



固高科技
GOOGOLTECH

PARADOX 派动智能
Smart Your Machine

四旋翼飞行仿真器 实验指导书

四旋翼系列
GHP3001-II

V1.00



www.googoltech.com.cn

© 2022 Googol Technology. All rights reserved

版权声明

固高科技有限公司版权所有. 固高科技保留所有
版权以及相关的知识产权。

GHP3001-II 型四旋翼飞行仿真器实验教程

在版权法保护下，在没有固高科技的书面许可下，任何人都不能直接的或是间接的复制、生产、加工 本产品以及附属产品。

声明

固高科技保留在没有预先通知的情况下修改产品或其特性的权利。固高科技并不承担由于使用产品不当而产生的直接或是间接的伤害或损坏的责任。

商标

[Windows](#) 和 [Microsoft](#) 是 Microsoft 的注册商。

[IPM](#) 和 [IPM Motion Studio](#) 是 Technosoft 公司的商标。

[MATLAB](#) 是 MathWorks 公司的商标。

[LabVIEW](#) 是美国国家仪器（NI）公司的商标。

联系我们

固高科技（香港）有限公司

香港九龙观塘伟业街 108 号丝宝国际大厦 10 楼
1008-09 室

Tel.: (852) 2358 1033, (852) 2719 8310

Fax: (852) 2719 8399

<http://www.googoltech.com>

固高科技（深圳）有限公司

深圳市南山区高新科技园深港产学研基地西二楼 Tel: (86) 755 2697 0817; (86) 755 2697 0835
Fax: (86) 755 2697 0846

<http://www.googoltech.com.cn>

固高派动（东莞）智能科技有限公司

广东省东莞市松山湖工业东路 6 号固高科技园 A 栋 9 楼
9F,A#Buldg.No.6, Industry East Road,Songshan
Lake,Dongguan,Guangdong,China （Googol Park）

Tel: 0769-38809218

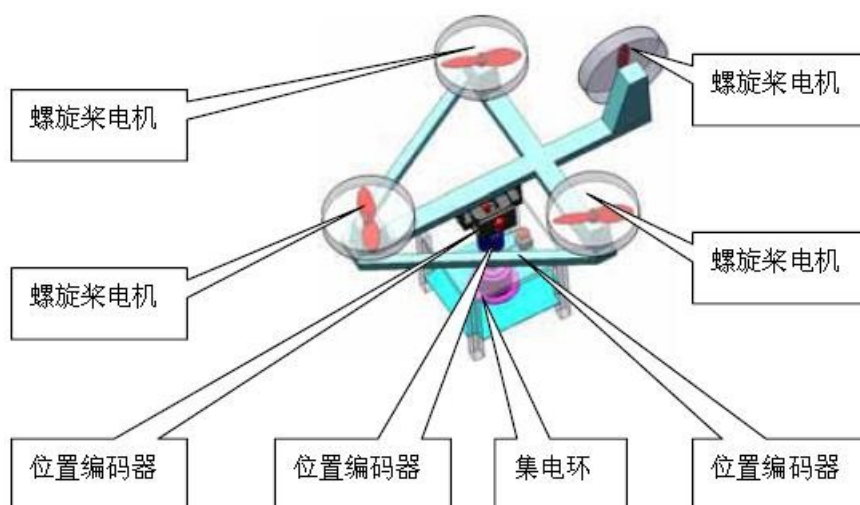
Fax: (86) 769-38853560-8106

<https://www.paradoxtech.cn/>

前言

四旋翼飞行仿真器是由安装在万向节轴杆上的四个带螺旋桨的电机提供动力。前、左、右三个电机驱动其相应的螺旋桨来实现直升机本体的俯仰运动；左、右两个电机驱动相应的螺旋桨来实现直升机本体的滚动运动；后电机驱动其相应的螺旋桨来实现直升机本体的偏航运动。

当构成闭环系统时，通过安装在平台上的三个编码器检测飞行仿真器机体的俯仰、滚动、偏航三种姿态角，反馈入控制器计算四个电机的控制量——电机电压，驱动相应电机运动，带动螺旋桨产生升、降力，以此来实现直升机的俯仰、滚动和偏航三种姿态的精确定位。系统示意图如下图示：



飞行仿真器的底座上装有集电滑环，因此机身可以绕底座任意角度的旋转而不会使连线纠缠在一起，同时也减少了摩擦。

飞行仿真器是四输入三输出的系统。四输入为：前、后、左、右三个电机的电压；三输出为：俯仰角、滚动角、偏航角。所以可以选用现代控制理论的状态反馈或者LQR线性二次型最优控制理论设计调节器。本实验指导书采用LQR线性二次型最优控制法，通过本实验，用户可以学习LQR调节器的设计和实现方法；观察选择不同状态量时对系统性能的影响情况；结合性能指标仿真和测试，分析影响系统性能指标提高的因素。

目 录

第一章 系统硬件简介	1
1.1 系统简介	1
1.2 注意事项	4
第二章 Matlab 环境配置及工具箱简介	5
2.1 Matlab 编译器及实时内核安装	5
2.2 配置 Simulink 实时控制运行环境	6
2.3 工具箱接口说明	7
2.4 Real Time Control 模块说明	12
第三章 机理法建立四旋翼飞行仿真器的数学模型	14
第四章 系统硬件检测	20
第五章 PID 控制器设计	25
第六章 四旋翼仿真系统 LQR 控制	31
6.1 四旋翼 LQR 六状态控制	31
6.2 四旋翼 LQR 九状态控制	37

第一章 系统硬件简介

1.1 系统简介

1. 系统组成

四旋翼飞行仿真系统平台主要由三部分组成：运动控制卡、手持急停及四旋翼本体。系统硬件结构运动控制卡如图 1.1 所示、手持急停如图 1.2 所示、四旋翼本体如图 1.3 所示。



图 1.1 运动控制卡



图 1.2 手持急停



图 1.3 四旋翼本体

2. 硬件接口简介

1) 运动控制卡接口简介

其接口为 68Pin 母头，连接外部扩展端子板

运动控制卡



2) 手持急停接口简介

主要用于系统上电，及紧急情况急停



3) 四旋翼本体接口简介

共三个接口：220V 电源接口、连接手持急停接口及端子板接口



3. 系统上下电步骤

1) 上电步骤

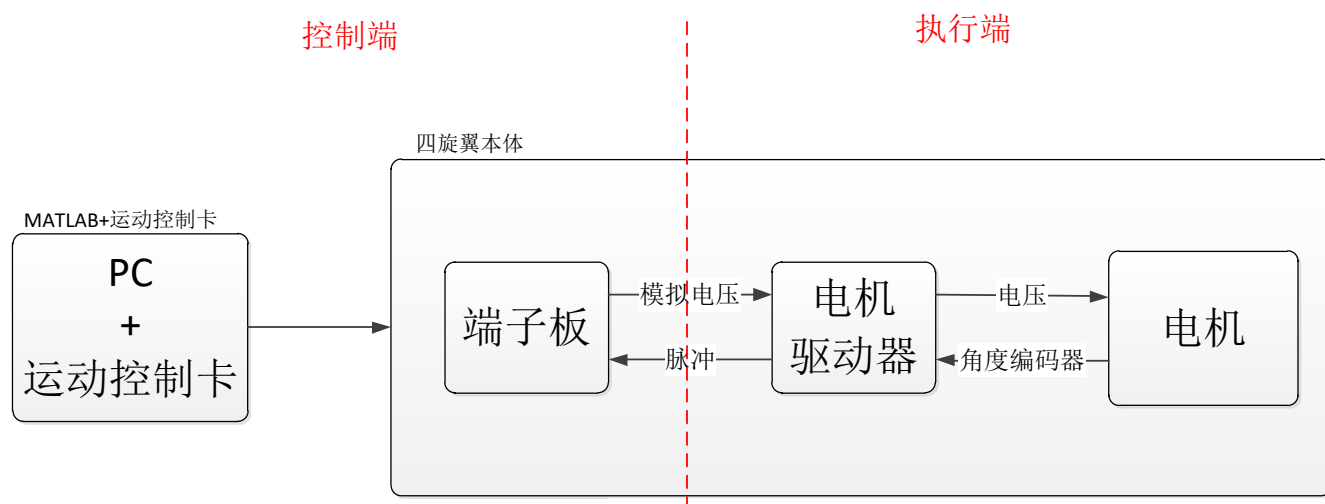
- a) 220V 电源线连接至四旋翼本体;
- b) 打开空气开关;
- c) 松开手持急停, 摁下手持急停上的 Power 键, 上电完成;

