

# 1. 实验名称及目的

## 1.1. 实验名称

自定义无人机(以六旋翼为例)控制算法开发（仅限完整版及以上版本）

## 1.2. 实验目的

基于 RflySim 中四旋翼无人机底层控制算法开发的框架，扩展到六旋翼无人机的底层控制算法开发，以此指导用户实现自定义无人机的底层控制算法开发。

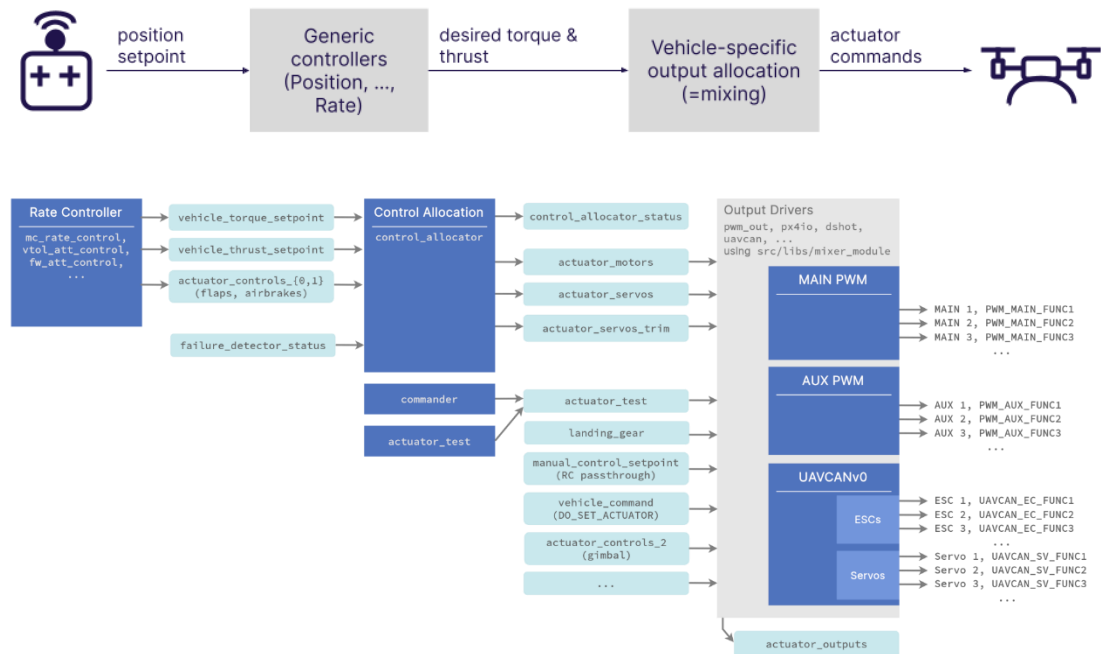
## 1.3. 关键知识点

### 1.3.1. 关键知识点 1: PX 4 软件系统控制分配框架

PX4 架构保证了核心控制器中不需要针对机身布局做特别处理。混控指的是把输入指令（例如：遥控器打右转）分配到电机以及舵机的执行器（如电调或舵机 PWM）指令。对于固定翼的副翼控制而言，每个副翼由一个舵机控制，那么混控的意义就是控制其中一个副翼抬起而另一个副翼落下。同样的，对多旋翼而言，俯仰操作需要改变所有电机的转速。将混控逻辑从实际姿态控制器中分离出来可以大大提高复用性。

特定的控制器发送一个特定的归一化的力或力矩指令（缩放至-1..+1）给混控器，混控器则相应地去设置每个单独的执行器。控制量输出驱动程序（比如：UART,UAVCAN 或者 PWM）则将混控器的输出所放为执行器实际运行时的原生单位，例如输出一个值为 1300 的 PWM 指令。

此外，PX 4 抽象了输出功能到特定硬件输出的映射。这意味着任何电机或伺服机构都可以分配给几乎任何物理输出。



---

更多详细信息可见：[https://docs.px4.io/v1.13/en/concept/control\\_allocation.html](https://docs.px4.io/v1.13/en/concept/control_allocation.html)

