

1. 实验名称及目的

实验平台&实验流程(定点控制实验): 本实验首先总体介绍本书使用的硬件平台和软件平台, 然后详细介绍仿真平台。其中, 仿真平台包括仿真 1.0 和仿真 2.0 使用的 MATLAB/Simulink 整体模块、多旋翼非线性系统模型、硬件在环仿真整体模块以及实飞实验模块。通过本章各模型和模块的介绍, 读者能够初步了解基于半自主飞控的多旋翼远程控制实验平台的基本构成, 掌握各个软件与硬件的基本功能与使用方式, 为后续的实验打下基础, 提高学习效率。

实验流程中总体介绍实验的流程, 然后以一个效果良好的定点控制实验为例, 详细介绍平台的使用方法和实验的基本操作流程。

2. 实验原理

针对设计所需要的仿真实验平台, 如下图所示。仿真 1.0、仿真 2.0 和硬件在环仿真三者的区别在于多旋翼模型模块不一样: 仿真 1.0 的多旋翼模型模块内部主要包含通过“系统辨识”实验得到的传递函数模型, 即设计模型; 仿真 2.0 的多旋翼模型模块内部包含一个给定的非线性模型和 RflySim3D 显示模块; 硬件在环仿真的多旋翼模型模块内部包含了与 CopterSim 联合仿真的通信接口。

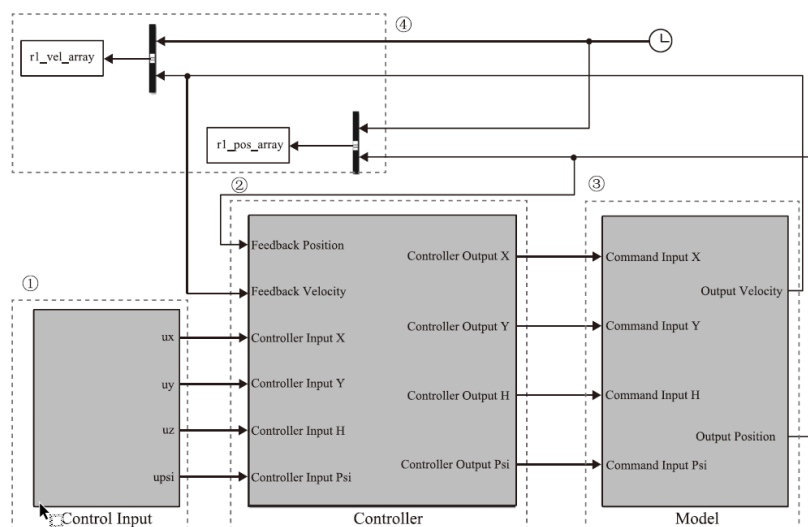


图 2.3 整体模型, Simulink 模型详见 “e0\sim1.0\sample0.slx”

打开“e0\sim1.0\sample0.slx”，如图 2.3 所示，该模型包含四个大模块，分别为：虚线框①——“Control Input”，期望输入模块，用于输入期望的飞行器位置信息；虚线框②——“Controller”，控制器模块，用于设计控制器，实现飞行器的有效位置控制；虚线框③——“Model”，飞行器仿真模型模块，用于接收控制器输出的控制量，并输出飞行器的位置和速度信息；虚线框④——飞行器状态数据收集模块，该模块拥有两个变量——“r1_pos_array”与“r1_vel_array”，分别用于收集单次仿真全过程中的位置与速度反馈，并以矩阵的形式保存到 MATLAB 工作空间。矩阵的每一行信息均是以“时间，水平前向通道位置/速度，水平侧向通道位置/速度，高度通道位置/速度，偏航通道角度/偏航角速率”方式自左而右排列的一组值，矩阵的列按时间顺序自上而下排列。以变量“r1_pos_array”为例，其存储信息形式如表 2.1 所示。

表 2.1 r1_pos_array 变量存储信息形式

时间	水平前向通道位置	水平侧向通道位置	高度通道位置	偏航通道角度
----	----------	----------	--------	--------

硬件在环模块内部如图 2.12 所示，该模块是 Simulink 模型和 RflySim 平台（Copter-Sim）之间的通信桥梁，接收 Simulink 模型中发出的控制指令并返回硬件在环仿真过程中多旋翼的位置、速度和欧拉角等数据。图中虚线框①用于将需要发送的数据打包成统一的数据结构作为虚线框②的输入。虚线框②是 simulink 和 CopterSim 通信的核心模块，包含两个作用，一是将硬件在环仿真需要的数据发送给 CopterSim，一是从 CopterSim 中接收反馈信息。硬件在环仿真的 Simulink 文件保存在每一章的设计实验文件夹中，与软件仿真使用的文件区别在于用硬件在环模块替换了多旋翼模型模块。这是因为硬件在环仿真的多旋翼模型在 CopterSim 中。

