



智能无人集群系统开发与实践

基于RflySim工具链的全栈开发案例

第2讲 实验工具链配置



课程资源

- 本讲PPT公益课视频地址为：
- 软件的使用与配置：<https://www.bilibili.com/video/BV1UL4y1F7NL>
- 硬件的使用与配置：<https://www.bilibili.com/video/BV1qY4y187NZ>
- 更多课程资源：https://space.bilibili.com/3493283546269949?spm_id_from=333.1007.0.0
- 或使用手机扫码观看：



软件的使用与配置



硬件的使用与配置



RflySim更多教程



大纲

1. 工具链安装包获取
2. 软、硬件简介和使用
3. 基础实验例程
4. 进阶实验例程
5. 后续章节学习方法
6. 总结



飞思实验室



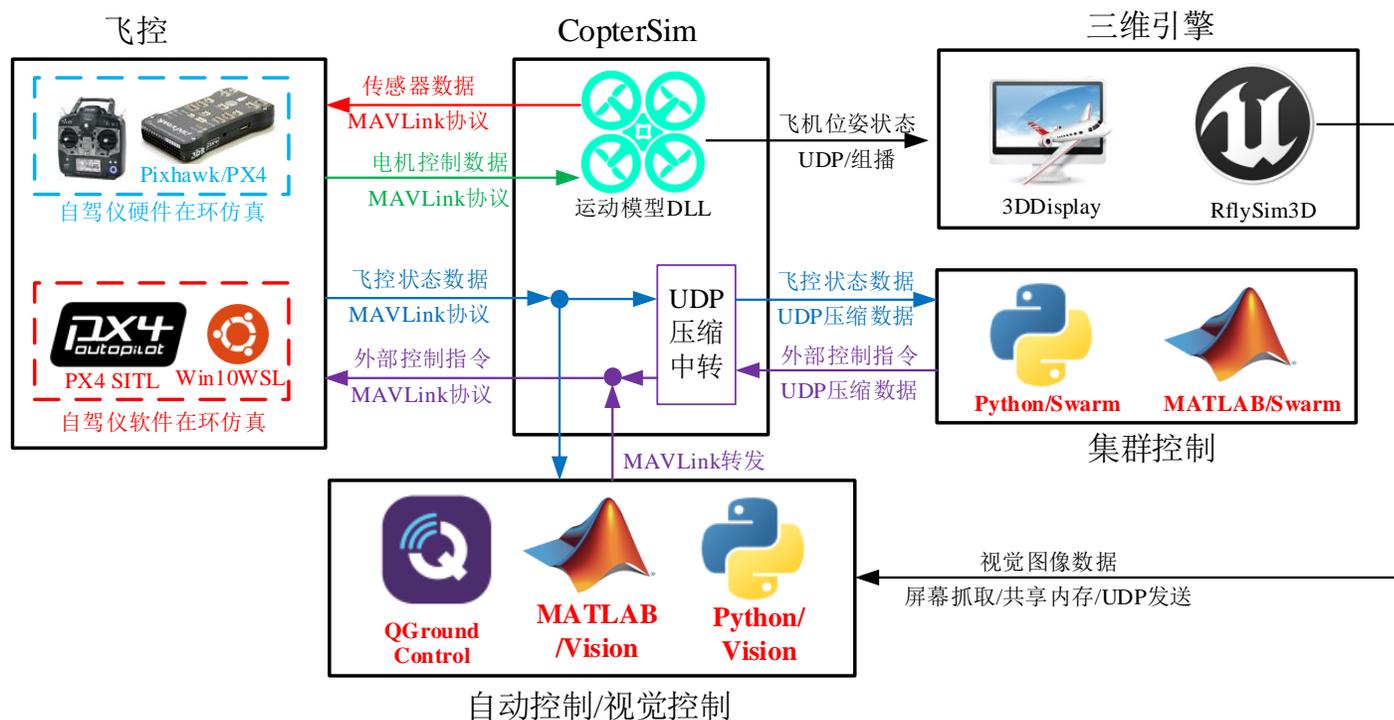
RflySim教程



1.工具链安装包获取

1.0 RflySim工具链简介:

RflySim是一套专为科研和教育打造的Pixhawk /PX4和MATLAB/Simulink生态系统或工具链，采用基于模型设计（Model-Based Design, MBD）的思想，可用于无人系统的控制和安全测试。





1.工具链安装包获取

1.1 安装包获取

RflySim工具链下载请到：<https://rflsim.com/download.html> 中，如右图所示，填写相关信息后即可直接获取下载链接，具体安装步骤可见安装包内HowToInstall.pdf文件。（付费版本，请咨询 service@rflsim.com ，详见：[RflsimVersions.pdf](#)）。

5. 软件下载

注意：当前为中文页面，发送的下载链接为中文版，如需英文版，请切换英文页面

请填写好联系方式，我们会将最新的【RflySim工具链】(安装包: RflySimToolchain-v3.04-20240619.iso, 在线文档: <https://rflsim.com>)下载链接发送到您的邮箱。由于邮箱回复存在一定延迟，请您耐心等待几分钟再去检查收件箱；如果您没有收到邮件，请检查邮箱的垃圾箱；如果仍然没有发现邮件，请您更换邮箱地址或者直接发送邮件到 service@rflsim.com 来获取下载链接。

您也可以微信扫码右侧二维码，关注飞思实验室官方公共号，发送咨询文字来获取下载链接、加入技术交流群或咨询购买完整版平台等。



姓名 *

邮箱 *

公司 *

主题

留言 *

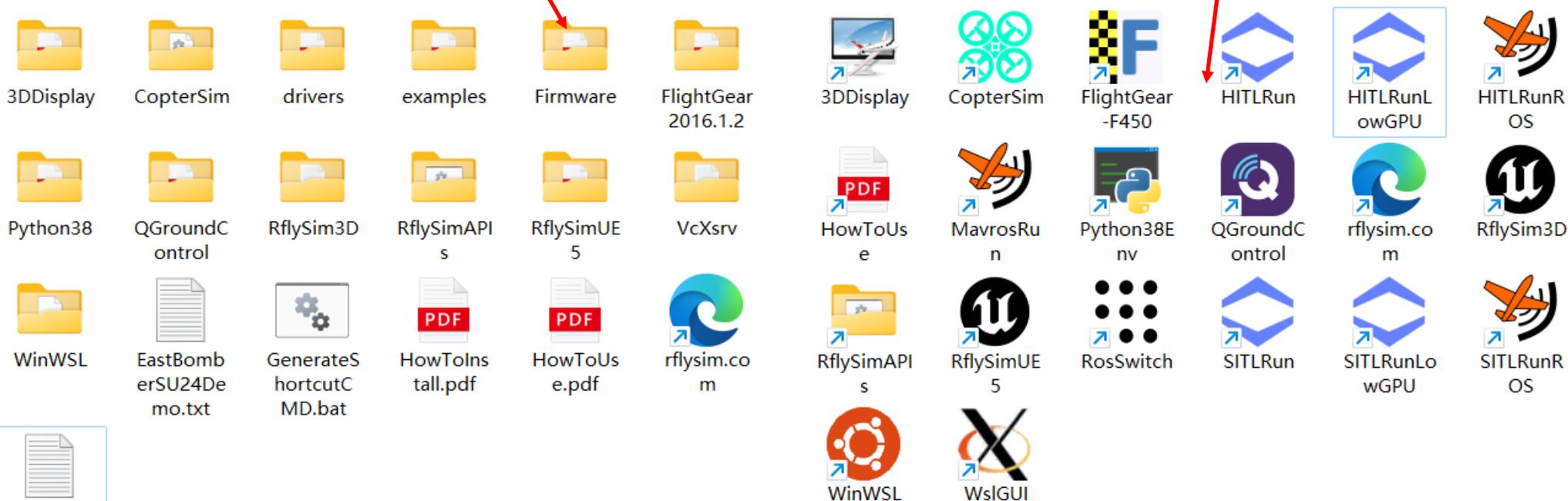
获取下载链接



1.工具链安装包获取

1.2 安装后Windows效果

- 如下图所示，在安装目录（默认是C:\PX4PSP）下可以得到一系列的文件夹，其中“**RflySimAPIs**”文件夹是高级功能的接口教程文件夹，最为重要。
- 如右图所示，在桌面RflyTools文件夹内可以得到一系列的快捷方式。



RflySimAdv3Full-V3.04-20240605.txt

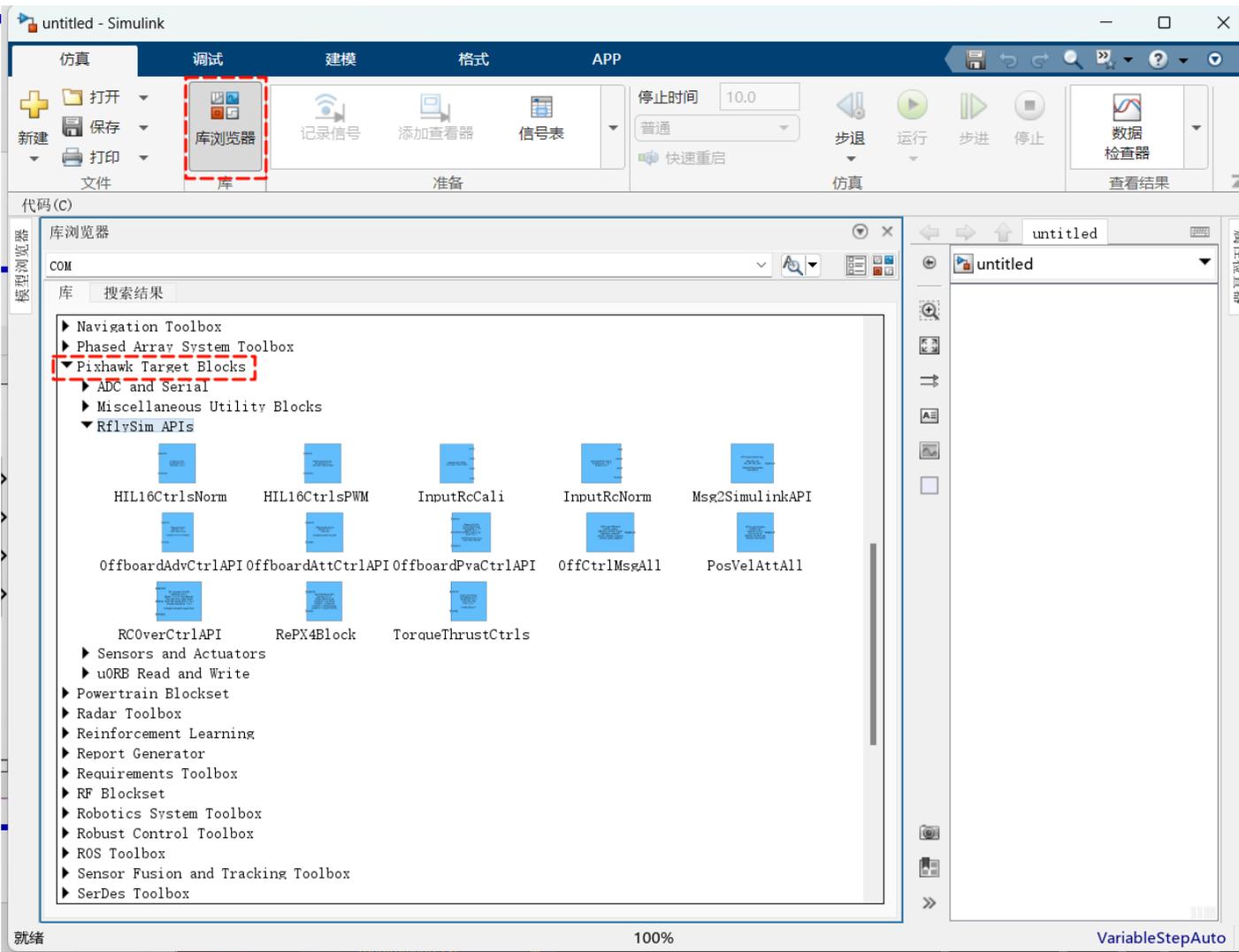


1.工具链安装包获取

此功能仅面向底层飞控算法开发用户，视觉和集群算法开发用户不需要查看。

1.2 安装后Windows效果

- 打开MATLAB，任意新建一个Simulink程序，进入库浏览器（Library browser）页面。
- 如右图，向下翻可以看到Pixhawk Target Blocks的工具箱，说明安装成功。
- 本功能针对底层飞控算法开发，支持Simulink设计飞控算法，并生成代码上传到Pixhawk中，进行硬件在环仿真和真机实验。

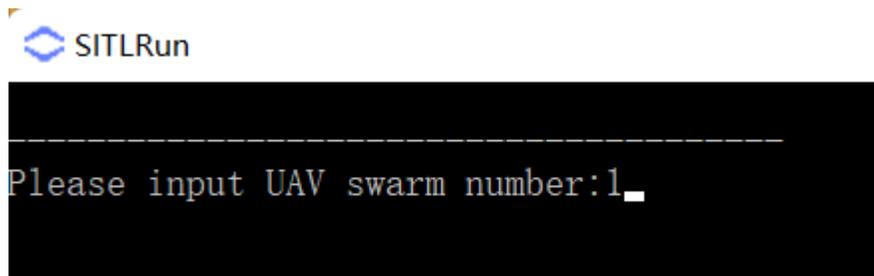
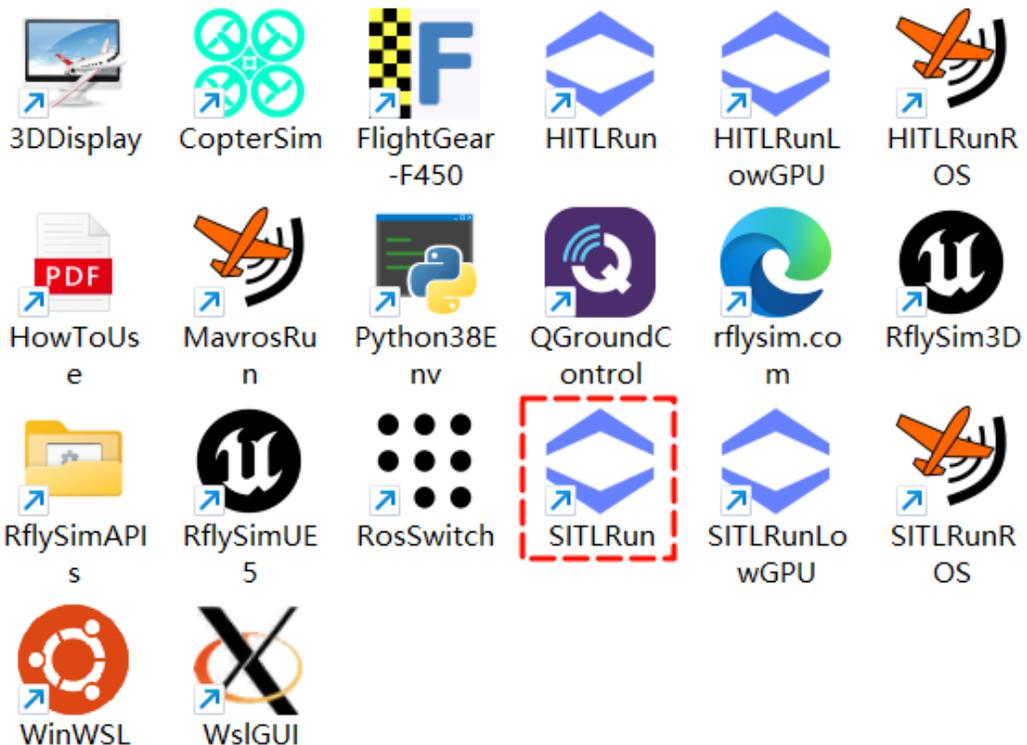




1.工具链安装包获取

1.2 安装后Windows效果

- 进入桌面“RflyTools”文件夹，双击“SITLRun”快捷方式，并输入1，再回车。
- 等到RflySim3D显示“*** EKF 3DFixed”（CopterSim上也会显示），表示飞控已经初始化完毕，可以开始控制自主飞行。



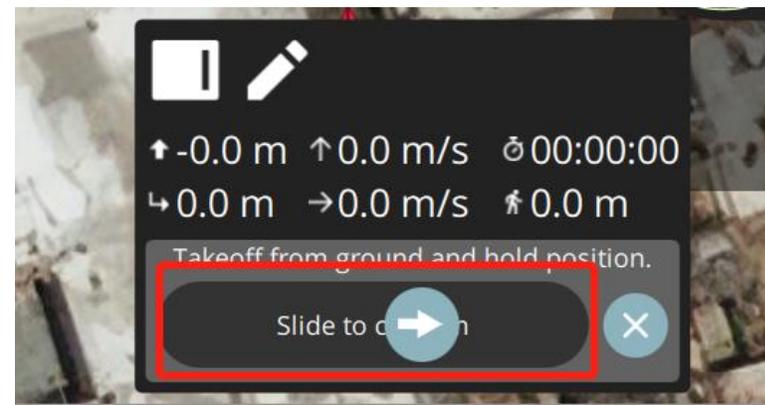
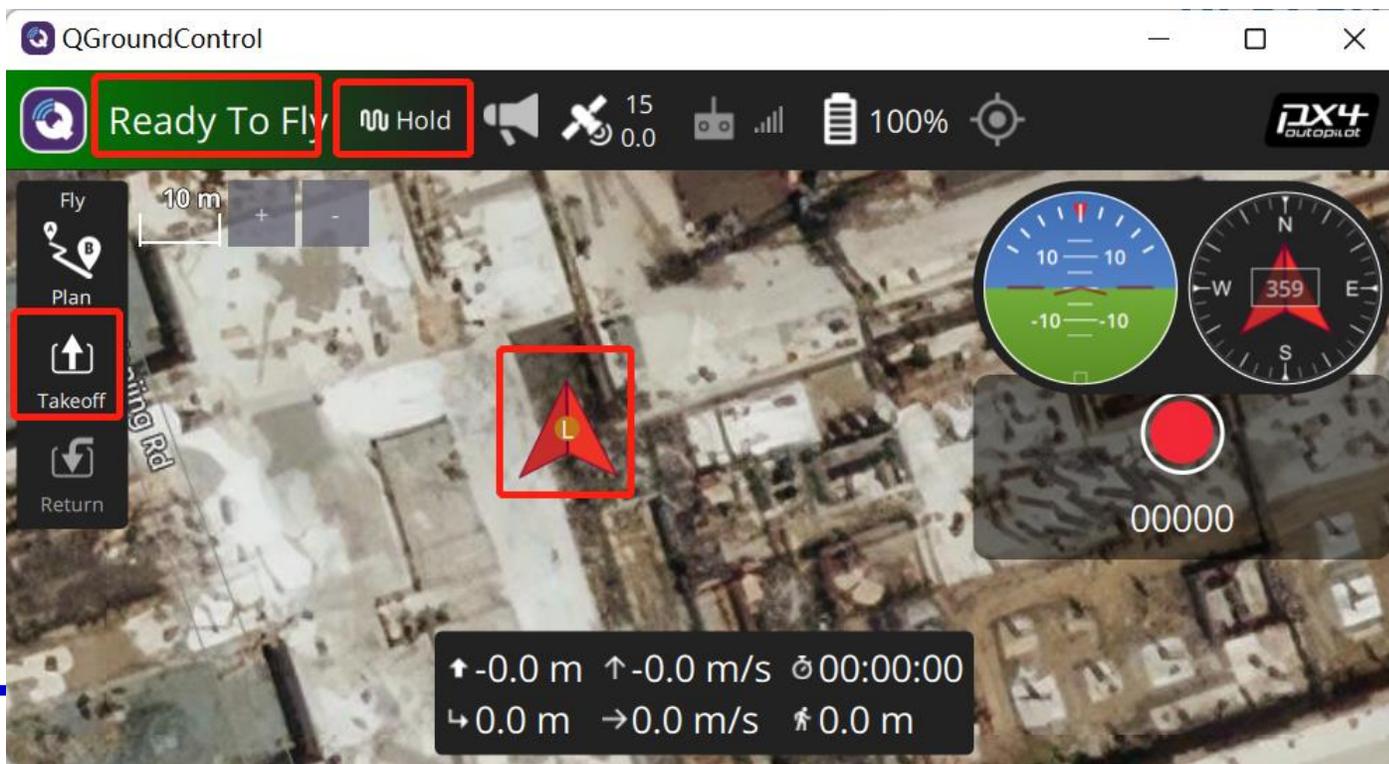


