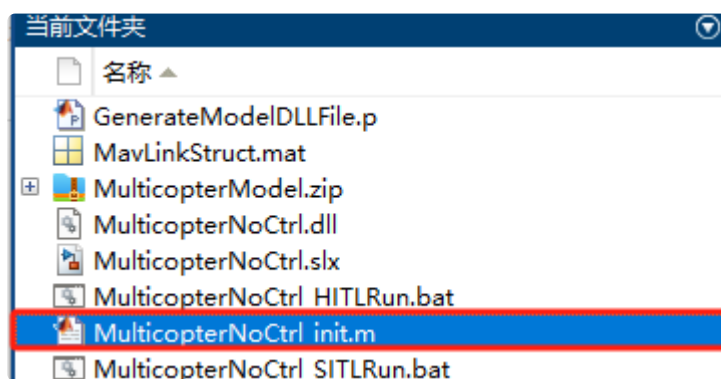
`MulticopterNoCtrl_SITLRun.bat` 脚本中设置机架如下：

```
set PX4SITLFrame=hexa_x
```

5.2. 必做实验：适配八旋翼模型

Step 1: 修改.m文件

在Matlab中打开“`MulticopterNoCtrl_init.m`”文件。



修改模型参数，根据AirFrames文档将3Dtype修改为int16(8)，该参数将决定模型在UE中的显示；将uavtype修改为int16(8)，该参数决定了力和力矩的计算。

```

%% 关键参数

% 下面这个参数决定了飞机的显示样式
ModelParam_3DType = int16(8);
% 注意：这里需要和RtFlySim3D的XML文件中的ClassID相匹配。

% 这里和力和力矩的计算有关
ModelParam_uavType = int16(8);

% 初始位姿参数，这两个变量名必须存在，CopterSim才能修改飞机的初始位置
ModelInit_PosE=[0,0,0]; % 飞机初始位置，单位米
ModelInit_AngEuler=[0,0,0];% 飞机初始姿态，单位弧度
% 注意1：当ModelInit_PosE参数存在时，CopterSim会根据UI上的x和y坐标来修改ModelInit_PosE[0]和ModelInit_PosE[1]的值，来配置飞
% ModelInit_PosE[2]初始高度的值，会被CopterSim中读取的地图txt文件中的高度修改，不可以手动配置。
% ModelInit_AngEuler[2]会被CopterSim会根据UI上的yaw输入填充，从而改变飞机初始偏航角
% 初始滚转角和俯仰角ModelInit_AngEuler[0]和ModelInit_AngEuler[1]由这里设置的值决定，CopterSim无法修改。

% 注意2：你也可以不使用ModelInit_PosE和ModelInit_AngEuler变量名，这样CopterSim就没法修改你的初始位置，在集群仿真时，所有飞

% 下面的参数，决定了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。
% 注意1：由于Simulink使用的地球大气模型不支持海平面以下的输入，ModelParam_envAltitude取值必须为负。
% 注意2：你也可以不使用ModelParam_GPSLatLong和ModelParam_envAltitude变量名，这样CopterSim无法从外部（地图txt中读取）修改GP

```

修改模型的质量参数与转动惯量参数，将以下参数在四旋翼的基础上都乘以2倍，四旋翼适配八旋翼时旋翼数量增加了2倍，以下参数相应修改。

```

%% 6DOF模块相关参数
% 飞机质量
ModelParam_uavMass=1.515*2;
% 转动惯量
ModelParam_uavJ= [0.0211*2,0,0;0,0.0219*2,0;0,0,0.0366*2];
ModelInit_VelB=[0,0,0];
ModelInit_RateB=[0,0,0];

```

修改启用的电机个数，四旋翼适配八旋翼后启用的电机数量也由四个增加为八个，故将uavMotNumbs由int8(4)修改为int8(8)。

```

%% 电机模型参数
ModelParam_uavMotNumbs = int8(8);
%ModelParam_ControlMode = int8(1); % 整型 1表示Auto模式，0表示Manual模式
ModelParam_motorMinThr=0.05;
ModelParam_motorCr=842.1;
ModelParam_motorWb=22.83;
ModelParam_motorT= 0.0214;%0.0261;
ModelParam_motorJm =0.0001287;
ModelParam_rotorCm=2.783e-07;
ModelParam_rotorCt=1.681e-05;
ModelInit_RPM = 0; %Initial motor speed (rad/s)
ModelInit_Inputs = [0 0 0 0 0 0 0 0 0 0 0 0 0 0];