### 1.3.2. 关键知识点 2: PX4 混控器文件定义

PX4 中的混控器的其中一个作用是连接上层应用模块（例如姿态控制器解算的输出量）与底层硬件的执行机构输出（对应电机、舵机的 PWM 值）。混控器模块相当于隔离了上层应用模块与底层的硬件接口，在编写上层应用模块时不用担心控制量是输出给哪个电机，同时更改不同的混控器配置文件可以适应不同的机型而无需更改上层应用模块的代码。具体代码见：`*\PX4PSP\Firmware\ROMFS\px4fmu_common\mixers`

PX4 软件系统中混控器文件的默认设置可在文件夹`*\PX4PSP\Firmware\ROMFS\px4fmu_common\init.d\airframes`中查阅。同时，也可在 SD 卡的`*/etc/mixers/`目录中同名的混控器文件进行覆盖。PX4 将混控器文件中被命名为 `xxxx.main.mix` 的文件映射到主(MAIN)输出，被命名为 `yyyy.aux.mix` 的文件映射到辅助(AUX)输出，其中前缀(`xxxx/yyyy`)取决于机架和机架的配置。通常，MAIN 和 AUX 输出对应于 MAIN 和 AUXPWM 输出，但当启用时，这些可能被加载到 UAVCAN(或其他)总线中。

主混合器文件名(前缀 `xxxx`)在机身配置中使用 `setmixerxxxx` 设置，如：`airframes/4011_dji_f450` 调用 `setmixerquad_x` 来加载主混控器文件 `quad_x.MAIN.mix`。AUX 混控器文件(前缀 `yyyy`)取决于机身设置或默认值，如：

MIXER\_AUX 可用于设置加载哪个 `yyyy.aux.mix` 文件。如：`*\PX4PSP\Firmware\ROMFS\px4fmu_common\init.d\airframes\13006_vtol_standard_delta` 文件中，`setMIXER_AUXvtol_delta` 设置为该机架后，加载的混控器文件为：`*\PX4PSP\Firmware\ROMFS\px4fmu_common\mixers\vtol_delta.aux.mix`。

对于多旋翼和固定翼机型，若未设置 MIXER\_AUX 选项，则默认加载的混控器文件为：`*\PX4PSP\Firmware\ROMFS\px4fmu_common\mixers\pass.aux.mix`。（注：`pass.aux.mix` 是遥控器的直通混控器文件，它将 4 个用户自定义的遥控器通道的值(使用 `RC_MAP_AUXx/RC_MAP_FLAPS` 参数设置)传递到 AUX 输出的前四个输出。）

垂直起降无人机 (VOTL) 若进行相关设置则加载所设置的 `yyyy.aux.mix`；若未进行设置，则只加载 MIXER 设置的混控器文件。

通用控制模式启用(且输出模式设置为 AUX)将覆盖机架中 MIXER\_AUX 设置混控器文件，并加载`*\PX4PSP\Firmware\ROMFS\px4fmu_common\mixers\mount.aux.mix` 上的 AUX 输出。

注：上述所述的混控文件加载通过文件：`*\PX4PSP\Firmware\ROMFS\px4fmu_common\init.d/rc.interface` 实现。

Mixer 文件是定义一个或多个 Mixer 定义的文本文件:一个或多个输入与一个或多个输出之间的映射。主要有四种类型的定义:`multiroto mixer, helicopter mixer, summing mixer, and nul l mixer`。

➤ **multiroto mixer:** 定义输出 4、6 或 8 的+型或 x 型旋翼载具。

- 
- **helicoptermixer**: 定义直升机斜盘伺服器和主电机 ESCs 的输出(尾桨是一个单独的混合器)。
  - **summingmixer**: 将零或多个控制输入组合成单个执行器输出。输入被缩放, 混合函数在应用输出缩放器之前对结果求和。
  - **summingmixer**: 产生一个输出为零的执行器输出(当不处于故障安全模式时)。