2) 下电步骤

- a) 程序停止运行后, 摁下手持急停的 Power 键, 然后摁下急停键 (如发生紧急情况, 立刻摁下急停按钮);
- b) 关闭空气开关;
- c) 断开 220V 电源连线;

4. 系统架构

1) 系统原理框图



2) 系统运行原理

通过 PC 机上的 MATLAB 软件平台让运动控制卡发送给驱动器模拟电压信号，然后驱动器驱动电机旋转，旋转产生向上或者向下的力，通过三个角度编码器检测并反馈四旋翼系统的俯仰角度、滚动角度及偏航角度，通过设计控制器，使四旋翼平衡在某个角度

1.2 注意事项

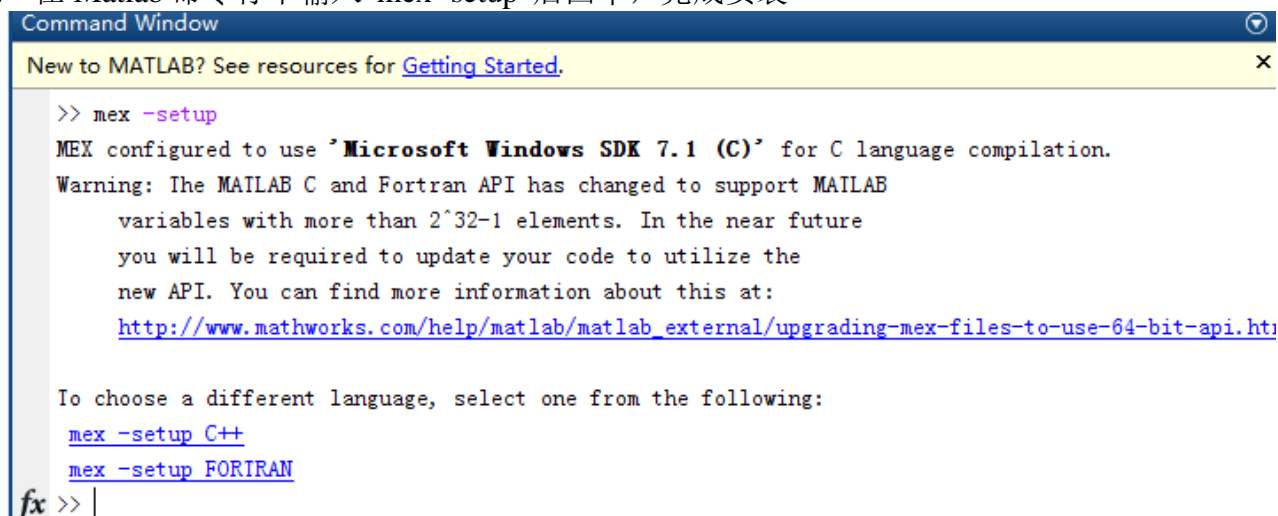
1. 所加的干扰不能太大；
2. 倾斜的时候请不要再加扰动；
3. 正、反桨升力不同，所以抗干扰能力也有所不同；
4. 调节配重块，在相同参数下，静平衡时性能最好。
5. 螺旋桨较软，不要频繁切换电机方向；
6. 系统快速性不能太快，否则会变形碰触外罩。

第二章 Matlab 环境配置及工具箱简介

2.1 Matlab 编译器及实时内核安装

1. 编译器安装

- 1) 在 Matlab 命令行中输入“mex -setup”后回车，完成安装

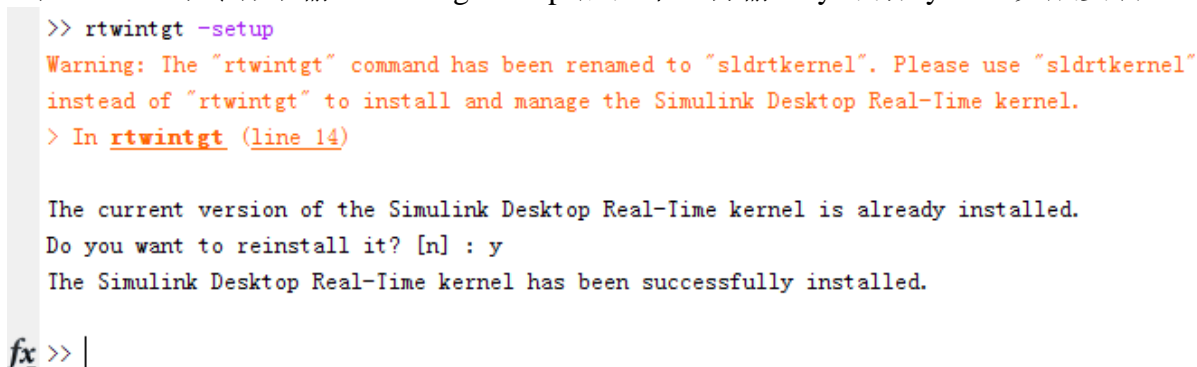


```
Command Window
New to MATLAB? See resources for Getting Started.
>> mex -setup
MEX configured to use 'Microsoft Windows SDK 7.1 (C)' for C language compilation.
Warning: The MATLAB C and Fortran API has changed to support MATLAB
variables with more than 2^32-1 elements. In the near future
you will be required to update your code to utilize the
new API. You can find more information about this at:
http://www.mathworks.com/help/matlab/matlab\_external/upgrading-mex-files-to-use-64-bit-api.html

To choose a different language, select one from the following:
mex -setup C++
mex -setup FORTRAN
fx >> |
```

