

# 多旋翼无人机远程控制实践

## 第2&3章 实验平台和实验流程

全 权

qq\_buaa@buaa.edu.cn

自动化科学与电气工程学院



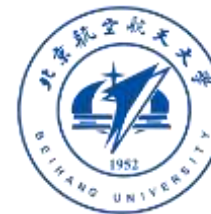
北京航空航天大学  
BEIHANG UNIVERSITY



可靠飞行控制研究组

RELIABLE FLIGHT CONTROL GROUP

# 大纲



**1.总体介绍**

**2.仿真平台**

**3.实飞平台**

**4.实验流程**

**5.平台优势**

# 1.总体介绍

## □硬件平台

- 室内定位系统
- 计算机/工作站
- 带半自主飞控的多旋翼飞行器
- 自驾仪系统



图. 自

# 1.总体介绍

## □软件平台



图. FlightGear

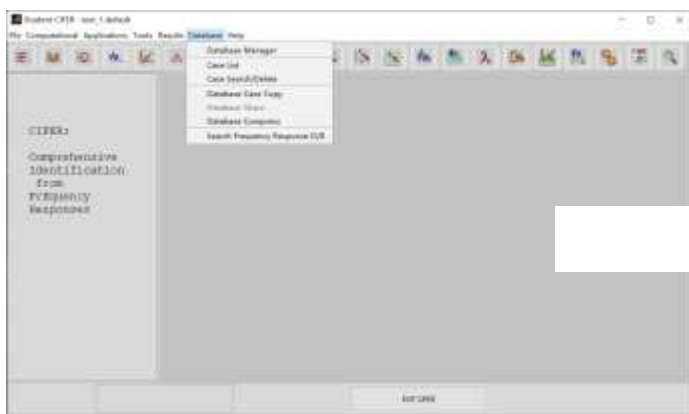


图. CIFER工具箱



图. MATLAB



图. RflySim3D



图. CopterSim



# 1.总体介绍

## □软件平台

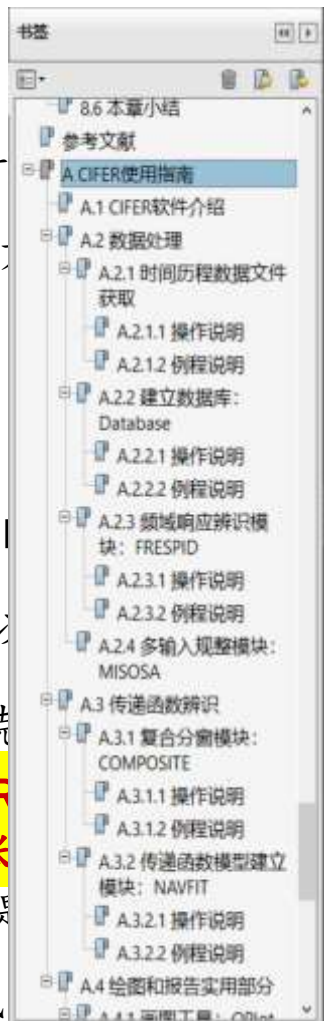
### ■ FlightGear飞行模拟器

一款非常受欢迎的开源飞行模拟器，接收Simulink发送的飞行状态，并模拟飞机的飞行状态。

### ■ CIPHER工具箱

- Comprehensive Identification
- 线性调频Z变换、多输入处理以及
- 有效滤除各种干扰的影响，克服模型估计作用的缺陷
- 得到满意的能用于操作性能提升的控制系统设计的多输入多输出线

Tischler M, R  
验案例[M]. 张



### 附录 A CIPHER 使用指南

CIPHER 软件是适用于系统辨识的高效专业软件，它采用频域响应来进行系统辨识，获得的结果不仅精确而且符合实际的要求。附录的主要目的在于满足本书第四章“多输入系统辨识实验”章节的要求，介绍 CIPHER 学生版的简单功能，如要更加深入理解 CIPHER 软件以及多输入系统辨识方法请参考 [11] 和 [16]。本章主要通过数据处理、传递函数辨识以及绘图和实用工具三个部分介绍基于 CIPHER 软件的系统辨识过程。同时，在 CIPHER 软件各个功能介绍完之后，我们将通过一个实例展示该模块操作的过程和结果。

#### A.1 CIPHER 软件介绍

在第四章的实验内容中，我们不需要进行参数辨识以及状态空间模型验证，因此忽略“DE-RIVID”模块和“VERIFY”模块。简化后的 CIPHER 主要流程如图 A.1 所示，将传递函数的操作主要分为数据处理、传递函数辨识以及绘图和报告实验三个部分。下面对三个部分进行简要介绍。

##### (1) 数据处理

数据处理部分主要包括时间历程数据获取、频域响应辨识“FRESPID”和单输入重整“MISOSA”三个部分。时间历程数据包含时间变量、输入变量和输出变量，它是将辨识的飞行器或者模型生成的单通道或者耦合通道的输入输出数据。获得时间历程数据之后，如果是单通道的输入输出，则进入“FRESPID”模块进行频域响应辨识；如果是耦合通道的输入输出，则进入“MISOSA”模块进行频域响应辨识。

##### (2) 传递函数辨识

传递函数辨识部分主要包括复合分频“COMPOSITE”模块和频域响应拟合“NAVFIT”模块。“COMPOSITE”模块可以获得精确的频域响应辨识结果，主要用来将经过“FRESPID”模块和“MISOSA”模块处理后的数据加权结合在一起，用以消减数据中处理各个部分的缺点，同时将各自的优点综合起来，提高辨识的准确度。“NAVFIT”模块是将符合频率范围要求的数据，经过相关操作，辨识得到传递函数模型。

##### (3) 绘图和实用工具

绘图和实用工具部分可以分为方向“RMS”模块，计算频域辨识结果带宽与裕量“BANDWIDTH”模块，计算穿越频率“CROSSOVER”模块、干扰抑制带宽“DRB”模块以及绘图“QPlot”模块。通过这些模块，读者可以知道频域辨识的一些图表以及相关数据结果。

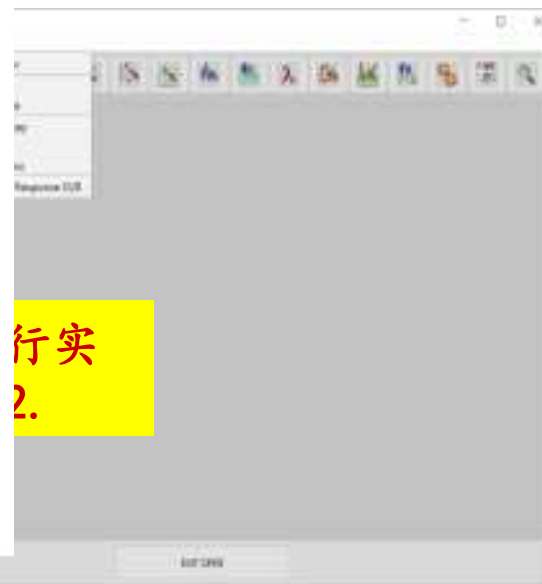
#### A.2 数据处理

##### A.2.1 时间历程数据文件获取

FlightGear



LIGHT SIMULATOR



行实  
2.

# 1.总体介绍

## □软件平台

### ■ CopterSim 实时运动仿真软件

北航可靠飞行控制研究组开发的针对Pixhawk PX4自驾仪平台的处理器在环仿真软件，可以在软件中建立模型，通过USB串口与Pixhawk自驾仪连接来实现硬件在环仿真，达到室内模拟室外飞行测试的效果。

### ■ RflySim3D 三维可视化视景软件

北航可靠飞行控制研究组开发的多旋翼三维显示软件，通过UDP接收CopterSim的飞行数据来实时显示多旋翼的姿态与位置。CopterSim仿真程序与RflySim3D显示软件两者共同构成了硬件在环仿真平台，两者分布式独立运行机制保证了多机多视角仿真任务的完成。

[https://rflysim.com/zh/3\\_Using/HILSimulator.html](https://rflysim.com/zh/3_Using/HILSimulator.html)