每个混控器产生的输出量取决于混控器的类型和配置。如: **multirotormixer** 根据其机型可有 4、6 或 8 个输出, 而 **summingmixer** 或 **summingmixer** 只产生一个输出。可以在每个文件中指定多个混控器。输出顺序(将混合器分配给执行器)特定于读取混控器定义的设备, 对于 PWM, 输出顺序与声明顺序相匹配。

每个混控器文件最开始定义的语句为:

```
<tag>:<mixerarguments>
```

其中 tag 表示所选择的混控器类型, 如下:

```
R:Multirotormixer  
H:Helicoptermixer  
M:Summingmixer  
Z:Nullmixer
```

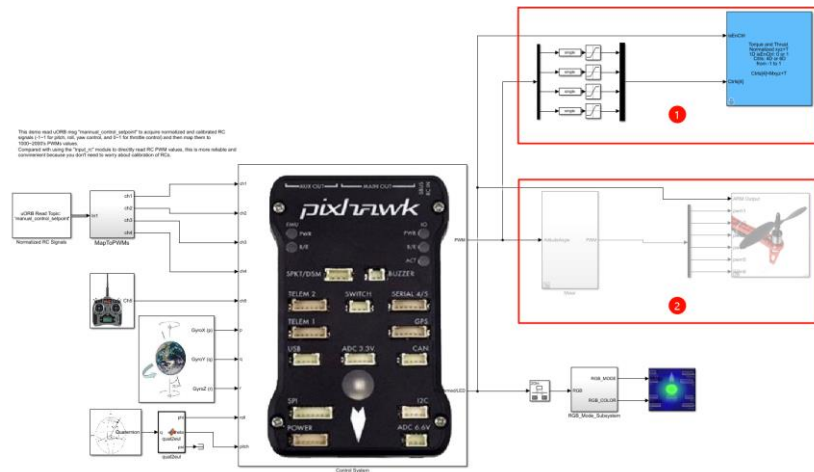
更多详细信息可见: [..\..\API.pdf](#) 中第 8 小节。

### 1.3.3. 关键知识点 3: 基于 RflySim 混控器切换

RflySim 工具链即可直接将控制器的力和力矩型号直接传递到 PX4 软件系统的混控器中, 来实现 PX4 软件中的混控器复用。也可以在 Simulink 模型中自定义混控器(控制分配)函数将力和力矩转换成电机的 PWM 值, 直接输出到电机中。在本实验中可以看到如下两种不同的控制逻辑的实现, 其中:

①、表示将控制器的输出直接发送力和力矩(归一化后的)控制量 `actuator_controls_0` 消息(或 PX4 v1.14 版本中的 `vehicle_torque/thrust_setpoint`), 并经过 PX4 自带的 Mixer 混控器计算电机转速后, 驱动给 CopterSim 的 `inPWMs` 输入, 控制无人机的运动。真机实飞时, 则直接驱动飞机进行飞行。具体介绍可见: [..\..\API.pdf](#) 中第 2.4.13 小节。注: **Hex\_AttCtrl\_HIL.slx** 该文件中默认为此模式, 推荐使用此模式。

②、表示在 Simulink 中自定义 Mixer 函数模块, 将控制器的输出进一步转换成 PWM 信号, 通过图示这个模块发送 PWM 信号到 PX4IO 的输出端口以控制电机转动。具体介绍可见: [..\..\API.pdf](#) 中第 2.3.3 小节。



同时，需要注意的是：本实验适配的是六旋翼无人机机型，因此其动力学模型也应该使用六旋翼无人机模型，实验中所使用的 DLL 动力学模型文件来源于：[..\..\..\4.RflySimModel\2.AdvExps\2\\_MultiModelCtrl\4.HexModelCtrl\Readme.pdf](https://github.com/RoboStack/4.RflySimModel/2.AdvExps/e2_MultiModelCtrl/4.HexModelCtrl/Readme.pdf)。模型详细搭建可见该实验中的 Readme 文件。