1.工具链安装包获取

注：如果直接运行bat脚本出现飞机抖动，请用鼠标右键并使用管理员方式打开bat脚本！这样可以获取更高的运行优先级。

1.2 安装后Windows效果

- 进入QGroundControl软件，看到飞机进入“Hold”模式，点击“Takeoff”按钮。
- 会弹出确认滑块，将其拖到最右侧，开始自动起飞。
- 如果飞机能离地起飞，说明工具链配置正确。





大纲

1. 平台的简介与安装
2. 软、硬件简介和使用
3. 基础实验例程
4. 进阶实验例程
5. 后续章节学习方法
6. 总结



飞思实验室



RflySim教程



2. 软、硬件简介和配置

2.1 软硬件总体介绍

- RflySim工具链作为一个工具链，包含了众多运用于无人系统开发的软件，如：Python 3.8环境、MATLAB / PSP工具箱、FlightGear、QGroundControl等；同时，针对无人系统的自主开发的RflySim3D、CopterSim等。
- 除了软件系统，RflySim硬件系统有：地面计算机/机载计算机、机架系统、动力系统、遥控器系统、自驾仪系统等等。

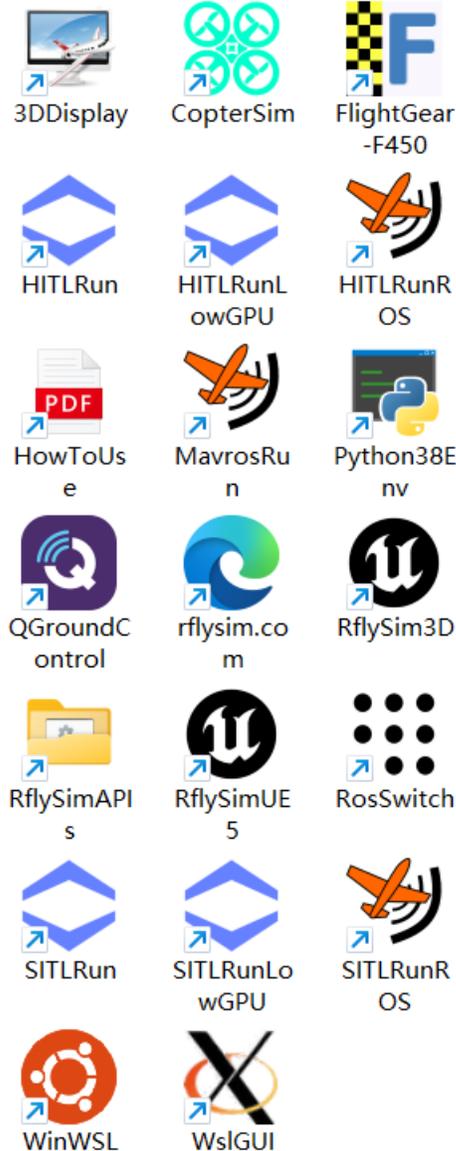




2. 软、硬件简介和配置

2.2 RflySim核心软件介绍

- **3DDisplay**: 该软件为RflySim工具链初始版本中使用的三维显示软件，可以适配更低性能的电脑，但是仿真效果欠佳。
- **CopterSim**: 本工具链核心仿真软件，运行多旋翼运动动态模型，并连同其他软件构成软/硬件在环仿真。
- **FlightGear飞行模拟器**: 一款非常受欢迎的开源飞行模拟器软件，可以通过UDP接收Simulink发送的飞行状态，方便地观测Simulink仿真时飞机的飞行状态。
- **HILRun/SILRun、HITLRunLowGPU/SITLRunLowGPU**: 一键快速启动脚本，可以快速开启所有视觉/集群相关软件，并完成所需配置。若运行过程中卡顿，请使用HITLRunLowGPU/SITLRunLowGPU一键启动脚本。
- **SITLRunROS/HITLRunROS**: 启动指定数量的SITL/HITL飞机，并开启对应的mavros节点。
- **MavrosRun**: 快速启动指定数据的mavros节点。先运行SITLRun\HITLRun开启指定数量的飞机（使用MAVLink_Full）通信模式，然后再运行MavrosRun.bat并输入飞机数据，就能开启对应数量的mavros节点。

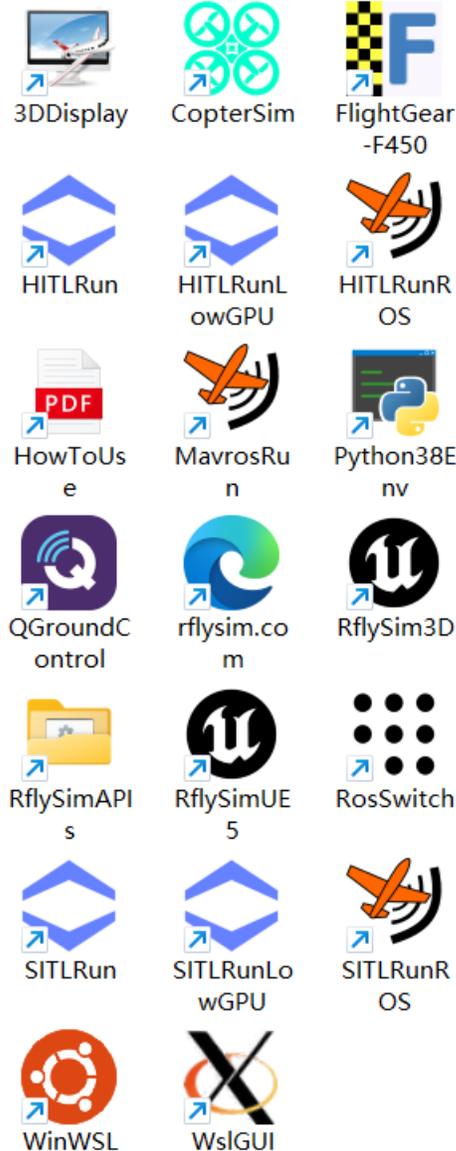




2. 软、硬件简介和配置

2.2 RflySim核心软件介绍

- **Python38Env**: RflySim工具链中内置的Python 3.8版环境启动快捷方式，包含OpenCV等库的一个Python环境。
- **QGrounControl (QGC) 地面站**: 包含配置飞控参数和控制飞机起飞、降落、航线等功能。用户可以阅读如下网址来学习软件使用:<https://docs.qgroundcontrol.com/master/en/index.html>
- **rflysim.com**: RflySim工具链官方网站快捷链接。
- **RflySim3D/RflySimUE5**: 本工具链核心三维显示软件，基于Unreal Engine 4 (UE4, 虚幻4, 完整版支持UE5) 引擎开发，具备高逼真虚拟现实显示效果。
- **RflySimAPIs例程文件夹**: 包含了本课程的所有例程和源码，覆盖了单/多机控制、集群飞行、视觉控制等。注: 该文件夹中包含有旧版RflySim工具链中的PPT文件夹所有文件。
- **RosSwitch**: WinWSL环境的ROS一键切换脚本，双击RosSwitch输入1或2来切换ROS1或ROS2环境。

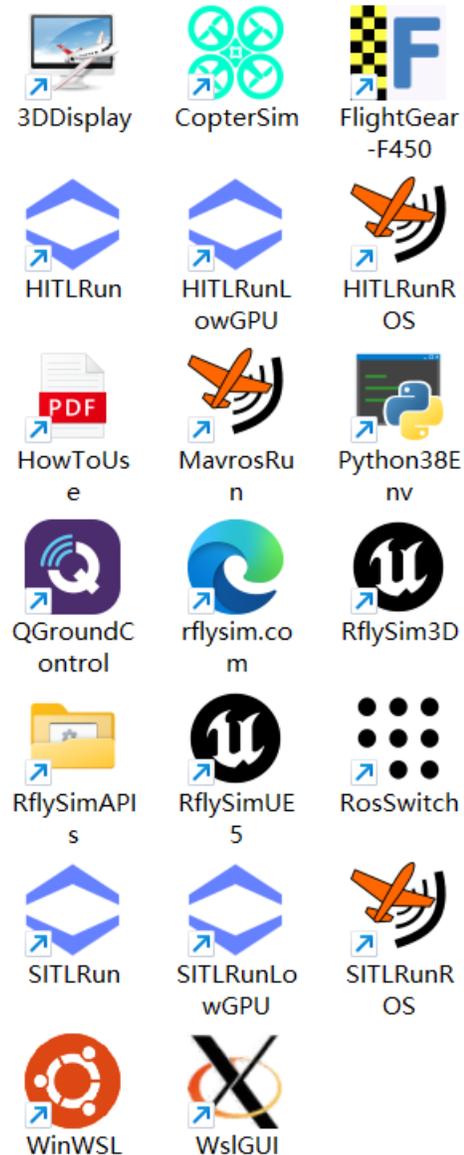
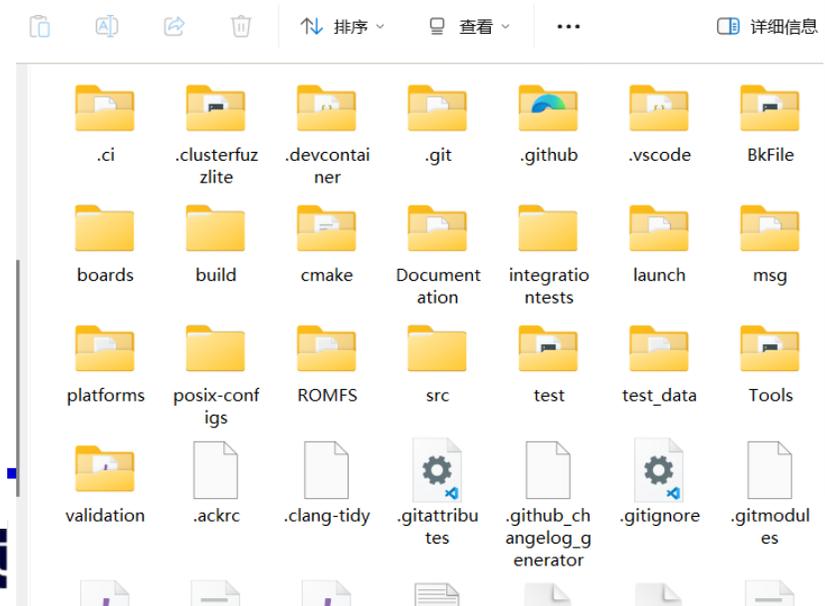
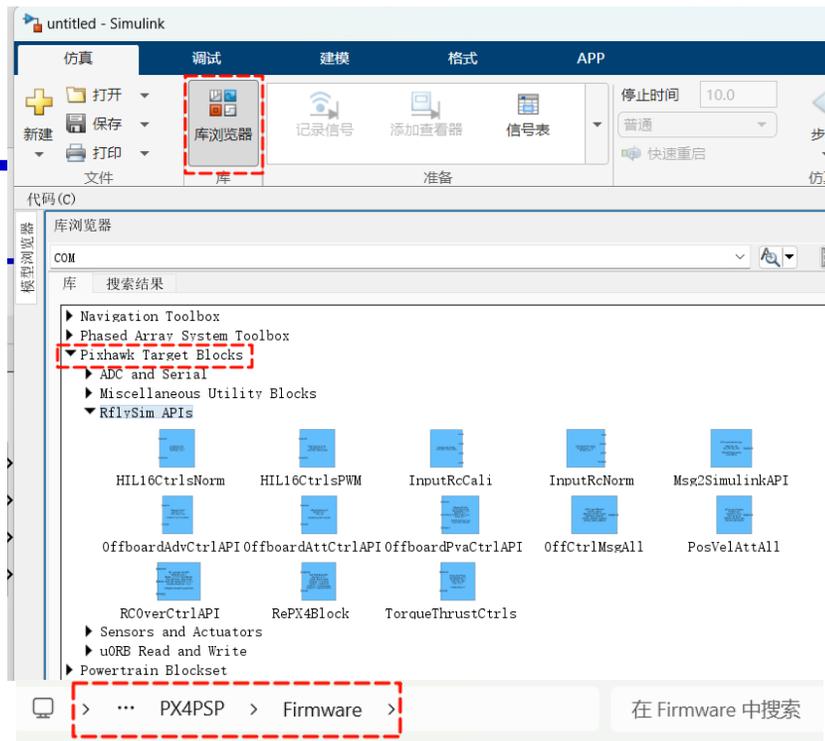




2. 软、硬件简介和配置

2.2 RflySim核心软件介绍

- **WinWSL编译器**：用于编译固件和软件在环仿真。
- **WslGUI**：WinWSL系统的可视化显示程序。
- **Pixhawk Support Package (PSP) 工具箱**：Mathworks公司官方为Pixhawk自驾仪推出一个工具箱，用于将Simulink中设计的控制算法生成C代码并编译上传到Pixhawk自驾仪硬件。
- **PX4 Firmware固件源代码**：PX4 是一款开源飞行控制软件系统，它运行在Pixhawk系列自驾仪硬件平台上，构成了Pixhawk PX4自驾仪软硬件平台，是目前世界范围内广泛应用的开源无人机自驾仪。
- **VS Code/Eclipse/VS**：用于代码阅读、编辑与编译。





2. 软、硬件简介和配置

2.3 RflySim核心硬件介绍

- 地面计算机/机载计算机。
- 机架系统：机臂、机身、起落架等。
- 动力系统：电机、电调、电池、螺旋桨等
- 遥控器系统：遥控器发射器、接收机、充电器、锂电池等。
- 自驾仪系统：自驾仪/飞控、GPS模块、电源模块、USB数据线、数传模块、机载AI视觉/集群计算机。
- 动捕系统：完整的动捕系统，可用于算法验证、运动规划、集群控制、人机交互、轨迹回放、步态分析、智慧沙盘等。



机架系统



动力系统





2. 软、硬件简介和配置

2.4 软、硬件系统关系与配置

- 工具链除了软件，还包含了硬件在环仿真和真机实验的部分。
- 底层飞控算法开发的流程为：底层飞控开发 → MATLAB软件仿真 → 自动代码生成 → 飞控软件在环仿真 → 飞控硬件在环仿真 → 室外真机实验。

底层飞行控制算法开发



自动代码生成与固件编译



硬件在环仿真



室外或室内实验





2. 软、硬件简介和配置

2.4 软、硬件系统关系与配置

一键安装脚本的作用主要有以下几点：

- ① **首次安装时**，将工具链一键部署到系统中（使用默认配置，全选“是”即可），并完成相关配置。
- ② **后续使用中**，再次运行安装脚本，可以修改编译命令、编译器、固件版本、还原软件等。（不需要还原的项目选择“否”，会根据情况更新配置，节省时间）
- ③ **下载新安装包后**，直接运行安装脚本（选择“自动”，会需要更新的内容），再点击确认，开始升级。





2. 软、硬件简介和配置

一键安装脚本选项详解：

1) **工具包安装路径**。本工具链的所有依赖文件都会安装在本路径下，大约需要70G的空间。默认安装路径是“C:\PX4PSP”，如果C盘空间不够可以选择其他盘符下的路径。注意：路径名称必须正确，且只能用纯英文的路径，否则会导致编译失败。

2) **PX4固件编译命令**。主要对应底层控制器开发需求，并使用代码生成功能，需要根据飞控硬件来选择编译命令（注：顶层视觉和集群算法开发用户不需要配置，保持默认即可）。默认为“px4_fmu-v6x_default”对应Pixhawk V6X或Pixhawk V6X mini飞控。除此之外，RflySim工具链将长期支持所有可适配PX4软件系统的飞控，具体的长期适配的飞控型号及编译命令可见：

<https://rflysim.com/doc/zh/B/2.Pixhawk.html>。

注：第一次安装完成后，除了重新运行本安装脚本，另一种针对不同的Pixhawk硬件板子想更换不同的编译命令（例如换成px4_fmu-v3_default）的方法，只需要在MATLAB中输入命令：PX4CMD('px4_fmu-v3_default') 或者使用命令：PX4CMD px4_fmu-v3_default



```
fx >> PX4CMD px4_fmu-v3_default
```



2. 软、硬件简介和配置

一键安装脚本选项详解

3) **PX4固件版本**。PX4源代码每年都会进行更新，目前最新的固件版本为1.14。随着固件版本的升级，功能会逐渐增加，支持的新产品也越多，但是对旧的一些自驾仪硬件的兼容就会变差。本实验课程推荐使用Pixhawk V6X飞控，对应的编译指令为“px4_fmuv6x_default”，选用的固件版本**PX4-1.13.2**。

4) **PX4固件编译器**。由于PX4源代码的编译依赖于Linux编译环境和相关组件，本工具链提供了三套编译环境来实现Windows平台下对Linux编译环境的模拟，它们分别是：基于Windows Subsystem for Linux (WSL) 的编译环境WinWSL编译器、基于Msys2的Msys2Toolchain编译环境和基于Cygwin的CygwinToolchain编译器。注意，如果需要编译 \geq PX4-1.8版本以上固件，请选择CygwinToolchain编译器；编译 \leq PX4-1.8版本的固件，可选择Msys2Toolchain编译器。基于Msys2或Cygwin的本地编译器，而且部署方便，但是编译效率较低。对于Windows 10 1903及以上的系统版本，推荐安装WinWSL编译器，这种方式可以大大加快编译速度，而且兼容所有版本的PX4飞控固件。

3.PX4固件版本 (1: PX4-1.7.3, .., 6: PX4-1.12.3, 7: PX4-1.13.2, 8: PX4-1.14.*)

6

4.PX4固件编译器 (1: WinWSL编译器[\geq 1.9], 2: Msys2编译器[\leq 1.8], 3: Cygwin编译器[\geq 1.8])

1



2. 软、硬件简介和配置

一键安装脚本选项详解

5) **是否全新安装PSP工具箱**。如果该选项设置为“是”，会将PSP工具箱安装在本地MATLAB软件中。如果PSP工具箱已经安装过，则会对PSP工具箱进行全新安装。如果选择“否”，脚本对PSP工具箱不做任何更改（不会卸载掉安装的PSP工具箱或其他动作）。

6) **是否全新安装其他依赖程序包**。如果该选项设置为“是”，会将QGC地面站、CopterSim、RflySim3D、RflySimAPIs等软件部署在设定的安装路径上，并安装Pixhawk硬件的相关驱动程序，以及在桌面生成这些软件的快捷方式。如果安装路径上已经部署过相关依赖软件，选择“是”则会删除旧的安装包并进行全新重新安装。如果该选项设置为“否”则不做任何修改。

7) **是否全新配置固件编译器编译环境**。如果该选项设置为“是”，会将选定的编译器（WinWSL、CygwinToolchain或Msys2Toolchain）部署在设定的安装路径上，如果环境已经存在，则会清空旧的编译环境，进行还原与全新部署。反之，如果该选项设置为“否”则不会进行任何更改。

5.是否重新安装PSP工具箱(是：重装工具箱，否：跳过，自动：仅更新)

是

6.是否重新安装其他依赖程序包（CopterSim、RflySim3D、RflySimAPIs等）

是

7.是否重新配置固件编译器编译环境（是：全新安装编译器，否：跳过，自动：仅更新）

是

注：你也可以直接在(6)中指定个别你想要重装的应用名字（逗号分隔），可用选项包括：

CopterSim,drivers,FlightGear,QGroundControl,RflySim3D,RflySimAPIs,UE3DDisplay



2. 软、硬件简介和配置

一键安装脚本选项详解

8) 是否重新部署PX4固件代码。如果该选项设置为“是”，会将选定的PX4 Firmware源代码部署在设定的安装路径上，如果固件存在，会删除旧的固件文件夹，并进行全新部署。如果该选项设置为“否”则不会进行任何更改。

9) 是否全新编译固件。如果该选项设置为“是”，会对部署固件进行预编译，这样可以大大节省后续代码生成与编译的时间，同时可以检测环境安装是否正常。如果该选项设置为“否”则不会进行任何更改。