2. 实时内核安装

- 1) 在 Matlab 命令行中输入“rtwintgt -setup”后回车，再输入“y”或者“yes”，完成安装

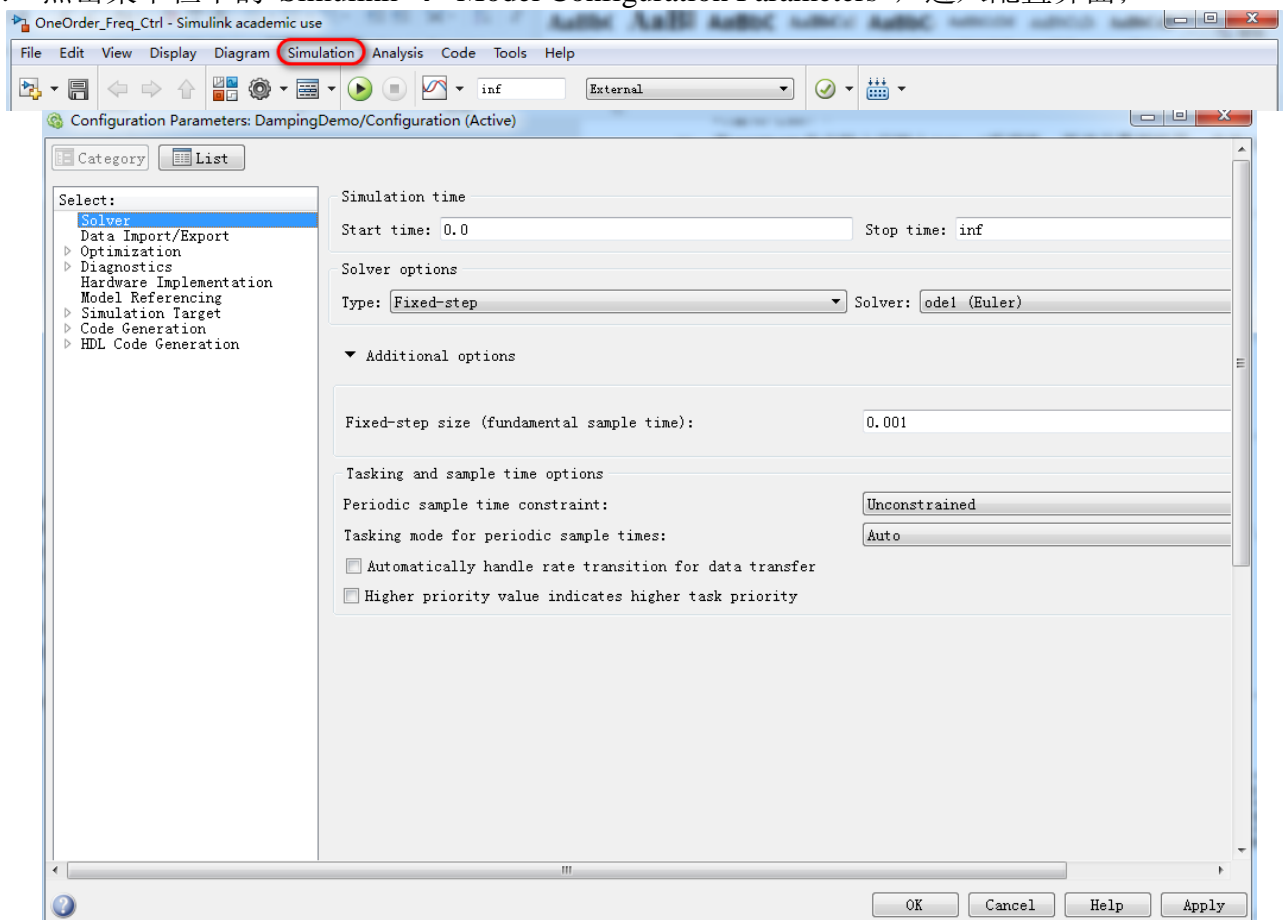


```
>> rtwintgt -setup
Warning: The "rtwintgt" command has been renamed to "sldrtkernel". Please use "sldrtkernel"
instead of "rtwintgt" to install and manage the Simulink Desktop Real-Time kernel.
> In rtwintgt (line 14)

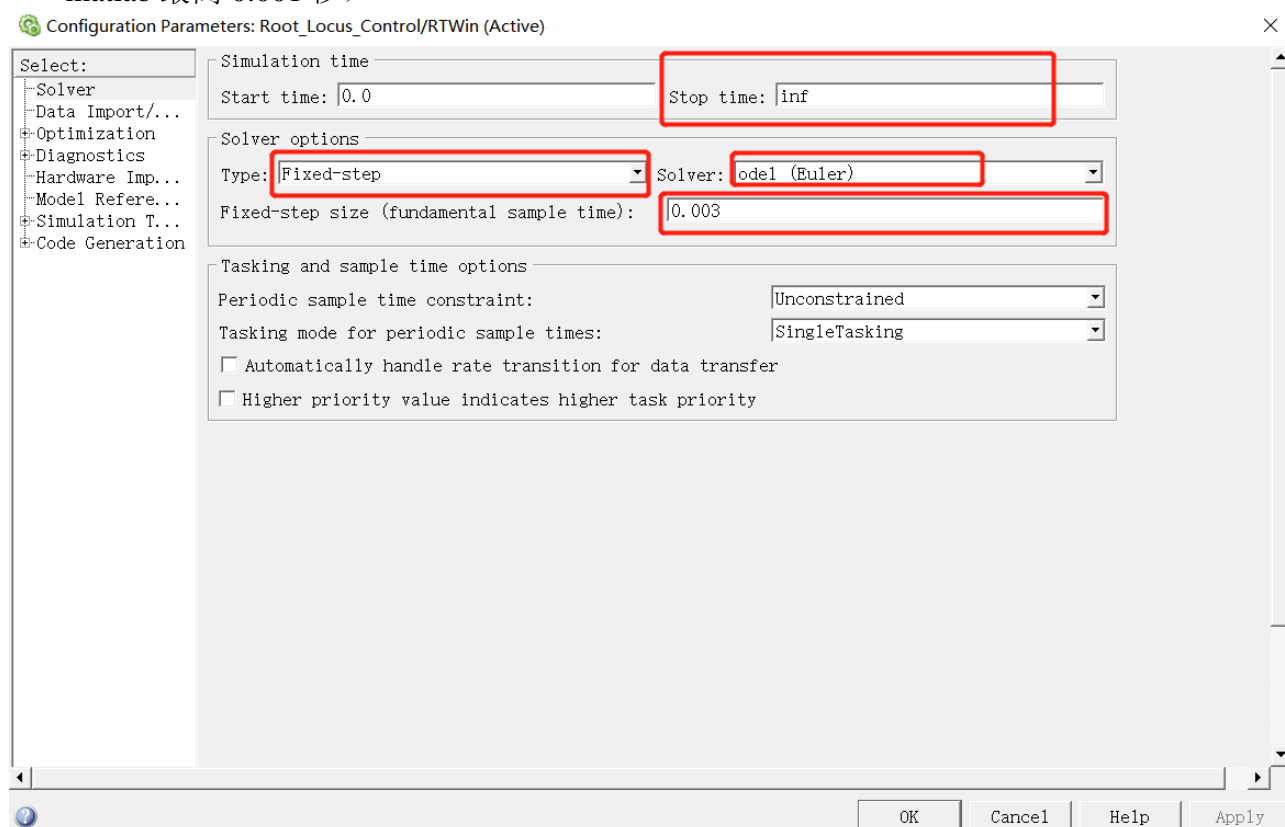
The current version of the Simulink Desktop Real-Time kernel is already installed.
Do you want to reinstall it? [n] : y
The Simulink Desktop Real-Time kernel has been successfully installed.
fx >> |
```