```

Step 2: 编译模型

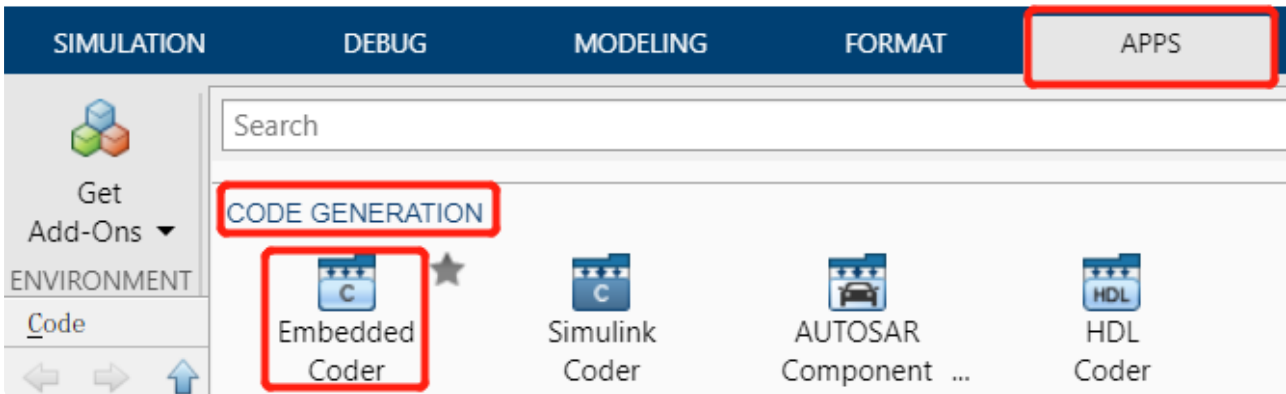
将.m文件修改保存后打开slx模型文件后编译。

对于MATLAB 2019a及之前版本，工具栏样式见下图，直接点击它的编译按钮“Build”即可。

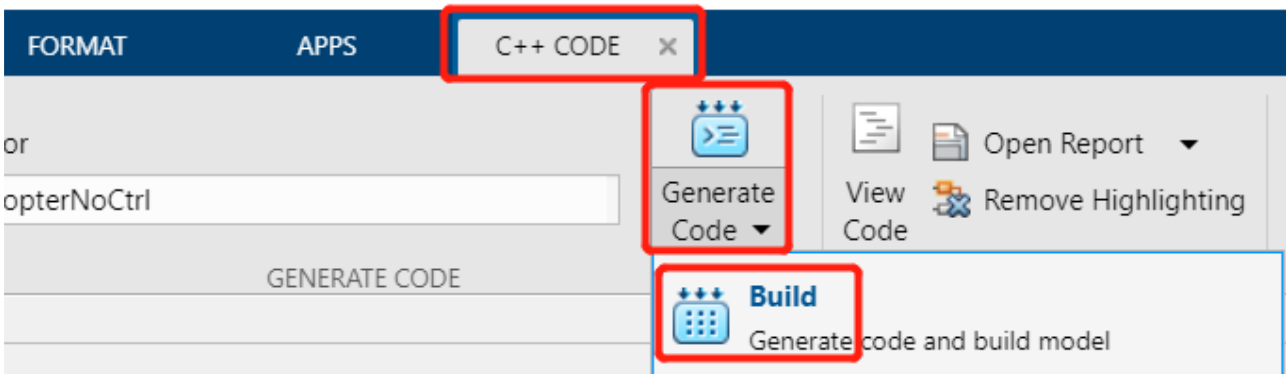


对于2019b及之后版本，点击APPS - CODE GENERATION - Embedded Coder才能弹出代码生成工具栏，在其中如下图所示点击“C++CODE” - “Generate Code” - “Build”按钮就能编译生成代码。