图. CIPHER工具箱



图. CIPHER工具箱

# 1.总体介绍

## □软件平台

### ■ 基于Simulink的控制器设计与仿真平台

包含一个高逼真的多旋翼非线性模型，可以用来仿真多旋翼的各种动态特性，并用FlightGear图形化实时展示多旋翼飞行轨迹与姿态等状态信息。



图. CIFER工具箱



# 1.总体介绍

## □软件平台



图. FlightGear

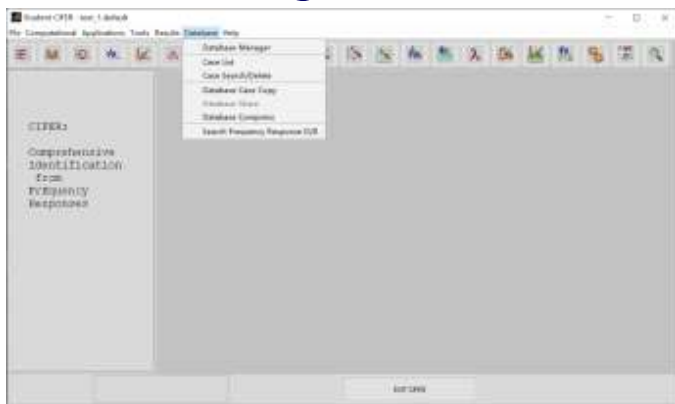


图. CIFER工具箱

硬件在环仿真  
仿真2.0



图. RflySim3D



图. CopterSim



## 2.仿真平台



### □整体模型

- 期望输入模块—Control Input
- 控制器模块—Controller
- 多旋翼模型模块—Model
- 飞行器状态数据收集模块

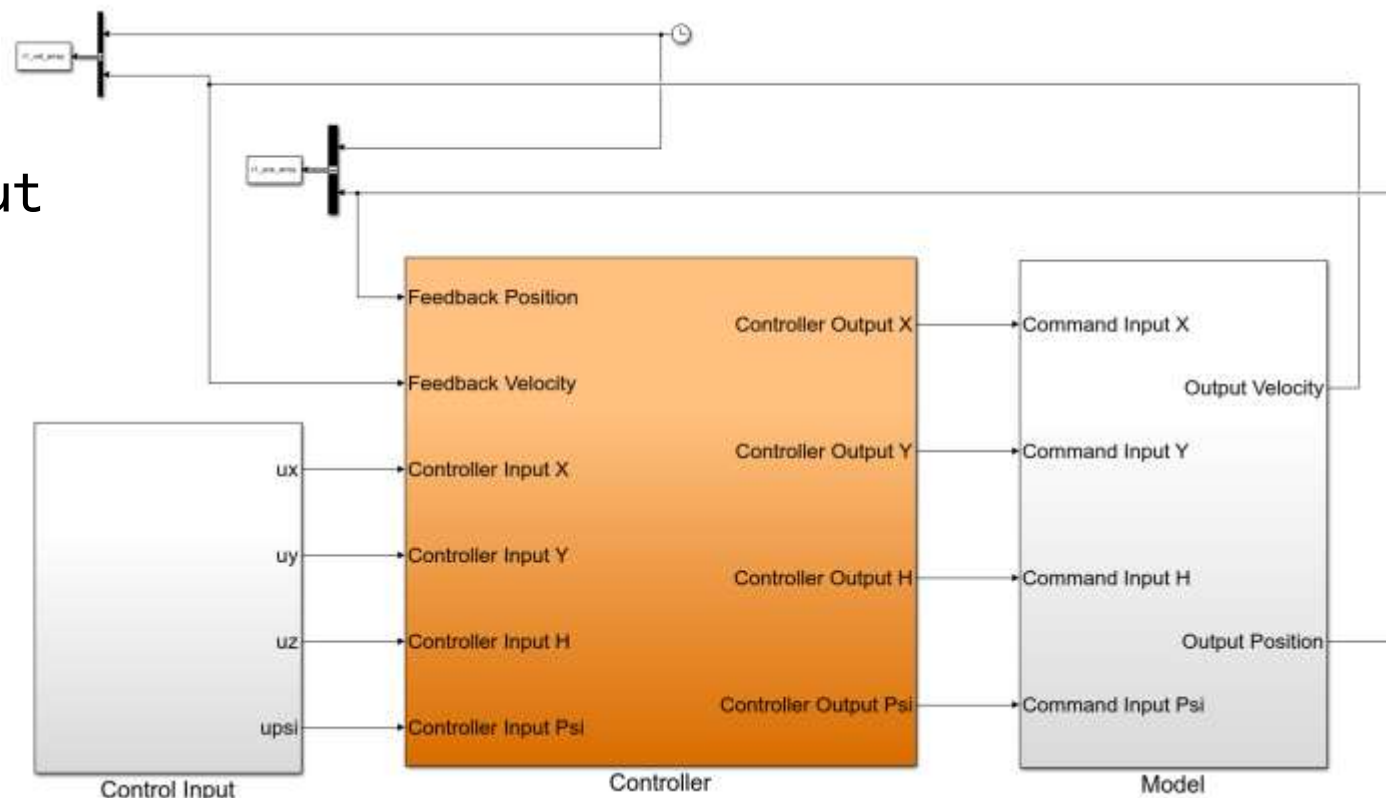


图. 整体模型

## 2.仿真平台

### □模块详解

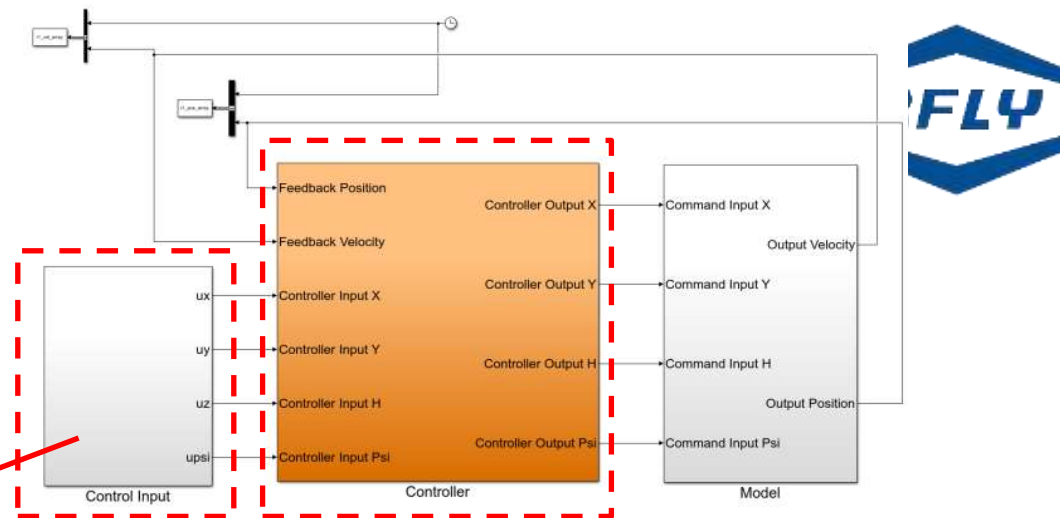
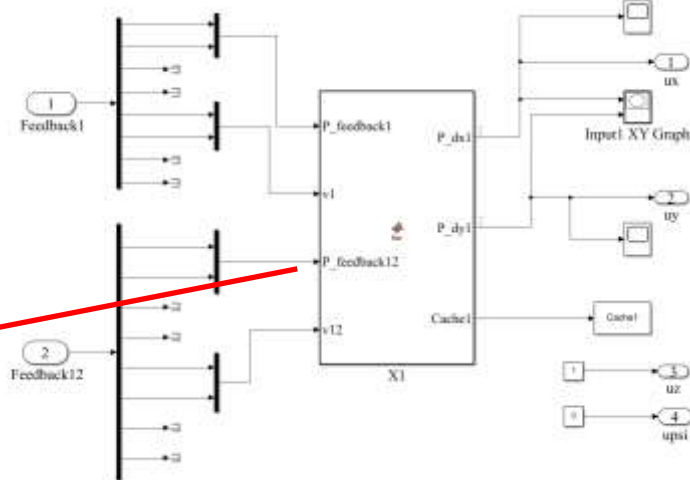
#### ■ 期望输入模块

```
ksi_wp_ksi=ksi-ksi_wp; % filtered error
ksi_0=P2+k0*v12;
ksi_0_ksi=ksi-ksi_0; % filtered error

% Smooth transition curve
d1=15;d2=25;
d0=k2/((d1-r_0_)^2+0.000001)/(d1+0.00
A=-2*d0/(d1-d2)^3;
B=-3/2*(d1+d2)*A;
C=-3*A*d1^2-2*B*d1;
D=d0-A*d1^3-B*d1^2-C*d1;

a=k1;
% Limit the range of repulsion field
```

图. 避障实验部分代码



#### ■ 控制器模块

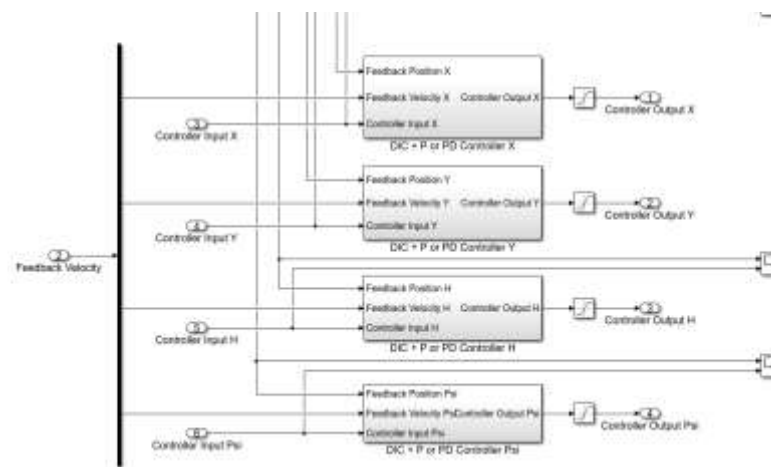


图. 跟踪控制器

# 2.仿真平台

## ■ 多旋翼模型模块

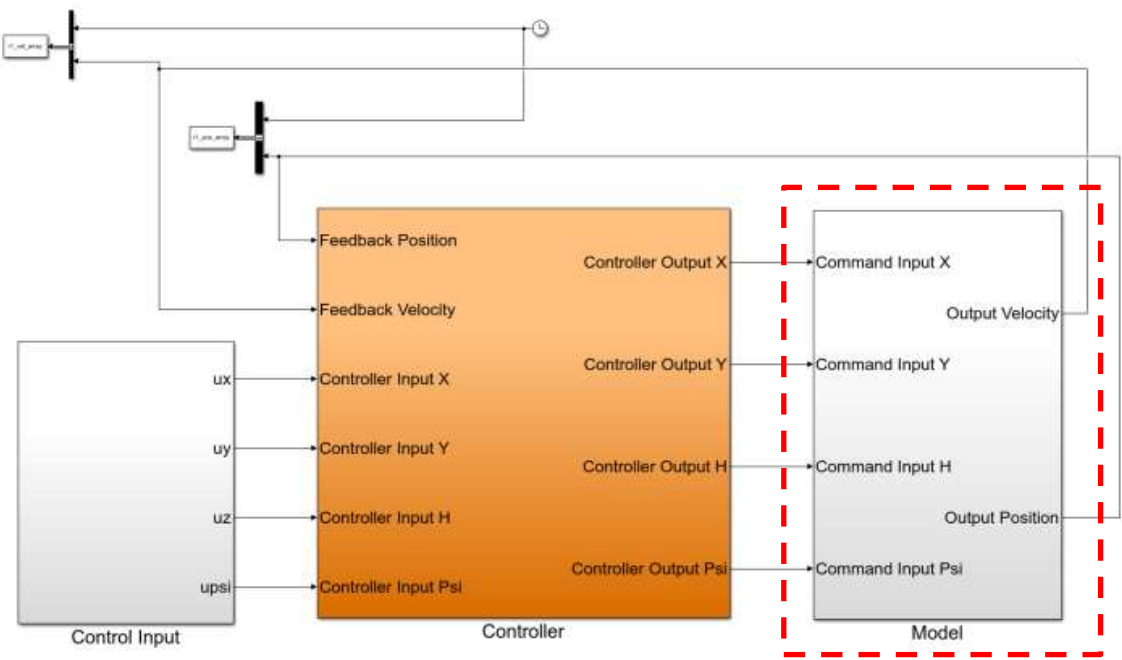


图. 整体模型



图. 仿真阶段流程

# 2.仿真平台

## ■ 多旋翼模型模块

1) 非线性系统模型——仿真2.0实验

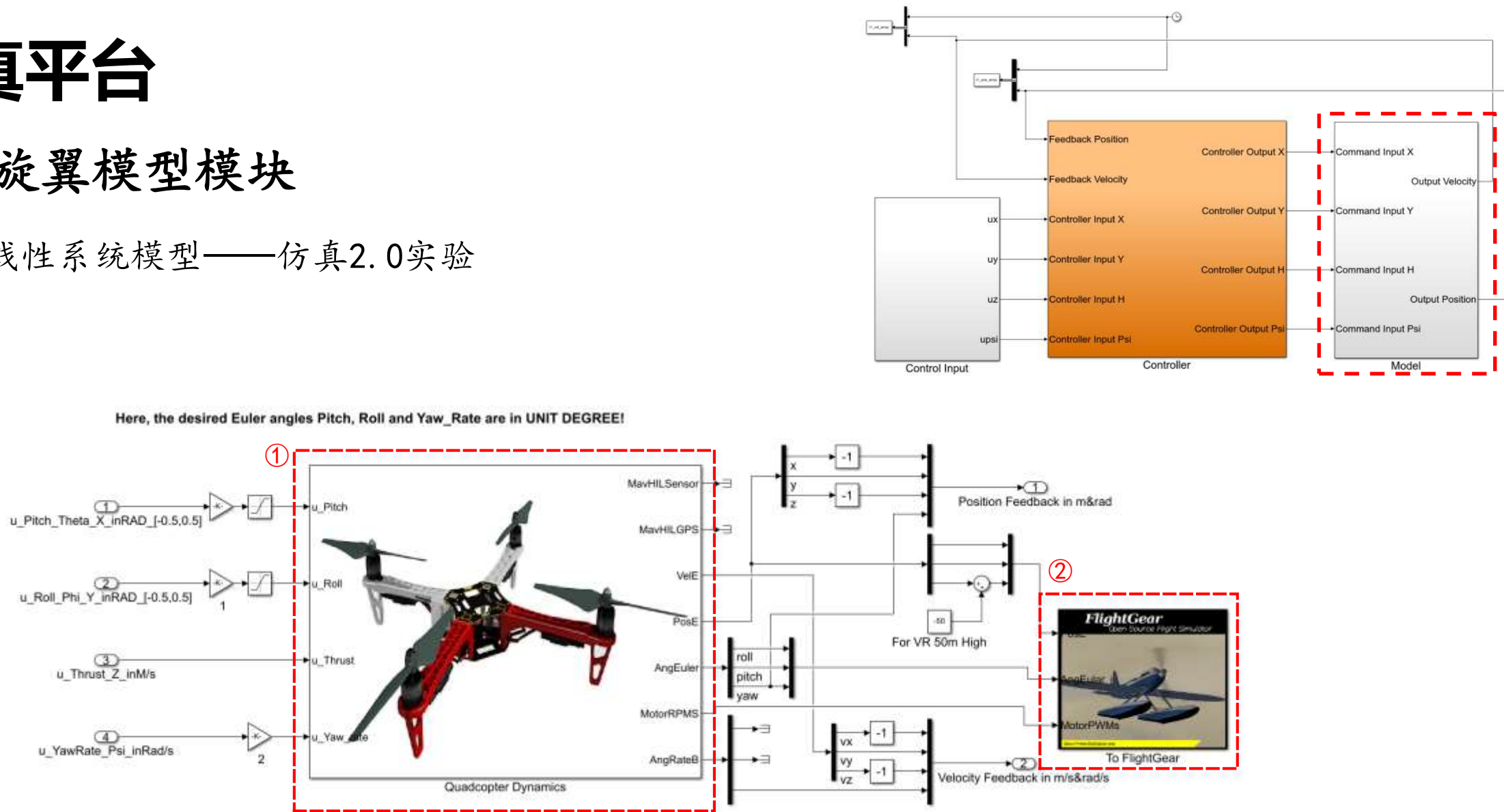


图. 多旋翼非线性系统模型，Simulink模型 “e0\Sim2.0\sample0.slx”