10) 是否屏蔽PX4自身控制器输出。如果该选项设置为“是”，会对Firmware中对电机的控制信号进行屏蔽，防止与生成代码发生冲突（注：本选项不会屏蔽PX4_SITL控制器的输出，因此可以正常进行软件在环仿真）。如果选择“否”，则不会进行对固件输出进行屏蔽，可以用于测试PX4自带的控制算法，因此如果要生成官方固件，本选项请选择“否”。

8.是否重新部署PX4固件代码（是：全新部署代码，否：跳过，自动：仅更新）

是

9.是否预先用选定命令编译固件（是：全新编译固件，否：跳过，自动：仅更新）

是

10.是否屏蔽PX4官方控制器输出(使用Simulink控制器选"是",使用PX4官方控制器选"否")

是



2. 软、硬件简介和配置

2.5 基于模型开发流程简介

(1) 软件在环仿真阶段

整个阶段都在MATLAB环境下进行，利用给定多旋翼仿真模型和例程，在Simulink中进行控制算法设计，并正确连接模型和控制器，确保输入输出信号与实际多旋翼系统一致。类似于实际多旋翼系统，多旋翼模型将传感器数据或状态估计信息（例如，姿态角、角速率、位置和速度等）发送给控制器，控制器将每个电机PWM控制指令发回给模型，从而形成一个软件在环仿真闭环系统。在本阶段，读者可以观察控制性能，自行修改或设计控制器来达到期望的性能需求。

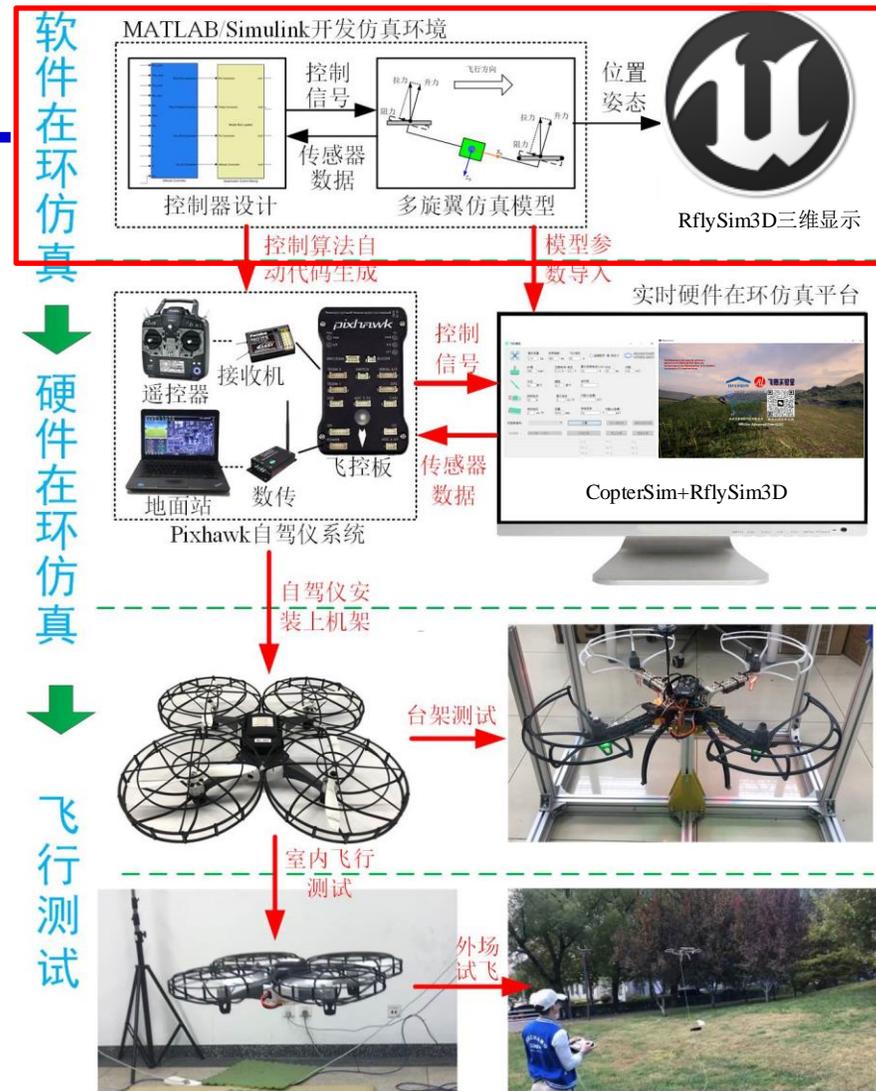


图. 实验流程图



2. 软、硬件简介和配置

2.5 基于模型开发流程简介

(2) 硬件在环仿真阶段

利用给定的模型和例程，进行实验。模型在硬件在环多旋翼飞行器仿真器里，而控制器上传到Pixhawk飞控硬件环境下，其中通讯过程是通过串口线直接连接。模型通过串口线将姿态角、姿态角速率、位置和速度发送给控制器，控制器通过串口线将每个电机PWM控制指令发回给模型，从而形成一个闭环。

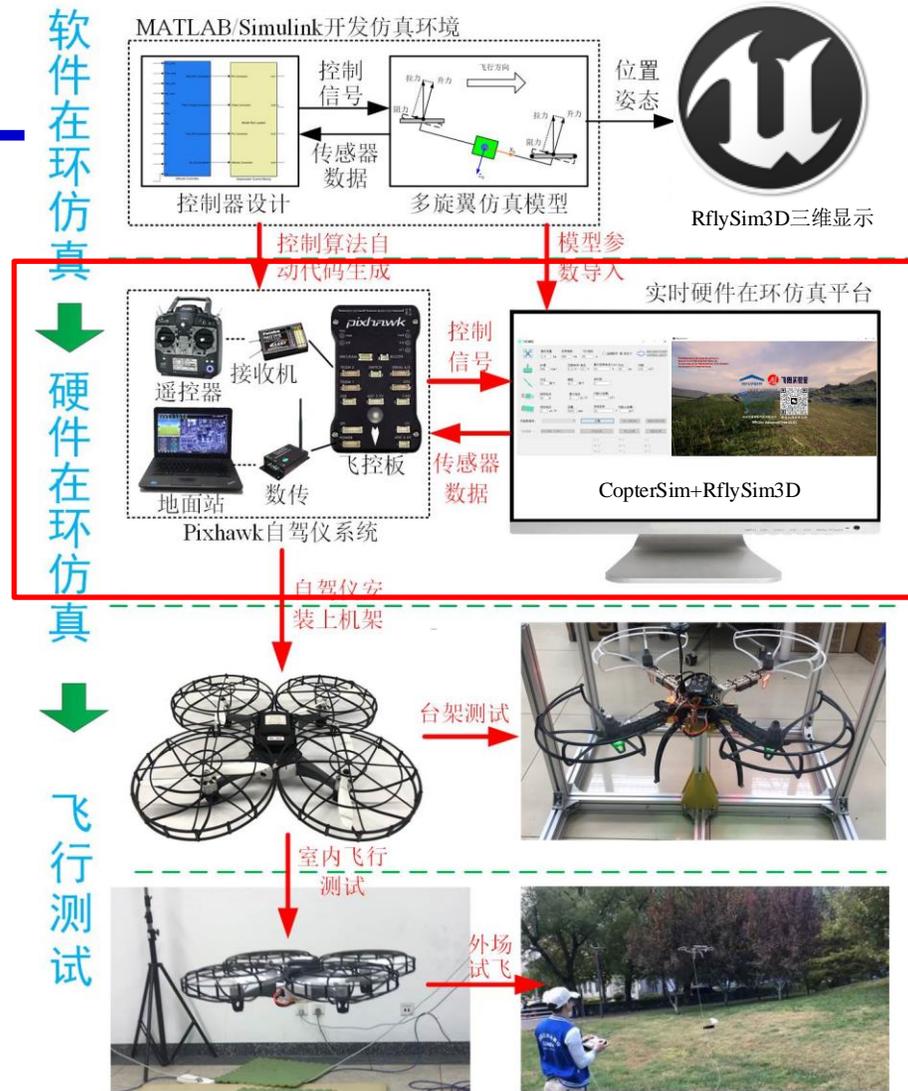


图. 实验流程图



2. 软、硬件简介和配置

2.5 基于模型开发流程简介

(2) 硬件在环仿真阶段

将Simulink多旋翼模型参数导入到CopterSim中，并将Simulink控制器算法生成代码下载到Pixhawk自驾仪，然后用USB实体信号线替代Simulink中的虚拟信号线。CopterSim将传感器数据（例如，加速度计、气压计、磁力计等）通过USB数据线发送给Pixhawk系统；Pixhawk系统中的PX4自驾仪软件将收到传感器数据进行滤波和状态估计，将估计的状态信息通过内部的uORB消息总线发送给控制器；控制器再通过USB数据线将每个电机的PWM控制指令发回给CopterSim，从而形成一个硬件在环仿真闭环。

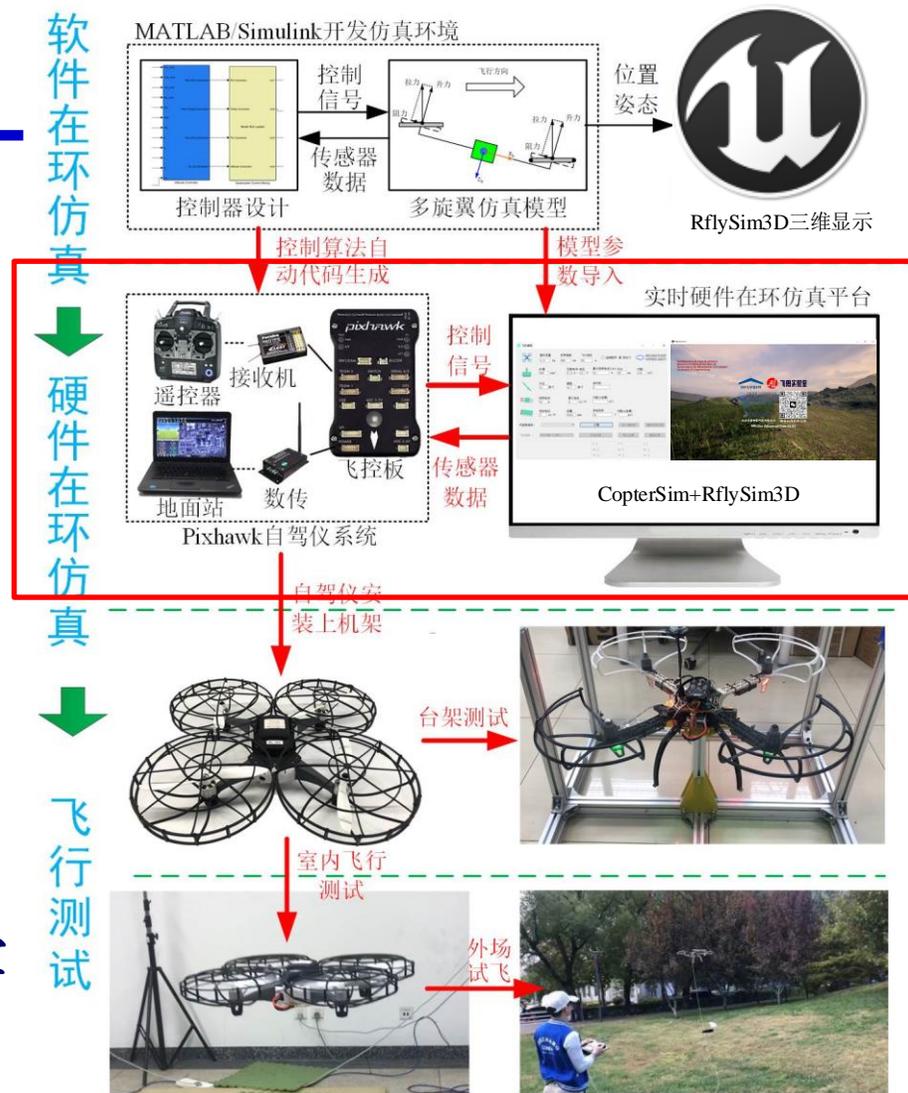


图. 实验流程图



2. 软、硬件简介和配置

2.5 基于模型开发流程简介

(2) 硬件在环仿真阶段

相对于软件在环仿真，硬件在环仿真中多旋翼模型运行速度与实际时钟是一致的，以此保证仿真的实时性，同时控制算法可以部署并运行在真实的嵌入式系统中，更加接近实际多旋翼系统。需要注意的是，实际硬件通讯中可能会存在传输延迟，同时硬件在环系统的仿真模型和控制器所运行环境也难免与软件在环系统存在一定差异，因此控制器的参数可能需要进一步调节来达到设计需求，这也恰恰反映实际中的情况。

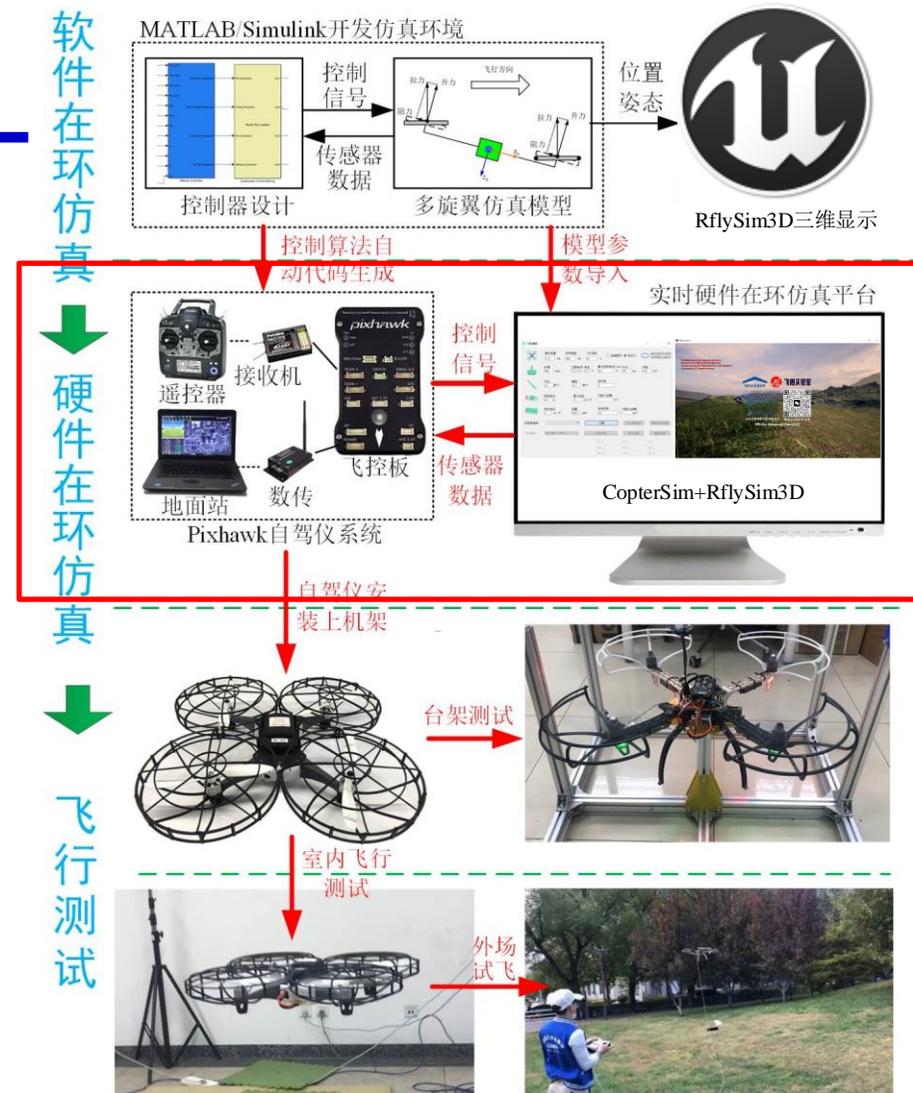


图. 实验流程图



2. 软、硬件简介和配置

2.5 基于模型开发流程简介

(3) 飞行测试阶段

在这个阶段，CopterSim的虚拟仿真模型进一步由真实多旋翼飞行器替代，传感器数据直接由传感器芯片感知飞行运动状态得到，控制器信号直接输出给电机，从而实现真实飞机的控制。需要注意的是，无论是硬件在环仿真还是软件在环仿真，其仿真模型都难以与真实飞机保持完全一致，因此进一步的参数调整也是必要的。





大纲

1. 平台的简介与安装
2. 软、硬件简介和使用
3. **基础实验例程**
4. 进阶实验例程
5. 后续章节学习方法
6. 总结



飞思实验室



RflySim教程



3.基础实验例程

3.0 学习思路

本章所有实验的目的旨在帮助用户实现对RflySim工具链中的各种软件、组件、插件以及相关API接口等资源的使用教学，让用户能够快速对RflySim工具链有初步的认识和理解，便于后期针对性实验的开发。因此，本章主要涉及到如下几个文件夹的例程实验：

- [0.ApiExps](#)文件夹：包含有各种主要软件的介绍、实验环境的搭建等教程；
- [1.BasicExps](#)文件夹：包含主要软件、API接口等使用性实验；
- [2.AdvExps](#)文件夹：相较于1.BasicExps文件夹，更加进阶的使用教程实验；
- [3.CustExps](#)文件夹：RflySim工具链中各组件的高级使用教程。



3.基础实验例程

3.0 学习思路—[0.ApiExps](#)文件夹

实验名称	简介	实验路径
RflySim工具链各组件简介	详细介绍RflySim工具链中的3DDisplay、CopterSim、QGroundControl、RflySim3D等组件。	0.ApiExps\e1_RflySimSoftwareReadme
固定翼无人机航迹飞行仿真实验	在QGC地面站中绘制固定翼无人机飞行航迹，来实现固定翼无人机起飞后按照航迹飞行。	0.ApiExps\e2_FWConfig
ReqCopterSim脚本接口验证	在平台进行仿真时可以通过一些接口实现更改CopterSim中的UDP模式，载具初始状态，载具模型的dll类型以及仿真地图等设置。	0.ApiExps\e3_ReqCopterSim
Ubuntu系统环境配置实验	包含有2个实验，主要讲解了VM_Ware虚拟机、普通计算机以及机载电脑中的Ubuntu环境配置。	0.ApiExps\e4_Vmware 、 0.ApiExps\e5_Ubuntu
无人系统开发软件配置实验	介绍Ubuntu系统下运用于无人系统开发过程中的各种常用软件系统，主要有：PX4、APM、ROS/ROS2、MAVROS等。	0.ApiExps\e6_PX4Build 、 0.ApiExps\e7_ApmBuild 、 0.ApiExps\e8_RosInstall 、 0.ApiExps\e9_Ros2Install 、 0.ApiExps\e10_Mavros



3.基础实验例程

3.0 学习思路—1.BasicExps文件夹

实验名称	简介	实验路径
CopterSim软件使用实验	熟悉CopterSim主界面“模型配置区”、“仿真功能区”、日志记录的基本使用方法。	1.BasicExps\e1_CopterSim-Usage 、 1.BasicExps\e2_DLL-Load 、 1.BasicExps\e14_Log-Get
RflySim3D软件使用实验	熟悉RflySim3D/RflySimUE5软件的快捷键和场景加载使用方法。	1.BasicExps\e3_RflySim3D-Shortcut-Instruct 、 1.BasicExps\e15_Scene-Load
Python3.8环境使用实验	熟悉RflySim工具链中内置的Python3.8环境的调用流程。	1.BasicExps\e4_Log-Reads-Python38Env 、
PX4软件使用实验	熟悉PX4软件系统的进行手动软件在环仿真和开发自定义PX4软件模块等方法。	1.BasicExps\e5_Manual-SIL 、 1.BasicExps\e12_PX4-App 、 1.BasicExps\e9_Build-Firmware
QGC地面站使用实验	熟悉QGC地面站的固件烧录、飞控指令识别以及航点绘制飞行等软件使用流程。	1.BasicExps\e10_Firmware-Upload 、 1.BasicExps\e16_Identify-Hardware-Command 、 1.BasicExps\e17_RoutePlanning
Bat脚本编写实验	熟悉RflySim工具链中Bat脚本的语言特点和编写规则。	1.BasicExps\e6_BAT-Startup
RflySim工具链关键接口使用实验	熟悉RflySim工具链中的代码自动生成、集群仿真等关键接口使用方法。	1.BasicExps\e7_Code-Generation 、 1.BasicExps\e8_SwarmAPI 、
教具配置实验	熟悉RflySim工具链中配套教具的配置使用流程和方法。	1.BasicExps\e11_RC-Config 、 1.BasicExps\e13_UAV-Config 、



3.基础实验例程

3.0 学习思路—[2.AdvExps](#)文件夹

实验名称	简介	实验路径
QGC显示共享内存图像实验	QGC显示RflySim3D模拟传感器的图像数据。	2.AdvExps\e1_QGCLoadimage



3.基础实验例程

3.1 “\0.ApiExps\e1_RflySimSoftware” RflySim内置应用的使用方法。



3DDisplay



API



CopterSim



FlightGear
-F450



HITLRun



HITLRunL
owGPU



HITLRunR
OS



HowToUse



MavrosRu
n



Python38
Env



QGroun
Control



rflysim.co
m



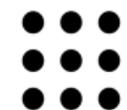
RflySim3D



RflySimAP
Is



RflySimUE
5



RosSwitch



SITLRun



SITLRunLo
wGPU



SITLRunR
OS



WinWSL



WsIGUI

e1_RflySimSoftwareReadme >

↑↓ 排序 ▾

≡ 查看 ▾

...