MulticopterCtrlVelocity/Force and Moment Model - Simulink

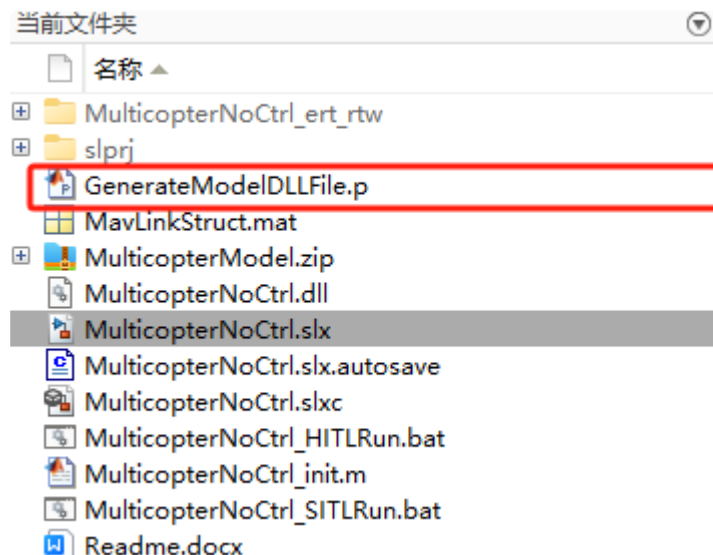


k



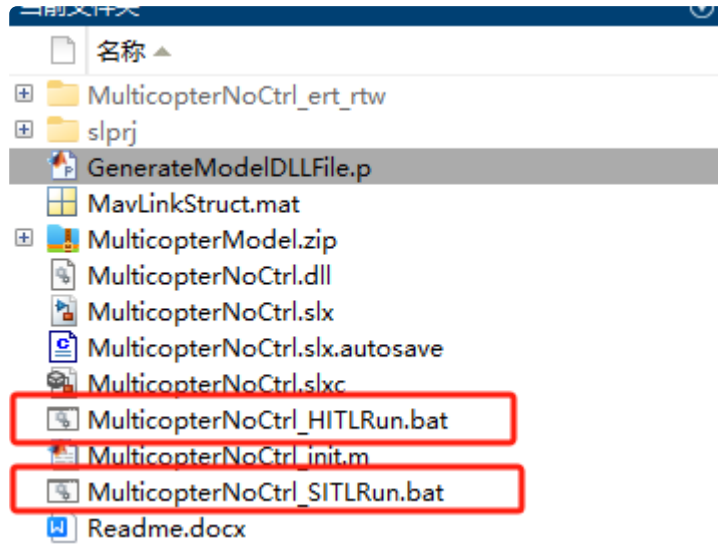
Step 3: 生成DLL文件

运行GenerateModelDLLFile.p文件将刚刚编译的模型生成动态链接库。