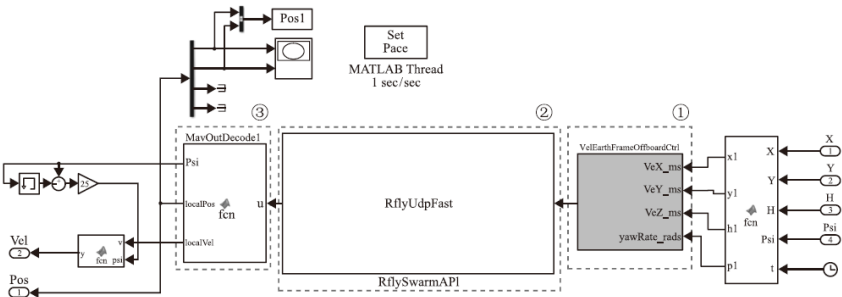


图 2.12 硬件在环模块内部

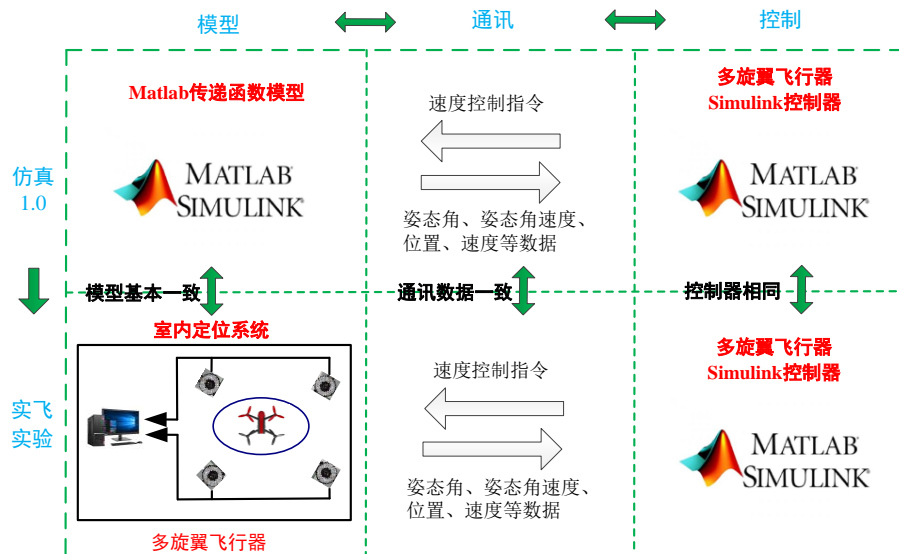


图 实飞阶段流程

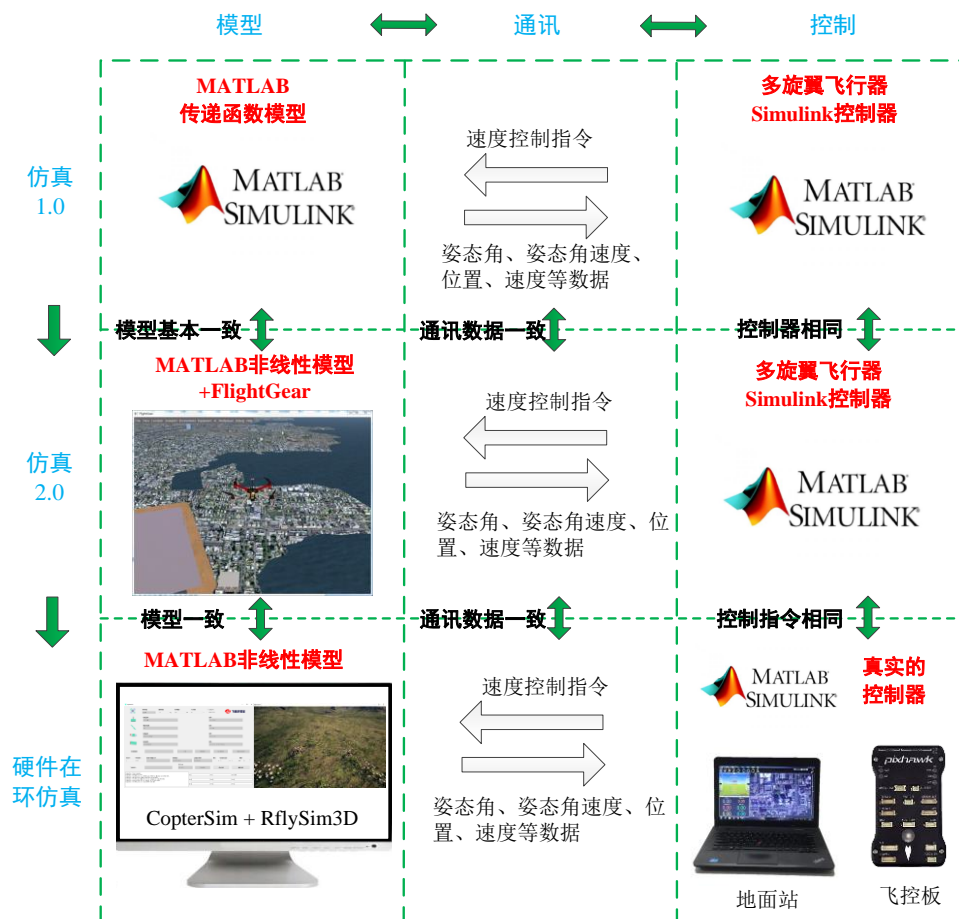


图 仿真阶段流程

3. 实验效果

实现在 Simulink 中直接控制四旋翼无人机飞行，在 RflySim3D 中显示飞行效果。

4. 文件目录

文件夹/文件名称		说明
sim1.0	Ch0F1.fig	水平前、侧及高度通道幅值图
	e0_plot.m	水平前、侧及高度通道幅值图绘制文件
	New_Export.mat	Simulink 仿真输出数据文件
	sample0_2017b.slx	期望轨迹(直线与圆)输入的仿真 1.0 文件
	sample1_2017b.slx	输入为(1,1,1,0)的仿真 1.0 文件
	sample2_2017b.slx	实飞试验的仿真 1.0 文件
	startSimulation.m	初始化参数文件
sim2.0	icon	图标及模型参数文件夹
	Ch0F1.fig	水平前、侧及高度通道幅值图
	Ch0F2.fig	水平前、侧及高度通道幅值图
	e0_plot.m	水平前、侧及高度通道幅值图绘制文件
	sample0_2017b.slx	非线性模型的仿真 2.0 文件
	sample1_2017b.slx	非线性模型的仿真 2.0 文件
	startSimulation.m	初始化参数文件
SoftwareSimExps	软件在环仿真例程，见 SoftwareSimExps\Readme.pdf	
HIL	sample0_2017b.slx	HIL 期望轨迹(直线与圆)输入的仿真 1.0 文件
	sample1_2017b.slx	HIL 输入为(1,1,1,0)的仿真 1.0 文件
	satgd.m	保方向的饱和函数
	startSimulation.m	初始化参数文件
	RflyUdpFast.mexw64	MATLAB 中通过 MEX 编译生成的 S 函数模块文件
	HITLRun.bat	一键启动硬件在环仿真脚本
Rfly	RflyData	飞行日志数据文件
	e0_plot.m	图像绘制程序
	start_tello.m	初始化文件，会自动运行 start.m 文件
	start.m	初始化文件
	satgd.m	保方向的饱和函数
	satgd_planning.m	
	build_ros_model.sh	MATLAB 自动生成的 ROS 代码
	sample1_2017b.slx	Simulink 程序文件