# 2.仿真平台



## ■ 多旋翼模型模块

### 1) 非线性系统模型——仿真2.0实验



图. 多旋翼非线性系统模型内部组成

- 模型输入：电机的PWM控制量
- 模型输出：多旋翼的状态和传感器信息
- 电机模型①：模拟电机动态；
- 力和力矩模块②：模拟螺旋桨拉力、机身气动力、自身重力以及地面支撑力等所有的外部力和力矩；
- 刚体运动动态模型③：计算多旋翼的速度、位置和姿态等运动学状态；
- 环境模型④：根据高度计算加速度、空气密度等数据；
- 故障模型⑤：主要用于注入模型不确定（质量和转动惯量有关的）和故障数据；
- 电池模型⑥：模拟电池的放电过程；
- 输出接口模块⑦：将数据打包成需要的格式。

# 2.仿真平台



## ■ 多旋翼模型模块

### 2) 线性模型——仿真1.0实验

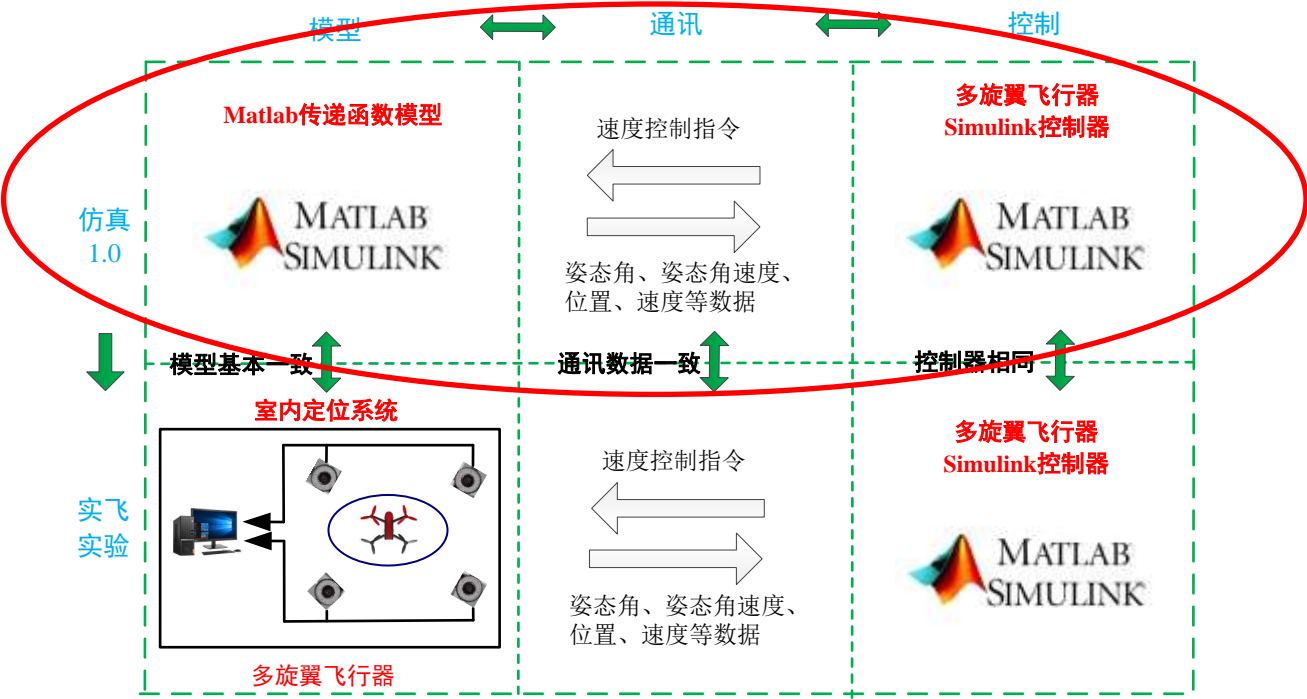


图. 实飞阶段流程

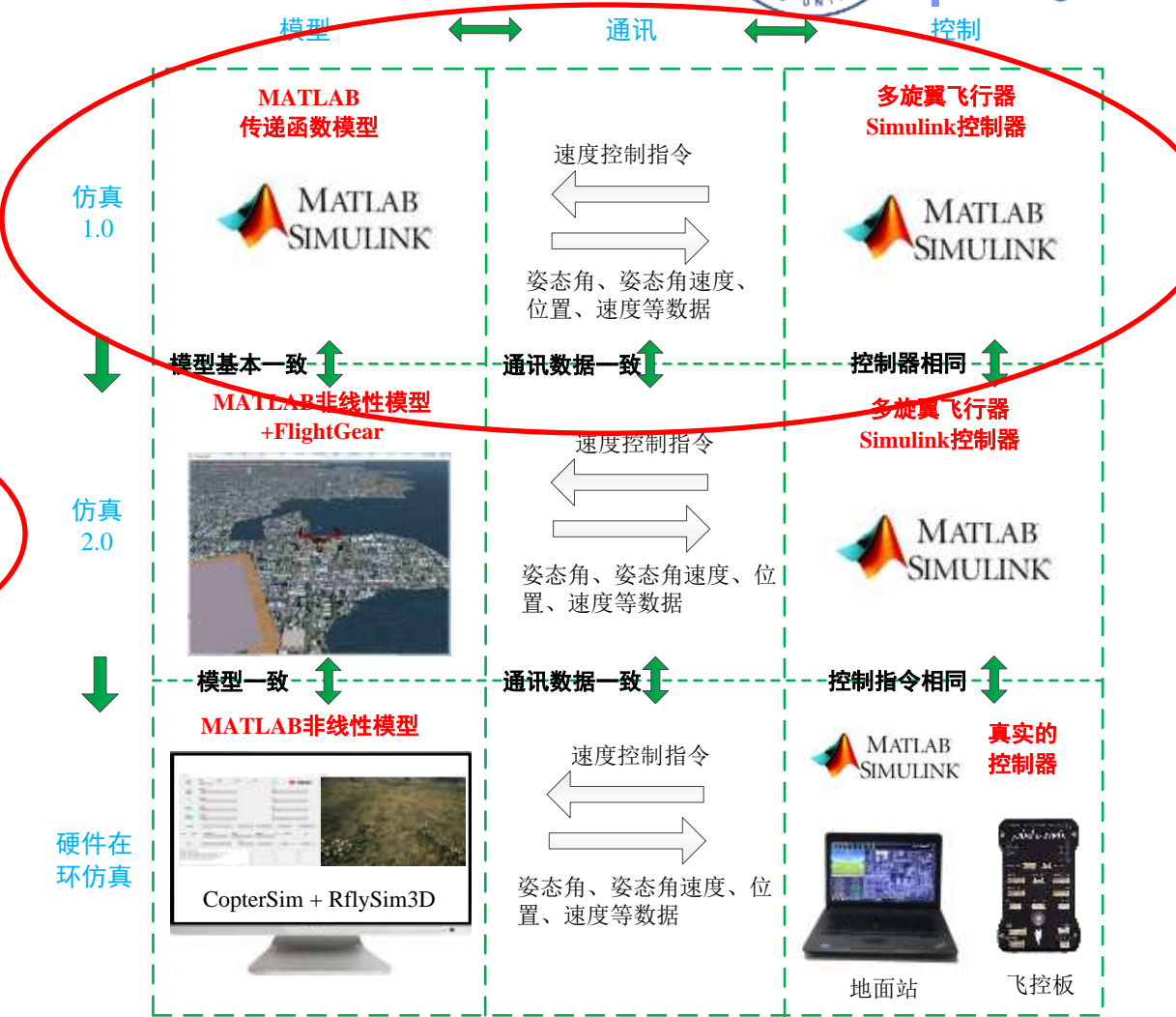