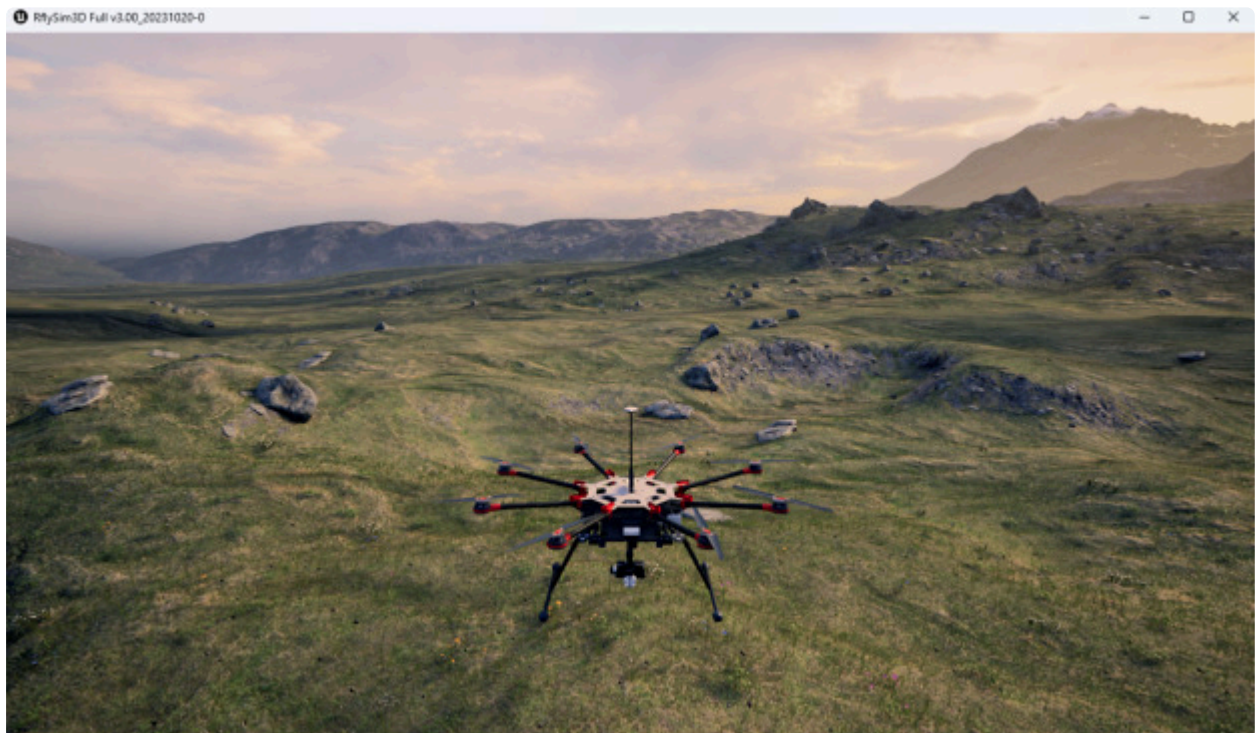


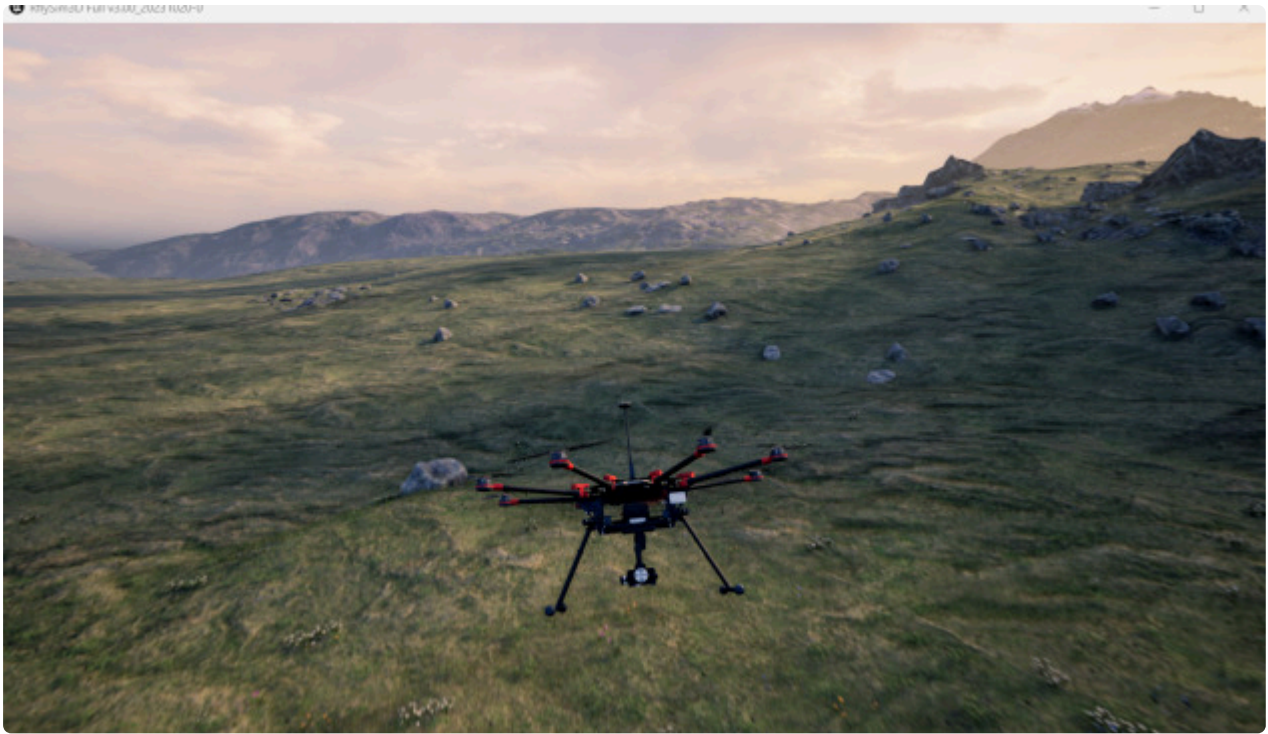
Step 4: 仿真过程

按照..\6.OctoX\Readme.pdf中的软硬件在环的操作分别运行软件在环脚本“[MulticopterNoCtrl_SITLRun.bat](#)”和硬件在环脚本“[MulticopterNoCtrl_HITLRun.bat](#)”。