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 ^①	1
2	RflySim 平台个人版	Pixhawk 6C ^②	1
3	MATLAB2017b 及以上	遥控器 ^③	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://doc.rflysim.com>

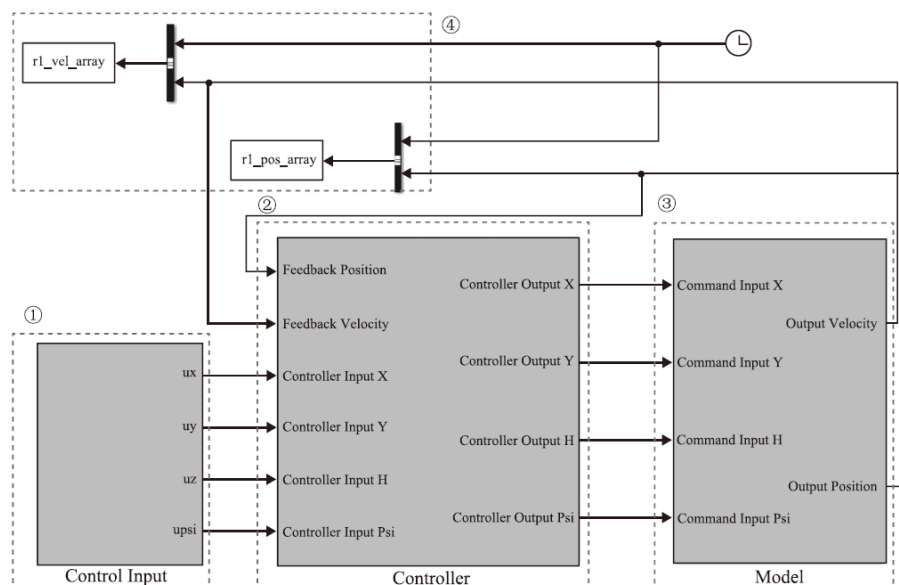
②：须保证平台安装时的编译命令为：px4_fmu-v6c_default，固件版本为：1.13.3。其他配套飞控请见：<http://doc.rflysim.com>

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：..\e11_RC-Config\Readme.pdf

6. 仿真 1.0(sim1.0)实验步骤

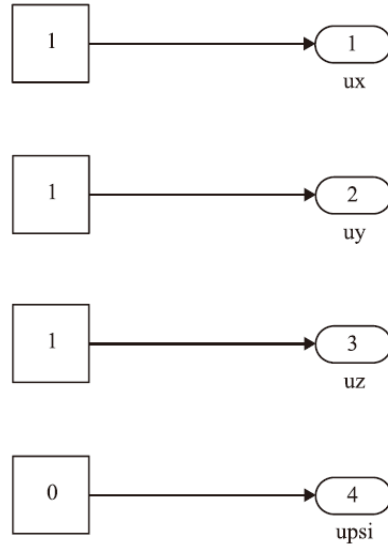
Step 1:

这里我们给出一个设计好的例子，见文件打开 [sim1.0\sample1_2017b.slx](#) 该文件，该模型中“Model”模块采用的是设计模型（线性模型）。请仔细观察和分析其中的子模块的实现方法，并进行功能完善。

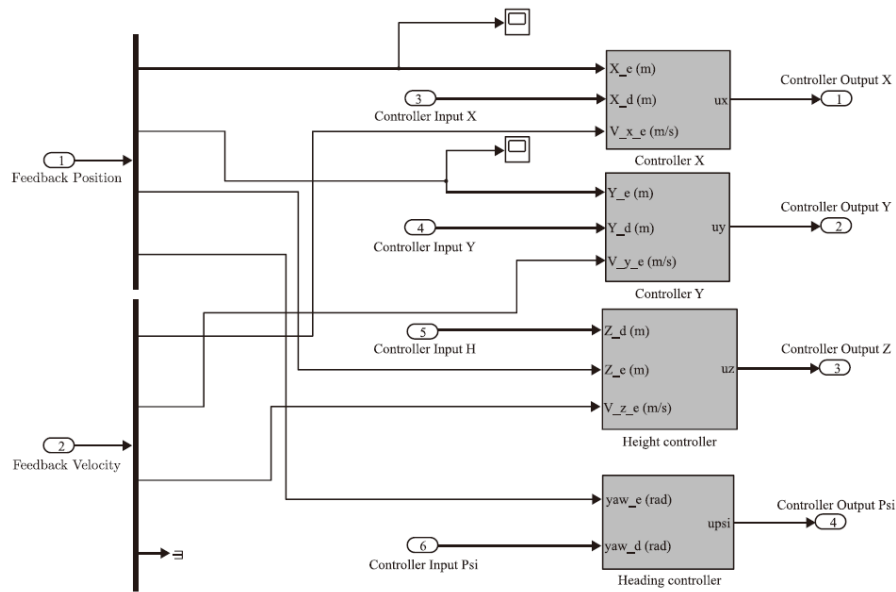


Step 2:

建立控制器子模块：打开上图中的“Controll Input”模块，设定相应的定点期望。



设置完定点期望后，打开上图中的“Controller”模块，查看各个通道的定点控制器形式和内容。如下图所示是控制器的内部形式。具体原理请见文献[4]第 2.2.2 节的控制器模块部分。

