

# 基于RePX4Block模块控制器切换实验

## 1. 实验目的

1. 在 控制器模型 (本例为 `Exp6_AttitudeSystemCodeGen3.slx`) 中正确接入 RePX4Block, 生成可下发的力/力矩控制。
2. 关闭平台侧对 PX4 的自动屏蔽 (如自动 Loiter/Arm 屏蔽), 保持控制链路直通。
3. 通过 RC 辅助通道 (建议 ch8) 实现 RePX4Block `mode` 0/2 切换, 完成 PX4 官方控制器与 Simulink 控制器的快速切换。
4. 验证切换日志: 在 QGC 中看到 "Switching to Simulink control!" / "Switching to PX4 control!" 提示, 确认固件替换生效。

## 2. 实验要求

- 软件要求: Windows 10 及以上版本; RflySim 工具链<sup>1</sup> (本实验使用1.13版本固件); 已正确安装 MATLAB/Simulink 及 Embedded Coder (版本需满足当前工程要求)。
- 硬件要求: 笔记本/台式电脑 1 台<sup>2</sup>, 以及 1 套 PX4 飞控 (硬件在环) 和配套遥控器 (至少 6 通道)。

## 3. 实验地址

例程目录: [\[RflySim 安装目录\]\RflySimAPIs\5.RflySimFlyCtrl\0.ApiExps\21.SwitchPX4Simulink](#)

- `Exp6_AttitudeSystemCodeGen3.slx`: 示例控制器模型, 含 RePX4Block 切换。
- `Init_control.m`: 初始化与编译入口脚本。
- `Firmware\6x_Start.bat`: 一键替换 `Firmware\src\modules\px4_simulink_app\PX4_TaskControl.c` 并编译固件, 提供切换提示日志。
- `Firmware\src\modules\px4_simulink_app\PX4_TaskControl.c`: 增加 uORB 订阅与 MAVLink 提示的定制文件。

# 4. 实验内容或步骤

## 4.1 步骤1: RePX4Block切换

打开设计好的Simulink控制器模型，这里是Exp6\_AttitudeSystemCodeGen3.slx

使用平台自带的TorqueThrustCtrls模块控制输出，勾选 PX4 的自动屏蔽选项，取消勾选自动 Loiter、自动 arm 等

For PX4 1.14 and above, the Ctrls will send uORB message: vehicle\_torque\_setpoint and vehicle\_thrust\_setpoint, and vehicle\_torque\_setpoint1 and vehicle\_thrust\_setpoint1 (2nd channel, for VTOL etc)

In configuration page:  
If "Auto arm" is checked, the module will send arm command when isEnCtrl is enabled  
If "Auto block PX4" is checked, the module will send signals to block PX4's torque and thrust (actuator\_controls\_0) message when isEnCtrl is enabled, which makes sure this Simulink controller can control the vehicle.  
If "Auto Loiter" is checked, the module will send cmd to unblock PX4 and switch to auto loiter mode when isEnCtrl is disabled, which makes sure the vehicle can switch to PX4 controller and stay in the air.

参数

3D Torque Mxyz + 1D Throttle or 3D Force

1D Thrust throttle T (T > 0 fly upward): Input: 4D Ctrls[4]  
 3D Thrust force Txyz (Txyz(3) < 0 fly upward): Input: 6D Ctrls[6]