## 2. 实验效果

实现六旋翼无人机的自稳模式下飞行。



## 3. 文件目录

文件夹/文件名称	说明
<a href="#">HexarotorModelCTRL.dll</a>	六旋翼无人机动力学 DLL 文件
<a href="#">HexarotorModelCTRL_HITLRun.bat</a>	硬件在环一键启动仿真脚本
<a href="#">Init_control.m</a>	控制参数文件
<a href="#">Hex_AttCtrl_HIL.slx</a>	控制器 Simulink 模型文件

<a href="#">icon</a>	控制器图标文件
----------------------	---------

## 4. 运行环境

序号	软件要求	硬件要求	
		名称	数量
1	Windows 10 1909 及以上版本	笔记本/台式电脑 <sup>①</sup>	1
2	RflySim 工具链	Pixhawk 6X 或 Pixhawk 6X mini <sup>②</sup>	1
3	MATLAB 2022b 及以上版本	遥控器 <sup>③</sup>	1
		遥控器接收器	1
		数据线、杜邦线等	若干

①：推荐配置请见：<https://rflysim.com/doc/zh/HowToInstall.pdf>

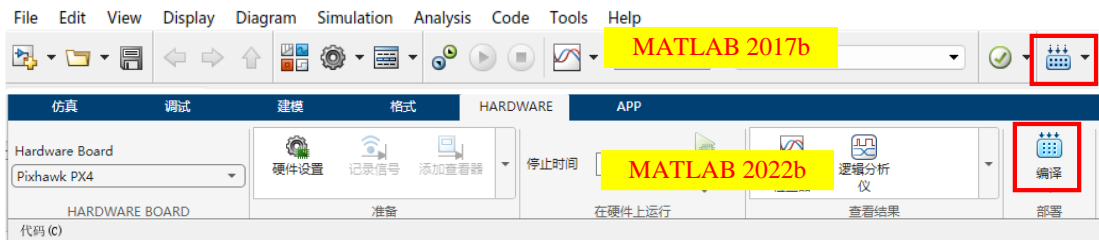
②：须保证平台安装时的编译命令为：px4\_fmuv6x\_default，固件版本为：1.12.3 及以上。其他配套飞控请见：<https://rflysim.com/doc/zh/B/2.Pixhawk.html>

③：本实验演示所使用的遥控器为：天地飞 ET10、配套接收器为：WFLY RF209S。遥控器相关配置见：<https://rflysim.com/doc/zh/B/3.1ET10.html>

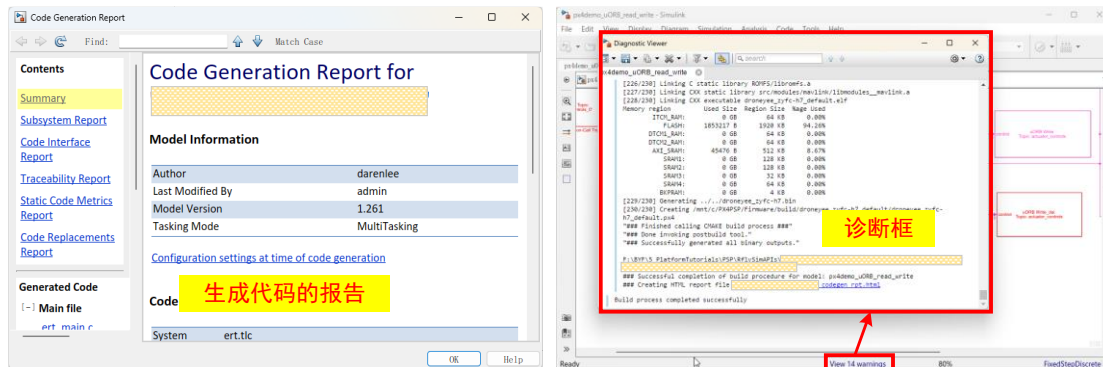
## 5. 实验步骤

### 5.1. 步骤 1：固件编译

在 MATLAB 中打开 `init_control.m` 文件并运行，等待运行完成之后，将自动打开 `HexAttCtrl_HIL.slx` 文件，点击 Simulink 中的编译按钮。