图. 仿真阶段流程

# 2.仿真平台

## ■ 多旋翼模型模块

### 2) 线性模型——仿真1.0实验

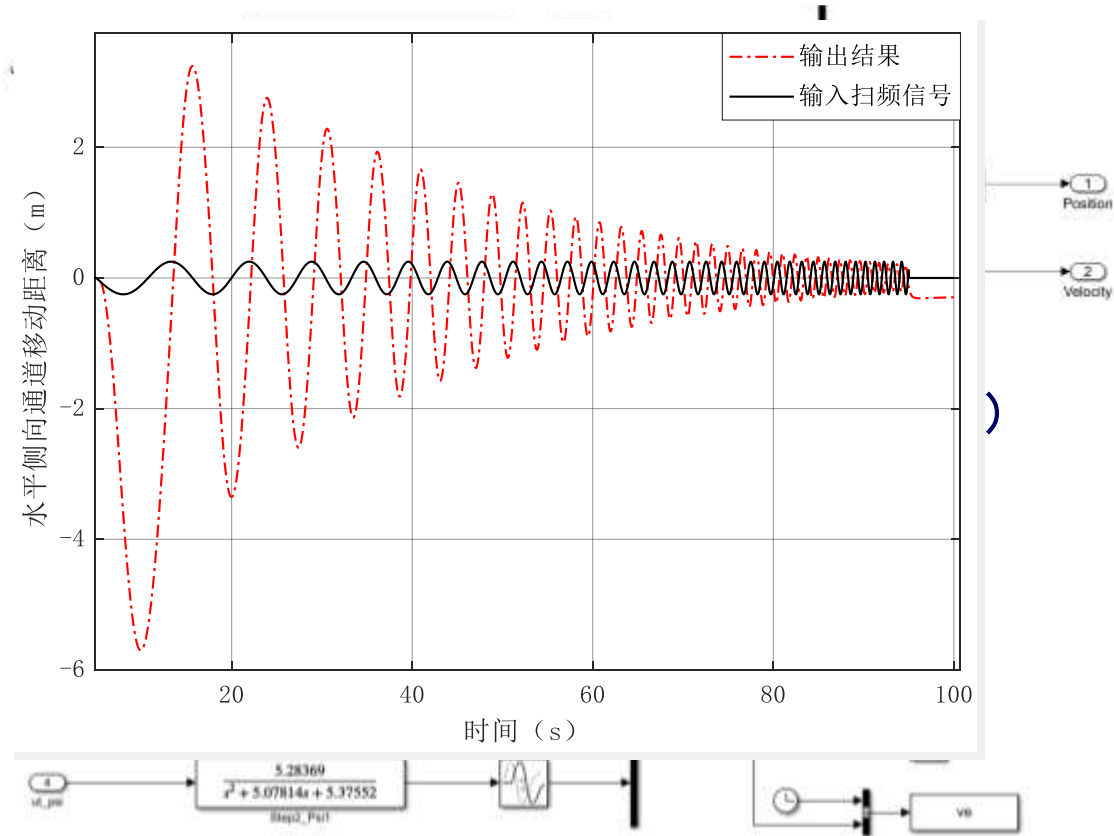


图. 实飞阶段仿真1.0（输入-位置/角度）



图. 后续课程内容设置

# 2.仿真平台

## ■ 多旋翼模型模块

### 3) 硬件在环接口模块——硬件在环仿真实验

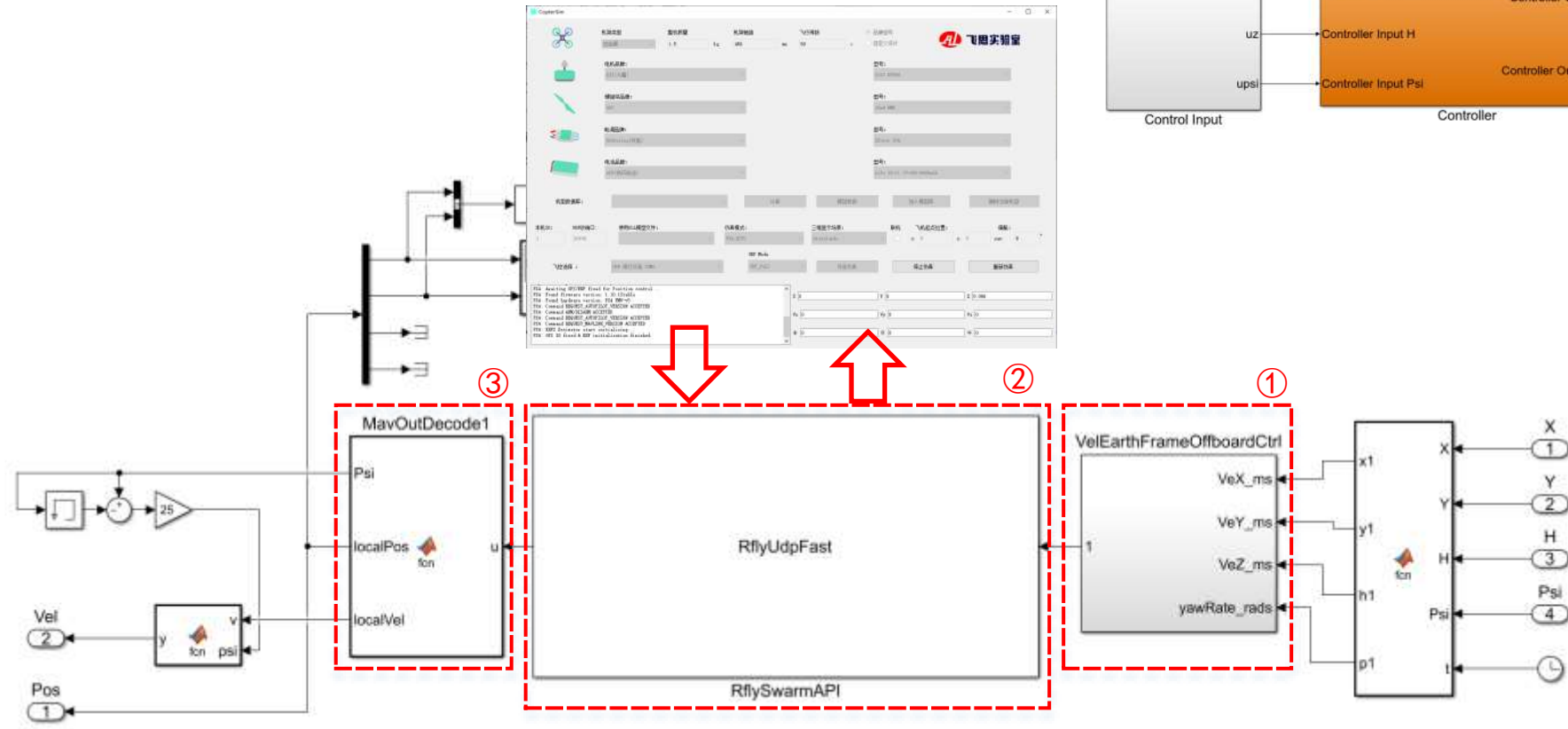


图. 硬件在环仿真接口，Simulink模型 “e0\Sim2.0\sample0.slx”



## 2.仿真平台

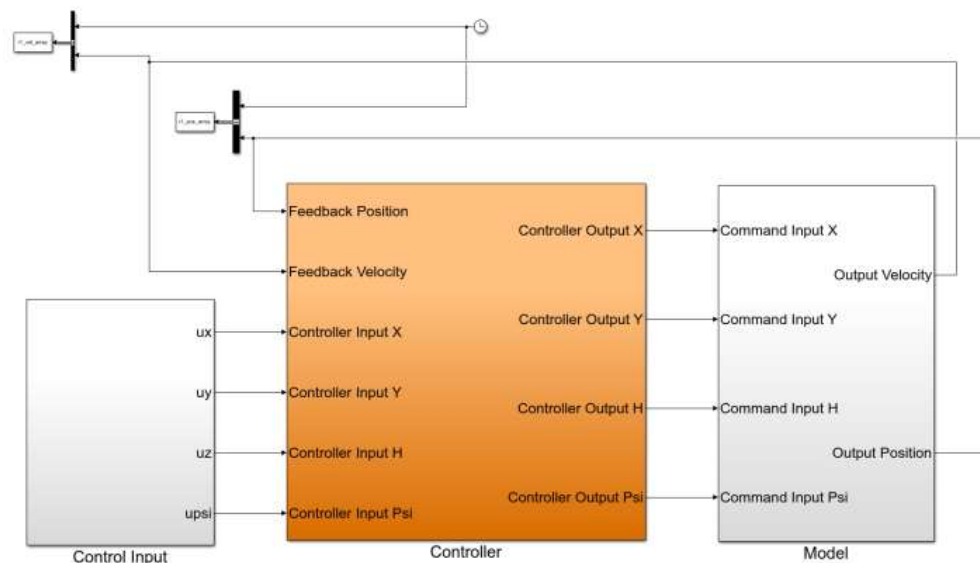


图. 单机整体模型

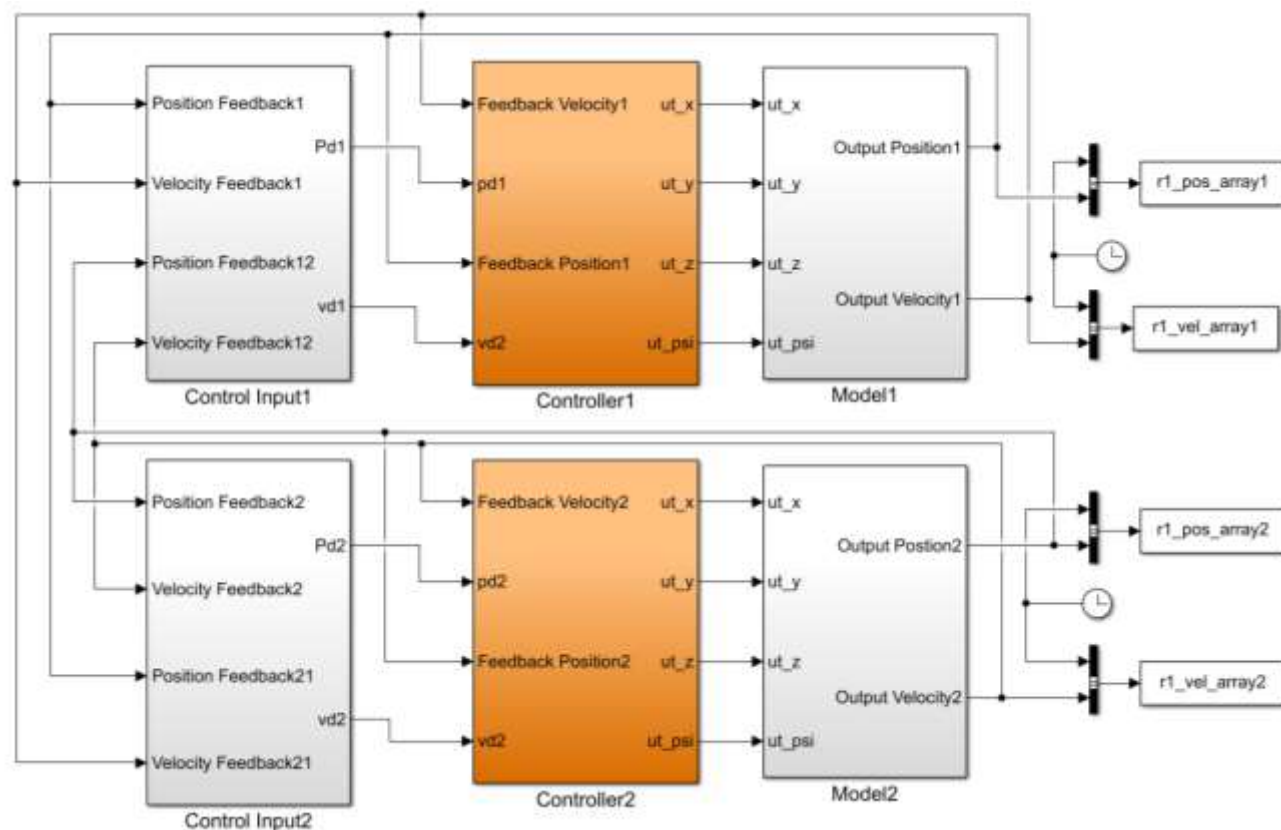
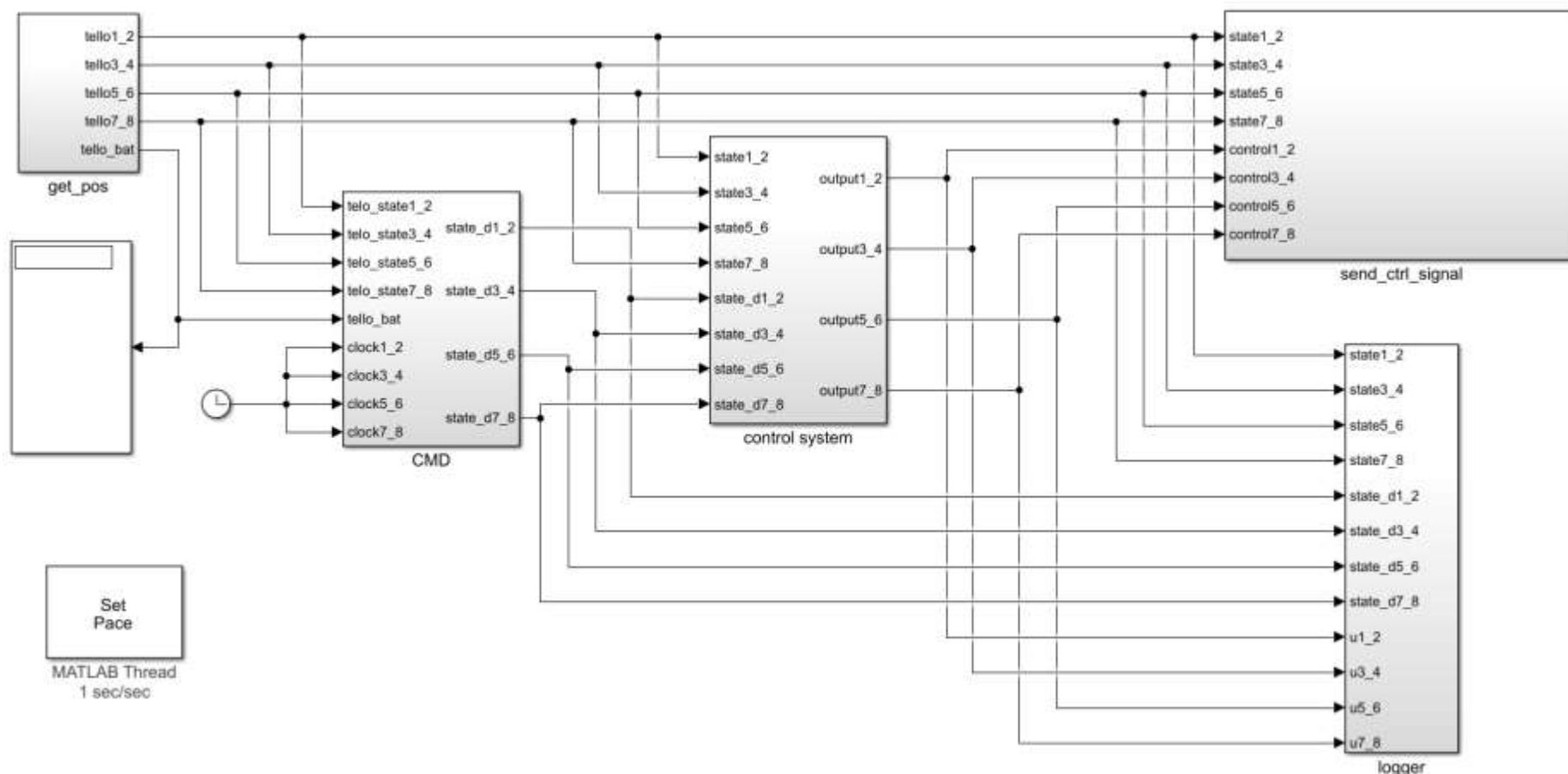