软硬件在环启动后均可以在UE中看到八旋翼模型，并可以响应QGC发送的运动指令。





注意：

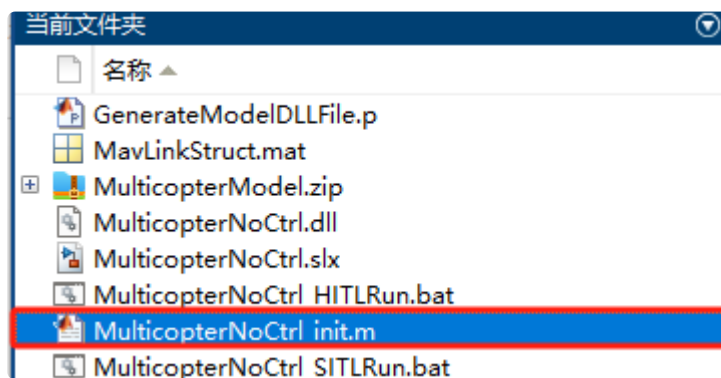
[MulticopterNoCtrl_SITLRun.bat](#) 脚本中设置机架如下：

```
set PX4SITLFrame=octo_x
```

5.3. 必做实验：适配四轴八旋翼模型

Step 1: 修改.m文件

在Matlab中打开 “[MulticopterNoCtrl_init.m](#)” 文件。



修改模型参数，根据AirFrames文档将uavtype修改为int16(10)，该参数将决定在UE中模型显示的类型。将uavtype修改为int16(10)，该参数决定了力和力矩的计算。

```
%% 关键参数
```

```
% 下面这个参数决定了飞机的显示样式
```

```
ModelParam_3DType = int16(10);
```

```
% 注意，这里需要和RflySim3D的XML文件中的ClassID相匹配。
```

```
% 这里和力和力矩的计算有关
```

```
ModelParam_uavType = int16(10);
```

```
% 初始姿态参数，这两个变量名必须存在，CopterSim才能修改飞机的初始位置
```

```
ModelInit_PosE=[0,0,0]; % 飞机初始位置，单位米
```

```
ModelInit_AngEuler=[0,0,0]; % 飞机初始姿态，单位弧度
```

```
% 注意1：当ModelInit_PosE参数存在时，CopterSim会根据UI上的x和y坐标来修改ModelInit_PosE[0]和ModelInit_PosE[1]的值，来配置飞机  
% ModelInit_PosE[2]初始高度的值，会被CopterSim中读取的地图txt文件中的高度修改，不可以手动配置。
```

```
% ModelInit_AngEuler[2]会被CopterSim会根据UI上的yaw输入填充，从而改变飞机初始偏航角
```

```
% 初始滚转角和俯仰角ModelInit_AngEuler[0]和ModelInit_AngEuler[1]由这里设置的值决定，CopterSim无法修改。
```

```
% 注意2：你也可以不使用ModelInit_PosE和ModelInit_AngEuler变量名，这样CopterSim就没法修改你的初始位置，在集群仿真时，所有飞
```

```
% 下面的参数，决定了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。
```

```
% 注意1：由于simulink使用的地球大气模型不支持海平面以下的输入，ModelParam_envAltitude取值必须为负。
```

```
% 注意2：你也可以不使用ModelParam_GPSLatLong和ModelParam_envAltitude变量名，这样CopterSim无法从外部（地图txt中读取）修改GP
```

修改模型的质量参数为2.7，由于四轴八旋翼的结构与传统八旋翼的区别，质量不是四旋翼质量的两倍：同理，转动惯量的修改也不是严格遵守八旋翼的规则，做出以下修改。

```
%% 6DOF模块相关参数
```

```
% 飞机质量：
```

```
ModelParam_uavMass=2.7;
```

```
% 转动惯量
```

```
ModelParam_uavJ= [0.0411,0,0;0,0.0419,0;0,0,0.0766];
```

```
ModelInit_VeIB=[0,0,0];
```

```
ModelInit_RateB=[0,0,0];
```