2.2 配置 Simulink 实时控制运行环境

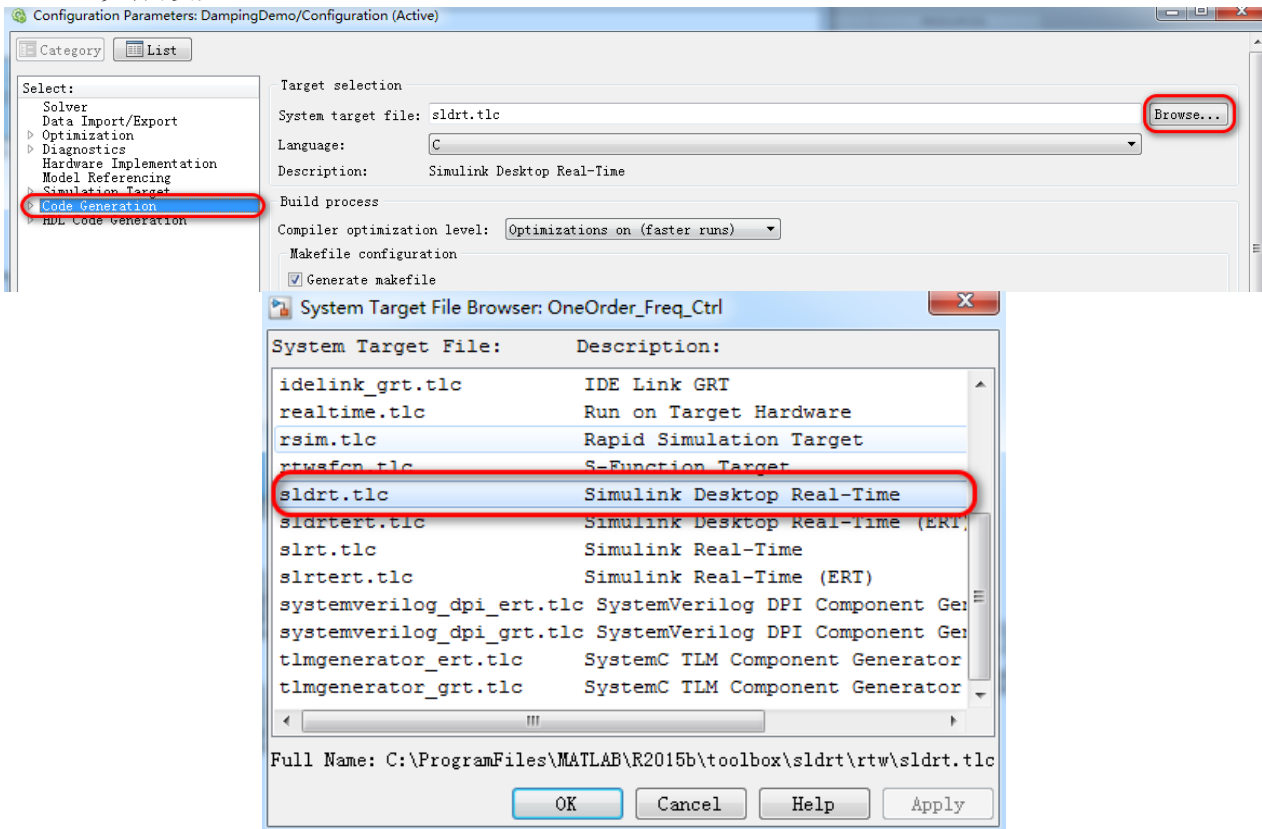
1. 点击菜单栏中的“Simulink”→“Model Configuration Parameters”，进入配置界面；



2. 点击左侧菜单“Solver”，设置“Start time”为“0”、“Stop time”为“inf”，选择“Solver options”下“Type”为“Fixed-step”，然后点击“Additional options”展开，设置“Fixed-step size”为“0.001”（这块设定了程序运行从 0 秒开始无限时间，手动停止程序，配置了固定采样时间 0.003 秒，matlab 最高 0.001 秒）



3. 点击左侧菜单栏“Code Generation”，点击“Browse...”,选择“sldrt.tlc”（配置实时内核），点击“OK”完成设置



2.3 工具箱接口说明

1. Matlab 中安装固高运动控制卡（GT/GTS）工具箱

- 1) 从MATLAB 找到安装文件路径



- 2) 在命令行中输入“gtmtb -setup”后回车。



- 3) 输入“yes”后回车或者直接回车进行安装，并等待安装完成。

```

命令窗口
>> gtmdb -setup

<(!)Thank you for using products from Googoltech.<(!)

Copyright 2016-gogogo The Paradox Intelligent Technology Co., Ltd. All Rights Reserved.

Googoltech matlab toolbox requires administrator-level permission for processing.
The toolbox requires the following package for executing:
1. "Real-Time Windows Target" or "Simulink Desktop Real-Time" toolbox, and
2. "mex" functionality
Please check your matlab installation for the details.

Attention: No Googoltech matlab toolbox was installed currently.
Still continue to install the Googoltech matlab toolbox (ver 1.32.1.20161123)? [yes] :
-gt>>installing Googoltech matlab toolbox ver 1.32.1.20161123, please wait...
-gt>>installing, please wait...
-gt>>a backup of old user data was stored in "C:\Users\...\Documents\MATLAB\backup\13122016-151559"
-gt>>product was installed in "C:\Users\...\Documents\MATLAB\ulmsip"
-gt>>product was installed in "C:\Users\...\Documents\MATLAB\ulm2ods"
-gt>>product was installed in "C:\Users\...\Documents\MATLAB\ulaxy"
-gt>>product was installed in "C:\Users\...\Documents\MATLAB\ulmsipt"
-gt>>product was installed in "C:\Users\...\Documents\MATLAB\h3dof"
-gt>>product was installed in "C:\Users\...\Documents\MATLAB\ddbb"
-gt>>product was installed in "C:\Users\...\Documents\MATLAB\fhover"
-gt>>product was installed in "C:\Users\...\Documents\MATLAB\gbb"
-gt>>done...
fx >> |

```

4) 在命令行输入“gtmdb -mex”。

```

命令窗口
>> gtmdb -mex

Attention:
Still continue to mex the Googoltech matlab toolbox? [yes] :

```

5) 输入yes 后按回车或直接按回车进行文件编译，并等待完成编译。

```

命令窗口
>> gtmdb -mex

Attention:
Still continue to mex the Googoltech matlab toolbox? [yes] :
-gt>>mexing 1.32.1.20161123.
MEX 已成功完成。
-gt>>mexing done...
fx >> |

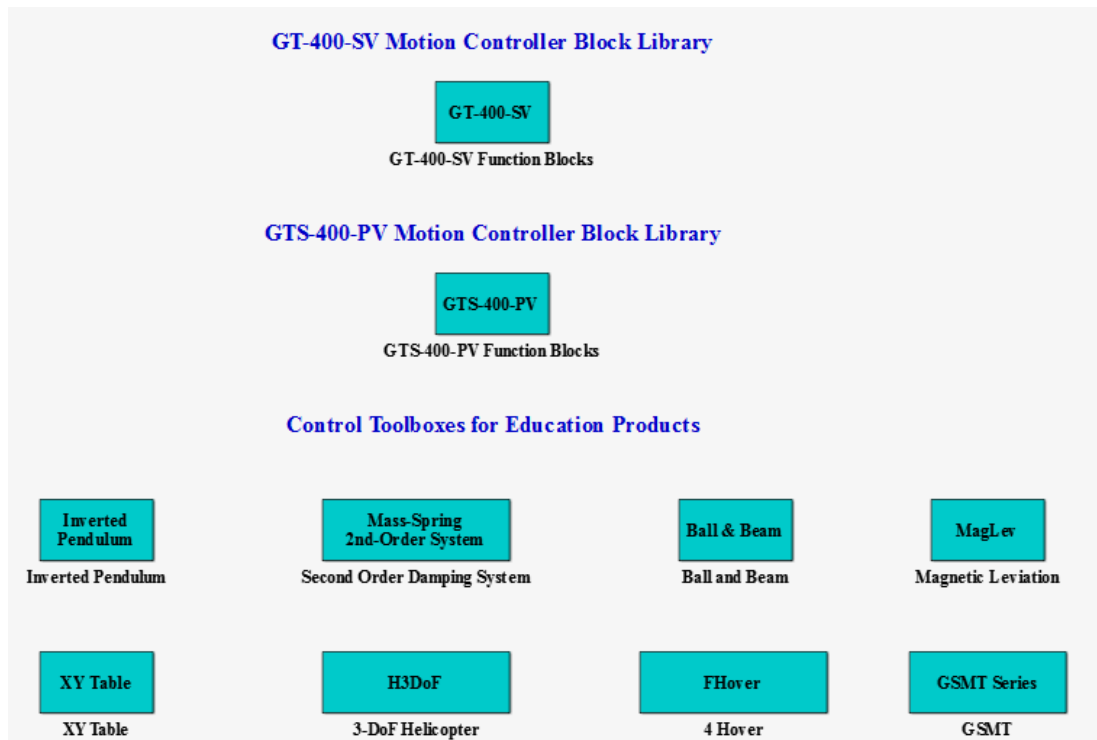
```