名称 ^

3DDisplay

CopterSim

drivers

examples

Firmware

FlightGear 2016.1.2

Python38

QGrounControl

RflySim3D

RflySimUE5

WinWSL





3.基础实验例程

3.2 “\0.ApiExps\e2_FWConfig” 固定翼仿真配置方法。

- 软件在环仿真配置
- 硬件在环仿真配置
- QGC航线绘制与上传
- 自动航路飞行功能





3.基础实验例程

3.3 “\0.ApiExps\e3_ReqCopterSim” 通过Python在线调整CopterSim初 始位置、仿真模式等。

- 使用ReqCopterSim库，来实现IP地址自动识别、初始位置调整、通信模式调整、地图和DLL模型的调整等
- 相比于Bat脚本提前写好的方法，用本接口在线调整的方式更为灵活，特别是在多机仿真时（多种载具，不同初始位置）



Public 成员函数

`__init__ (self)`

构造函数，初始化ReqCopterSim类的实例。开始监听20005端口以获取Copt

`getLocalIp (self)`

获取本地IP地址。

`isValidIp (self, address)`

检查给定的IP地址是否有效。

`updateSimMsg (self)`

更新模拟消息。

`getSimIpList (self)`

获取CopterSim IP列表。

`getSimIpID (self, CopterID=1)`

根据CopterID获取其IP地址。

`getHostIP (self)`

获取主机IP地址。

`sendReCopterSim (self, CopterID=1, isReqIP=1, UDP_mode=-1, isXyYaw=0, ...)`
请求初始化指定的CopterSim。

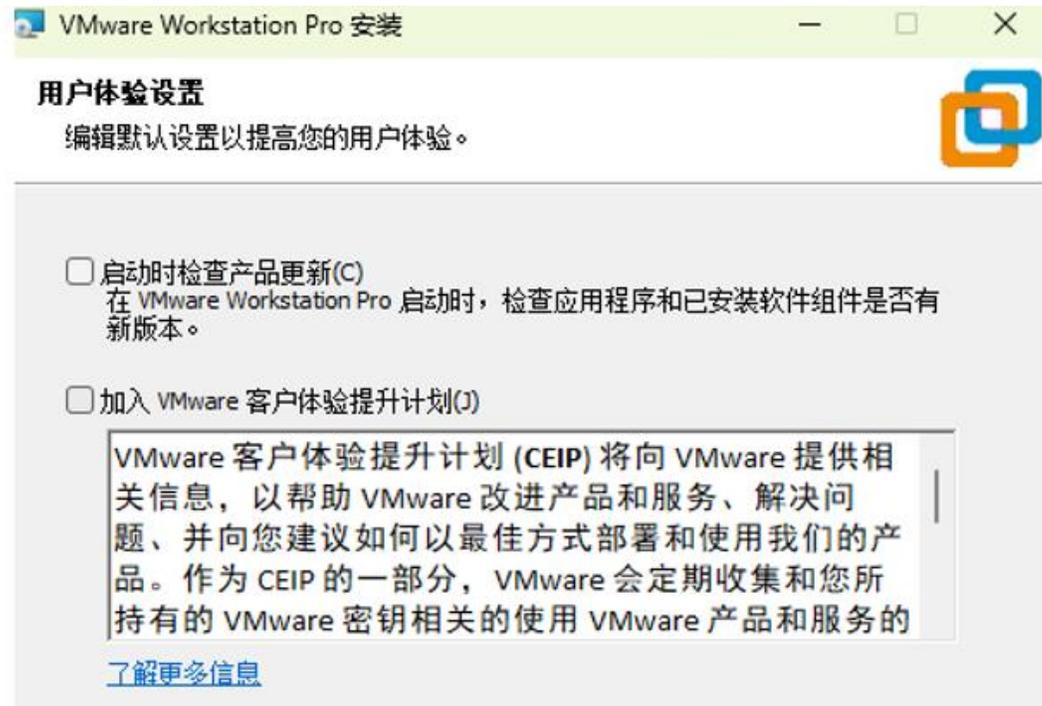
`sendReDllMap (self, CopterID=0, dllOrMap=-1, index=-1, name="")`



3.基础实验例程

3.4 “\0.ApiExps\e4_VMware” 虚拟机的使用与环境搭建。

- 虚拟机软件VMware的安装方法
- 在虚拟机中以Ubuntu 20.04为例，了解机载板卡中环境部署方法
- 在虚拟机中，安装并关联RflySim的依赖库，实现分布式仿真控制。
- 在虚拟机中，实现分布式视觉控制的例程。

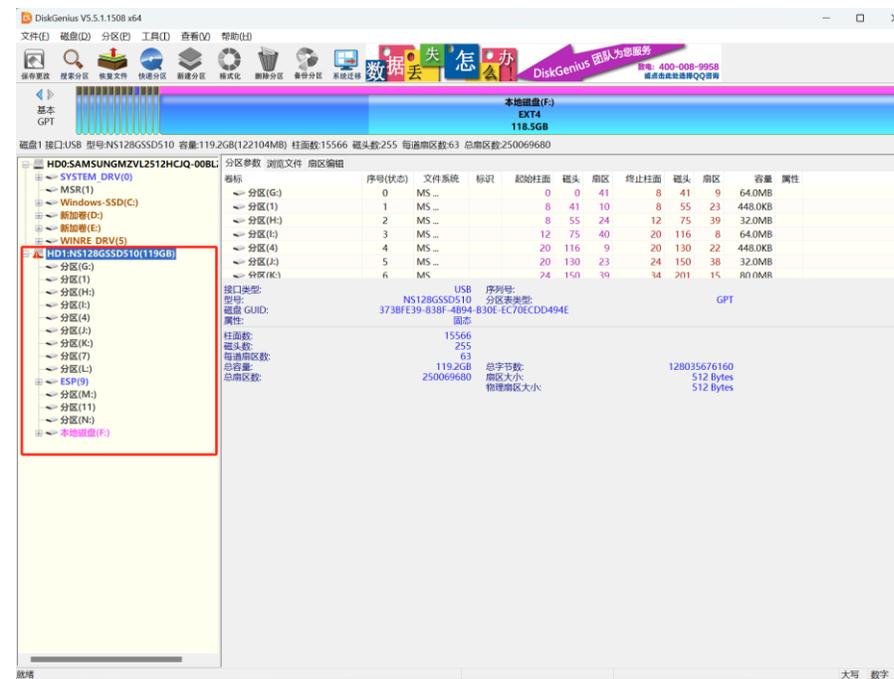
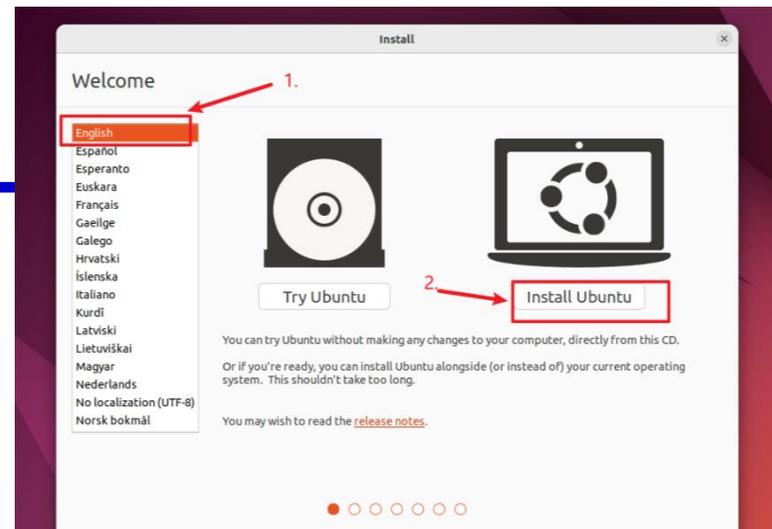




3.基础实验例程

3.5 “\0.ApiExps\e5_Ubuntu” Ubuntu虚拟机与机载板卡系统开发方法。

- 在虚拟机中以Ubuntu 20.04为例，了解机载板卡中环境部署方法
- 在虚拟机中，熟悉Linux的各种使用技巧，以及编程开发技巧。
- 在虚拟机中，安装并关联RflySim的依赖库，实现分布式仿真控制。
- 了解Ubuntu双系统的开发方法
- 了解机载板卡中，利用DiskGenius对整个硬盘拷贝镜像的方法。

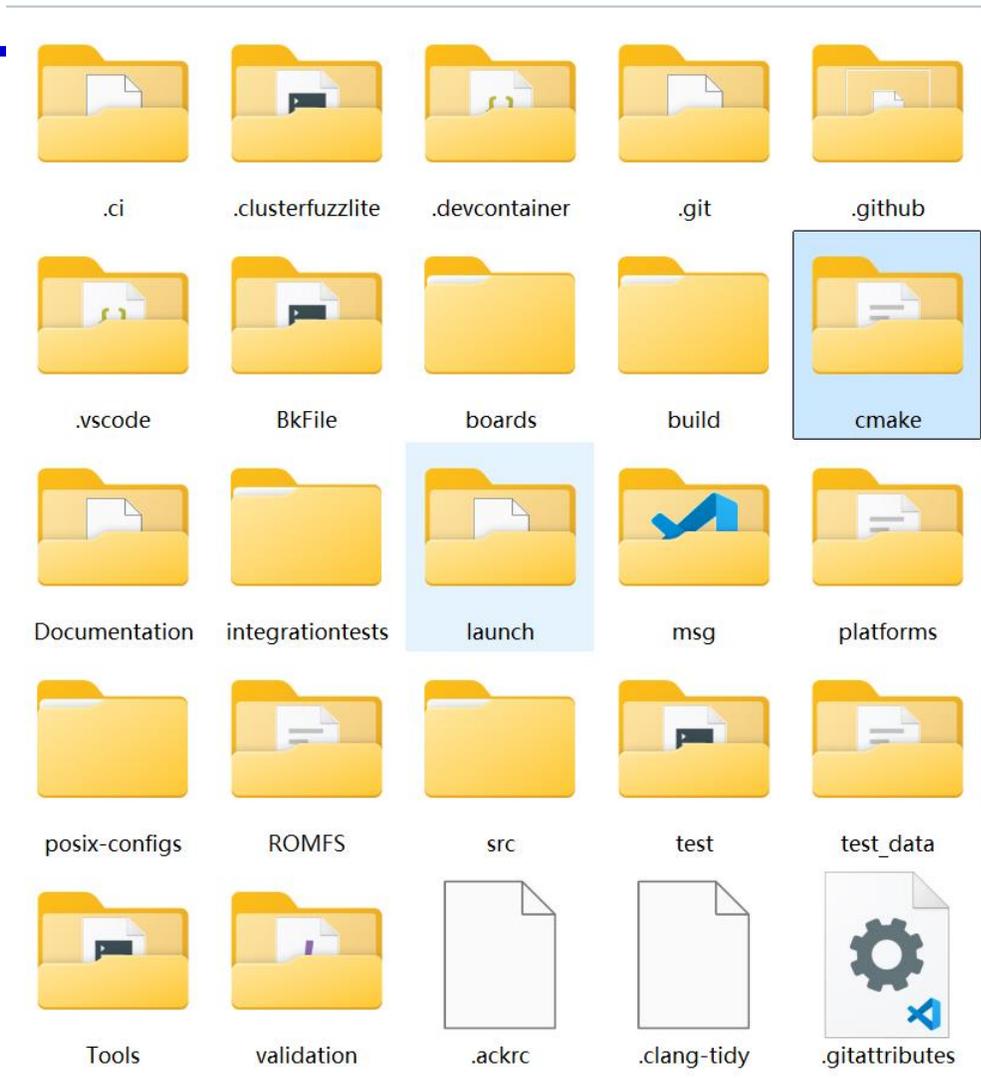




3.基础实验例程

3.6 “\0.ApiExps\e6_PX4Build” PX4编译环境部署与开发

- 了解PX4编译环境的部署方法
- 了解PX4代码的下载与编译方法
- 了解PX4源码的构架
- 了解PX4源码内开发独立App能
- 了解硬件在环模式下nsh和软件在环下pxh的调试方法。





3.基础实验例程

3.7 “\0.ApiExps\e7_ApmBuild” ArduPilot编译环境部署与开发

- `git clone --recursive https://github.com/ArduPilot/ardupilot.git`
- (可能有网络问题)
- `./waf configure --board CubeBlack`
- `./waf copter`

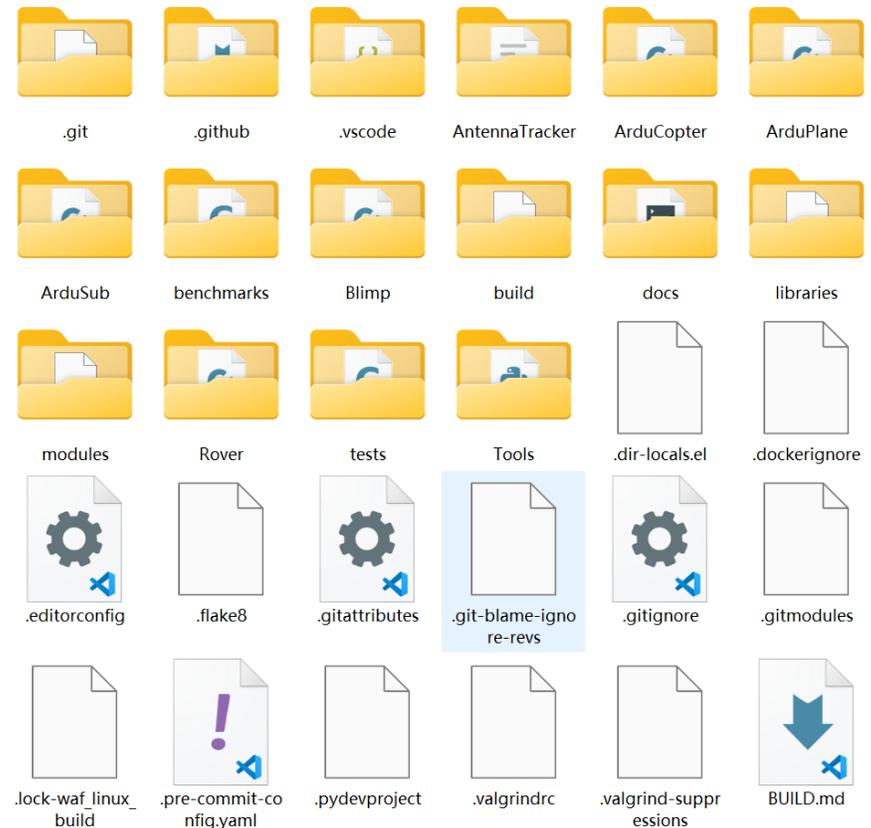
注：当项目越来越复杂，库与库之间是有十分复杂的依赖关系，人工去进行维护是十分困难的。这个时候就需要类似Make或者Waf工具来进行维护。
Waf编译工具相对于Make编译工具更加轻量化。

```

root@DaiPC: /mnt/e/Firmwar
Mx/compilers/GCC/mk/startup_stm32f4xx.mk CHIBIOS_PLATFORM_MK=os/hal/ports/STM32/STM32
F4xx/platform.mk MCU=cortex-m4 ENV_UDEFS=-DCHPRINTF_USE_FLOAT=1
Padded 12 bytes for bootloader.bin to 16384
Embedding file bootloader.bin:/mnt/e/FirmwareTest/ardupilot/Tools/bootloaders/CubeBlack_bl.bin
Embedding file defaults.parm:/mnt/e/FirmwareTest/ardupilot/build/CubeBlack/processed_defaults.parm
Embedding file hwdef.dat:/mnt/e/FirmwareTest/ardupilot/build/CubeBlack/hw.dat
Embedding file io_firmware.bin:Tools/IO_Firmware/iofirmware_highpolh.bin
Embedding file io_firmware_dshot.bin:Tools/IO_Firmware/iofirmware_dshot_highpolh.bin
[1195/1200] Linking build/CubeBlack/bin/arducopter
[1196/1200] Generating bin/arducopter.bin
[1197/1200] app_descriptor build/CubeBlack/bin/arducopter.bin
No APP_DESCRIPTOR found
[1198/1200] apj_gen build/CubeBlack/bin/arducopter.bin
[1199/1200] bin_cleanup build/CubeBlack/bin/arducopter.bin
[1200/1200] Generating bin/arducopter_with_bl.hex
Waf: Leaving directory '/mnt/e/FirmwareTest/ardupilot/build/CubeBlack'

BUILD SUMMARY
Build directory: /mnt/e/FirmwareTest/ardupilot/build/CubeBlack
Target      Text (B)  Data (B)  BSS (B)  Total Flash Used (B)  Free Flash (B)  External Flash Used (B)

```





3.基础实验例程

3.8 “\0.ApiExps\e8_RosInstall” ROS1环境安装与测试方法。

e9_Ros2Install: ROS2环境安装与测试方法，以及ROS1和ROS2的切换方案。

e10_Mavros: mavros的安装与ROS1/ROS2的配置方法，以及使用Offboard进行飞行测试的例程。

```
$ mavrosInstall.sh ×
E: > RflySimGit > rflysimapis_new > RflySimAPIs > 2.RflySimUsage > 0.ApiExps > e10_Mavros > 1.MavrosInstall > $ mavrosInst
1  #!/usr/bin/env bash
2
3  # 参考 https://docs.px4.io/main/en/ros/mavros\_installation.html
4
5  # 安装ROS1/mavros
6  sudo apt-get install ros-noetic-mavros ros-noetic-mavros-extras ros-noetic-mavros-msgs
7
8  # 替换node.launch和px4.launch使得支持多机连接
9  sudo cp -fp ./Ros1Launch/* /opt/ros/noetic/share/mavros/launch
10 # 如下代码，会启动/mavros1和/mavros2打头的消息，分别连上CopterSim1和CopterSim2
11 # roslaunch mavros px4.launch namespace:="mavros1" tgt_system:="1" fcu_url:="udp://:20101
12 # roslaunch mavros px4.launch namespace:="mavros2" tgt_system:="2" fcu_url:="udp://:20103
13
14 # 安装ROS2/mavros
15 sudo apt-get install ros-foxy-mavros ros-foxy-mavros-extras ros-foxy-mavros-msgs
16
17 # 解决ROS2/mavros px4.launch启动文件的bug，进行文件替换
18 sudo cp -fp ./Ros2Launch/* /opt/ros/foxy/share/mavros/launch
19 # 如下代码，会启动/mavros1和/mavros2打头的消息，分别连上CopterSim1和CopterSim2
20 # ros2 launch mavros px4.launch namespace:="mavros1" tgt_system:="1" fcu_url:="udp://:201
21 # ros2 launch mavros px4.launch namespace:="mavros2" tgt_system:="2" fcu_url:="udp://:201
```



大纲

1. 平台的简介与安装
2. 软、硬件简介和使用
3. 基础实验例程
- 4. 进阶实验例程**
5. 后续章节学习方法
6. 总结



飞思实验室



RflySim教程



4.进阶实验例程



扫码或点击二维码观看本
实验视频教程

4.1 CopterSim 使用实验

本实验可对CopterSim进行相关设置后求解出无人机的动力系统模型。主文件夹见“*\\PX4PSP\\RflySimAPIs\\2.RflySimUsage\\1.BasicExps\\e1_CopterSim-Usage”，具体实验操作见文件[1.BasicExps\\e1_CopterSim-Usage\\Readme.pdf](#)，实验效果如下(部分)：