图. 多机整体模型

# 3.实飞平台

## □MATLAB控制模型



# 3.实飞平台

## □MATLAB控制模型

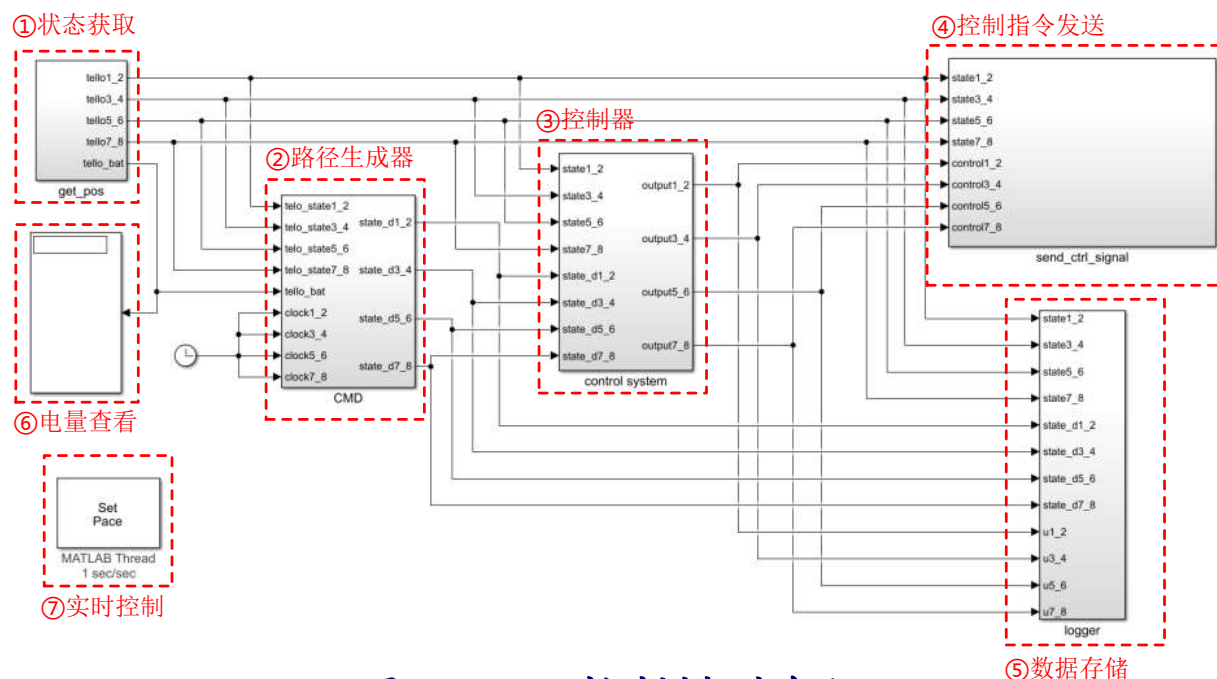


图. MATLAB控制模型介绍

- **状态获取模块:** 通过此接口模块，多旋翼可以接收室内定位系统反馈的飞行器位置和姿态信息。
- **路径生成器模块:** 此模块产生给飞行器的期望位置和偏航角，若需要执行其他的飞行任务，可以在此模块中修改期望的轨迹。
- **控制器模块:** 此控制器的输入是期望的位置，输出是期望的速度和偏航角速率。
- **控制指令发送模块:** 通过此接口模块，将控制器的输出发送给飞行器。
- **数据存储模块:** 将飞行器飞行过程中的实际状态和期望状态保存到MATLAB工作空间中，可用于分析飞行状态。
- **电量查看模块:** 可以看到每个飞行器的剩余电量百分比。
- **实时控制模块:** 此MATLAB模型的运行速度快于实时时间，加入这个实时模块保持MATLAB运行速度与实际时间一致。

# 3.实飞平台

## ■ 飞行器状态获取模块

- (1) UDP接收模块: UDP Receive
- (2) 数据解压模块: Byte Unpack
- (3) 本地IP端口号: TELL01\_REC\_PORT, 26000
- (4) 远端IP地址: UDP\_REC\_IP = '127.0.0.1'
- (5) 接收缓冲区大小和数据长度

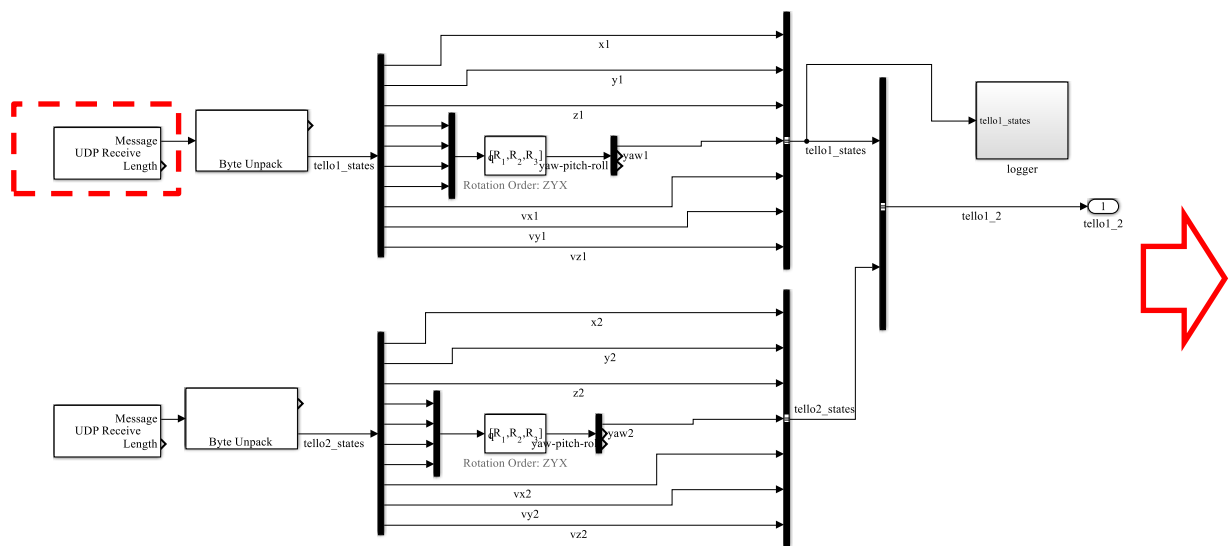


图. 飞行器状态获取模块内部

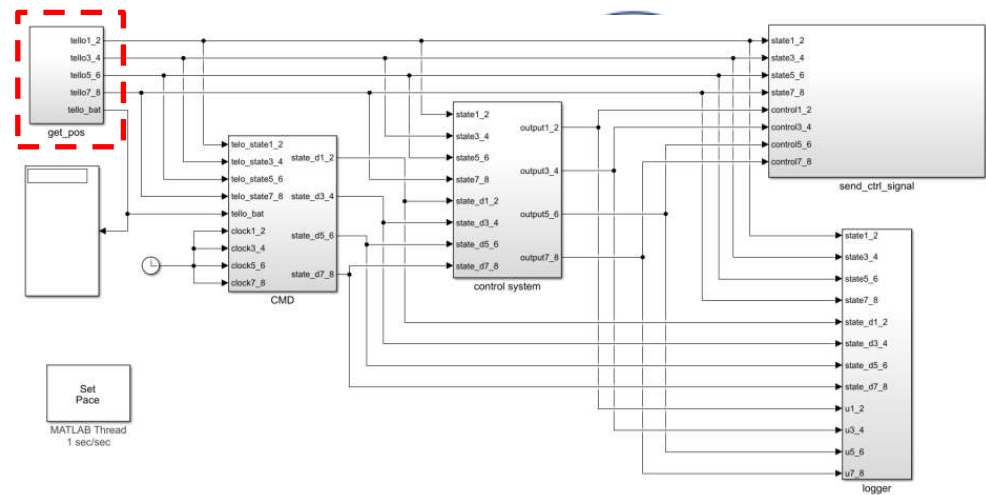


图. UDP接收模块设置



# 3.实飞平台

## ■ 控制指令发送模块

- (1) UDP发送模块：UDP Send
- (2) 数据打包模块：Byte Pack
- (3) 受控无人机IP端口号：UDP\_SEND\_IP
- (4) 受控无人机IP地址：TELL01\_SEND\_PORT