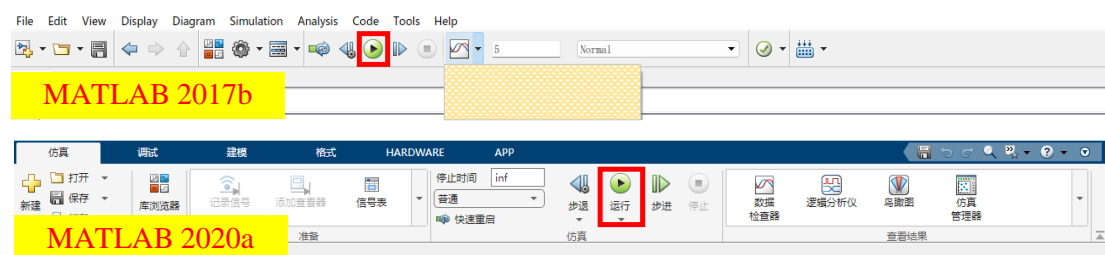


Step 3:

双击 MATLAB 文件 [sim1.0\startSimulation.m](#)，并单击工作界面中的“运行”(Run)按钮。

Step 4:

单击 Simulink 的 [sim1.0\sample1_2017b.slx](#)，工具栏“开始仿真”按钮开始仿真。



水平前向通道、水平侧向通道和高度通道的仿真结果可以通过运行文件 [sim1.0\le0_plot.m](#) 得到，如下图所示，这三个变量从起点位置迅速到达 (1, 1, 1) 位置，达到了预期效果。

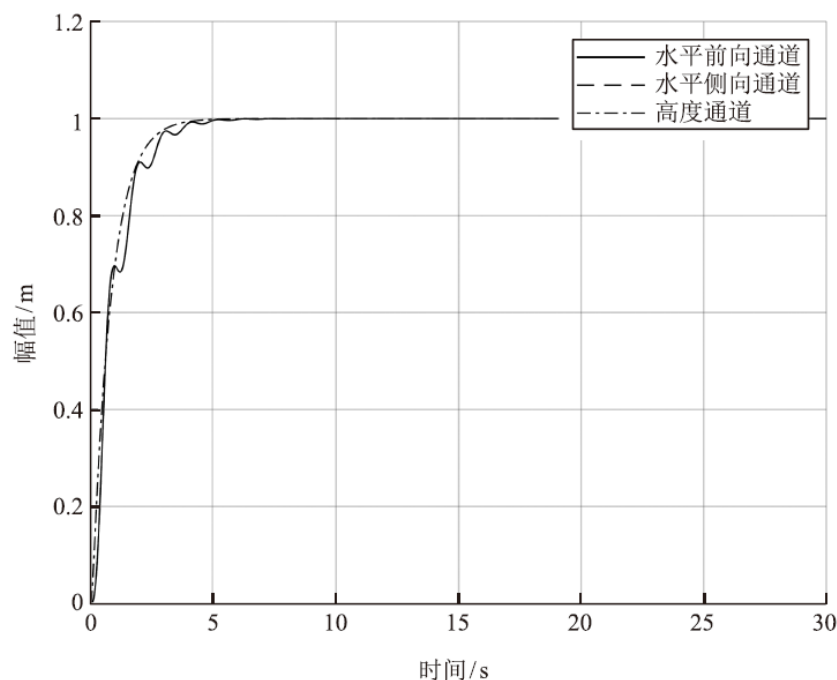


图 3.4 各通道位置响应曲线

7. 仿真 2.0(sim2.0)实验步骤

基于设计模型，已经完成了仿真 1.0，下一步需要在非线性模型上进行验证，即仿真 2.0。

Step 1:

此处控制器与[仿真 1.0 实验步骤](#)相同。

Step 2:

打开 [sim1.0\sample1_2017b.slx](#)，将“Model”模块里的传递函数模型替换成非线性模型，然后按要求设置期望输入。这里我们给出一个设计好的例子，见文件 [sim2.0\sample1_2017b.slx](#)，其与 [sim1.0\sample1_2017b.slx](#) 的区别在于此处的多旋翼模型为非线性模型。

Step 3:

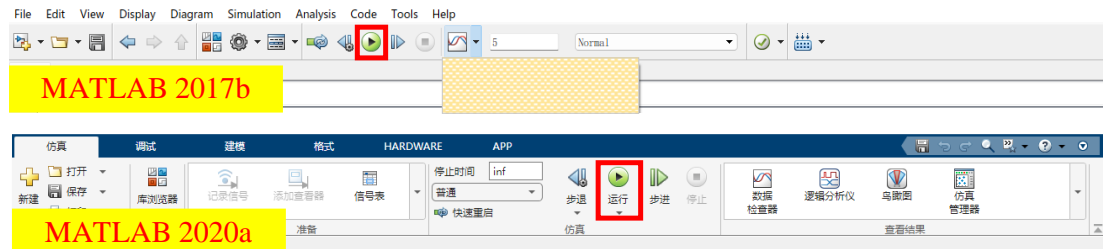
在桌面“*\桌面\RflyTools\RflySim3D.lnk”打开 RflySim3D 软件。

Step 4:

双击 MATLAB 文件 [sim2.0\startSimulation.m](#)，并单击工作界面中的“运行”(Run)按钮。

Step 5:

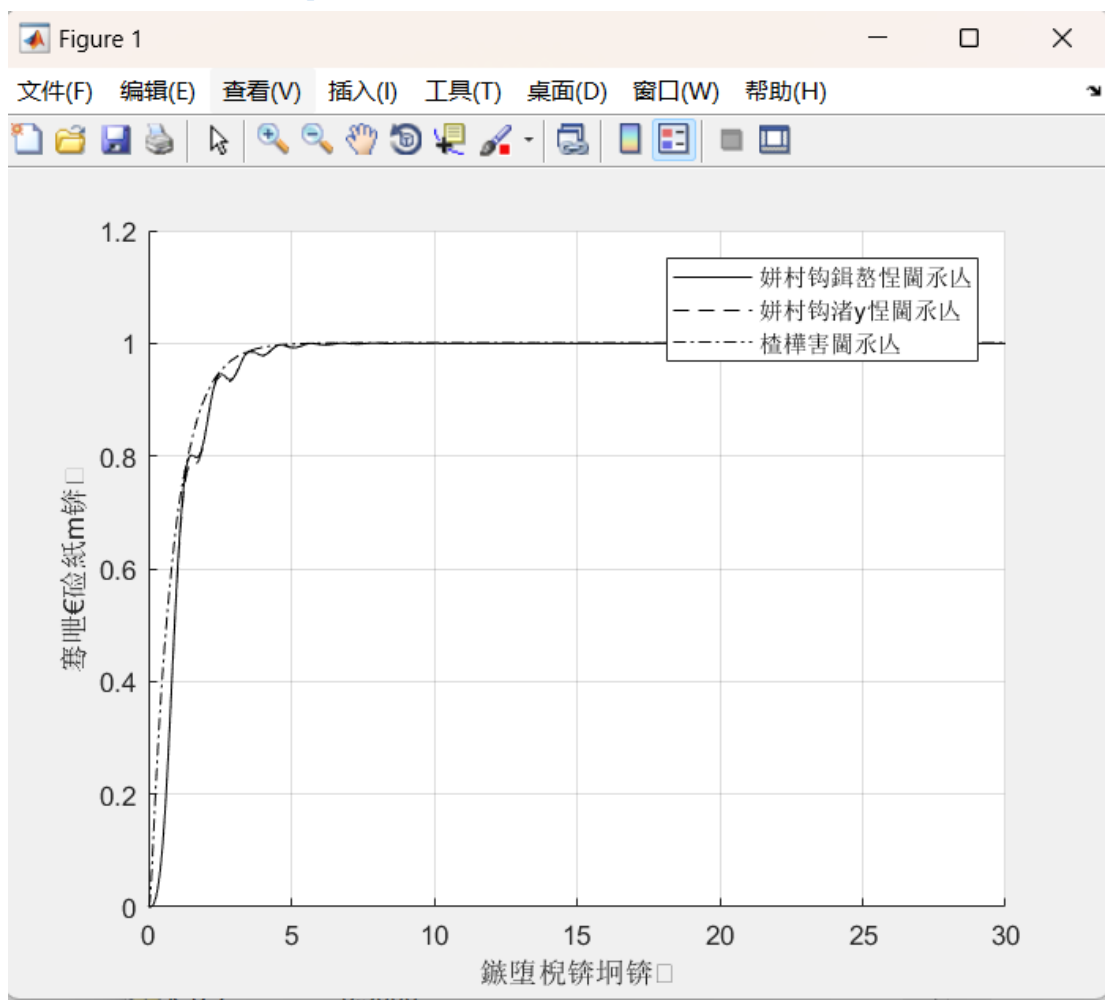
单击 Simulink [sim2.0\sample1_2017b.slx](#)，工具栏“开始仿真”按钮开始仿真。



此时，可以在 RflySim3D 中观察到：多旋翼到达指定定点。这说明控制器实现了预期要求。

Step 6:

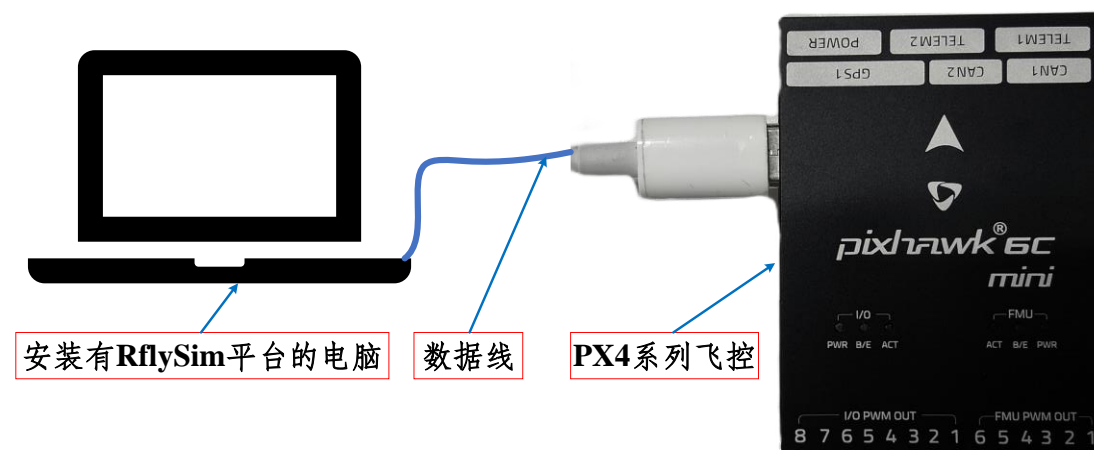
双击打开 [sim2.0\e0_plot.m](#)，并单击工作界面中的“运行”(Run)按钮。



8. 硬件在环仿真(HIL)实验步骤

Step 1:

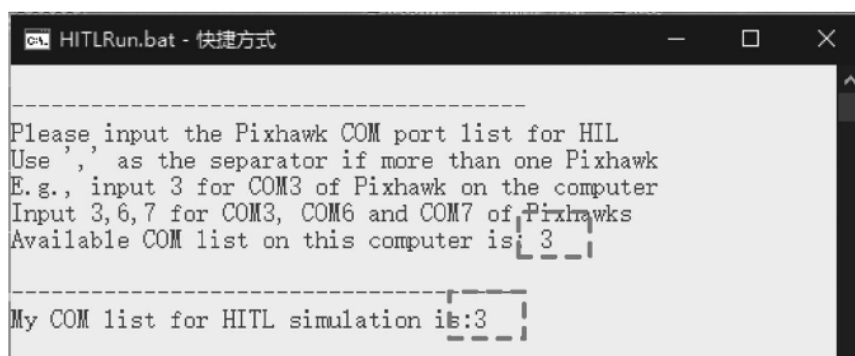
连接硬件：将 Pixhawk 自驾仪与计算机通过 USB 数据线连接。



Step 2:

选择 HIL 脚本：打开 [HIL\HITLRun.bat](#) 一键启动硬件在环仿真脚本，在弹出的命令行中。输入 PX4 系列自驾仪显示的串口号，如这里是“3”，只需把该串口号输入下方即可。

注：本脚本也可支持多机仿真，多机硬件在环仿真时，需要插入多个飞控到电脑中，双击“*桌面\RflyTools\HITLRun.lnk”后弹出的命令行中提示的串口号个数应与连接的自驾仪个数一致，需要在下方输入所有给出的串口号，每个串口号用逗号分隔开。



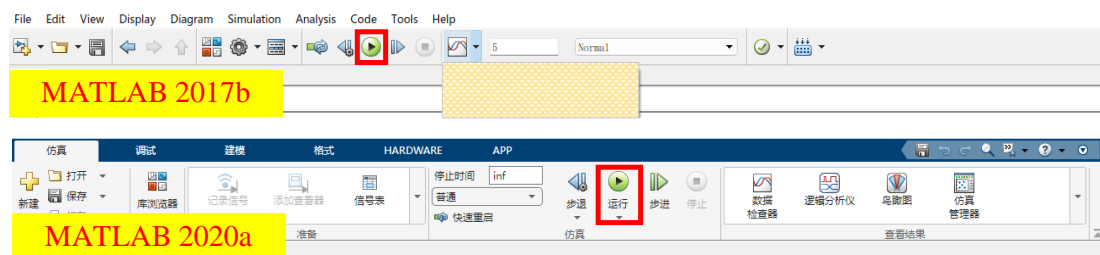
在填写完串口号后，按下回车键，系统会自动打开所有与硬件在环仿真相关的 RflySim3D、CopterSim 和 QGC 地面站。等待 CopterSim 的左下角状态框中显示：PX4: GPS 3D fixed & EKF initialization finished。

Step 3:

双击 MATLAB 文件 [HIL\startSimulation.m](#)，并单击工作界面中的“运行”(Run)按钮。

Step 4:

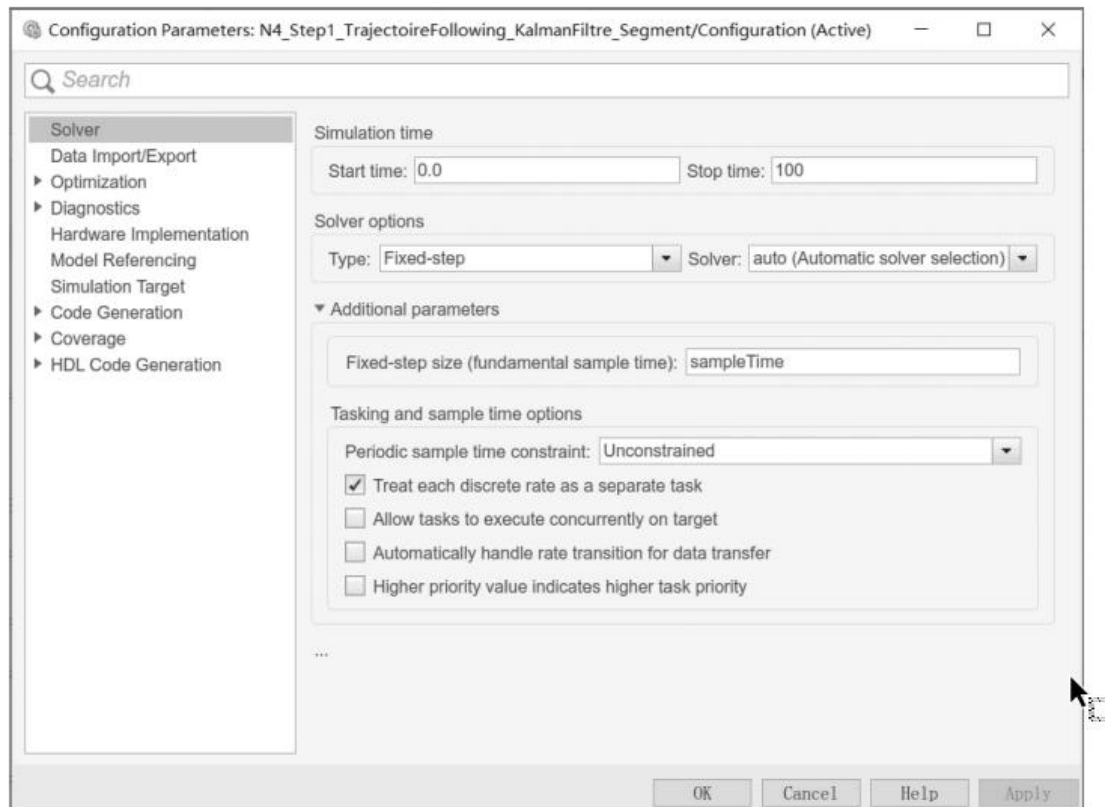
在 Simulink 中打开 [HIL\sample1_2017b.slx](#) 程序，单机“运行”。



可以实现自驾仪硬件在环仿真，仿真结果通过 RflySim3D 软件实时显示。



注意：1）本实验中的仿真模型均在 R2017b 版本中运行，所以需要计算机中的版本为 R2017b 及以上。2）所有的仿真模型均需要设定固定的仿真步长。在 Simulink 界面的上方菜单栏中单击“Model Configuration Parameters”进入如下图所示的界面，在其中单击“Solver”“Slover options”“Type”，在下拉菜单中选择“Fixed-Step”选项。单击其下方的“Additional Parameters”，在“Fixed-Stepsize”一栏中填入变量名“sample-Time”。这样就规定好了仿真模型的固定仿真步长，而变量“sampleTime”值设置在“startSimulation.m”文件中。