4.进阶实验例程



扫码或点击二维码观看本
实验视频教程

4.2 CopterSim 导入DLL实验

本实验根据提供一个Simulink的固定翼模型，一键导出为DLL文件，再加载CopterSim中，最后进行仿真。主文件夹见“*\PX4PSP\RflySimAPIs\2.RflySimUsage\1.BasicExps\e2_DLL-Load”，具体实验操作见文件[1.BasicExps\e2_DLL-Load\Readme.pdf](#)，实验效果如下(部分)：





4.进阶实验例程



扫码或点击二维码观看本
实验视频教程

4.3 RflySim3D快捷键与指令实验

本实验主要讲解了RflySim3D得基本操作、快捷键及快捷指令的使用。主文件夹见“*\PX4PSP\RflySimAPIs\2.RflySimUsage\1.BasicExps\e3_RflySim3D-Shortcut-Instruct”，具体实验操作见文件[1.BasicExps\e3_RflySim3D-Shortcut-Instruct\Readme.pdf](#)，实验效果如下(部分)

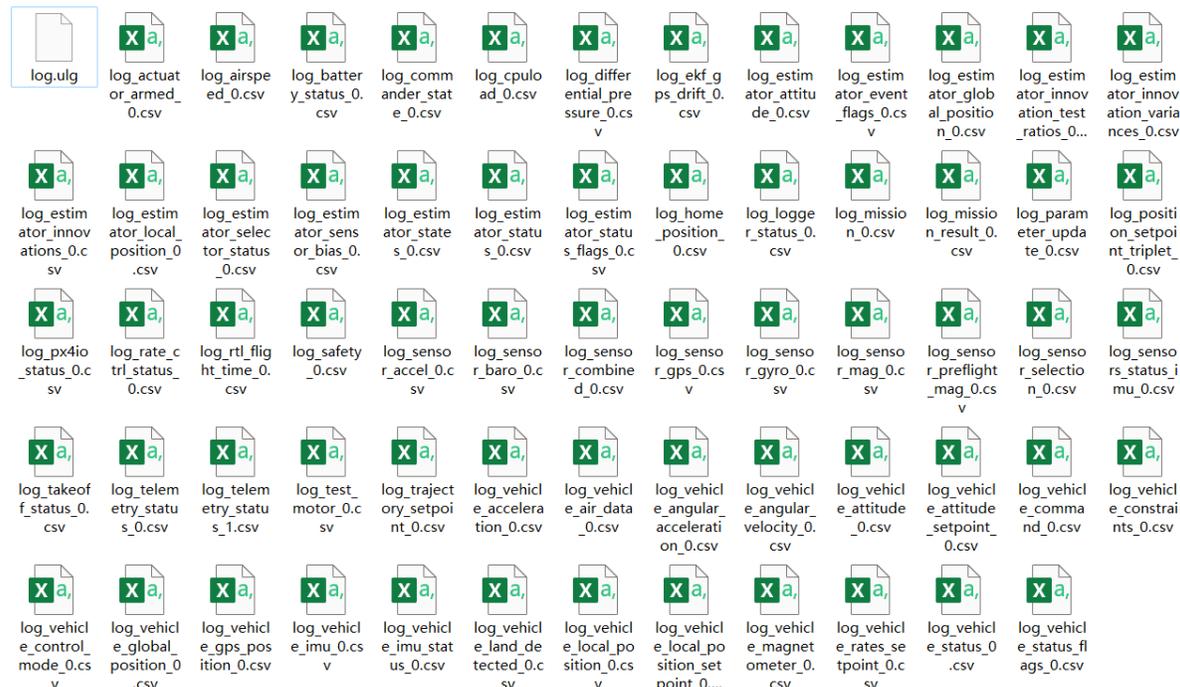




4.进阶实验例程

4.4 Python38Env读取飞行日志实验

本实验基于Python3.8环境，对飞行日志1.BasicExps\\e4_Log-Reads-Python38Env\\Readme.pdf，实验效果如下(部分):





4.进阶实验例程

4.5 手动软件在环仿真配置实验

本实验基于Win10WSL编译仿真指令，实现手动配置整个软件在环的仿真环境。

主文件夹见“*\PX4PSP\RflySimAPIs\2.RflySimUsage\1.BasicExps\e5_Manual-SIL”，

具体实验操作见文件[1.BasicExps\e5_Manual-SIL\Readme.pdf](#)，实验效果如下(部分):

```
program: none
model: none
world: none
src_path: /mnt/c/PX4PSP/Firmware
build_path: /mnt/c/PX4PSP/Firmware/build/px4_sitl_default
empty model, setting iris as default
SITL COMMAND: "/mnt/c/PX4PSP/Firmware/build/px4_sitl_default/bin/px4" "/mnt/c/PX4PSP/Firmware/build/px4_sitl_default"/etc
c -s etc/init.d-posix/rcS -t "/mnt/c/PX4PSP/Firmware/test_data
INFO [px4] Creating symlink /mnt/c/PX4PSP/Firmware/build/px4_sitl_default/etc -> /mnt/c/PX4PSP/Firmware/build/px4_sitl
default/tmp/rootfs/etc

PX4

px4 starting.

INFO [px4] Calling startup script: /bin/sh etc/init.d-posix/rcS 0
Info: found model autostart file as SYS_AUTOSTART=10016
INFO [param] selected parameter default file eeprom/parameters_10016
[param] parameter file not found, creating eeprom/parameters_10016
SYS_AUTOCONFIG: curr: 0 -> new: 1
SYS_AUTOSTART: curr: 0 -> new: 10016
CAL_ACC0_ID: curr: 0 -> new: 1310988
CAL_GYRO0_ID: curr: 0 -> new: 1310988
CAL_ACC1_ID: curr: 0 -> new: 1310996
CAL_GYRO1_ID: curr: 0 -> new: 1310996
CAL_ACC2_ID: curr: 0 -> new: 1311004
CAL_GYRO2_ID: curr: 0 -> new: 1311004
CAL_MAG0_ID: curr: 0 -> new: 197388
CAL_MAG1_ID: curr: 0 -> new: 197644
SENS_BOARD_X_OFF: curr: 0.0000 -> new: 0.0000
SENS_DPRES_OFF: curr: 0.0000 -> new: 0.0010
* SYS_AUTOCONFIG: curr: 1 -> new: 0
IMU_INTEG_RATE: curr: 200 -> new: 250
```



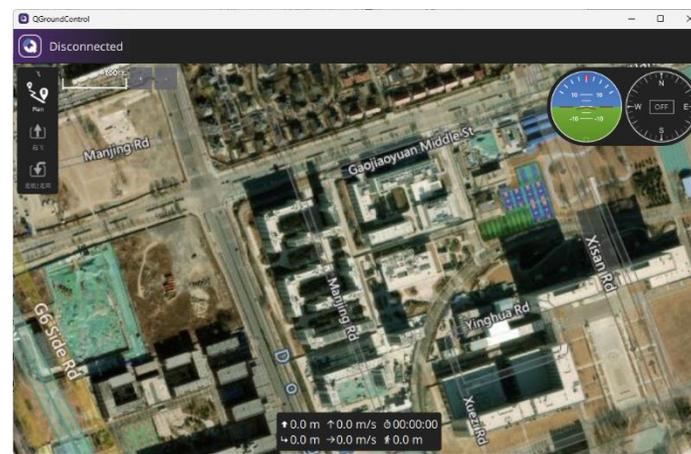
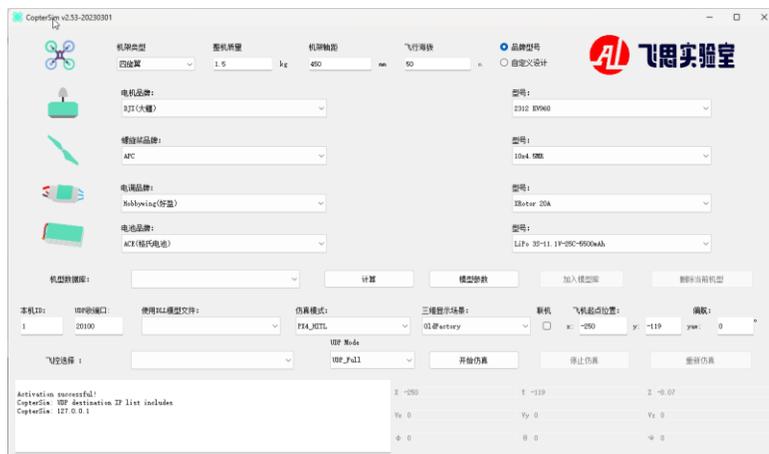
4.进阶实验例程



扫码或点击二维码观看本实验视频教程

4.6 BAT脚本启动组件实验

本实验基于Windows批处理语言，编写BAT脚本，实现RflySim中多个组件的一键。文件夹见“*\PX4PSP\RflySimAPIs\2.RflySimUsage\1.BasicExps\e6_BAT-Startup”，具体实验操作见文件[1.BasicExps\e6_BAT-Startup\Readme.pdf](#)，实验效果如下(部分):





4.进阶实验例程



扫码或点击二维码观看本
实验视频教程

4.7 MATLAB代码自动生成飞控固件实验

本实验基于RflySim工具链的MATLAB自动代码生成模块，在Simulink搭建完成的控制模型，可直接一键生成飞控固件。具体实验操作见文件[1.BasicExps\e7_Code-Generation\Readme.pdf](#)，实验效果如下(部分):

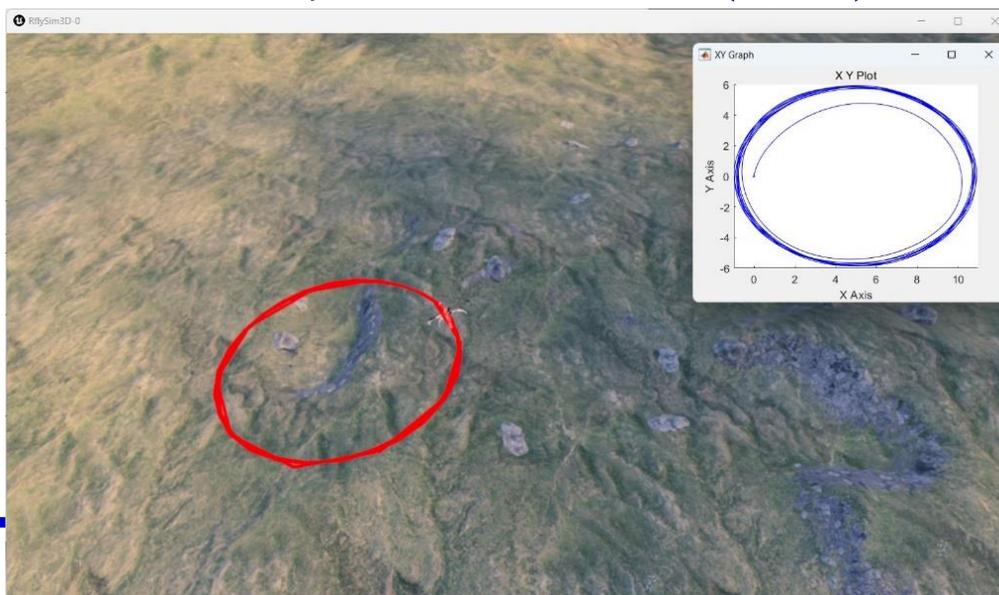
```
Stabilize_HIL
[196/225] Linking CXX static library src/modules/mavlink/libmodules__mavlink__original.a
[197/225] Linking CXX static library src/modules/sensors/vehicle_imu/libvehicle_imu.a
[198/225] Linking CXX static library src/modules/sensors/vehicle_magnetometer/libvehicle_magnetometer.a
[199/225] Linking CXX static library src/modules/sih/libmodules__sih.a
[200/225] Linking CXX static library src/modules/temperature_compensation/libmodules__temperature_compensation.a
[201/225] Linking CXX static library src/examples/fake_gps/libmodules__fake_gps.a
[202/225] Linking CXX static library src/modules/navigator/libmodules__navigator.a
[203/225] Linking CXX static library src/modules/flight_mode_manager/tasks/Auto/libFlightTaskAuto.a
[204/225] Linking CXX static library src/modules/flight_mode_manager/tasks/ManualAltitude/libFlightTaskManualAltitude.a
[205/225] Linking CXX static library src/modules/sensors/libmodules__sensors.a
[206/225] modules_mavlink_merging source
[207/225] Linking CXX static library src/modules/flight_mode_manager/tasks/AutoMapper/libFlightTaskAutoMapper.a
[208/225] Linking CXX static library src/modules/flight_mode_manager/tasks/AutoFollowMe/libFlightTaskAutoFollowMe.a
[209/225] Linking CXX static library src/modules/flight_mode_manager/tasks/ManualPosition/libFlightTaskManualPosition.a
[210/225] Linking CXX static library src/modules/flight_mode_manager/tasks/ManualAltitudeSmoothVel/libFlightTaskManualAltitudeSmoothVel.a
[211/225] Building CXX object src/modules/mavlink/CMakeFiles/modules__mavlink__dir/modules__mavlink__unity.cpp.obj
[212/225] Linking CXX static library src/modules/flight_mode_manager/tasks/AutoLineSmoothVel/libFlightTaskAutoLineSmoothVel.a
[213/225] Linking CXX static library src/modules/flight_mode_manager/tasks/ManualPositionSmoothVel/libFlightTaskManualPositionSmoothVel.a
[214/225] Linking CXX static library src/modules/flight_mode_manager/tasks/ManualAcceleration/libFlightTaskManualAcceleration.a
[215/225] Linking CXX static library src/modules/flight_mode_manager/tasks/Orbit/libFlightTaskOrbit.a
[216/225] Building CXX object src/modules/flight_mode_manager/CMakeFiles/modules__flight_mode_manager.dir/FlightTasks_generated.cpp.obj
[217/225] Building CXX object src/modules/flight_mode_manager/CMakeFiles/modules__flight_mode_manager.dir/FlightModeManager.cpp.obj
[218/225] ROMFS: generating image
[219/225] Linking CXX static library src/modules/flight_mode_manager/libmodules__flight_mode_manager.a
[220/225] Building C object ROMFS/CMakeFiles/romfs.dir/nsh_romfsimg.c.obj
[221/225] Linking C static library ROMFS/libromfs.a
[222/225] Linking CXX static library src/modules/mavlink/libmodules__mavlink.a
[223/225] Linking CXX executable droneeye_zyfc-h7_default.elf
Memory region      Used Size  Region Size  %age Used
ITCM_RAM:          0 GB         64 KB      0.00%
FLASH:           1856273 B      1920 KB     94.41%
DTCM1_RAM:        0 GB         64 KB      0.00%
DTCM2_RAM:        0 GB         64 KB      0.00%
AXI_SRAM:         44196 B        512 KB     8.43%
SRAM1:            0 GB         128 KB     0.00%
SRAM2:            0 GB         128 KB     0.00%
SRAM3:            0 GB         32 KB      0.00%
SRAM4:            0 GB         64 KB      0.00%
BKPRAM:          0 GB          4 KB       0.00%
[224/225] Generating ../droneeye_zyfc-h7.bin
[225/225] Creating /mnt/c/PX4PSP/Firmware/build/droneeye_zyfc-h7_default/droneeye_zyfc-h7_default.px4
#### Finished calling CMAKE build process ####
#### Done invoking postbuild tool. ####
#### Successfully generated all binary outputs. ####
```



4.进阶实验例程

4.8 Simulink集群控制接口

通过运行本例程文件夹中的bat脚本即可一键启动QGC、CopterSim及RflySim3D软件，运行Simulink之后，可看到无人机起飞并进入盘旋状态。具体实验操作见文件[1.BasicExps\e8_SwarmAPI\Readme.pdf](#)，文件夹见“*\PX4PSP\RflySimAPIs\2.RflySim Usage\1.BasicExps\e8_SwarmAPI”，实验效果如下(部分):





4.进阶实验例程

4.9 PX4固件编译

本实验通过Win10WSL完成对PX4固件的编译。具体实验操作见文件[1.BasicExps\e9_Build-Firmware\Readme.pdf](#)，文件夹见“*\PX4PSP\RflySimAPIs\2.RflySimUsage\1.BasicExps\e9_Build-Firmware”，实验效果如下(部分):

```
root@RFLYSIM: /mnt/c/PX4PSP
-- Detecting CXX compile features - done
-- Check for working C compiler: /root/gcc-arm-none-eabi-7-2017-q4-major/bin/arm-none-eabi-gcc
-- Check for working C compiler: /root/gcc-arm-none-eabi-7-2017-q4-major/bin/arm-none-eabi-gcc -- works
-- Detecting C compiler ABI info
-- Detecting C compiler ABI info - done
-- Detecting C compile features
-- Detecting C compile features - done
-- Found PythonInterp: /usr/bin/python3 (found suitable version "3.6.9", minimum required is "3")
-- build type is MinSizeRel
-- PX4 ECL: Very lightweight Estimation & Control Library v1.9.0-rc1-591-gb3fed06
-- Configuring done
-- Generating done
-- Build files have been written to: /mnt/c/PX4PSP/Firmware/build/px4_fmu-v5_default/external/Build/px4io_firmware
[330/1374] Performing build step for 'px4io_firmware'
[1/247] git submodule platforms/nuttx/Nuttx/nuttx
[9/247] git submodule platforms/nuttx/Nuttx/apps
[245/247] Linking CXX executable px4_io-v2_default.elf
Memory region      Used Size  Region Size  %age Used
flash:             60436 B    60 KB        98.37%
sram:               3856 B     8 KB         47.07%
[247/247] Creating /mnt/c/PX4PSP/Firmware/build/px4_fmu-v5_default/external/Build/px4io_firmware/px4_io-v2_default.px4
[1372/1374] Linking CXX executable px4_fmu-v5_default.elf
Memory region      Used Size  Region Size  %age Used
FLASH_ITCM:        0 GB      2016 KB      0.00%
FLASH_AXIM:        1913417 B  2016 KB      92.69%
ITCM_RAM:          0 GB      16 KB        0.00%
DTCM_RAM:          0 GB      128 KB       0.00%
SRAM1:             45748 B   368 KB       12.14%
SRAM2:             0 GB      16 KB        0.00%
[1374/1374] Creating /mnt/c/PX4PSP/Firmware/build/px4_fmu-v5_default/px4_fmu-v5_default.px4
```



4.进阶实验例程



扫码或点击二维码观看本
实验视频教程

4.10 飞控固件烧录实验

本实验介绍了RflySim工具链的两种不同固件烧录方式。文件夹见“*\PX4PSP\RflySimAPIs\2.RflySimUsage\1.BasicExps\e10_Firmware-Upload”，具体实验操作见文件[1.BasicExps\e10_Firmware-Upload\Readme.pdf](#)，实验效果如下(部分):

```
C:\WINDOWS\SYSTEM32\cmd  x  +  v  -  □  x
Loaded firmware for board id: 1010,0 size: 1855977 bytes (94.40%), waiting for the bootloader...

Attempting reboot on COM6 with baudrate=57600...
If the board does not respond, unplug and re-plug the USB connector.

Found board id: 1010,0 bootloader version: 5 on COM6
sn: 003e00413430510534303536
chip: 20036450
family: b'STM32H7[4|5]x'
revision: b'V'
flash: 1966080 bytes
Windowed mode: False

Erase : [===== ] 81.7%
```



4.进阶实验例程

4.11 遥控器配置实验

本工具链使用的遥控器推荐使用“美国手”的操纵方式，即左侧摇杆对应的油门与偏航控制量，而右侧摇杆对应滚转与俯仰。文件夹见“*|P X4PSP\RflySimAPIs\2.RflySimUsage\1.BasicExps\ e11_RC-Config”，具体实验操作见文件[1.BasicExps\ e11_RC-Config\Readme.pdf](#)，请结合本例程文件夹中的遥控器配置手册及其他资料对自己手中的遥控器进行配置。