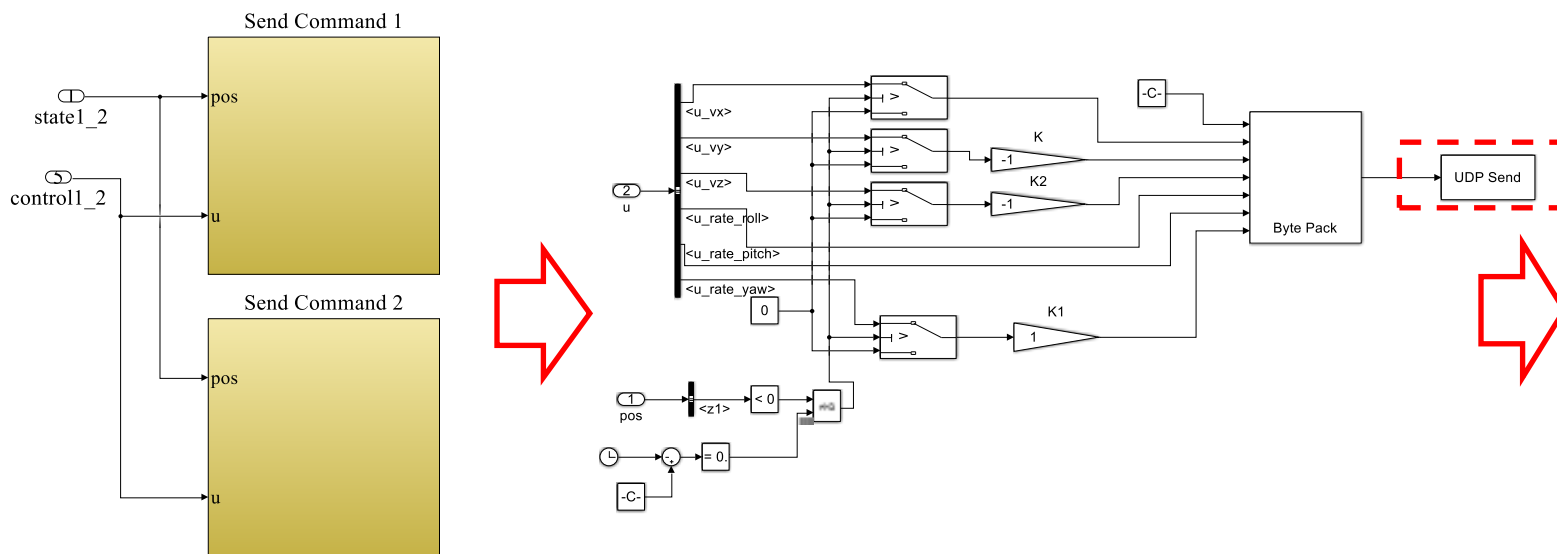


图. 控制指令发送模块内部

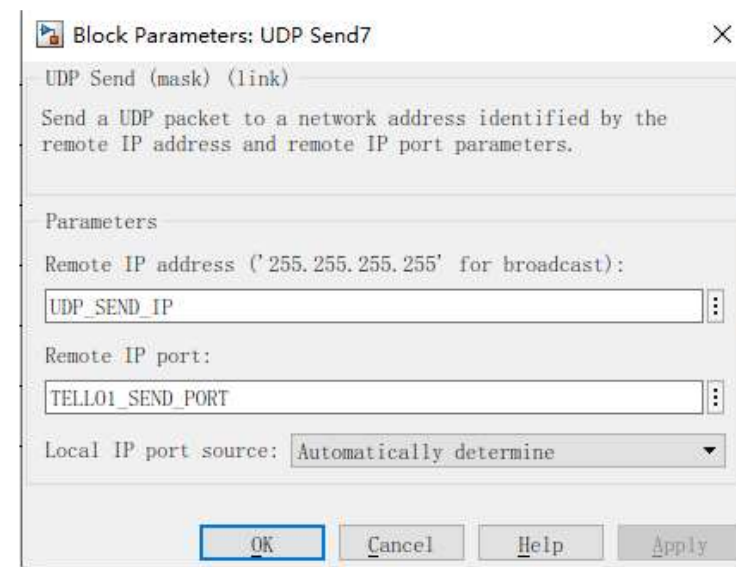
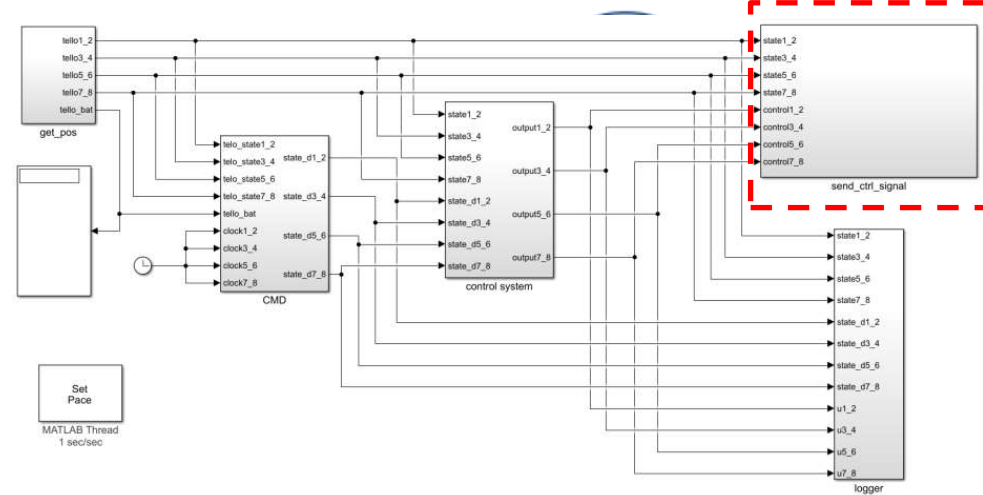


图. UDP发送模块设置

# 3.实飞平台

## ■ 增加或者减少飞行器的数量

- 1) 减少飞行器的数量不需要修改模型
- 2) 增加飞行器的数量
  - 复制对应数量模块
  - 修改IP地址和端口号

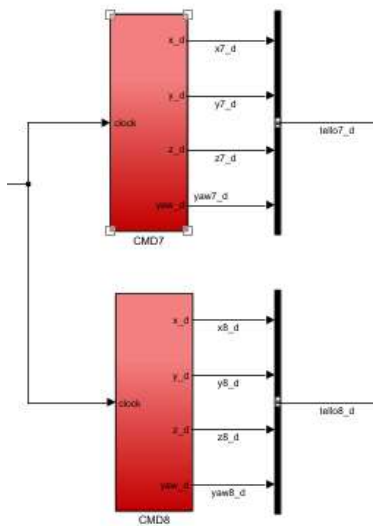


图. 期望输入

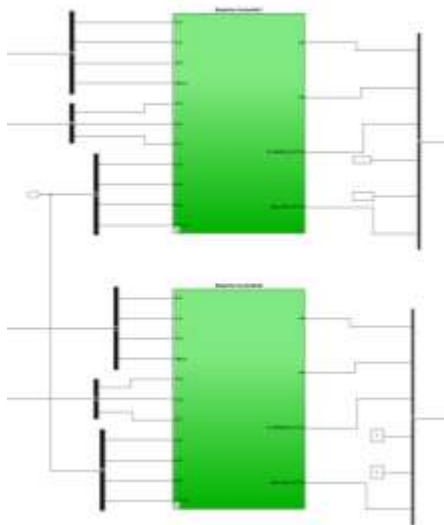


图. 控制器

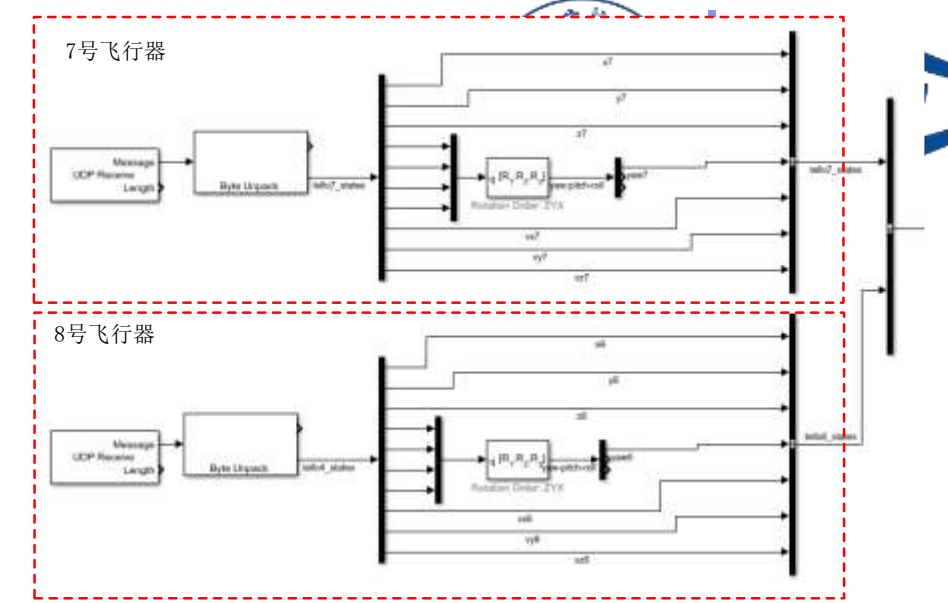


图. 接收模块

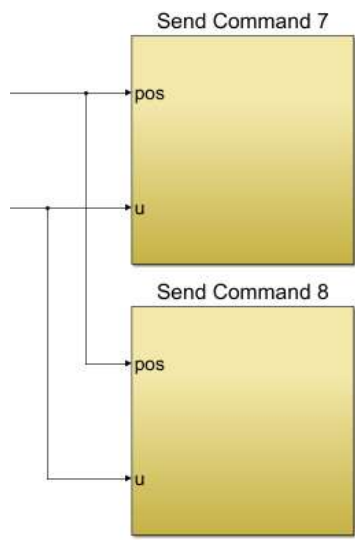


图. 发送模块

# 3.实飞平台

## □实飞环境和原理

### ■ Tello无人机和OptiTrack光学定位系统

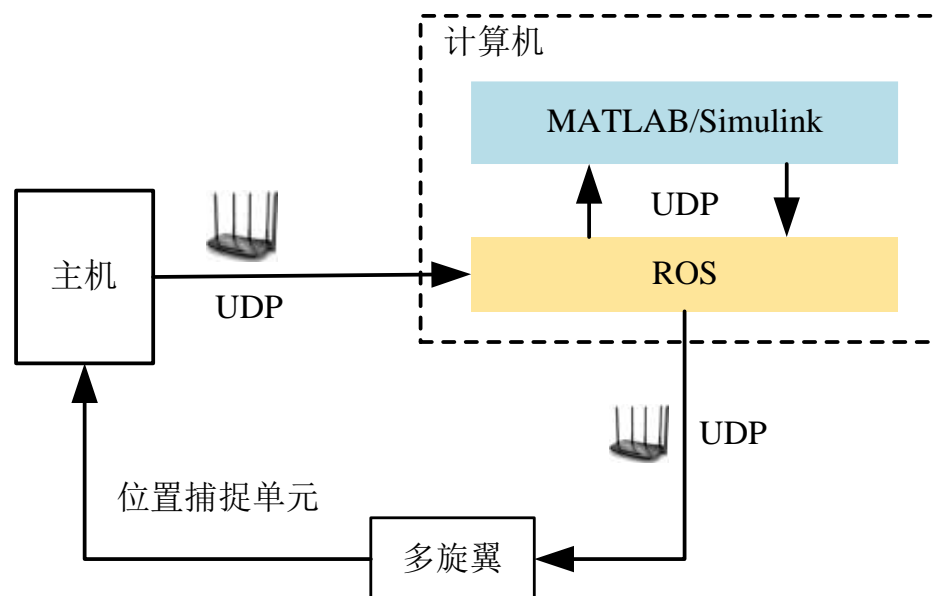


图. 实飞实验通信原理



图. 实飞环境

# 3.实飞平台

## □实飞流程



在Linux操作系统（Ubuntu16.04+ROS Kinetic）的终端中输入以下指令：

### (1) 启动 OptiTrack

```
roslaunch mocap_optitrack multi_rigidbody8.launch
```

### (2) 启动 tello\_driver

```
roslaunch tello_driver tello_node_all8.launch
```

### (3) 起飞Tello

```
roslaunch tello Tello_takeoff_all
```

### (4) 运行MATLAB/Simulink控制程序

运行“start\_tello.m”文件会自动打开“.slx”文件，点击Simulink文件的运行按钮，此时，Simulink生成的指令将会发给Tello。

### (5) 停止MATLAB/Simulink控制程序

点击Simulink文件的停止按钮。

### (6) 降落Tello

```
roslaunch tello Tello_land_all
```

### 注意：

- 在实验前可以将不同指令分别输入不同的终端中，快速实现无人机的起飞降落，防止发生未知情况；
- 在实验完成后，先结束Simulink程序再输入降落指令，防止多旋翼不受控制或者无法降落到地面。



# 4.实验流程

## □定点控制实验

实现多旋翼的定点位置控制，要求如下：

- 控制器输入：四个通道的定点指令
- 控制器输出：对多旋翼模型的直接控制指令，也就是速度和偏航角速率指令
- 期望效果：偏航角保持为0，从任意点到达（1,1,1）位置

表. 实验类型、目标和内容

目标	基础实验	分析实验	设计实验	实飞实验
熟悉开发平台	✓	✓	✓	✓
熟悉分析过程	×	✓	✓	✓
熟悉设计方法	×	×	✓	✓
仿真1.0	✓	✓	✓	✓
仿真2.0	✓	✓	✓	×
硬件在环仿真	×	×	✓	×
实飞实验	×	×	×	✓