注意

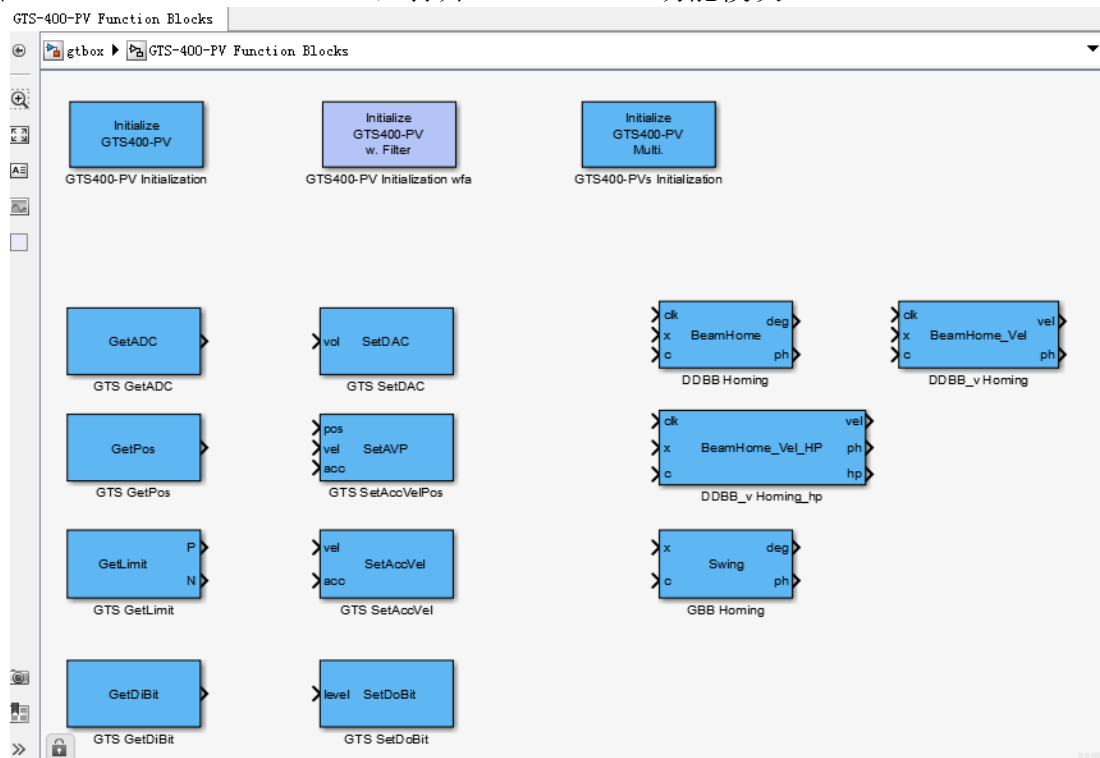
- 安装、编译完成后，需要重启 MATLAB

2. 工具箱模块介绍

1) 直接在命令窗口输入 gtbox 后回车，等待打开gtmtb 工具箱主页面：

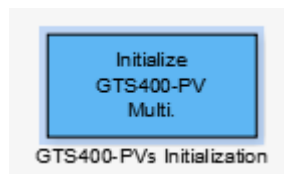


2) 双击“GTS-400-PV Function Blocks”，打开 GTS-400-SV 功能模块



3) GTS400-PVs Initialization 模块

功能：初始化控制器；



双击打开 GTS400-PVs Initialization 模块

GTS Multi. Initialize (mask) (link)

Block for card initialization

Parameters:

Card Number 1~4

Open/Closed Loop 0: closed-loop
1: open-loop
else: servo off

MotionMode reserved

Servo ON/OFF [axis1 axis2 axis3 axis4]
1: servo on
else: servo off

ADC Filter Time [aft1 aft2 aft3 aft4]
0~50

PID parameters [kp ki kd]
kp, ki, kd: 0~32767

Enc Reverse checked: reverse
unchecked: normal

Limit Reverse checked: reverse
unchecked: normal

Normal ADC Filter Axis1 Axis2 Axis3 Axis4

Open/Closed Loop

Servo ON/OFF

[1 1 0 0]

Card Number

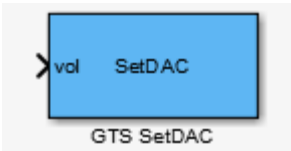
1

OK Cancel Help Apply

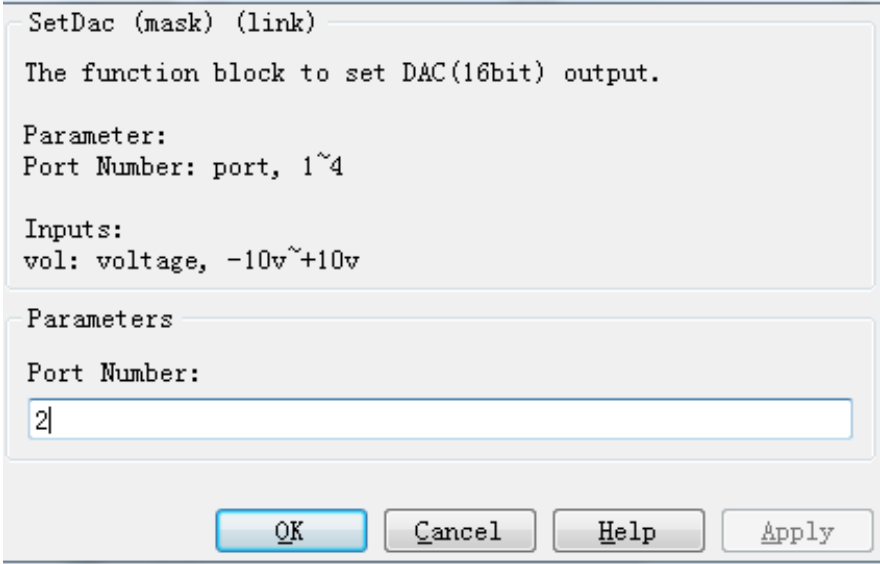
负责初始化GTS-400-PV 控制器

类别	标示	功能	默认值
参数	Card Number	表示选择控制卡卡号： 1:表示选择 1 号控制卡 2:表示选择 2 号控制卡 3:表示选择 3 号控制卡 4:表示选择 4 号控制卡	1
	Open/Closed Loop	表示控制器开、闭环控制。 0: 控制器闭环控制 1: 控制器开环控制 2: 控制器开环且轴不上伺服（多用于硬件测试）	2
	ServoON/OFF	表示各轴上下伺服 1: 上伺服 2: 下伺服	0
	Motion Mode	运动模式。 0: 梯形	0
	ADC Filter Time	表示各轴滤波器参数 范围 0~50	0
	PID parameters	[kp, ki, kd] 各轴使用控制器闭环时的 PID 参数 kp, ki, kd 参数为 0~32767, 但不能全为 0	[1,0,0] [1,0,0] [1,0,0] [1,0,0]
	ENC Reverse	表示编码器信号反向 不打勾: 正常 打钩: 取反	不打勾
	Limit Reverse	表示限位信号反向 不打勾: 正常 打钩: 取反	不打勾