Enable 2nd channel (for VTOLs): Ctrls should be 8D or 12D

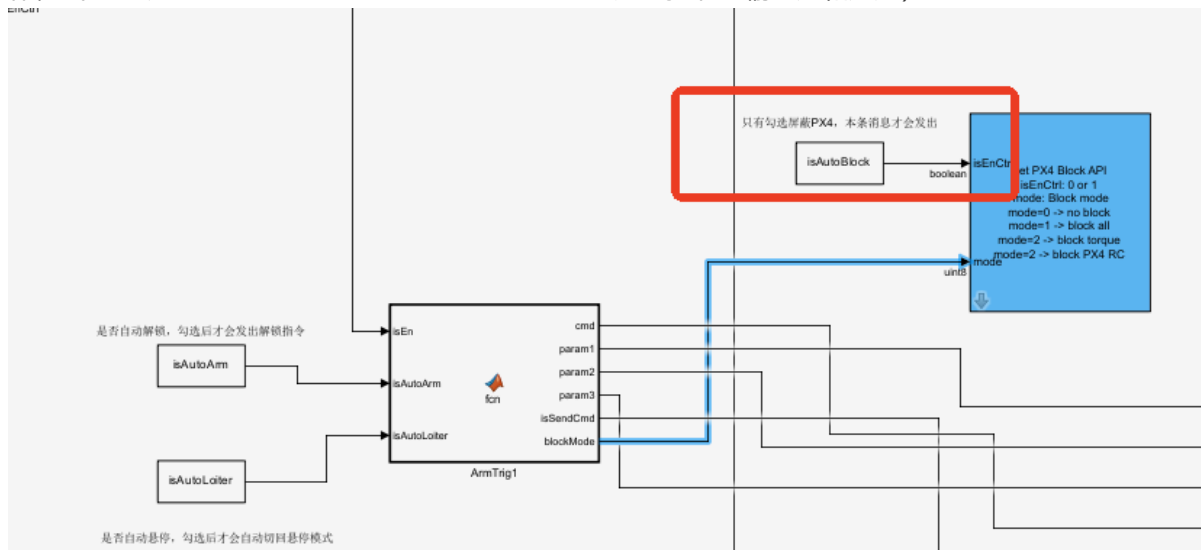
Option Functions

Auto arm  Auto block PX4  Auto Loiter

Sample Time (s) -1

注意: 本例程需要在安装脚本中, 禁用actuator\_controls\_0

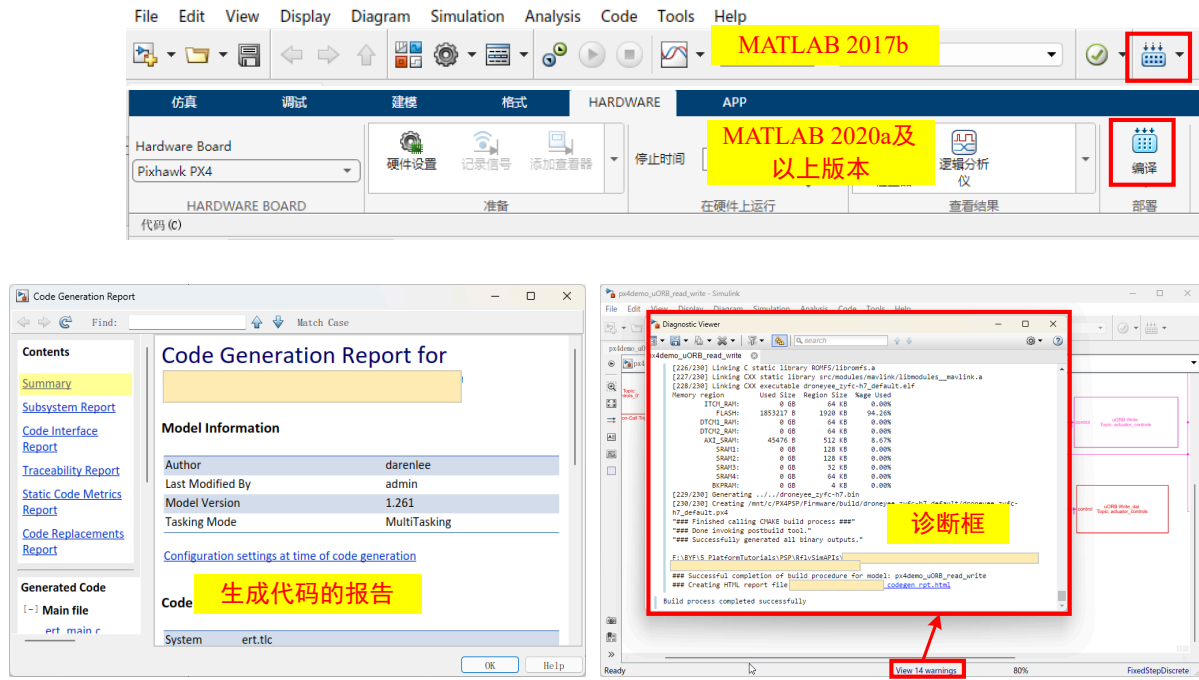
起到切换作用的是TorqueThrustCtrls模块内部的RePX4Block模块，isEnCtrl 这个端口需要输入一个布尔量（勾选 PX4 的自动屏蔽选项后这里给true，确保激活这个模块），mode端口需要用到uint8，这里只要取0、1用来切换PX4官方控制器和本Simulink生成的控制器，比如可以将PX4控制器中没有其它作用的一位遥控器辅助通道用来切换控制权（这里选择 Ch8 小于等于 1500: mode = 0 → PX4 官方控制器工作；当 Ch8 大于 1500: mode = 2 → 由 Simulink 生成的控制器输出力和力矩）。