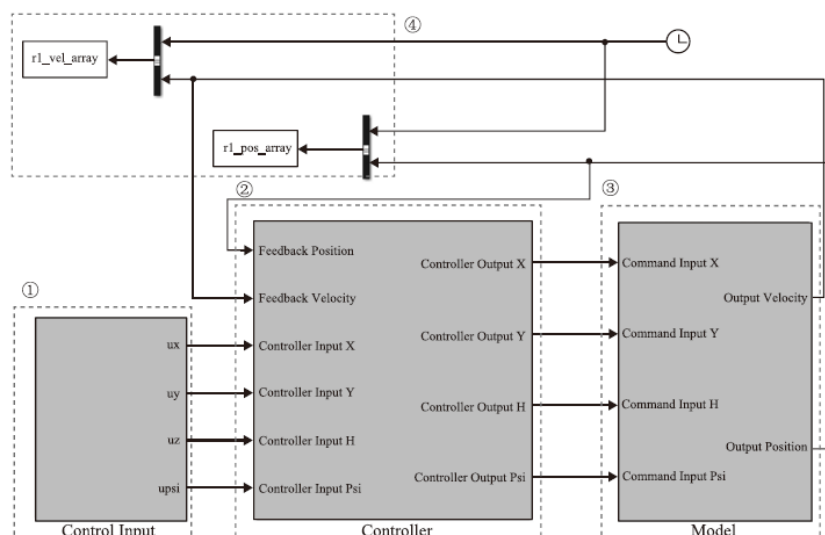


9. 实飞(Rfly)实验步骤

由于设计的控制器最终需要运用于真实的多旋翼上，因此，在实飞之前，进行基于真实的多旋翼模型的仿真实验非常必要。实飞实验的仿真 1.0 与仿真实验的仿真 1.0 区别就在于多旋翼模型，此处的模型是通过真实的多旋翼通道辨识获得的，其他模块与仿真实验一样。

Step 1:

新建一个 **Simulink** 文件，在其中设计多旋翼的定点控制器，设计要求与仿真阶段的要求一致。这里我们给出一个设计好的例子，见文件 [sim1.0\sample2_2017b.slx](#)，打开该文件后的 **Simulink** 框图如下图所示。请仔细阅读其中的子模块的实现方法，并进行功能的完善。



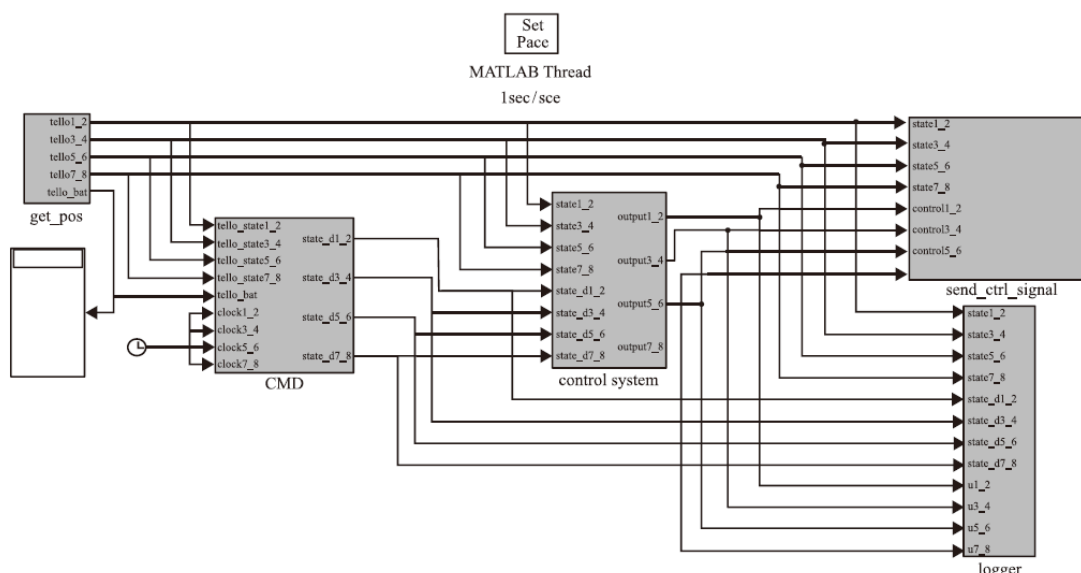
Step 2:

其余步骤请见[仿真 1.0 实验步骤](#)中的 [Step 2:](#)、[Step 3:](#)、[Step 4:](#)。开始仿真。水平前向通道、水平侧向通道和高度通道的仿真结果，这三个变量从 (0, 0, 0) 迅速到达 (1, 1, 1) 位置，达到了预期效果。

Step 3:

此处将上一节通过仿真测试的控制器用于真实的多旋翼上验证。

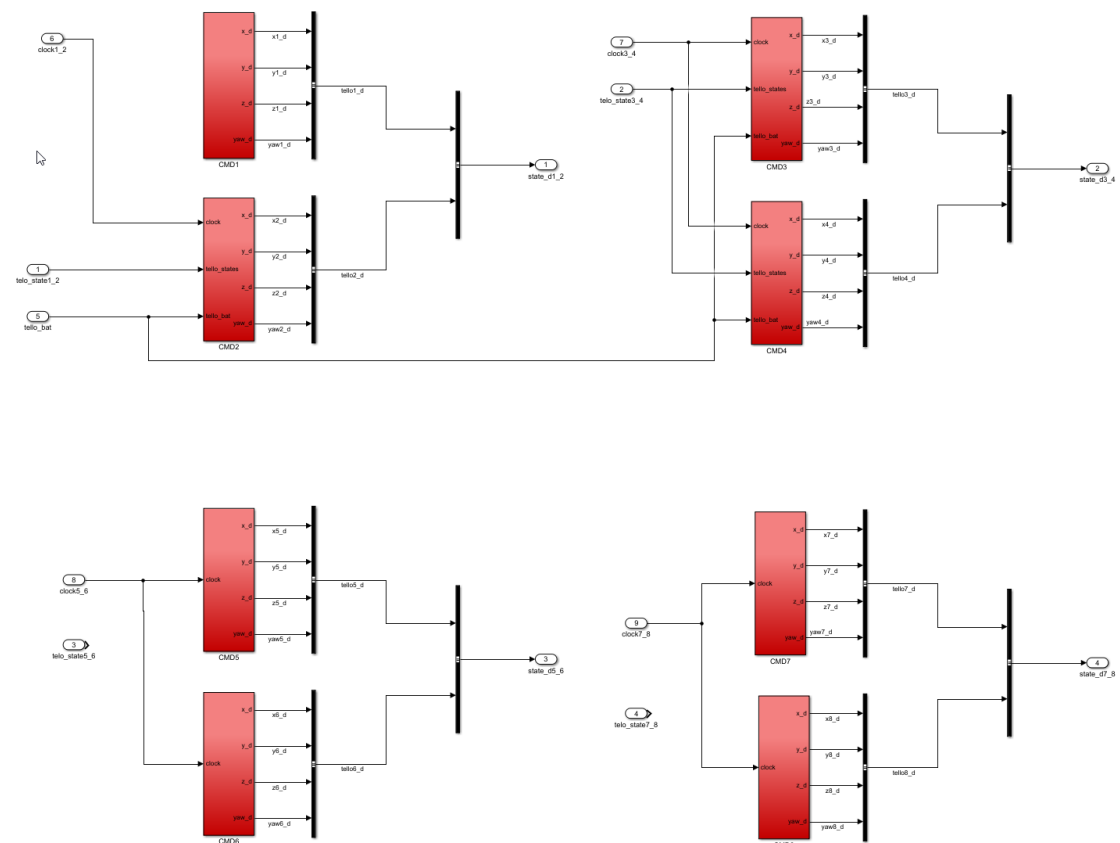
控制器设计：控制器与仿真 1.0 控制器相同，实飞实验相对复杂，可根据需要对控制参数进行微调。这里我们给出一个设计好的例子，见文件 [Rfly/sample1_2017b.slx](#)，打开该文件后的 Simulink 框图见下图。



Step 4:

建立控制器子模块：打开上图的“CMD”模块，在该模块中打开“CMD1”子模块，该模块用于设定相应的定点期望。设置完定点期望后，打开图 3, 15 中的“controlsystem”

模块内的“Baseline Controller Basic”子模块，查看各个通道的定点控制器形式和内容。



Step 5: 实飞准备

实飞环境如下图所示，按照以下步骤进行实验。1) 首先启动实飞实验所需要的各个设备，分别为：①用于进行数据通信的路山器；②用于实时获得多旋翼运动状态的室内定位系统；③用于作为 Simulink 模型控制载体的计算机；④用于飞行实验的多旋翼。



2) 在启动设备之后需要将所有设备连接到路由器提供的局域网内，实现设备之间

的通信。

3) 为了实现多旋翼运动状态的实时反馈,需要在室内定位系统中建立多旋翼的模型。

4) 在完成上述准备工作后,可以开始进行多旋翼的实飞实验。

Step 6:

首先查找多旋翼在当前局域网下分配的 IP 地址;修改仿真平台中“launch”文件的相应多旋翼 IP 地址;打开系统“Terminal”,启动实飞实验。

Step 7:

启动室内定位系统。在打开的系统“Terminal”中输入

```
ros launch mocap_optitrack multirigidb0dy8.launch
```

并运行。

Step 8:

连接多旋翼。在系统“Terminal”中输入

```
ros launch tell_driver tell_node.launch
```

并运行。

Step 9:

起飞多旋翼。在系统“Termmal”中输入:

```
roslaunch tello Tell_takeoff_all
```

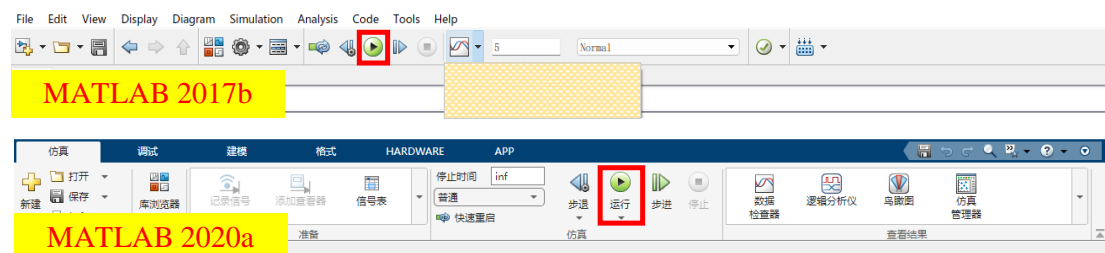
并运行。

Step 10:

运行 Simulink 仿真控制程序。首先打开 MATLAB/Simulink,运行“[Rfly\start_tello.m](#)”,启动文件加载相应数据,打开控制模型 [Rfly\sample1_2017b.slx](#)。

Step 11:

一切准备就绪后,单击 [Rfly\sample1_2017b.slx](#) 控制模型中 Simulink 工具栏“开始仿真”按钮开始仿真。



此时可以在飞行场地内观察到,多旋翼爬升一段时间后向左飞行,说明控制器实现预期要求。

Step 12:

单击 Simulink 中的“停止仿真”，然后在系统“Terminal”中输入：

```
roslaunch tello Tello_land_all
```

并运行，则多旋翼断电。

10.参考资料

- [1]. Quan Quan. Introduction to Multicopter Design and Control. Springer, Singapore, 2017
- [2]. 全权 杜光勋 赵峙尧 戴训华 任锦瑞 邓恒译 多旋翼飞行器设计与控制 [M] 电子工业出版社 2018.
- [3]. 全权 戴训华 王帅 多旋翼飞行器设计与控制 实践 [M] 电子工业出版社 2020.
- [4]. 全权 等.多旋翼无人机远程控制实践[M].电子工业出版社,2022.

11.常见问题

Q1: ****

A1: ****