修改启用的电机个数，四旋翼适配四轴八旋翼后启用的电机数量也由四个增加为8个，故将uavMotNumbs由int8(4)修改为int8 (8)。

```
%% 电机模型参数
```

```
ModelParam_uavMotNumbs = int8(8);
```

```
%ModelParam_ControlMode = int8(1); %整型 1表示Auto模式，0表示Manual模式
```

```
ModelParam_motorMinThr=0.05;
```

```
ModelParam_motorCr=842.1;
```

```
ModelParam_motorWb=22.83;
```

```
ModelParam_motorT= 0.0214;%0.0261;
```

```
ModelParam_motorJm =0.0001287;
```

```
ModelParam_rotorCm=2.783e-07;
```

```
ModelParam_rotorCt=1.681e-05;
```

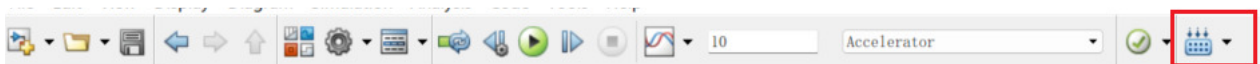
```
ModelInit_RPM = 0; %Initial motor speed (rad/s)
```

```
ModelInit_Inputs = [0 0 0 0 0 0 0 0 0 0 0 0 0 0];
```

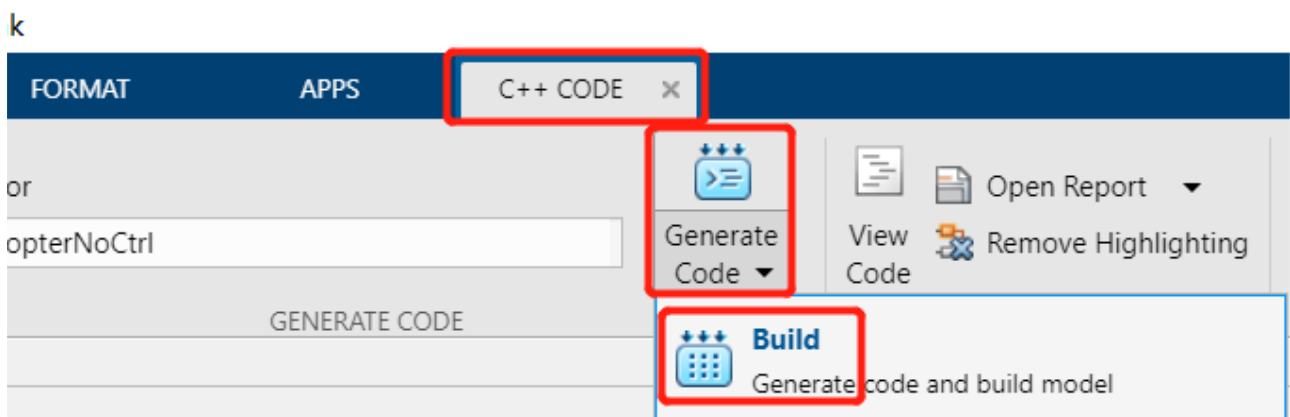
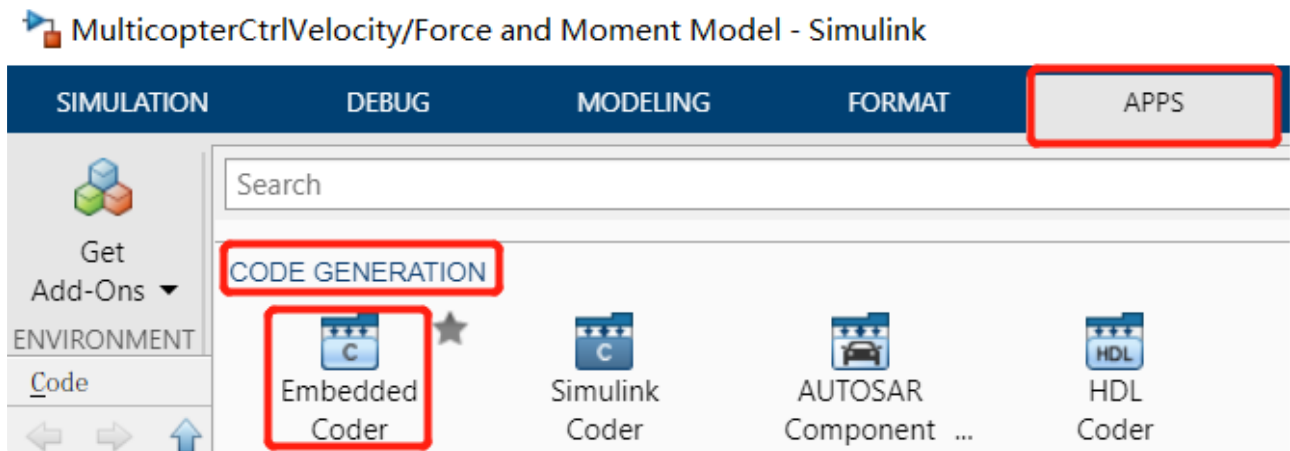
Step 2：编译模型