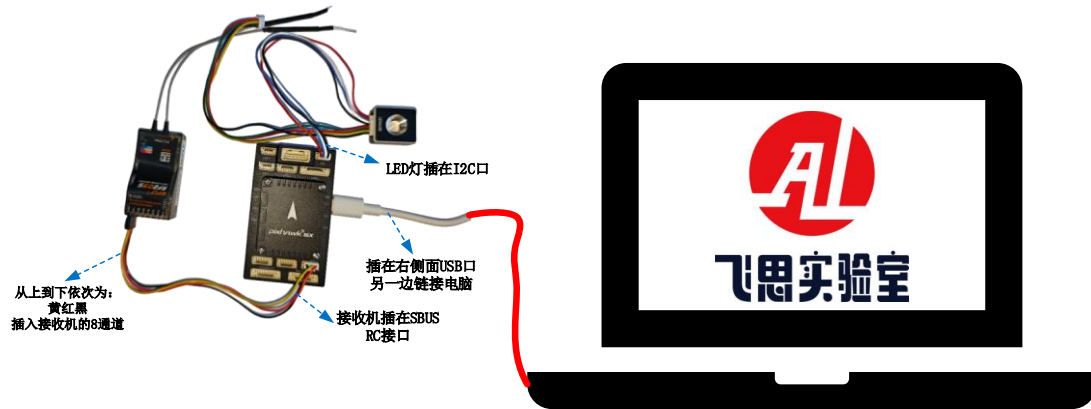


在 Simulink 的下方点击 View diagnostics 指令，即可弹出诊断对话框，可查看编译过程。在诊断框中弹出 Build process completed successfully，即可表示编译成功，左图为生成的编译报告。