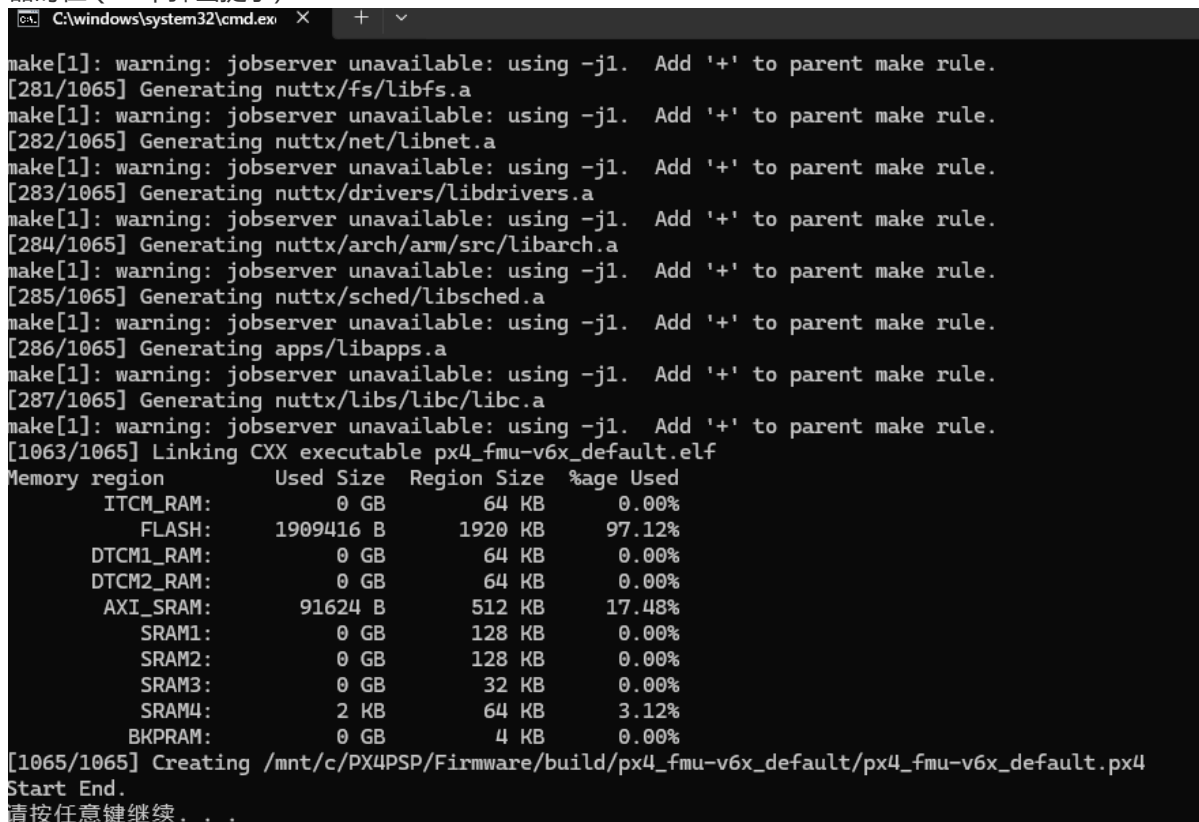
## 4.2 步骤2: HITL实验运行流程

### 飞控固件生成及烧录

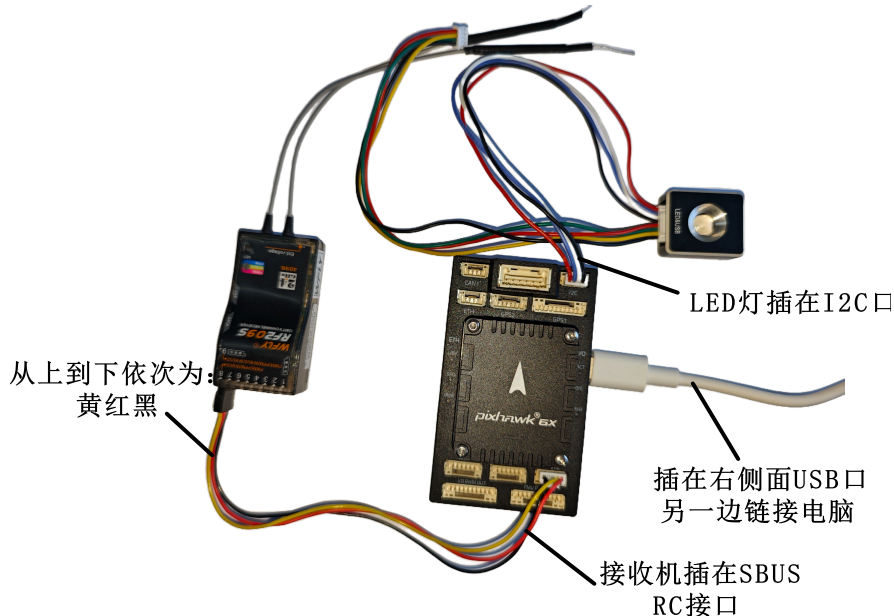
在 MATLAB 中打开 Init\_control.m 文件, 点击运行。按下图所示单击编译按钮, 对 Exp6\_AttitudeSystemCodeGen3.slx 文件进行编译, 编译完成后会生成代码报告, 在 Simulink 的下方点击 View diagnostics 指令, 即可弹出诊断对话框, 可查看编译过程。在诊断框中弹出 Build process completed successfully, 即可表示编译成功, 下图为生成的编译报告。



编译成功后, 双击运行 "Firmware\6x\_Start.bat" 替换掉修改过的 "Firmware\s\c\modules\px4\_simulink\_app\PX4\_TaskControl.c" 并重新编译固件 (目的是切换控制器时在 QGC 中弹出提示)