将.m文件修改保存后打开slx模型文件后编译。

对于MATLAB 2019a及之前版本，工具栏样式见下图，直接点击它的编译按钮“Build”即可。

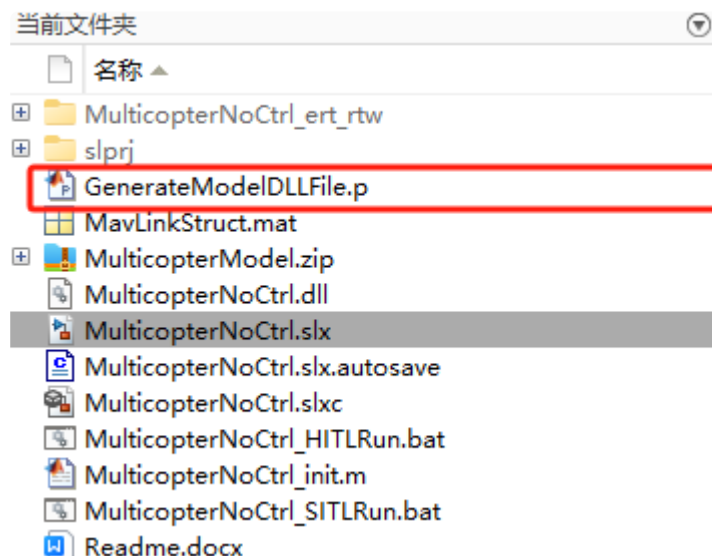


对于2019b及之后版本，点击APPS - CODE GENERATION -Embedded Coder才能弹出代码生成工具栏，在其中如下图所示点击“C++CODE” - “Generate Code” - “Build” 按钮就能编译生成代码。



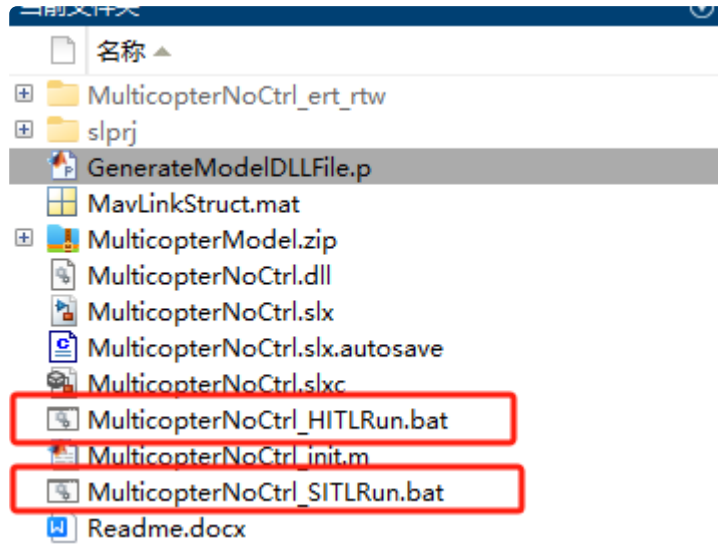
Step 3: 生成DLL文件

运行GenerateModelDLLFile.p文件将刚刚编译的模型生成动态链接库。



Step 4: 仿真过程

按照..\5.OctoCoxRotor\Readme.pdf软硬件在环的操作分别运行软件在环脚本“[MulticopterNoCtrl_SITLRun.bat](#)”和硬件在环脚本“[MulticopterNoCtrl_HITLRun.bat](#)”。