4) GTS SetDAC 模块
功能：设置轴通道模拟量输出；

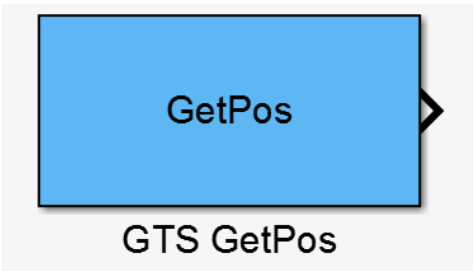


双击打开 GTS SetDAC 模块：

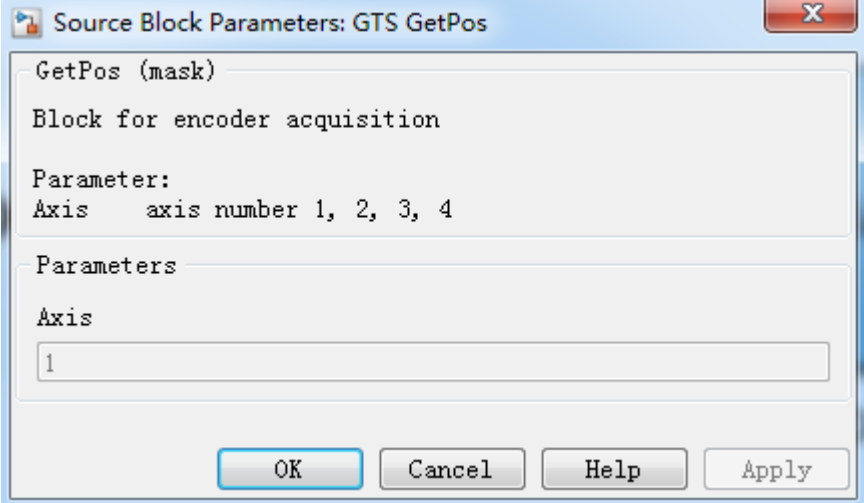


负责设定通道的模拟量输出			
类别	标示	功能	默认值
参数	Port Number	当前轴号，取值1、2、3、4	1
输入	vol	DAC（16bit）输出值，单位为伏特，取值为-10~+10	/