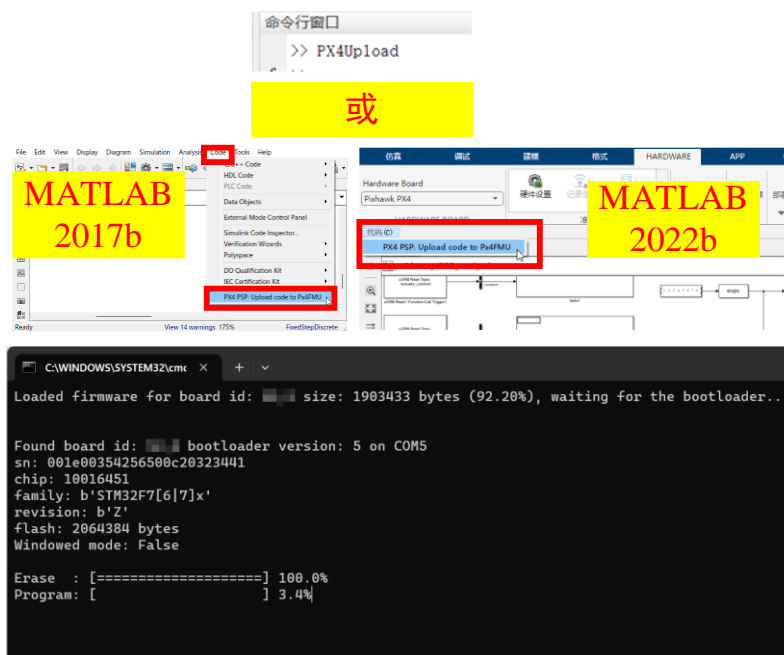


## 5.2. 步骤 2：固件上传

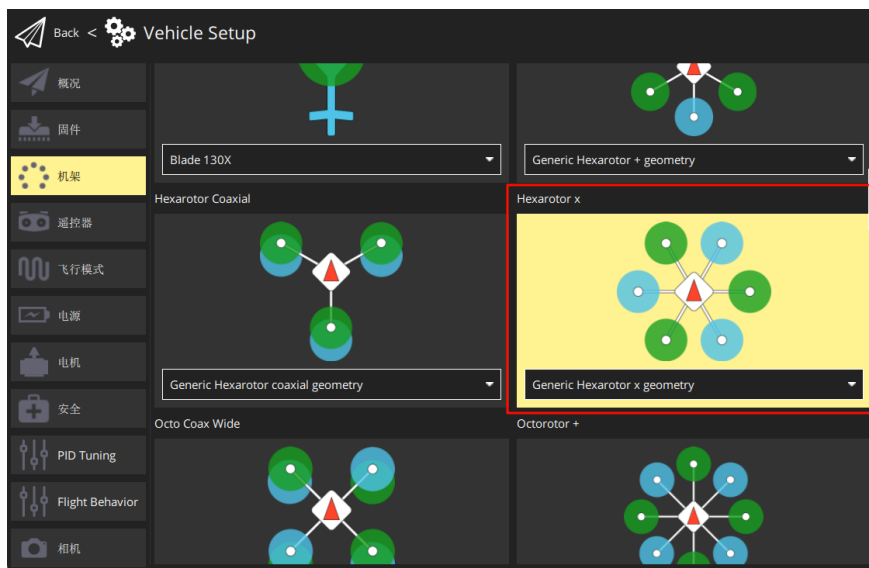
用 USB 数据线链接飞控与电脑。在 MATLAB 命令行窗口输入：PX4Upload 并运行或点击 PX4 PSP: Upload code to Px4FMU，弹出 CMD 对话框，显示正在上传固件至飞控中，等待上传成功。



在 MATLAB 命令行窗口输入：PX4Upload 并运行或点击 PX4 PSP: Upload code to Px4 FMU，弹出 CMD 对话框，显示正在上传固件至飞控中，等待上传成功。