软硬件在环启动后均可以在UE中看到四轴八旋翼旋翼模型，并可以响应QGC发送的运动指令。





注意：

[MulticopterNoCtrl_SITLRun.bat](#) 脚本中设置机架如下：

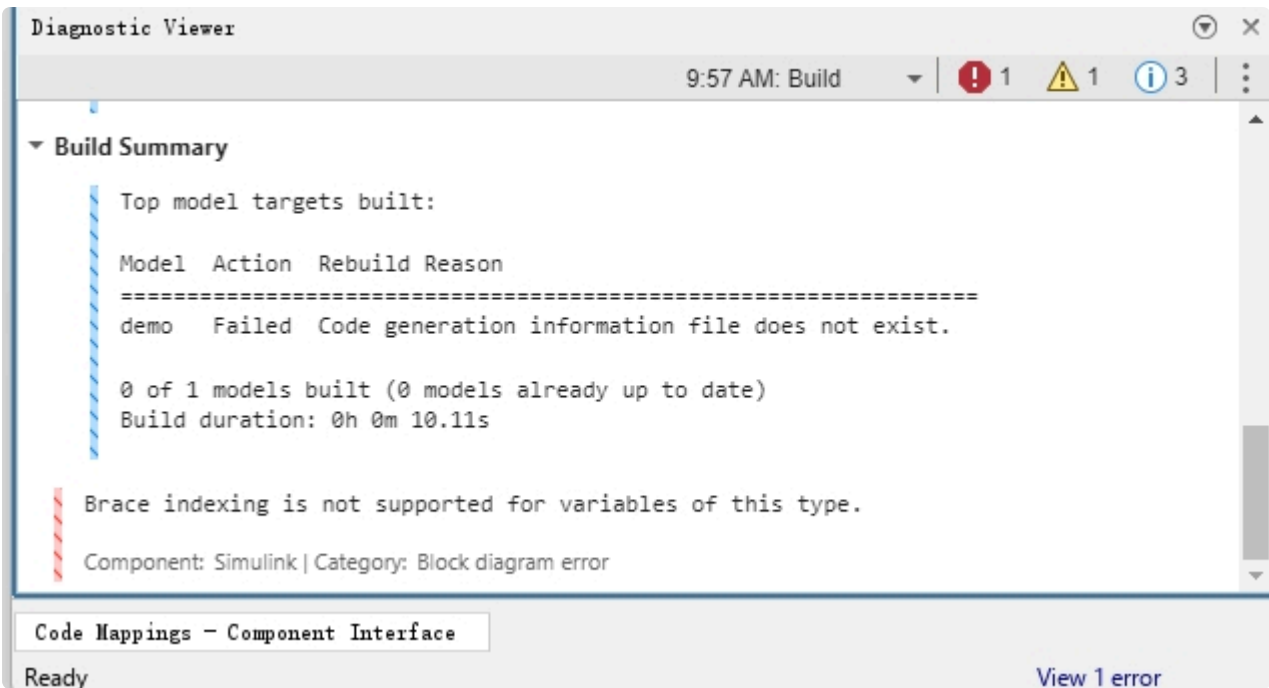
```
set PX4SITLFrame=octo_cox
```

6. 参考资料

1. PX4PSP\RflySimAPIs\4.RflySimModel\API.pdf中的Simulink载具建模模板介绍
- 2.

7. 常见问题

Q1: 未正确安装visual studio c++编译环境并配置mex，导致Simulink文件编译失败



A1: 首先将低于当前MATLAB版本的Visual Studio C++编译环境安装到VS默认安装目录，然后在MATLAB的命令行窗口中输入指令“mex -setup”，一般来说会自动识别并安装上支持的编译器，命令行显示“MEX 配置使用 ‘Microsoft Visual C++ 2017’ 以进行编译”的字样说明安装正确。详细环境配置参考” [RflySim平台安装目录]\RflySimAPIs\4.RflySimModel\API.pdf “中的环境配置



Q2: 编译报错，无法加载库文件



A2: 这可能是由于安装平台时PX4PSP工具箱未更新到最新版, 更新RflySim安装包后按照如下配置重新安装平台即可