重新编译成功后，按下图所示用 USB 数据线连接飞控与电脑。



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。

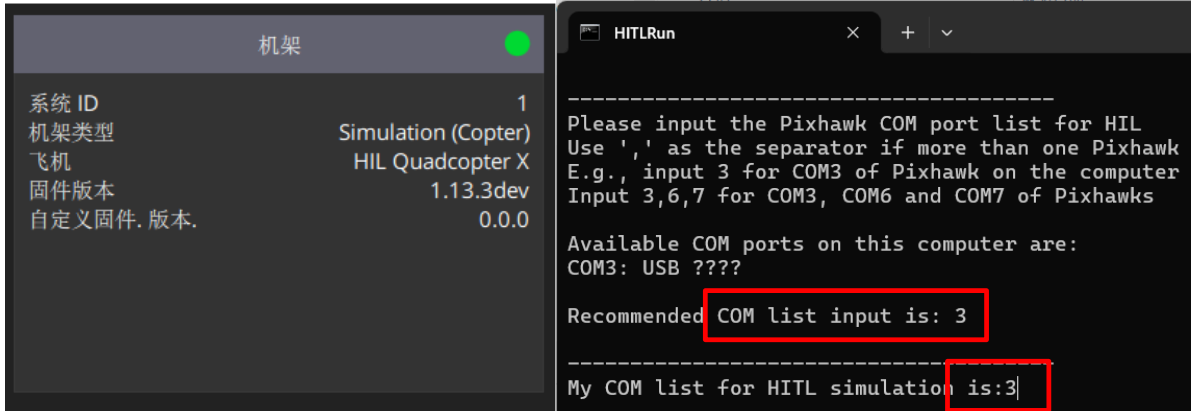


## 启动仿真环境

上传成功后，双击打开"\*\桌面\RflyTools\HITLRun.Ink"或"\*\PX4PSP\RflySimAPIs\HITL

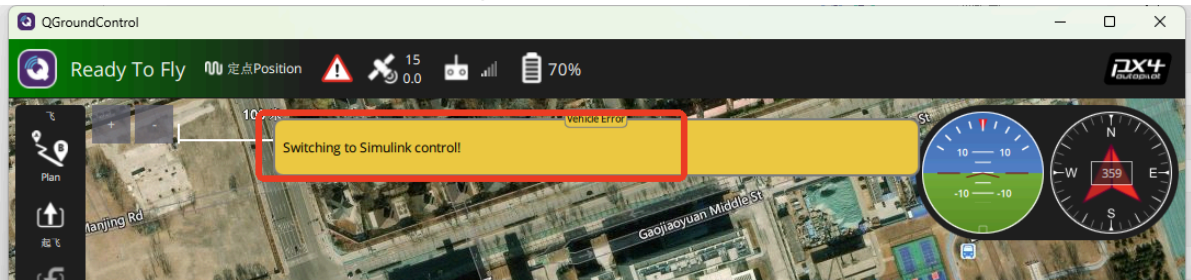
Run.bat"文件，在弹出的 CMD 对话框中输入插入的飞控 Com 端口号，即可自动启动 RflySim3D、CopterSim、QGroundControl 软件，等待 CopterSim 的状态框中显示：PX4: GPS 3D fixed & EKF initialization finished。

打开QGC地面站，连接上自驾仪，（1）进入“Airframe” 标签，确保模型处于“HIL Quadcopter X”机架模式（2）进入“Flight Modes”标签页，确认“Emergency Kill switch channel”切换开关为CH5。即可在QGroundControl 中设置飞机起飞等操作。

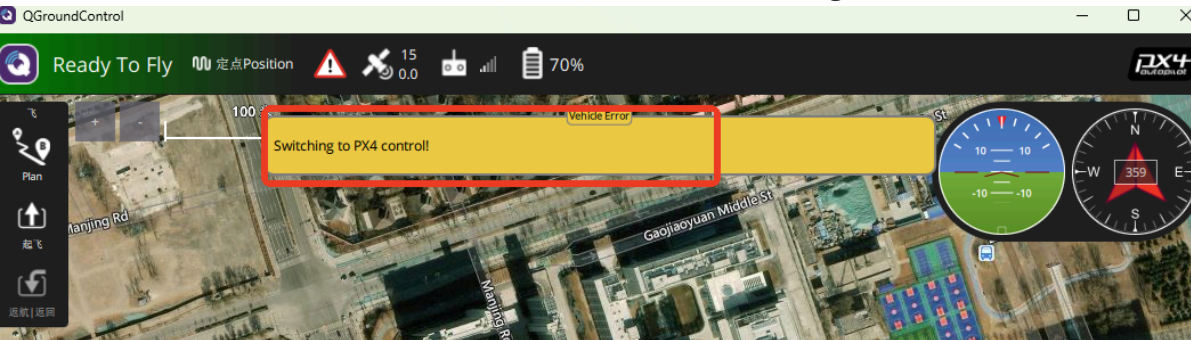


## 切换控制器

通过QGC解锁起飞之后（此时用的PX4控制器），在 RflySim3D 中即可看到飞机正常起飞，通过CH1~Ch4通道调整飞机姿态和高度。此时通过CH8通道进行0/1模式切换，即可切换至本Simulink生成的控制器，此时QGC中会弹出提示 Switching to Simulink control!



再次拨回Ch8，即可切换至PX4控制器，此时QGC中会弹出提示 Switching to PX4 control!