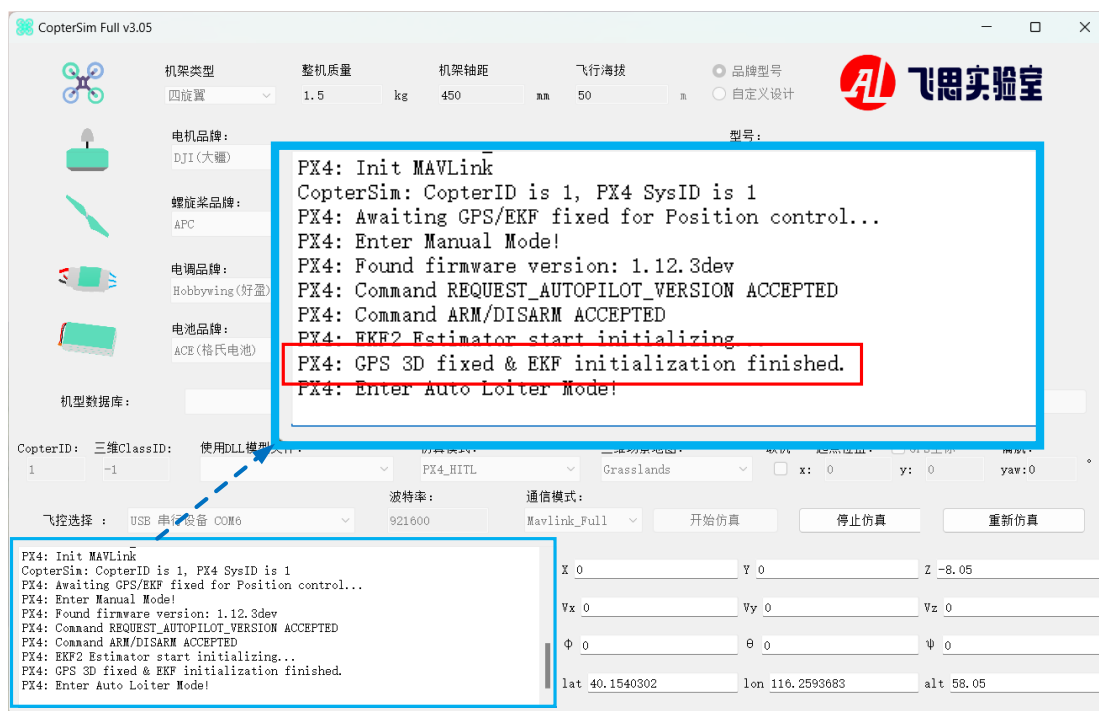


上传成功后，打开 QGroundControl 软件，确认机架为如下设置：



### 5.3. 步骤 3：启动硬件在环仿真

双击打开 [HexarotorModelCTRL\\_HITLRun.bat](#) 文件，在弹出的 CMD 对话框中输入插入的飞控 Com 端口号，即可自动启动 RflySim3D、CopterSim、QGroundControl 软件，等待 CopterSim 的状态框中显示：PX4: GPS 3D fixed & EKF initialization finished。即可在 QGroundControl 中设置飞机起飞等操作。



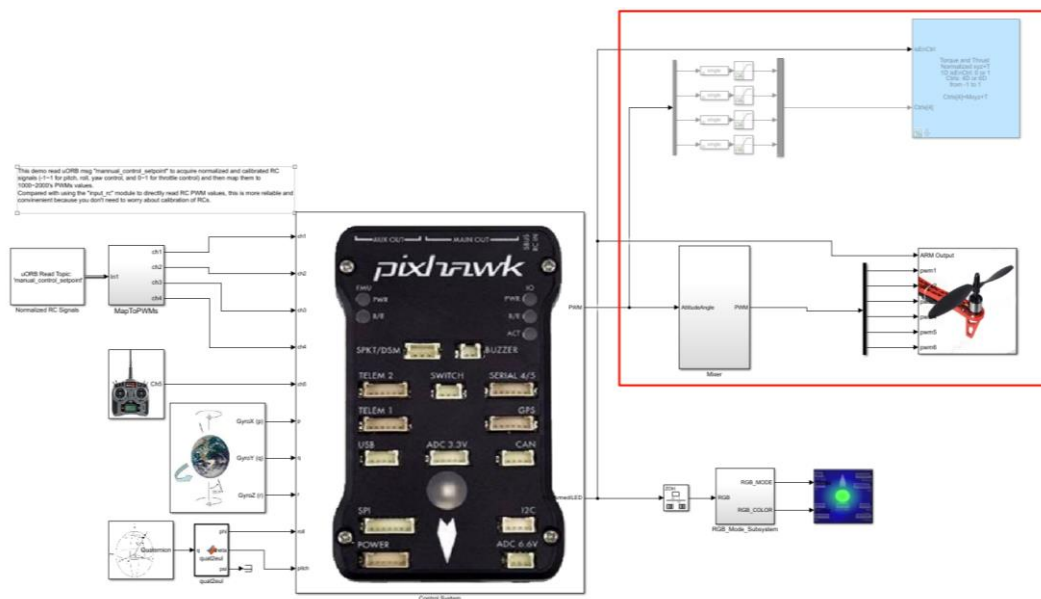
### 5.4. 步骤 4：实验效果

拨动遥控器上的 CH5 通道到高电位，飞控中的控制器将切换到 Simulink 中搭建的控制器输出并自动解锁无人机，通过遥控器控制六旋翼无人机正常起飞。



## 5.5. 步骤 5：切换为自定义 Mixer 函数模型

将 [Hex\\_AttCtrl\\_HIL.slx](#) 模型文件中，控制器输出部分进行替换，将力和力矩模块进行注释，PWM 和 Mixer 模块取消注释，具体操作如下图所示。



重复上述 [步骤 1：固件编译](#)~[步骤 3：启动硬件在环仿真](#)，拨动遥控器上的 CH5 通道到高电位，即可解锁无人机，可以通过遥控器控制六旋翼无人机正常起飞。

## 6. 参考资料

[1] 无。

## 7. 常见问题

Q1: \*\*\*

A1: \*\*\*