# 4.实验流程

## ■ 仿真阶段流程-仿真1.0

### (1) 步骤一：选择多旋翼模型

Model模块采用设计模型（线性模型）

### (2) 步骤二：建立控制器子模块

Control Input模块，为每个通道设置位置控制器

### (3) 步骤三：设置期望输入

Control Input模块，设定相应的定点期望

### (4) 步骤四：开始仿真

- 运行“startSimulation.m”参数初始化文件
- 运行Simulink程序

### (5) 步骤五：分析结果

运行文件“e0\_plot.m”得到结果，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。

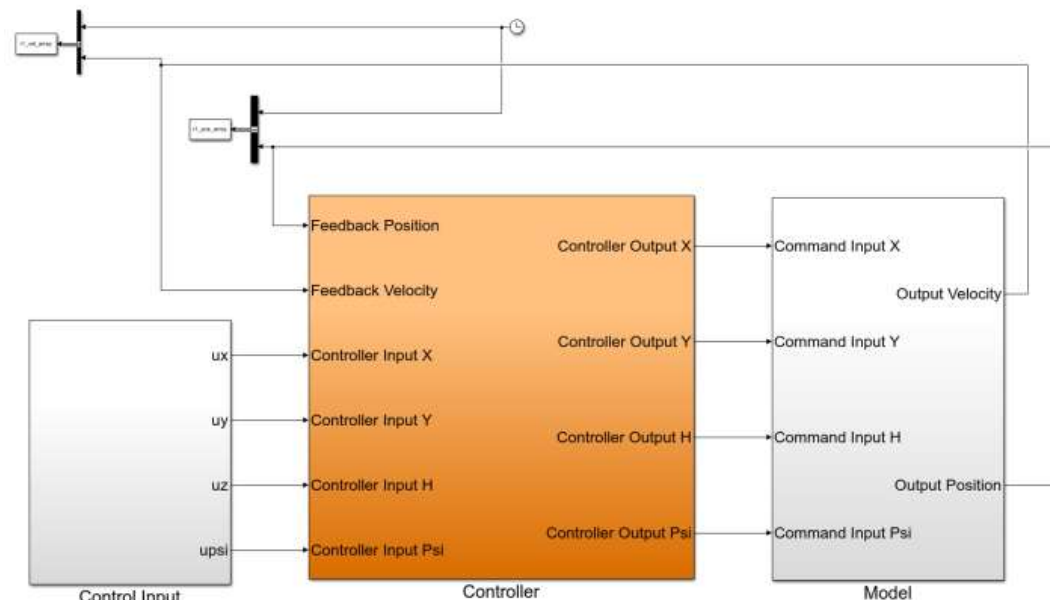


图. 仿真1.0整体模型

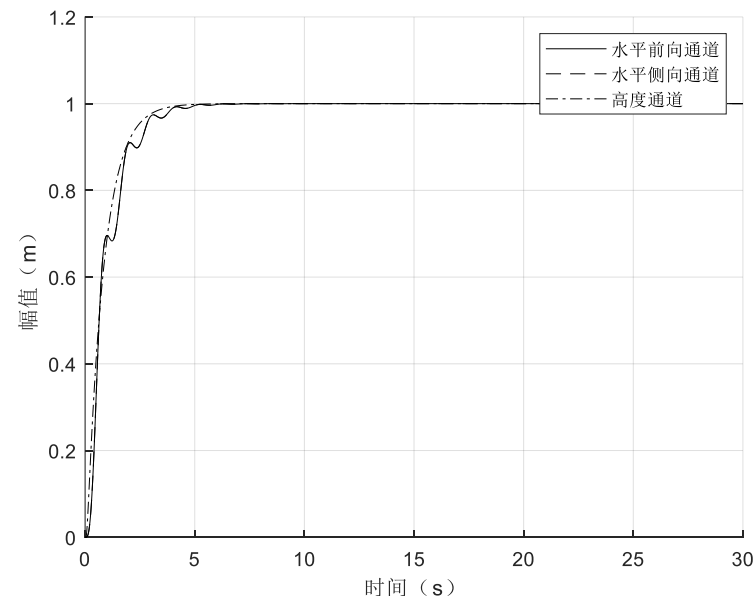


图. 仿真1.0结果

# 第三章实验流程

仿真实验：仿真1.0

# 4.实验流程

## ■ 仿真阶段流程-仿真2.0

### (1) 步骤一：选择多旋翼模型

Model模块，采用非线性系统模型

### (2) 步骤二：建立控制器子模块

Control Input模块，与仿真1.0相同

### (3) 步骤三：设置期望输入

Control Input模块，与仿真1.0相同

### (4) 步骤四：开始仿真

- 运行“FlightGear-Start.bat”打开FlightGear
- 运行“startSimulation.m”参数初始化文件
- 运行Simulink程序

### (5) 步骤五：分析结果

- 在FlightGear中观察到：多旋翼爬升一段时间后到达指定定点。
- 运行文件“e0\_plot.m”得到结果，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。

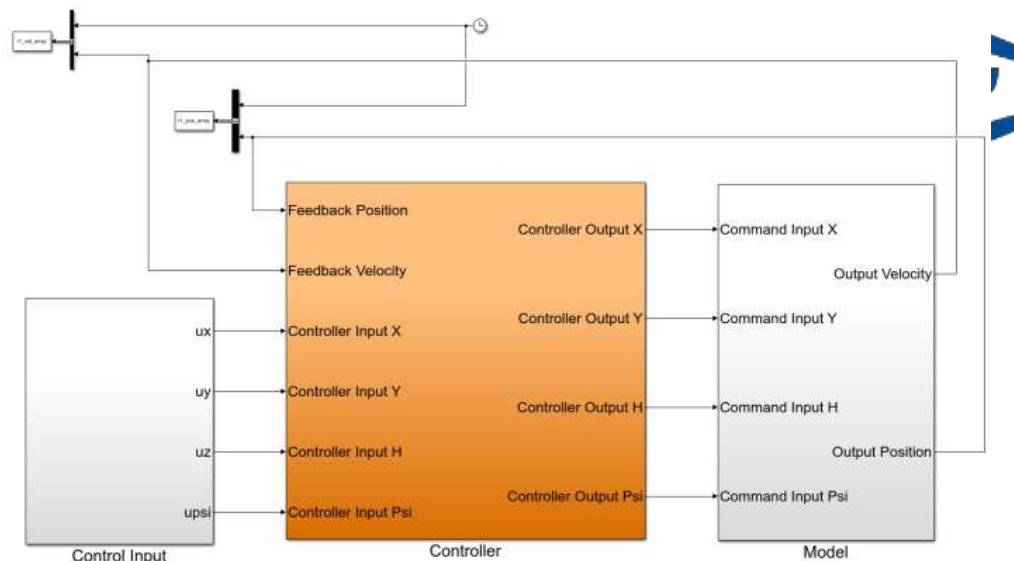


图. 仿真2.0整体模型

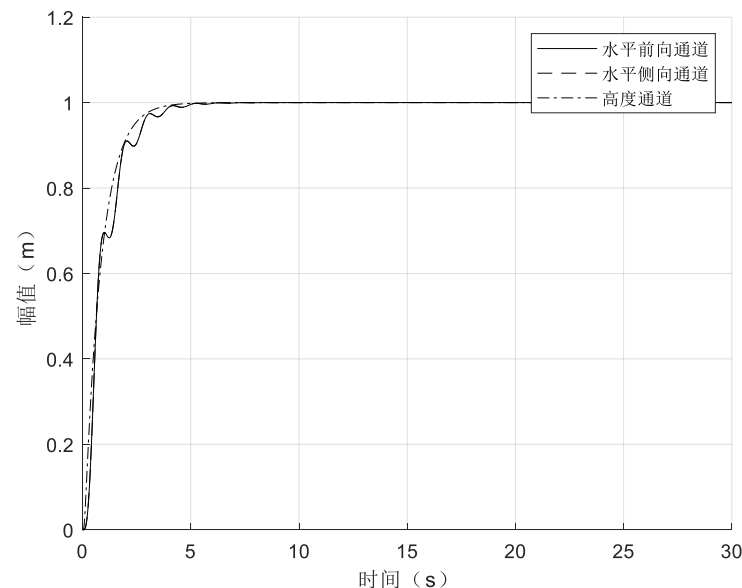


图. 仿真2.0结果



# 第三章实验流程

仿真实验：仿真2.0

# 4.实验流程



## ■ 仿真阶段流程-硬件在环仿真

### (1) 准备一：连接硬件

将自驾仪与计算机通过USB数据线连接。

### (2) 准备二：选择HIL脚本

a. 对于单机仿真，连接单个自驾仪，打开

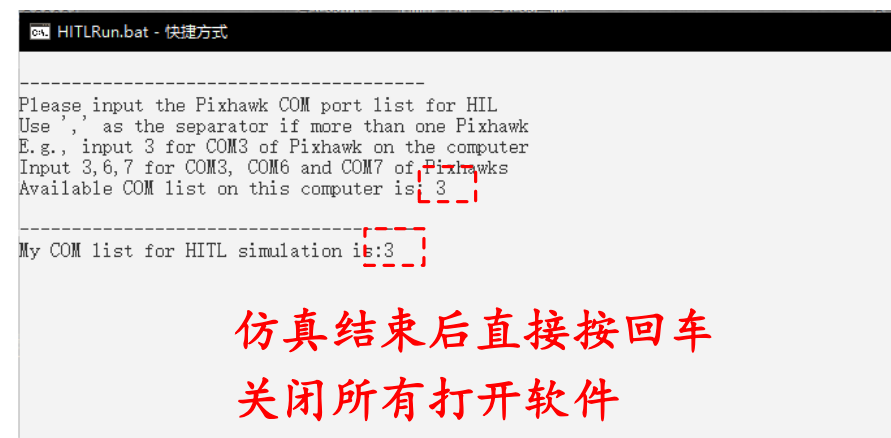
“HITLRun”脚本文件，输入显示的端口号。

b. 对于多机仿真，连接多个自驾仪连接计算机，同时按照路径“.\PX4PSP\RflySimAPIs\SimulinkSwarmAPI”找到名称为“HITLRunUdpFull.bat”的文件，双击打开，输入所有给出的串口号，每个串口号用逗号分隔开。