## 5. 关键知识点

### 5.1. 关键知识点1： HITL（Hardware-In-the-Loop）仿真

通过将真实飞控硬件接入仿真环境，实现“真机+虚拟环境”的闭环测试。

### 5.2. 关键知识点2： RePX4Block 的作用

在 Simulink 模型中与 PX4 控制接口对接，实现控制指令、模式切换等信号的接入与分发。

## 5.3. 关键知识点3: PX4 自动屏蔽机制

部分平台输出模块默认对 PX4 信号进行屏蔽, 需要在 Option Functions 中关闭相关选项, 保证飞控信号直通。

## 5.4. 关键知识点4: 遥控通道到逻辑模式的映射

利用 RC 通道的模拟量, 通过简单门限逻辑转换为 0/1 模式信号, 实现硬件开关控制软件模式。  
PX4\_TaskControl.c 的修改 (便于理解模式切换日志的实现) :

```
1 // 新增 uORB 订阅与 MAVLink 日志输出 (监听 actuator_armed_rfly)
2 static int rfly_logger_thread(int argc, char *argv[]) {
3     int sub = orb_subscribe(ACTUATOR_RFLY_ORB_ID);
4     orb_set_interval(sub, 100);
5     ...
6     if (msg.prearmed && !is_ctrl_rfly) {
7         mavlink_log_critical(&mavlink_log_pub, "Switching to Simulink
control!");
8     } else if (!msg.prearmed && is_ctrl_rfly) {
9         mavlink_log_critical(&mavlink_log_pub, "Switching to PX4 control!");
10    }
11 }
12
13 // 在任务回调里首次打印存活日志, 并启动上述日志线程
14 if (rfly_logger_task < 0) {
15     rfly_logger_task = px4_task_spawn_cmd("rfly_log", ..., rfly_logger_thread,
NULL);
16 }
```

更细节的差异点:

- 自定义 `actuator_armed_rfly` 结构与 ORB 宏: 补齐 uORB 声明, 方便在 PX4 mixer 中的 `prearmed` 标志变化被订阅并触发提示。
- 新增全局句柄: `mavlink_log_pub` 用于输出到 QGC, `rfly_logger_task` 用于确保日志线程只启动一次。
- 线程逻辑:
  - `orb_set_interval(sub, 100)` 降低轮询频率; `px4_poll` 超时 500 ms, 避免 CPU 忙轮询。
  - 监听 `prearmed` 上升/下降沿分别打印 "Switching to Simulink control!" / "Switching to PX4 control!"。
  - 变更时额外输出 `armed/prearmed/ready_to_arm/lockdown` 状态行, 便于调试。
- 任务回调增强: 首次进入打印 "simulink task alive", 并懒加载日志线程 (spawn 失败时输出 critical)。其余调度、Ext-Mode、关停逻辑与备份版一致。

## 6. 参考资料

1. [RflySim 官方文档](#)
2. [PX4 官方文档](#)
3. [RflySim安装路径]\RflySimAPIs\1.RflySimIntro\2.AdvExps\2.FCUIntro (飞控固件恢复与基础 HITL 流程)

# 7.常见问题

---

## Q1: 如何实现PX4控制器与Simulink控制器之间的切换?

---

A1: 在控制器模型(例如Exp6\_AttitudeSystemCodeGen3.slx)中使用RePX4Block模块来实现控制器切换功能。具体方法是:

- 使用RePX4Block模块的mode端口, 该端口接受uint8类型的值
- 通常设置mode=0时启用PX4官方控制器, mode=2时启用Simulink生成的控制器
- 可以使用遥控器的一个辅助通道(如CH8)作为切换开关: 当CH8小于等于1500时设为mode=0, 当CH8大于1500时设为mode=2
- 同时需要确保isEnCtrl端口输入true以激活模块

## Q2: 如何在切换时看到日志提示信息?

---

A2: 为了让控制器切换时能在QGroundControl(QGC)中看到明确的日志提示, 需要进行以下操作:

- 编译Simulink模型后, 运行Firmware\6x\_Start.bat脚本来替换并重新编译固件
- 修改后的固件包含在PX4\_TaskControl.c文件中增加了uORB订阅和MAVLink日志输出功能
- 当控制器切换发生时, 将在QGC中看到"Switching to Simulink control!"或"Switching to PX4 control!"的提示信息

## Q3: 如何正确配置飞控固件和硬件环境?

---

A3: 实验需要正确的软硬件配置才能正常运行:

- 软件要求: Windows 10及以上操作系统, RflySim工具链(推荐1.13版本固件), MATLAB/Simulink及Embedded Coder(需满足工程版本要求)
- 硬件要求: 一台电脑, 一套PX4飞控(硬件在环)和至少6通道的配套遥控器
- 固件烧录流程: 编译Simulink模型→运行6x\_Start.bat重新编译固件→使用USB数据线连接飞控与电脑→在MATLAB中运行PX4Upload命令上传固件到飞控

---

1. <https://rflysim.com/> 

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