5) GTS GetPos 模块
功能：捕获设定轴的编码器值；



双击打开 GTS GetPos 模块：

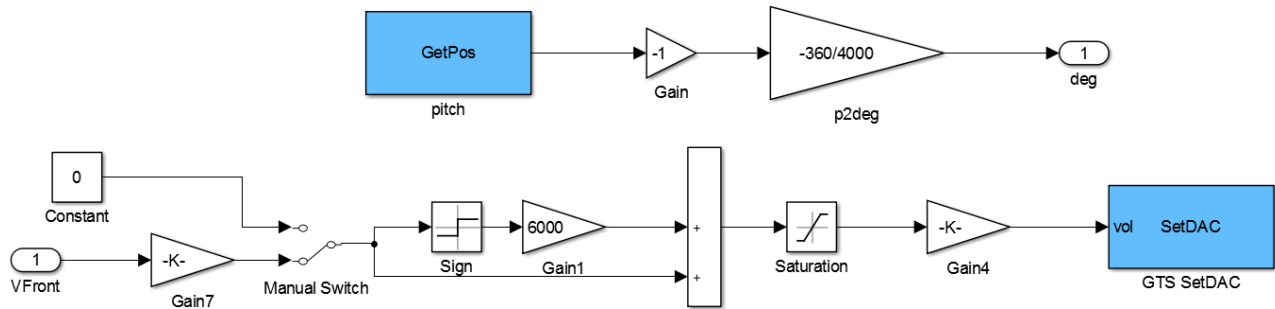


负责捕获设定轴的编码器值			
类别	标示	功能	默认值
参数	Axis	当前轴号，取值1、2、3、4	1
输出	/	设定轴的编码器数值，单位脉冲	/

2.4 Real Time Control 模块说明

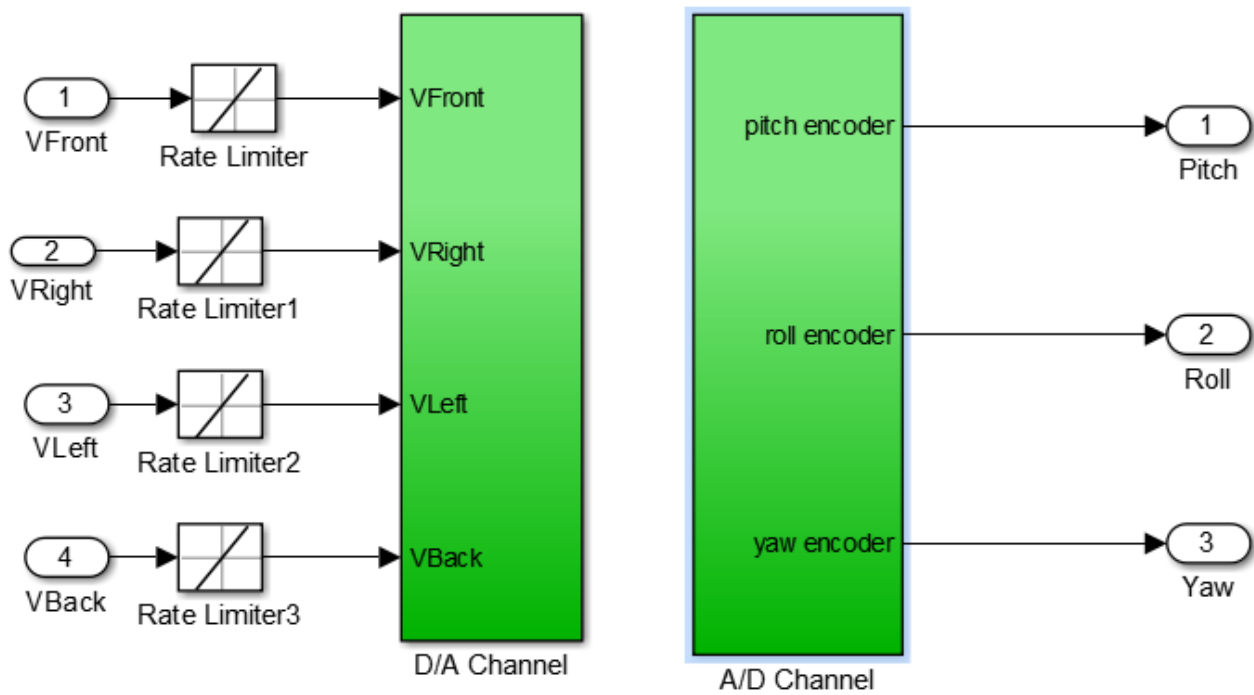
1. 仅俯仰控制的程序的 Real Time Control 模块说明

双击 Real Time Control 模块，打开如下图所示：

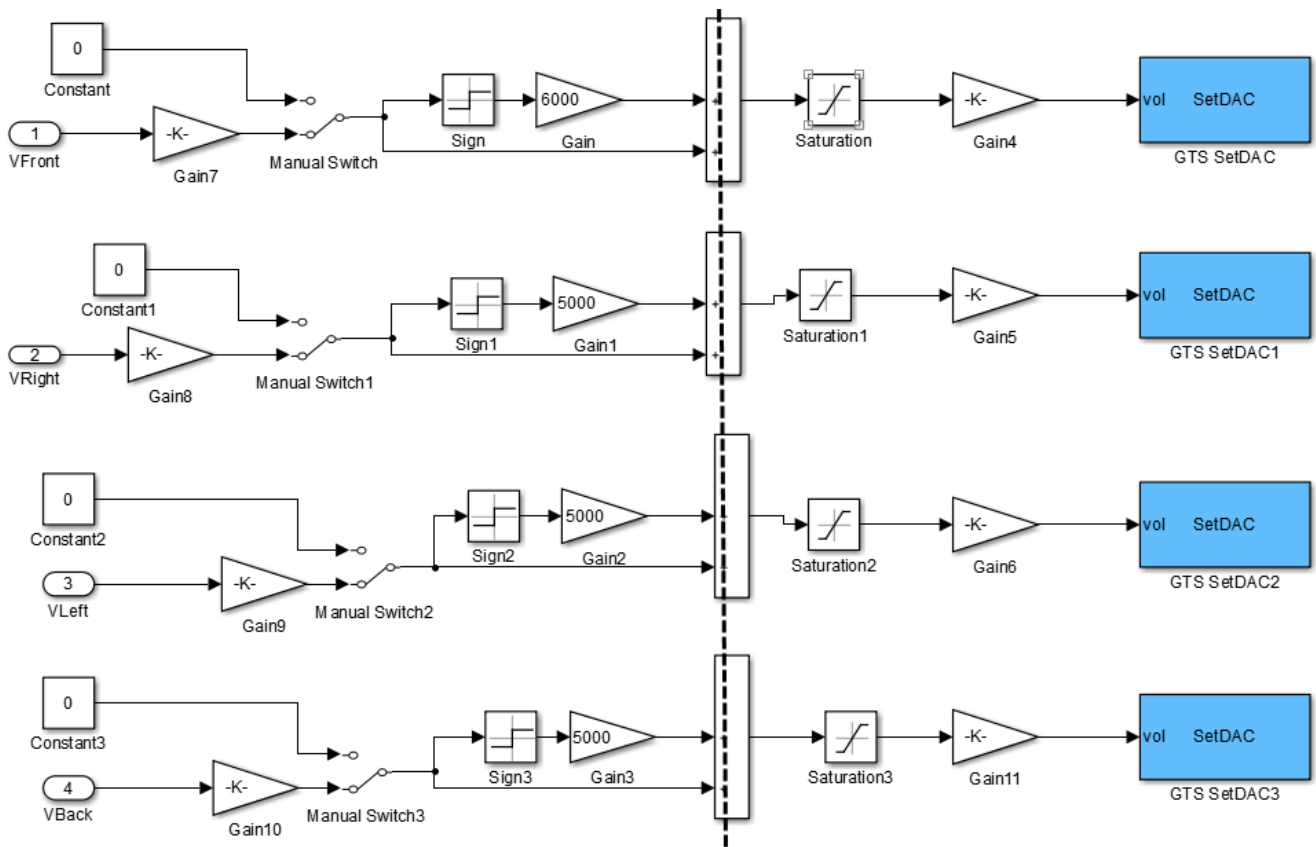


- 1) 蓝色模块均为运动控制卡模块，详见第二章 2.3 节；
- 2) Cain 模块调整编码器反馈方向，使其方向与建模定义方向一致；
- 3) P2deg 模块把脉冲转换成角度；
- 4) Saturation 模块左侧为死区补偿；
- 5) Saturation 模块右侧为运动控制卡饱和输出时对应电机最大转速；

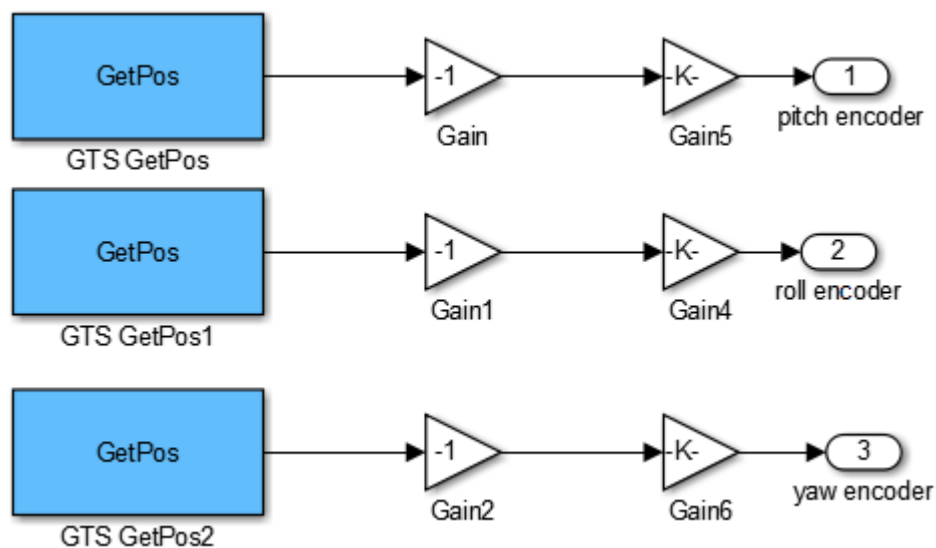
2. 全姿态控制程序的 Real Time Control 模块说明



- 1) VFront、VRight、VLeft 及 VBack 模块为控制器输出电压；
- 2) Rate Limiter、Rate Limiter 1、Rate Limiter2 及 Rate Limiter3 为限幅模块；
- 3) 双击 D/A Channel 模块，如下图所示：



- 以虚线为界左侧为克服死区功能，右侧为控制卡饱和输出对应电机的最高转速；
- 蓝色模块为运动控制卡模块，详见第二章 2.3 节；
- 双击 A/D Channel 模块，如下图所示



- Gain、Gain1 及 Gain2 模块为调整编码器反馈方向；
- Gain4、Gain5 及 Gain6 模块为把脉冲转换成角度；
- 蓝色模块为控制卡模块，详见第二章 2.3 节；

第三章 机理法建立四旋翼飞行仿真器的数学模型

(一)实验目的

1. 了解机理法建模；
2. 掌握控制系统稳定性分析的基本方法；

(二)实验设备

1. GHP3001-II 四旋翼飞行仿真器
2. PC (MATLAB 平台)

(三)实验原理

1. 建立三维坐标系：

四旋翼飞行仿真器可以建立如下图 1.1 的坐标系：

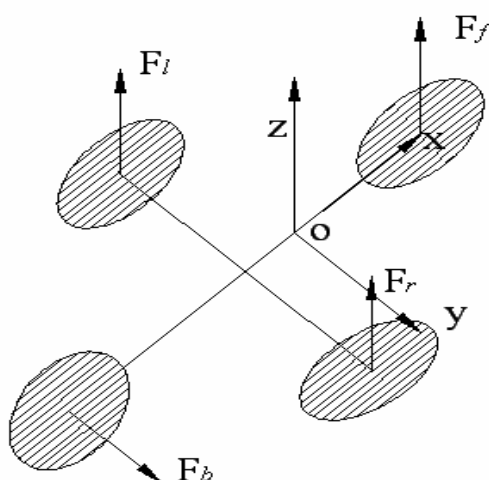


图 1.1 坐标系建立

其中，坐标原点位于支撑点，指向正前方电机的轴为 X 轴，指向右侧电机的轴为 Y 轴，采用左手定则确立坐标系 Z 轴正方向。定义正前方、左、右电机转动，带动螺旋桨运动产生的力，与 Z 轴同向为正向。

尾部电机转动带动螺旋桨运动产生的力，与 Y 轴同向为正向。其中 F_f 为前向电机的升力， F_l 为左侧电机的升力， F_r 为右侧电机的升力。 F_b 为尾部电机的力。