油门：控制上下运动，对应固定翼油门杆
偏航：控制机头转向，对应固定翼方向舵
俯仰：控制前后运动，对应固定翼升降舵
滚转：控制左右运动，对应固定翼副翼



4.进阶实验例程

4.12 飞控板载应用开发实验

本实验主要介绍如何创建并运行你的第一个板载应用程序，掌握 PX4 应用程序开发所需的所有基本概念和 API。文件夹见“*\PX4PSP\RflySimAPIs\2.RflySimUsage\1.BasicExps\e12_PX4-App”，具体实验操作见文件[1.BasicExps\e12_PX4-App\Readme.pdf](#)，实验效果如下(部

```
QGroundControl
Back < Analyze Tools
日志下载 Provides a connection to the vehicle's system shell.
地理标记图像
Mavlink 控制台
MAVLink 检测
振动

NuttShell (NSH) NuttX-10.1.0
nsh> my_app
INFO [my_app] Welcome to RflySim!
nsh>
```



4.进阶实验例程

4.13 四旋翼无人机配置实验

本实验介绍了飞思X450无人机平台的组成结构和简单配置等。文件夹见“*\PX4 PSP\RflySimAPIs\2.RflySimUsage\1.BasicExps\e13_UAV-Config”，具体实验操作见文件[1.BasicExps\e13_UAV-Config\Readme.pdf](#)，实验效果如下(部分):



飞思X450飞行器	
尺寸	折叠尺寸: 760*760*220mm
对称电机轴距	455mm
机体重量(不含电池)	1134g
最大载重	1800g
机体材料	碳纤维、尼龙
通讯接口	WiFi
电池	6s-9000mAh, 0.934kg
定位系统下悬停精度	垂直: $\pm 0.05m$; 水平: $\pm 0.05m$
最大上升速度	2m/s
最大下降速度	2m/s
最大水平飞行速度	10m/s
最大起飞海拔高度	5000m
续航时间(空载)	38分钟
工作环境温度	-20° C至 50° C
基本配置	光流传感器、激光定高传感器、独立外磁
机载板卡	NX、nano
可选载荷配置	CSI相机、T265、D435IS1激光雷达
通讯板卡	rockpi、nx板卡、数传、nanopi、4G模块、RTK



4.进阶实验例程



扫码或点击二维码观看本
实验视频教程

4.15 RflySim3D三维场景加载实验

本实验主要讲解了RflySim3D三维场景创建何加载流程。主文件夹见“*\\PX4PS
P\\RflySimAPIs\\2.RflySimUsage\\1.BasicExps\\e15_Scene-Load”，具体实验操作见文件[1.
BasicExps\\e15_Scene-Load\\Readme.pdf](#)，实验效果如下(部分):

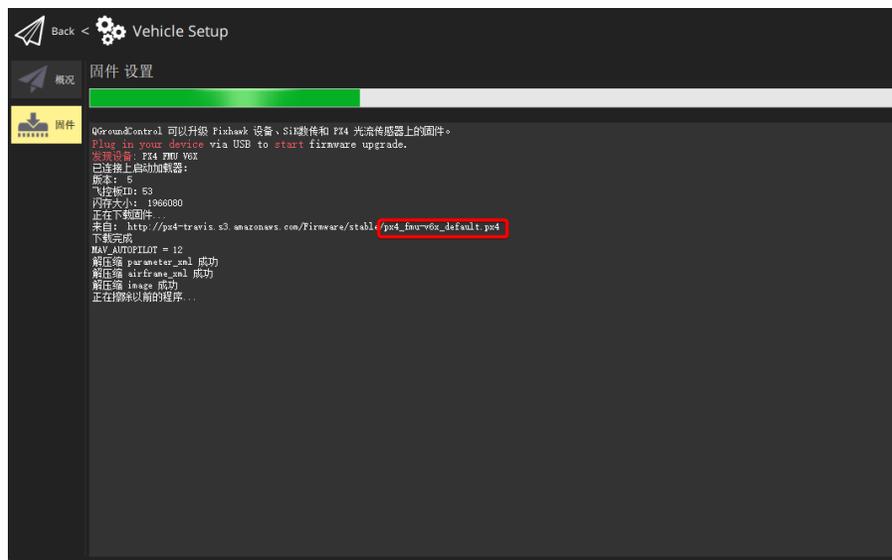




4.进阶实验例程

4.16 Pixhawk硬件编译命令识别实验

本实验针对不同的飞控硬件，介绍一种通过QGroundControl来识别不同飞控硬件的编译命令。具体实验操作见文件[1.BasicExps\e16_Identify-Hardware-Command\Readme.pdf](#)，实验效果如下(部分):

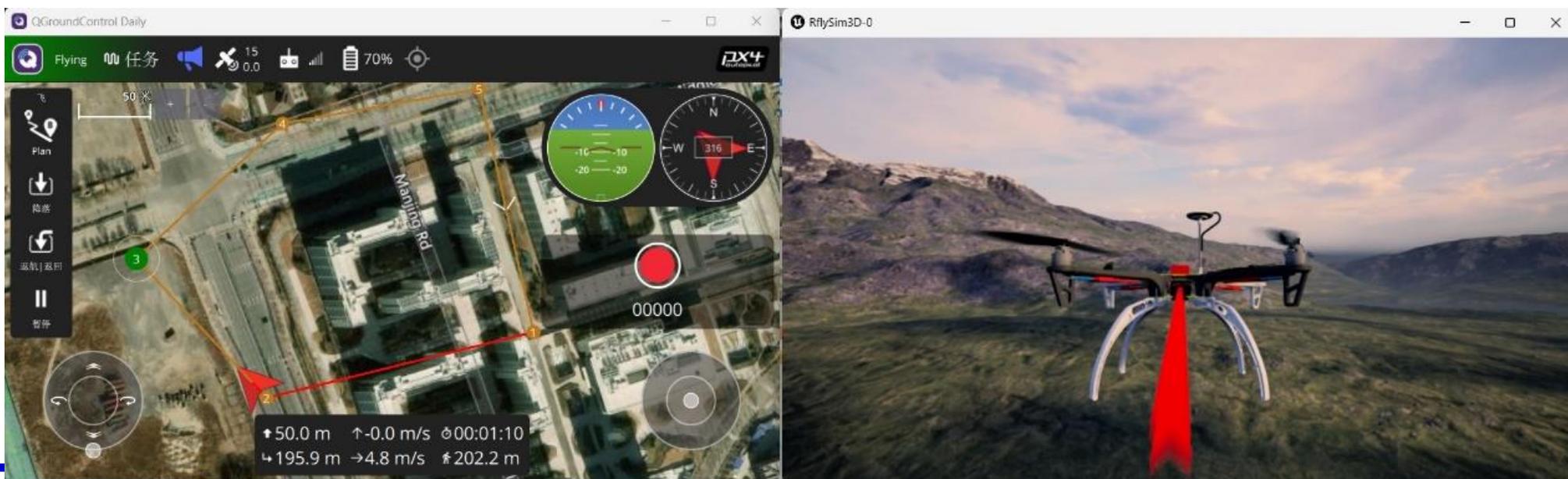




4.进阶实验例程

4.17 HIL航线绘制实验

本实验介绍了在QGroundControl中利用航线规划功能，绘制航线实现规定航线下的飞行的方法。具体实验操作见文件[1.BasicExps\e17_RoutePlanning\Readme.pdf](#)，实验效果如下(部分):

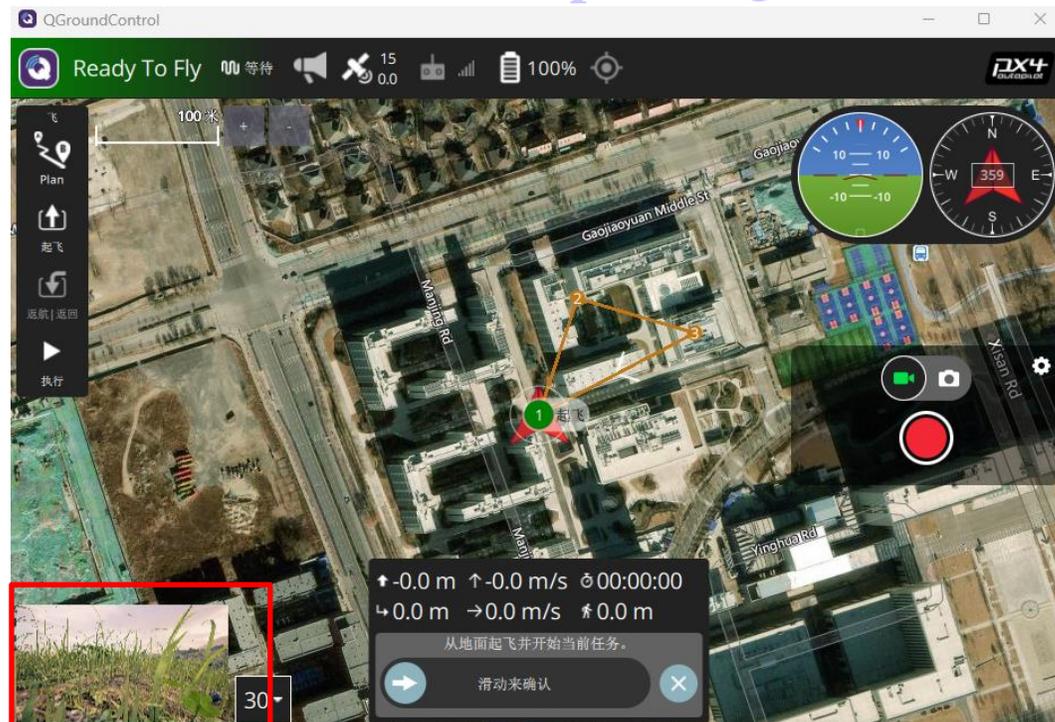




4.进阶实验例程

4.17 QGC显示共享内存图像实验

本实验介绍了在QGC从指定的共享内存中获取数据，进行解析后将图像显示在QGC界面中。具体实验操作见文件 [2.AdvExps\e1_QGCLoadimage\readme.pdf](#)，实验效果如下(部分):





大纲

1. 平台的简介与安装
2. 软、硬件简介和使用
4. 进阶实验例程
- 5. 后续章节学习方法**
6. 总结



飞思实验室



RflySim教程



5. 后续章节学习方法

5.1 配套书籍



《多旋翼飞行器设计与控制》：本书讲授多旋翼设计、动态模型建立、状态估计、控制和决策等方面的基础知识。涉及到空气流体力学、电机、电路、材料结构、理论力学、以及导航、制导与控制各个学科的基础知识，具有基础性和系统性两个特色。因此，有利于读者将已学知识融会贯通，着重培养学生解决问题的综合能力。本书覆盖了多旋翼飞行器设计的大部分内容，共十五章，包括多旋翼飞行器基础知识、布局、动力系统、建模、感知、控制和决策等部分。旨在将多旋翼飞行器工程实践中应用的设计原则组织起来，并强调基础概念的重要性，具有基础性、实用性、综合性和系统性等特点。本书可以用作高年级本科生以及研究生教材，或者作为该领域研究的入门指南，还可以作为多旋翼飞行器工程师的自学教材。

《多旋翼飞行器设计与控制实践》：本书分为实验平台和实验任务两大部分，其中实验平台依托为本书特别设计的RflySim工具链。RflySim工具链利用目前的先进开发理念“基于模型开发（Model-Based Design）”流程，将多旋翼飞行器、Pixhawk自驾仪，以及MATLAB Simulink编程语言紧密联系在一起。实验任务共包括循序渐进的8个实验：动力系统设计、动态建模、传感器标定、滤波器设计、姿态控制器设计、定点位置控制器设计、半自主控制模式设计及失效保护逻辑设计，完成多旋翼飞行器设计与控制实践。本书适合对无人机特别是多旋翼飞行器设计与控制感兴趣的读者，也可以把本书部分实验作为专业课程的实践环节。



5. 后续章节学习方法

5.1 配套书籍



《多旋翼飞行器从原理到实践》：本书是多旋翼飞行器的一本入门级教材，也是一本从原理到操作实践的手册。它首先介绍了多旋翼飞行器的基本概念、飞行原理、发展历史等内容；然后，对其系统组成以及机架、动力系统、通信系统、飞行控制系统等重要组成部分进行了详细介绍；最后，介绍了组装调试、操控和维护、行业应用和发展等知识。总得来说，本书做到了理论讲解和实践操作的有机融合，做到了内容的全面与细致，也在处处讲解安全操作，培养读者的安全意识。我们也为本书配备了大量教学和实践操作视频，帮助读者更好地学习和掌握本书内容。

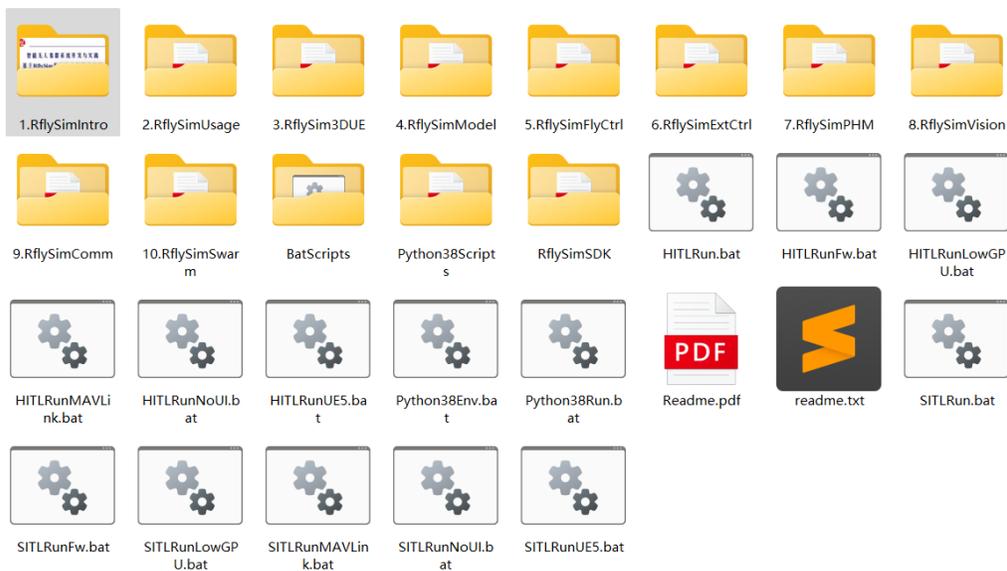
《多旋翼飞行器远端控制实践》：本书瞄准多旋翼无人机的应用实践，旨在降低入门门槛，为更多不具备多旋翼专业知识和开发背景的高职、低年级本科生或相关从业人员提供理论与实践相结合的综合参考教科书，让读者在学习“自动控制原理”等专业课之后，运用所学专业技术基础课及专业课知识，进行控制系统的详细设计，使读者在综合运用专业理论解决工程问题方面得到实际锻炼。



5. 后续章节学习方法

5.2 资源框架

RflySim 工具链涉及无人系统开发的三维场景搭建、无人系统的动力学模型建立、底层控制、智能感知、健康评估、网络模拟、集群控制等等。安装完成RflySim工具链之后，进入“【安装目录】\PX4PSP\RflySimAPIs”文件夹中，如右图所示，具体每章解释如右表所示。



序号	章节名称	简介	例程目录名	本章详解	配套课件	API	例程检索
1	第1讲-绪论	RflySim工具链简介、版本区别、安装及各功能特点。	1.RflySimIntro	Intro.pdf	PPT.pdf	API.pdf	Index.pdf
2	第2讲-实验平台配置	RflySim工具链配置流程、核心组件的使用方法 & 实验流程等。	2.RflySimUsage	Intro.pdf	PPT.pdf	API.pdf	Index.pdf
3	第3讲-三维场景建模与仿真	RflySim3D软件的构架和功能、无人系统三维建模与场景开发软件的使用等。	3.RflySim3DUE	Intro.pdf	PPT.pdf	API.pdf	Index.pdf
4	第4讲-载具运动建模与仿真	无人系统载具的控制模型搭建、RflySim工具链模型开发流程等。	4.RflySimModel	Intro.pdf	PPT.pdf	API.pdf	Index.pdf
5	第5讲-位姿控制与滤波估计	本章包含大量无人系统底层开发例程，提供代码生成与下载功能，可以将设计的 Simulink 控制算法一键生成PX4固件，并烧录自驾仪中。实现Sim2Real的基础性实验流程。	5.RflySimFlyCtrl	Intro.pdf	PPT.pdf	API.pdf	Index.pdf
6	第6讲-外部控制与轨迹规划	本章通过外部控制接口对智能体发送命令，去实现更上层的轨迹规划等控制功能。	6.RflySimExtCtrl	Intro.pdf	PPT.pdf	API.pdf	Index.pdf
7	第7讲-安全测试与健康评估	本章针对无人系统开发中的软件单元和集成验证、嵌入式软件和硬件验证、软硬件集成验证到整机集成与测试验证的过程。实现对上述所有的开发阶段进行故障注入与安全测试。	7.RflySimPHM	Intro.pdf	PPT.pdf	API.pdf	Index.pdf
8	第8讲-视觉感知与避障决策	本章讲述视觉传感器与相关理论，如：载体与各传感器坐标系，视觉控制的常见传感器等；介绍Linux、ROS、MAVROS等相关视觉开发的环境配置方式和RflySim工具链的视觉接口。	8.RflySimVision	Intro.pdf	PPT.pdf	API.pdf	Index.pdf
9	第9讲-通信协议与集群组网	无人系统组网的方式与现状、RflySim工具链中的集群通信的系统架构以及无人系统组网的仿真例程。	9.RflySimComm	Intro.pdf	PPT.pdf	API.pdf	Index.pdf
10	第10讲-集群控制与博弈对抗	围绕多智能体的无人系统集群控制开发，介绍了集群编队、任务规划、博弈等技术，重点讲述RflySim工具链无人机集群系统的分布式控制框架和基于MATLAB/Python的集群控制接口，提供基于蚂蚁算法的多无人机任务规划、多无人机的编队、曲线管道控制、大规模无人机集群控制等案例，帮助读者理解集群控制的原理和实现方式。	10.RflySimSwarm	Intro.pdf	PPT.pdf	API.pdf	Index.pdf