自驾仪：Pixhawk

固件版本：1.10.1。



仿真结束后直接按回车  
关闭所有打开软件

图. 输入串口号



图. 单机仿真脚本文件

需要更多资料，请访问<https://rflysim.com>或  
<http://doc.rflysim.com/>中的“集群控制”开发部分。

# 4.实验流程

## ■ 仿真阶段流程-硬件在环仿真

### (1) 步骤一：选择多旋翼模型

Model模块，采用**硬件在环仿真接口模块**

### (2) 步骤二：建立控制器子模块

Control Input模块，与仿真1.0相同

### (3) 步骤三：设置期望输入

Control Input模块，与仿真1.0相同

### (4) 步骤四：开始仿真

- 运行**硬件在环仿真脚本文件**打开仿真软件
- 运行“startSimulation.m”参数初始化文件
- 运行Simulink程序开始仿真

### (5) 步骤五：分析结果

- 在**RflySim3D**中观察到：多旋翼爬升一段时间后到达指定位置
- 运行文件“e0\_plot.m”得到结果

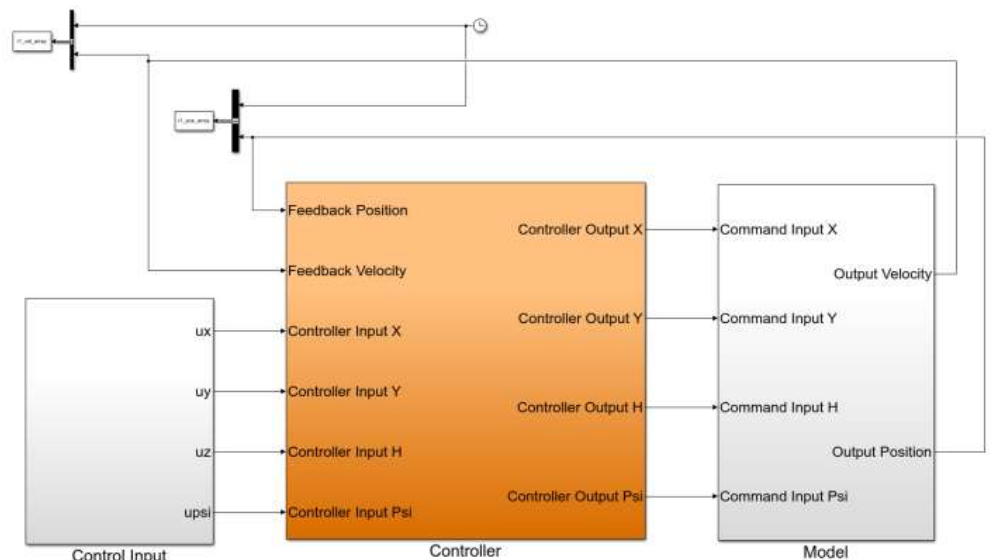


图. 硬件在环仿真整体模型

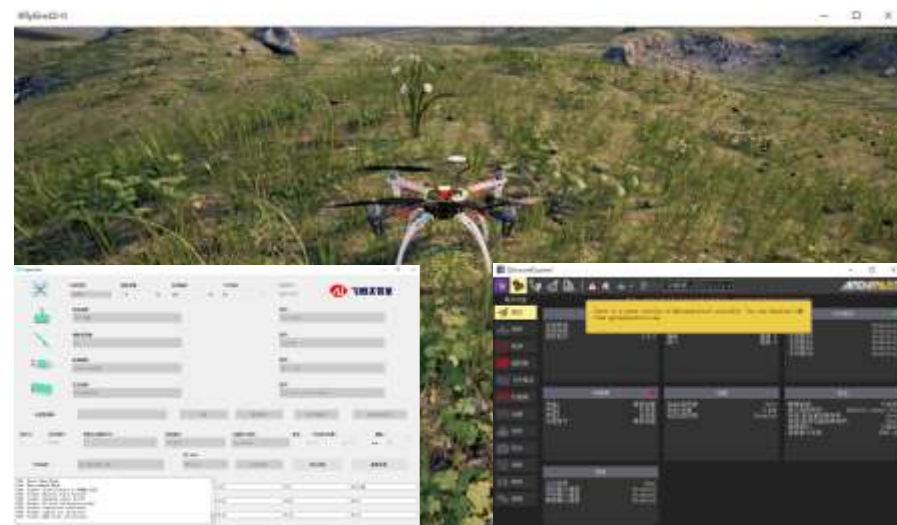


图. 仿真效果界面





# 4.实验流程

## ■ 实飞阶段流程-仿真1.0

### (1) 步骤一：选择多旋翼模型

Model模块采用设计模型（线性模型）

### (2) 步骤二：建立控制器子模块

Control Input模块，为每个通道设置位置控制器

### (3) 步骤三：设置期望输入

Control Input模块，与仿真阶段仿真1.0相同

### (4) 步骤四：开始仿真

- 运行“startSimulation.m”参数初始化文件
- 并运行Simulink程序

### (5) 步骤五：分析结果

运行文件“e0\_plot.m”得到结果

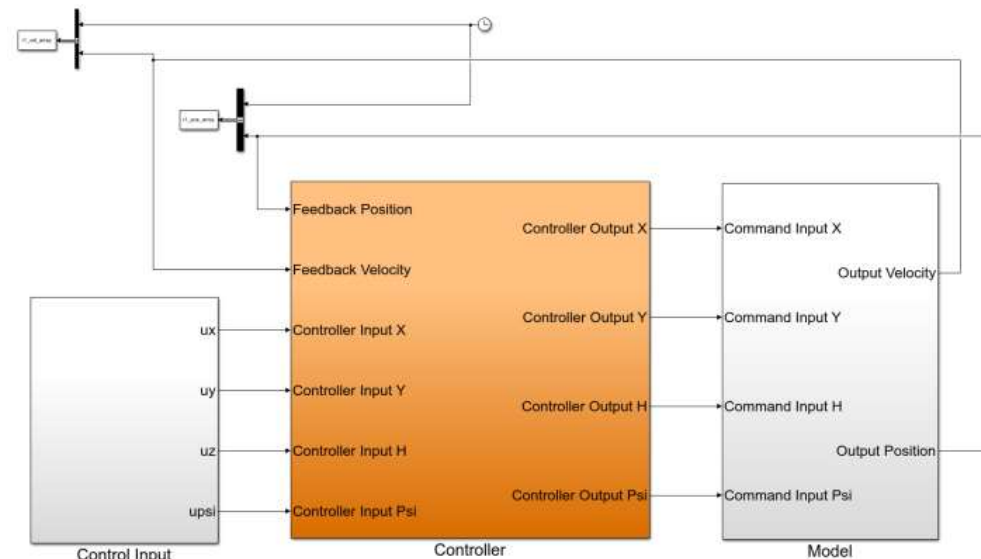


图. 实飞实验仿真1.0整体模型

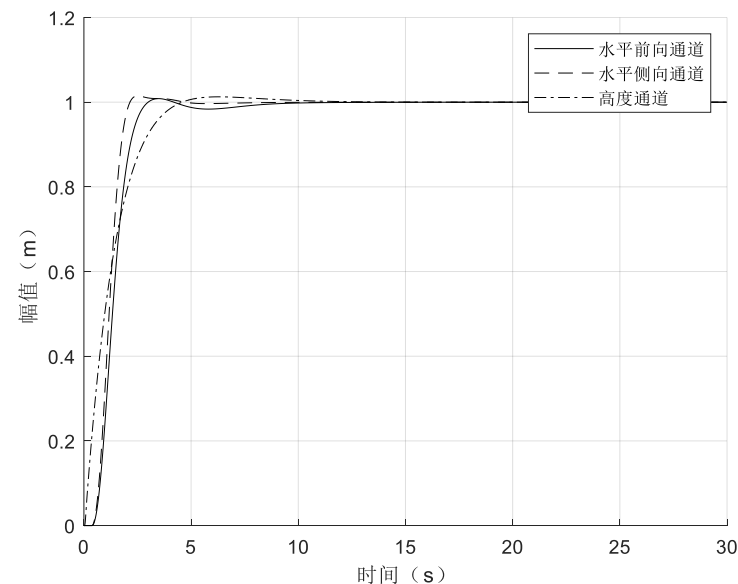


图. 仿真1.0结果

# 第三章实验流程

实飞实验：仿真1.0

# 4.实验流程

## ■ 实飞阶段流程-实飞实验

(1) 步骤一：建立控制器子模块

Control system-Baseline Controller Basic模块

(2) 步骤二：设置期望输入

CMD-CMD1子模块，与仿真1.0相同

(3) 步骤三：实飞准备

启动设备：路由器、室内定位系统、工作站、多旋翼...

(4) 步骤四：开始仿真

运行“start\_tello.m”参数初始化文件,运行Simulink程序开始实飞

(5) 步骤五：结束实验并分析结果

观察到多旋翼到达指定位置,运行文件“e0\_plot.m”得到结果

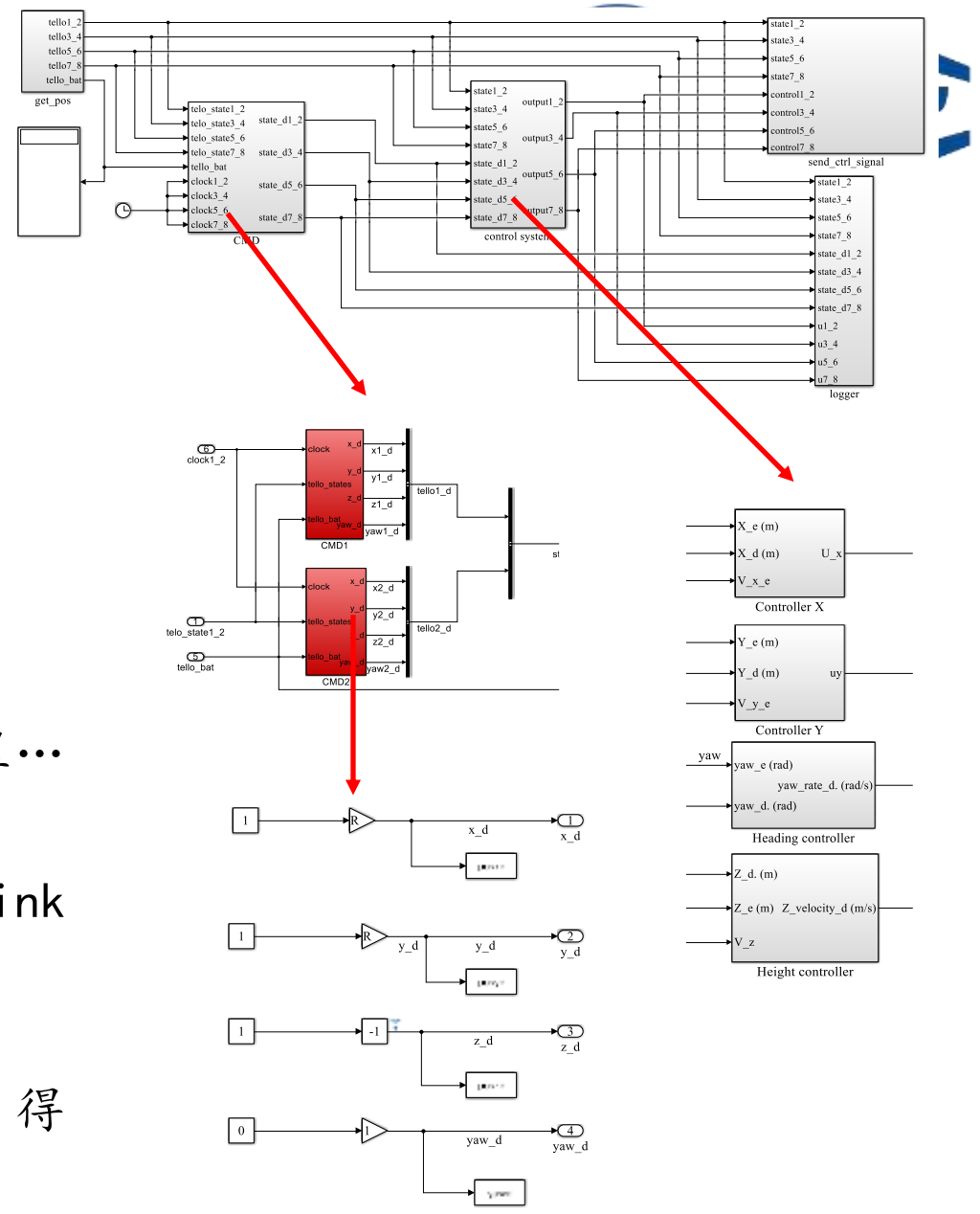


图. 设置期望输入和控制器

# 实飞环境及操作流程介绍

- 室内定位系统: Optitrack
- 主机: Win10系统、运行Motive
- 工作站: Ubuntu系统、运行MATLAB R2018b
- 多旋翼无人机: Tello无人机



# 5.平台优势



**(1) 仿真阶段：**熟悉整个开发过程，减少实验时间以及过程中带来的风险和不确定

**(2) 实飞阶段：**在控制器设计完成后，可以通过仿真1.0阶段测试，之后使用设计的控制代码直接控制真实的飞行器，来验证及评估所设计的控制算法的性能

**(3) 平台接口：**读者（初学者、学生或者工程师）可以利用自己所学知识快速进行控制器设计与验证

**(4) 完整代码：**所有案例均有代码和注释，并通过反复验证

**(5) 指导书籍：**所有原理能够自包含，所有的实验都具有详细的步骤

**(6) 教学视频：**所有的仿真和实飞实验都录制了操作视频

# 致谢

感谢可靠飞行控制研究组同学  
为本节课程准备做出的贡献



杨兰江



# 谢谢！

全 权

qq\_buaa@buaa.edu.cn

可靠飞行控制研究组



<http://rfly.buaa.edu.cn>