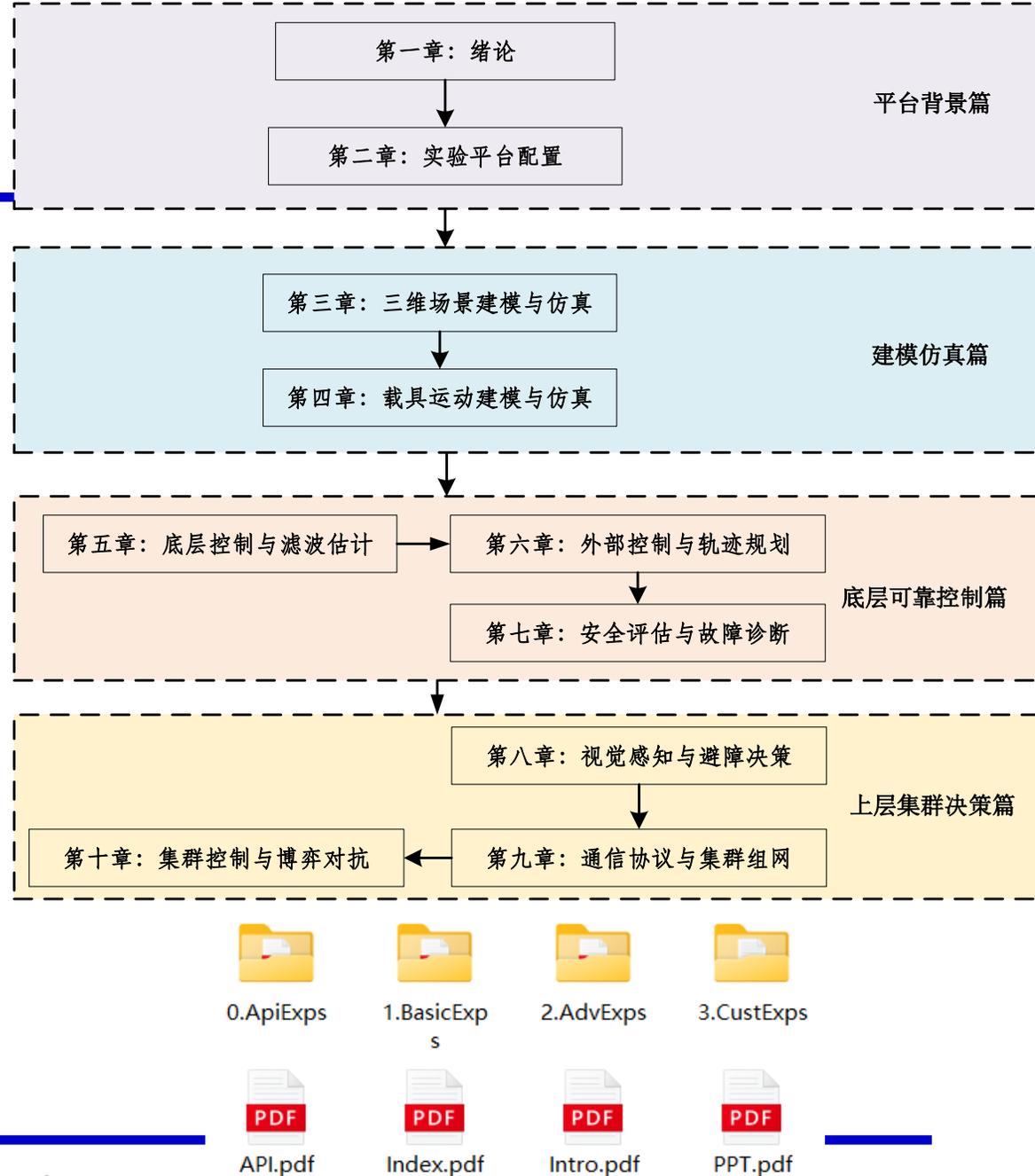
5. 后续章节学习方法

5.2 资源框架

所有章节之间的联系如右图所示。

同时，每讲的例程文件夹内部结构如下表所示，不同的文件夹中存有不同难度的实验，旨在帮助用户循序渐进的学习本讲相关内容。

序号	名称	文件夹/文件
1	基础接口例程文件夹	0.ApiExps
2	基础例程文件夹	1.BasicExps
3	进阶例程文件夹	2.AdvExps
4	定制例程文件夹	3.CustExps(仅完整版)
5	本讲接口说明文件	API.pdf
6	本讲介绍文件	Intro.pdf
7	本讲配套课件文件	PPT.pdf
8	本讲所有例程检索文件	Index.pdf





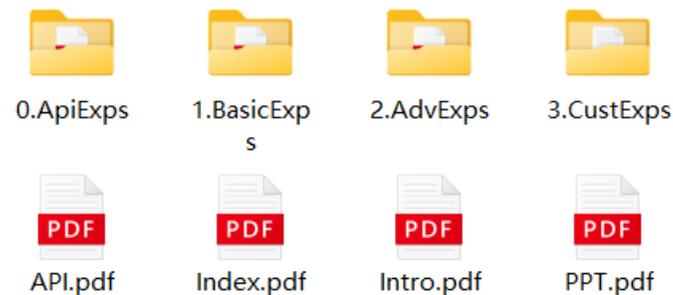
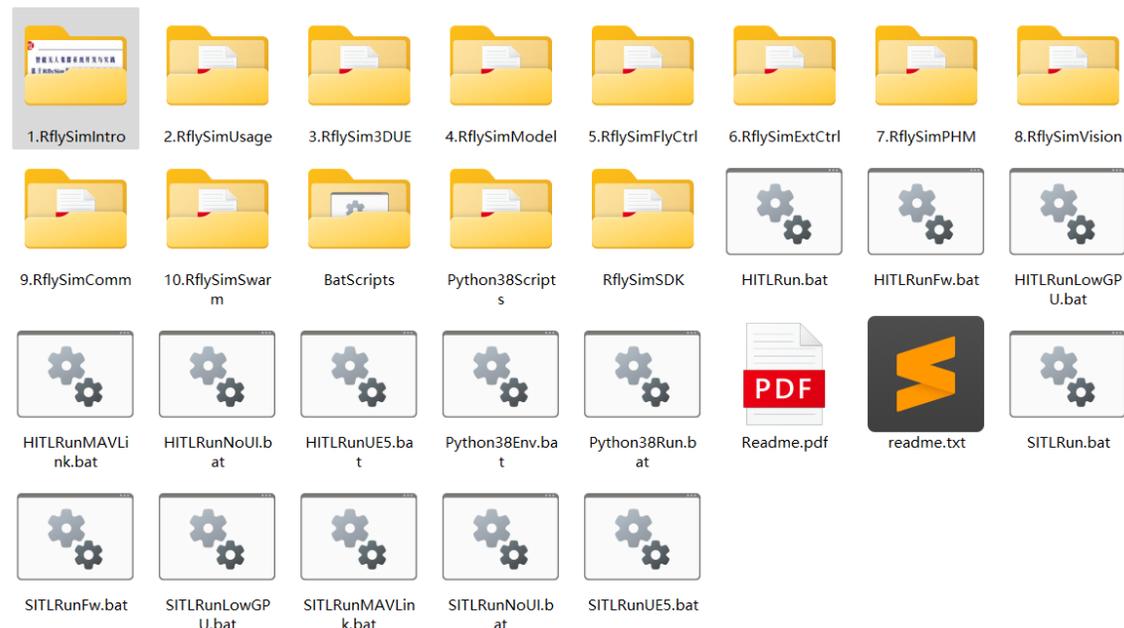
5. 后续章节学习方法

5.2 资源框架

RflySim 工具链涉及无人系统开发的三维场景搭建、无人系统的动力学模型建立、底层控制、智能感知、健康评估、网络模拟、集群控制等等。安装完成RflySim工具链之后，进入“【安装目录】\PX4PSP\RflySimAPIs”文件夹中，如右图所示，

同时，每讲的例程文件夹内部结构如下表所示，不同的文件夹中存有不同难度的实验，旨在帮助用户循序渐进的学习本讲相关内容。

序号	名称	文件夹/文件
1	基础接口例程文件夹	0.ApiExps
2	基础例程文件夹	1.BasicExps
3	进阶例程文件夹	2.AdvExps
4	定制例程文件夹	3.CustExps(Full version only)
4	本讲接口说明文件	API.pdf
6	本讲介绍文件	Intro.pdf
7	本讲配套课件文件	PPT.pdf
8	本讲所有例程检索文件	Index.pdf





5. 后续章节学习方法

5.3 实验工具链配置与使用

RflySim工具链包含了众多在进行无人系统建模、仿真、算法验证等开发过程中所涉及到的软件，其中，核心组件有CopterSim、QGroundControl、Rfly Sim3D/RflySimUE5、Python38Env、WinWSL子系统、SITL/HITLRun一键运行脚本、MATLAB自动代码生成工具箱、Simulink集群控制接口、PX4 Firmware源码、RflySim配套资料文件以及配套硬件系统。用户通过对这些核心组件的学习即可快速上手无人系统的开发和测试工作。本章开发相关API接口地址为：[API.pdf](#)。本章所有例程检索为：[Index.pdf](#)。

底层飞行控制算法开发



自动代码生成与固件编译



硬件在环仿真



室外或室内实验

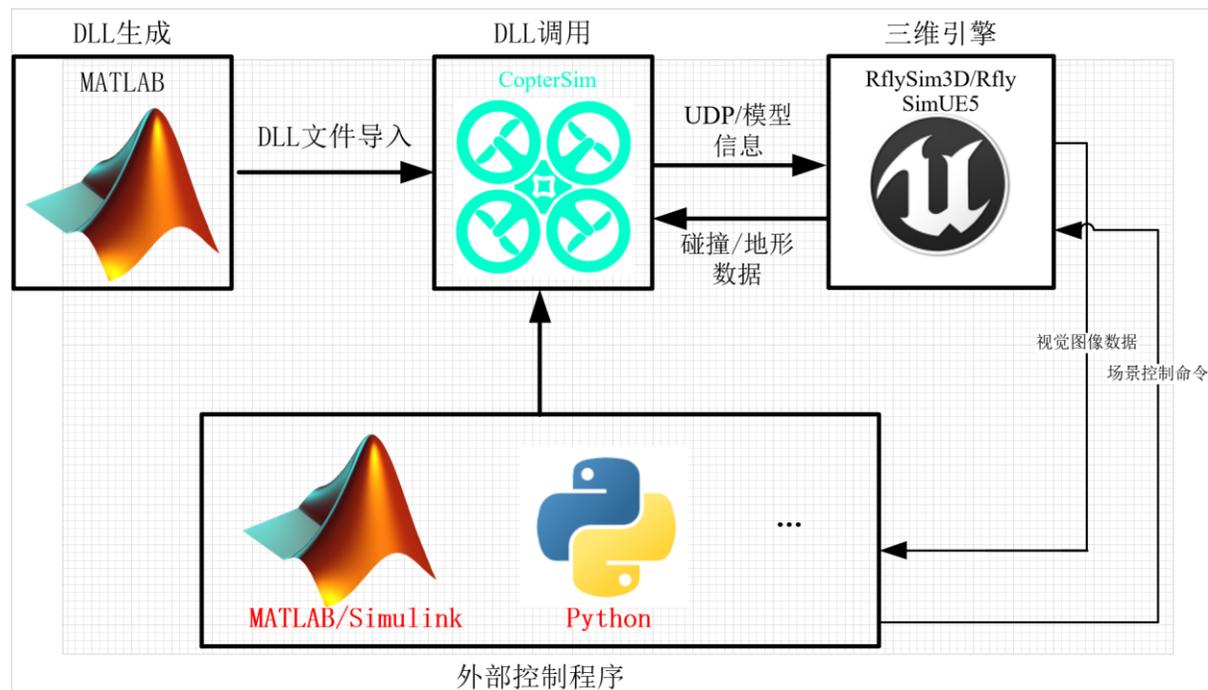




5. 后续章节学习方法

5.4 三维场景建模与仿真

RflySim3D是RflySim三维仿真平台，CopterSim会根据从Pixhawk（或者PX4 SITL）传入的电机控制数据解算出无人机当前的状态（主要是位置、姿态数据），随后会将这些数据发送给RflySim3D，而RflySim3D会将这些数据应用至场景里相应的无人机上，从而使我们能更直观的看到无人机的状态。RflySim3D还支持通过XML文件进行一些配置，主要是用XML配置无人机的构型（四旋翼、六旋翼、固定翼等）、模型在列表中的优先级、飞机的名字、飞机的初始位置与姿态、各致动器（一般是旋翼）的初始位置、姿态、材质、旋转轴、运动模式，还可以定义摄像机的位置，还可以定义一些障碍组件（例如柱子、圆环）等等。本章开发相关API接口平台地址为：[..\3.RflySim3DUE\API.pdf](#)。本章所有例程检索为：[..\3.RflySim3DUE/Index.pdf](#)。

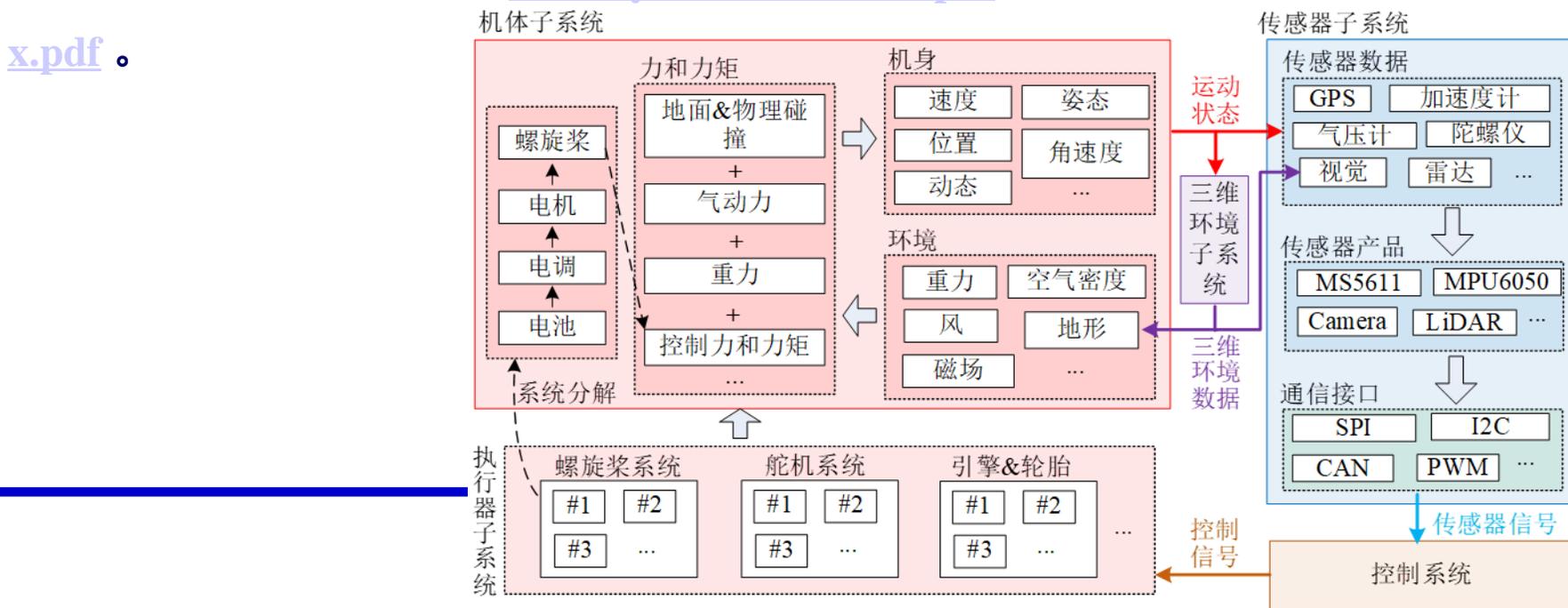




5. 后续章节学习方法

5.5 载具运动建模与仿真

无人载具系统统一建模框架将整个无人载具系统分解为两个部分：机身系统与控制系统。机身系统与控制系统之间进行着传感器数据与控制信号。而机身系统又可以细分为四个子系统：机体子系统、执行器子系统、三维环境子系统和传感器子系统。在整个建模框架中，机身系统需要进行高精度建模，并且在实时仿真计算机中实现，最后连接控制系统软件或硬件，构成软件在环仿真或硬件在环仿真闭环。本章开发相关API接口平台地址为：[..\4.RflySimModel\API.pdf](#)。本章所有例程平台地址为：[..\4.RflySimModel\Index.pdf](#)。





5. 后续章节学习方法

5.6 位姿控制与滤波估计

RflySim采用基于模型设计（Model-Based Design, MBD）的思想，可用于无人系统的控制和安全测试。通过以下五个阶段：建模阶段、控制器设计阶段、软件在环仿真阶段（Software-In-the-Loop, SIL）、硬件在环仿真阶段（Hardware-In-the-Loop, HIL）和实飞测试阶段。通过MATLAB/Simulink的自动代码生成技术，控制器能够被方便地自动下载到硬件中，用于HIL仿真和实际飞行测试。本章开发相关API接口平台地址为：<..\5.RflySimFlyCtrl\API.pdf>。本章所有例程平台地址为：<..\5.RflySimFlyCtrl\Intro.pdf>。

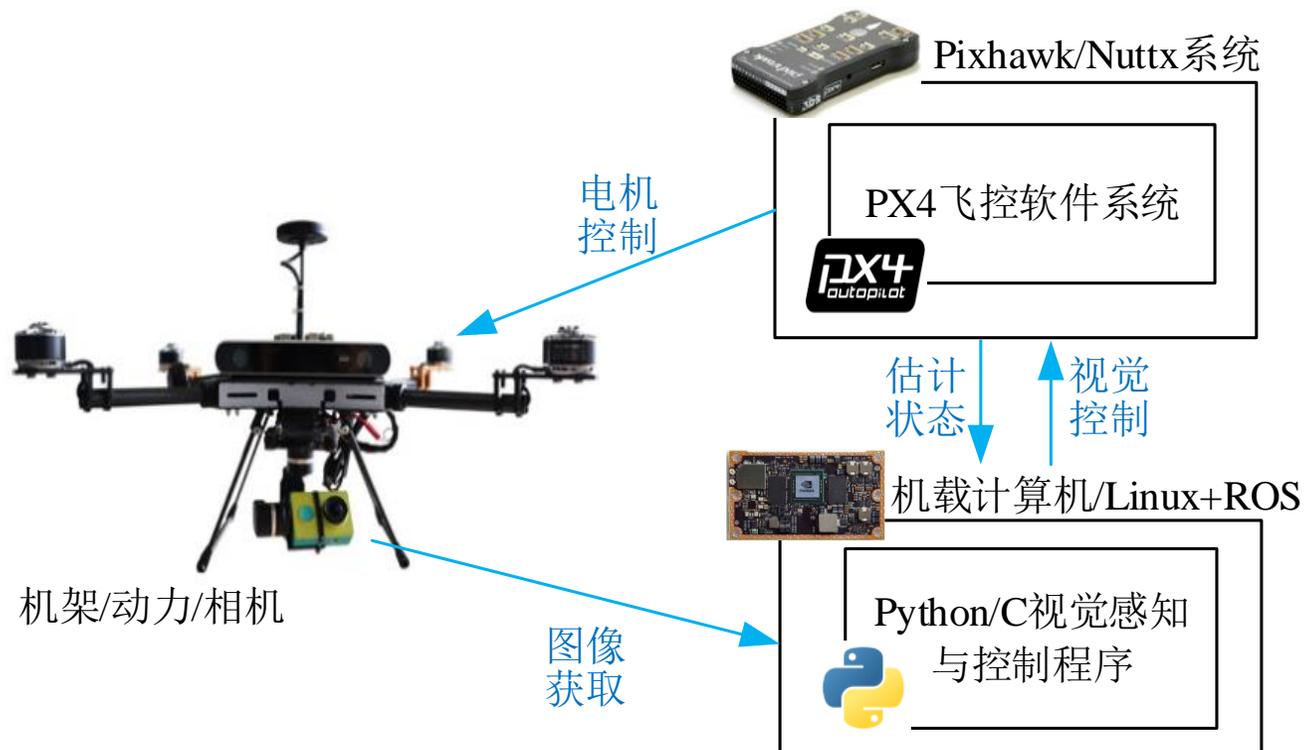




5. 后续章节学习方法

5.7 外部控制与轨迹规划

无人机作为一种具有自主飞行能力的飞行器，其外部控制能力对于实现精确的飞行任务和安全的飞行操作至关重要。通常常见的控制方式有：遥控器控制、地面站控制、半自主控制、和通过计算机相应接口控制。其中外部控制理论主要涉及飞行器在空中的稳定性，一般为理解为对无人机的位置和姿态控制，对此我们将深入探讨无人机的飞行控制刚体模型，也就是制导模型。本章开发相关API接口平台地址为：[..\6.RflySimExtCtrl\API.pdf](#)。本章所有例程平台地址为：[..\6.RflySimExtCtrl\Index.pdf](#)。

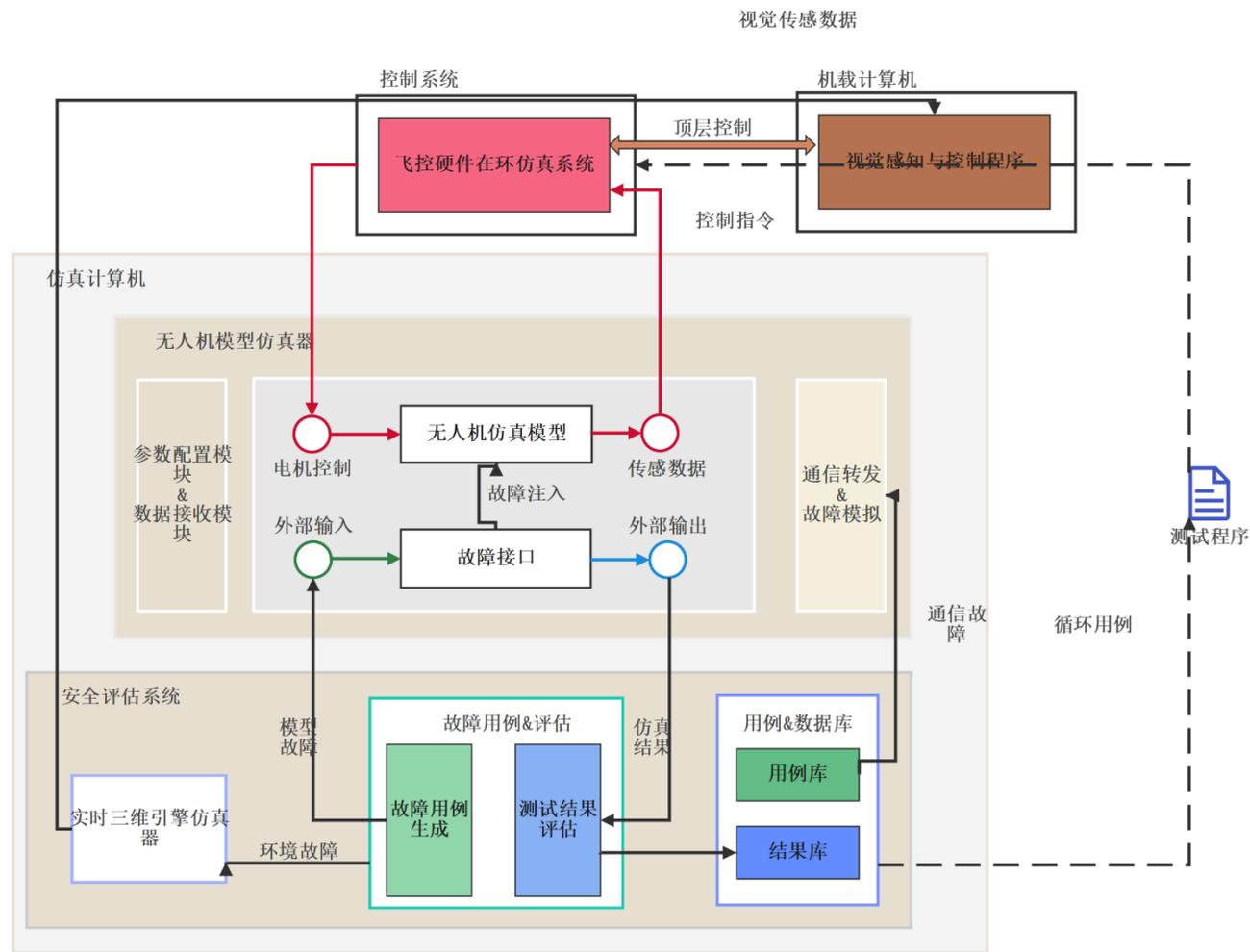




5. 后续章节学习方法

5.8 安全测试与健康平台

RflySim故障注入架构由实物模块、仿真模块与评估模块组成。实物模块由飞控硬件组成，负责与仿真计算机连接，接收来自外部的控制指令并作出姿态响应，组成半物理仿真闭环，可通过飞控进行硬件在环的实时故障注入。仿真模块由CopterSim、RflySim3D、QGC组成，负责整机的故障消息发送和三维故障注入，进行实时的故障模拟。评估模块负责输出故障注入之后的安全状况。本章开发相关API接口平台地址为：[..\7.RflySimPHM\API.pdf](#)。本章所有例程平台地址为：[..\7.RflySimPHM\Index.pdf](#)。

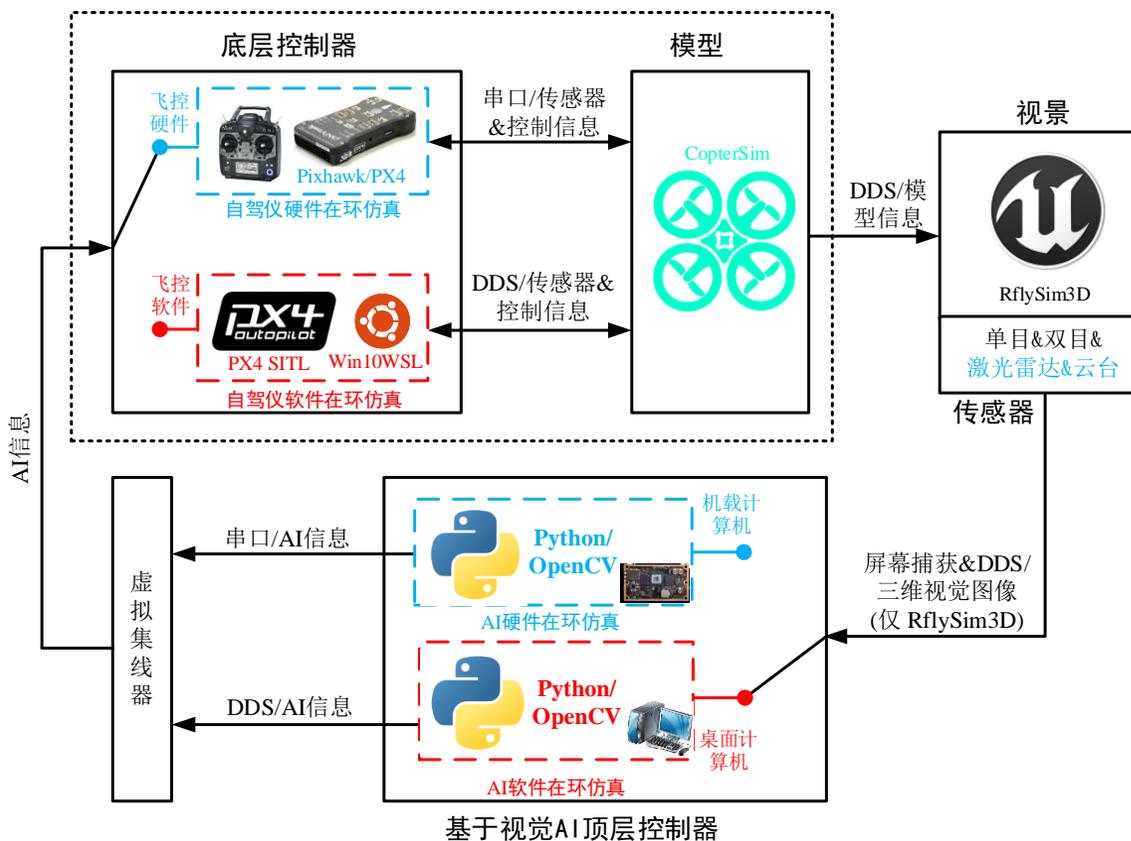




5. 后续章节学习方法

5.9 视觉感知与避障决策

RflySim工具链支持外部传感器接入，我们将这些传感器数据分为两类：一类直连飞控的外部传感器（磁罗盘、差分GPS、光流测速等），另一类是直连机载计算机的视觉传感器（双目、Lidar、深度相机等）飞控传感器通过Simulink等程序直接生成传入Pixhawk飞控，视觉传感器通过三维环境引擎生成，随图像传入机载计算机。RflySim提供传感器基本参数与安装位置的深度相机传感器模块SDK，用户可以通过设置相关参数对无人机载视觉传感器进行设计朝向、焦距、视场角等，自定义机载视觉模块；根据用决策的输入输出接口协议，提供决策系统输入/输出接口。本章开发相关API接口平台地址为：<..\8.RflySimVision\API.pdf>。本章所有例程平台地址为：<..\8.RflySimVision\Index.pdf>。

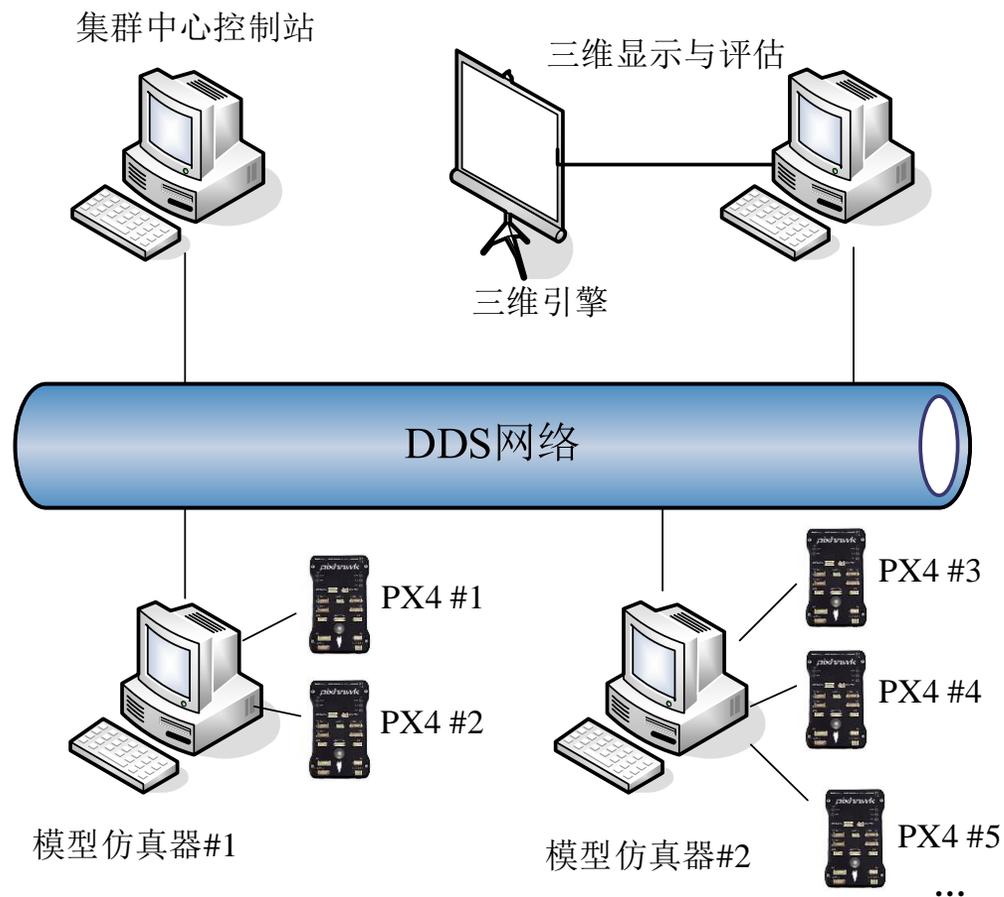




5. 后续章节学习方法

5.10 通信协议与集群组网

RflySim采用分布式组网架构，不同的仿真模型可以运行在同一台电脑或者不同的电脑上。打开多个模型仿真器并连接多个Pixhawk/PX4自驾仪硬件就可以形成多无人集群的仿真环境。由于单台电脑的性能是有限的，可以通过局域网内多台电脑间进行相互通信的形式进一步扩展整体飞机数量。本章开发相关API接口平台地址为：<..\9.RflySimComm\API.pdf>。本章所有例程平台地址为：<..\9.RflySimComm\Index.pdf>。

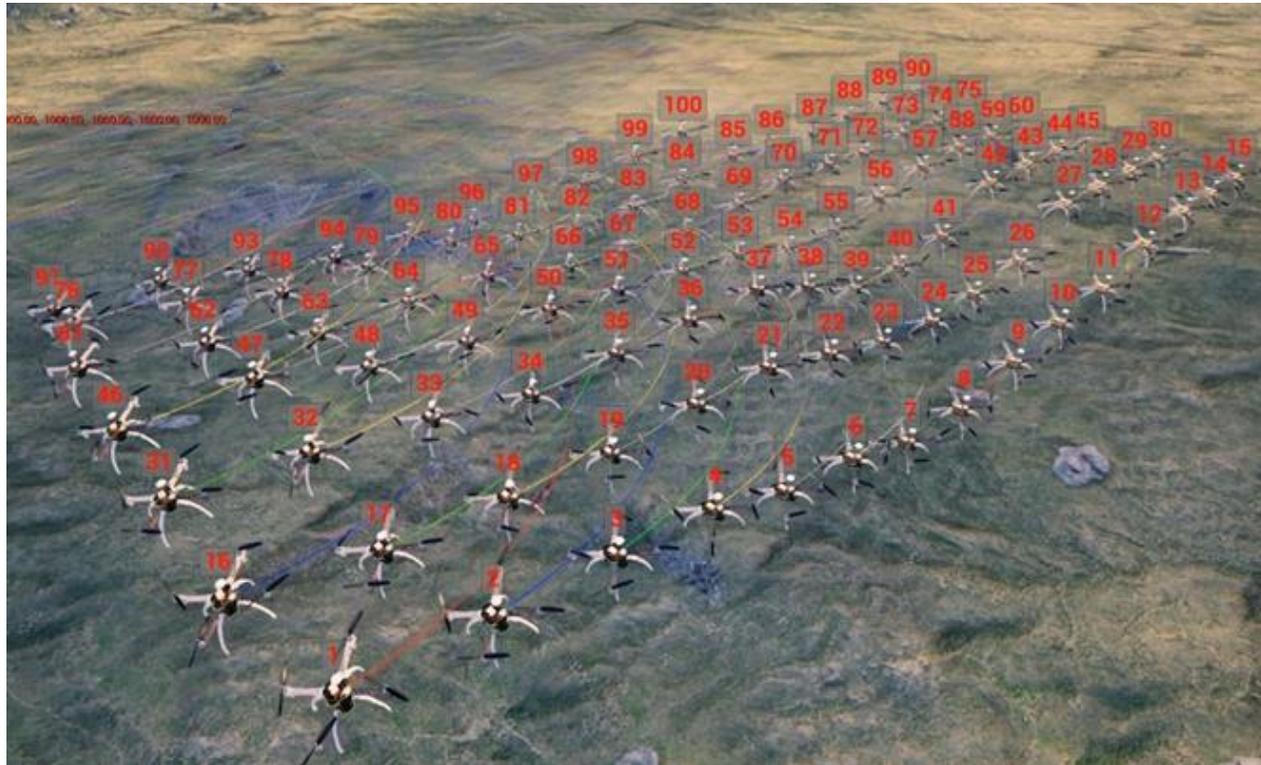




5. 后续章节学习方法

5.11 集群控制与博弈对抗

RflySim支持一键启动多机集群仿真功能，支持MATLAB/Simulink、Python端集群仿真开发，支持多架软件在环、硬件在环及软、硬件相结合的虚实集群仿真，支持局域网内多台电脑的分布式集群仿真。同时，随着飞机数量的增加，网络通信负载越来越大，为了在有限带宽下实现更多数量的无人机集群仿真，需要对通信进行优化，目前工具链的数据协议主要有两种：MAVLink数据和UDP压缩结构体，基于这两种数据协议，RflySim提出了5种压缩的数据协议，实现上百架的无人机集群仿真。本章开发相关API接口平台地址为：[..\10.RflySimSwarm\API.pdf](#)。本章所有例程平台地址为：[..\10.RflySimSwarm/Index.pdf](#)。

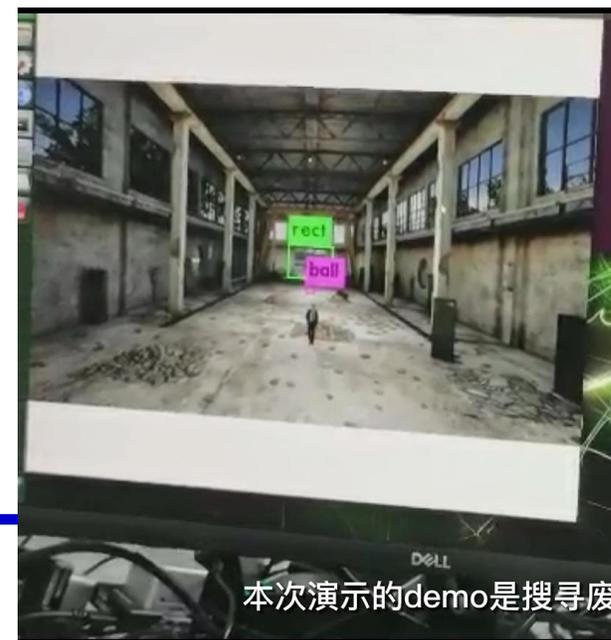
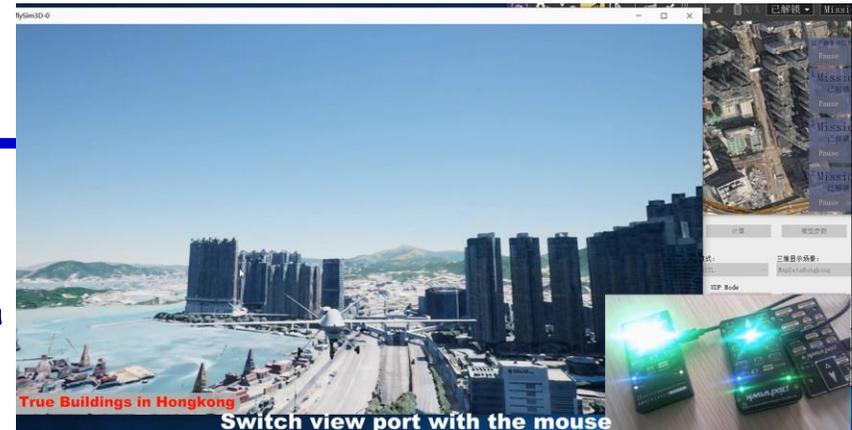




5. 后续章节实验预览

5.11 完整版视频展示

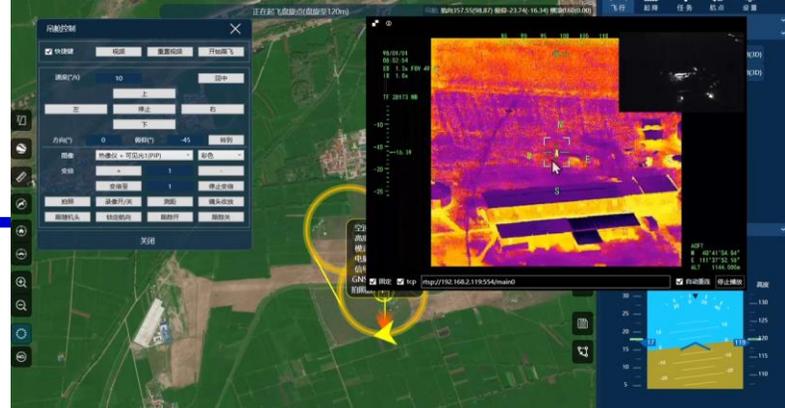
- 视频1: 多固定翼跑道自动起飞与航路硬件在环仿真测试
- <https://www.bilibili.com/video/BV13a411i7sH?p=16>
- 视频2: 全球指定场景的硬件在环仿真 (长沙为例)
- <https://www.bilibili.com/video/BV13a411i7sH?p=17>
- 视频3: 数字孪生&虚拟现实, 香港城市地图导入与单机硬件在环测试
- <https://www.bilibili.com/video/BV13a411i7sH?p=18>
- 视频4: 香港城市地图楼间穿越集群飞行测试
- <https://www.bilibili.com/video/BV13a411i7sH?p=19>
- 视频5: 带视觉的多机集群硬件在环仿真测试系统
- <https://www.bilibili.com/video/BV13a411i7sH?p=20>



本次演示的demo是搜寻废



5. 后续章节实验预览



5.11 完整版视频展示

- 视频6：红外光电吊舱仿真实验
- https://www.bilibili.com/video/BV1z1421z7Az?share_source=copy_web
- 视频7：360激光点云传感器仿真实验
- https://www.bilibili.com/video/BV1PZ42177W9?share_source=copy_web
- 视频8：倾转旋翼无人机硬件在环仿真实验
- https://www.bilibili.com/video/BV1eb421b7pn?share_source=copy_web
- 视频9：直升机软、硬件在环仿真实验
- https://www.bilibili.com/video/BV1TQ4y1c7yv?share_source=copy_web
- 视频10：水下潜航器硬件在环仿真与水下视觉SLAM算法验证
- https://www.bilibili.com/video/BV1Xi421R7Bu?share_source=copy_web





大纲

1. 平台的简介与安装
2. 软、硬件简介和使用
3. 进阶实验例程
5. 后续章节学习方法
6. 总结



飞思实验室



RflySim教程



6. 总结

- RflySim平台介绍和一键安装，并对平台的软、硬件的配置和使用进行介绍。
- 平台配置实验旨在通过实验的方式，帮助学员快速入门RflySim配置和使用。
- 对后续章节的主要内容进行了概括性的预览，帮助学员全面认识本平台包含的所有例程和教学资料。

如有疑问，扫描下方二维码或<https://doc.rflysim.com/>查询更多信息。



RflySim更多教程



扫码咨询与交流



飞思RflySim技术交流群



谢谢！