四旋翼飞行平台的物理参数如下表所示：

物理量	数值	符号	单位	备注
各个电机臂长	0.52	L_f	米	
各个单臂质量	1	m	公斤	包括：电机、罩子、螺旋桨、螺栓
配重	0.945	m_1	公斤	
配重臂长	0.5	L_1	米	
电压、升力比	15	K_{fc}	牛/伏	
俯仰转动惯量		J_p		
滚动转动惯量		J_r		

偏航转动惯量		J_y		
左右臂与X轴夹角	60		度	
前向电机电压		V_f	伏	
右侧电机电压		V_r	伏	
左侧电机电压		V_l	伏	
后侧电机电压		V_b	伏	
前向螺旋桨力		F_f	牛顿	
右侧螺旋桨力		F_r	牛顿	
左侧螺旋桨力		F_l	牛顿	
后侧螺旋桨力		F_b	牛顿	
俯仰角		P		
滚动角		R		
偏航角		Y		

2. 三个姿态角的力矩平衡方程

1) 建立三个输出姿态角的力矩平衡方程，并做以下假设：

- 假设系统处于静平衡状态，三个姿态角均为零，并忽略摩擦力、电机阻尼力矩；
- 忽略电机达到给定转速的时间；
- 假设螺旋桨正反转时产生的力一样。

2) 俯仰角力矩平衡方程：

俯仰角静平衡受力分析如下图 1.2 所示，定义俯仰角为前电机向下运动夹角为正。

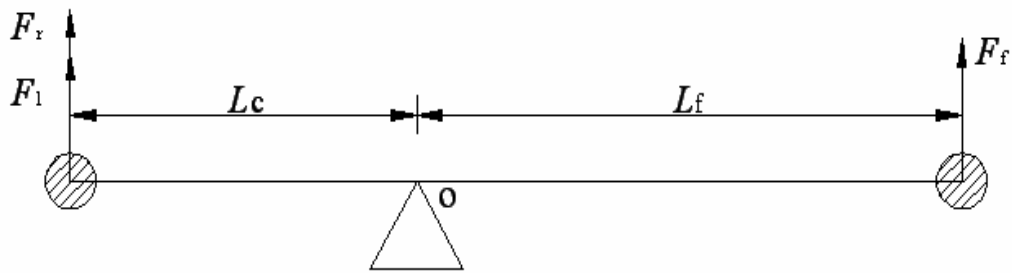


图 1.2 俯仰角静平衡受力分析

当左右电机力向上，前向电机力向下时，俯仰向正角度方向运动，则建立如下方程：

$$J_p \ddot{P} = -F_f L_f - F_l L_c - F_r L_c$$

其中：

$$L_c = L_f \cos(60^\circ) = 0.5L_f$$

$$F = K_{fc} V$$

所以可化简为：

$$\ddot{P} = -\frac{K_{fc} L_f}{J_p} (V_f + 0.5V_r + 0.5V_l) \quad (1-1)$$

3) 滚动角力矩平衡方程:

滚动角静平衡受力分析如下图 1.3 所示, 定义滚动角为右侧电机向下运动夹角为正

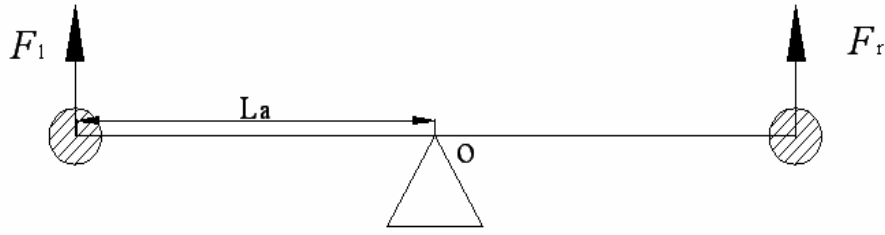


图 1.3 滚动角静平衡受力分析

当左右电机电力差使本体滚动角度为正向运动, 则建立如下方程:

$$J_r \ddot{R} = F_l L_a - F_r L_a$$

其中:

$$L_a = L_f \sin(60^\circ) = 0.866 L_f$$

$$F = K_{fc} V$$

所以可化简为:

$$\ddot{R} = -\frac{K_{fc} L_f}{J_r} (-0.866 V_l + 0.866 V_r) \quad (1-2)$$

4) 偏航角力矩平衡方程

定义偏航角为绕 Z 轴顺时针运动角度为正。只有尾部电机转动影响偏航角。则建立如下方程:

$$J_y \ddot{Y} = F_b L_f$$

其中:

$$F = K_{fc} V$$

所以可化简为:

$$\ddot{Y} = \frac{K_{fc} L_f}{J_y} (V_b) \quad (1-3)$$

5) 三个姿态的转动惯量计算

为保证系统静平衡, 在前向电机臂上增加了配重, 设各个部分质量近似为一个质点, 则三个姿态角的转动惯量可以由下述三式确定:

$$J_p = 2mL_f^2 + 2m(0.5L_f)^2 + m_1 L_1^2 = 0.9113 \quad (1-4)$$

$$J_r = 2m(0.866L_f)^2 = 0.405 \quad (1-5)$$

$$J_y = 2mL_f^2 + 2mL_f^2 + m_1 L_1^2 = 1.3065 \quad (1-6)$$

6) 力电压比 K_{fc}

K_{fc} 可以通过实验法测出, 通过给定不同的电压值, 测试对应输出的力矩为多大, 经测量约为 15

7) 三个姿态角转矩平衡方程

把式 (1-4) (1-5) (1-6) 分别带入式 (1-1) (1-2) (1-3) 可得:

$$\ddot{P} = -8.3848 V_f - 4.1151 V_r - 4.1151 V_l \quad (1-7)$$

$$\ddot{R} = -16.6667 V_l + 16.6667 V_r \quad (1-8)$$

$$\ddot{Y} = 5.9703 V_b \quad (1-9)$$

3. 仅考虑俯仰角控制

俯仰角静平衡受力分析如下图 1.4 所示，定义俯仰角为前电机向下运动夹角为正。

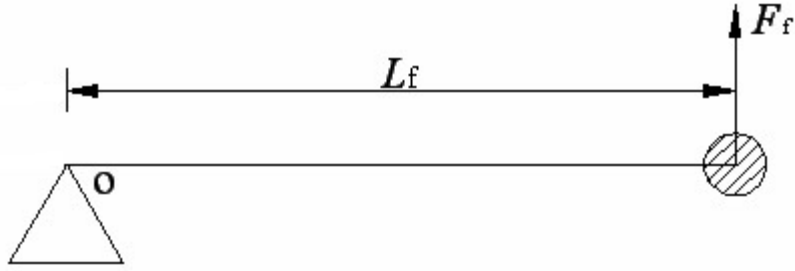


图 1.4 俯仰角静平衡受力分析

当左右电机力向上，前向电机力向下时，俯仰向正角度方向运动，则建立如下方程：

$$J_p \ddot{P} = -F_f L_f$$

其中：

$$F = K_{fc} V$$

带入可得：

$$\frac{\ddot{P}}{V_f} = -\frac{K_{fc} L_f}{J_p} \quad (1-10)$$

由上述可知： $K_{fc} = 15$ 、 $L_f = 0.52$ 、 $J_p = 0.9113$ 代入式 (1-10) 中可得：

$$\frac{\ddot{P}}{V_f} = -8.56 \quad (1-11)$$

即输出俯仰角速度与输入电压的关系为：

$$\frac{\ddot{P}}{V_f} = 8.56$$

所以输出角度与输入电压的关系为：

$$\frac{P}{V_f} = \frac{8.56}{s^2}$$

即四旋翼俯仰角控制数学模型为：

$$\frac{P}{V_f} = \frac{8.56}{s^2} \quad (1-12)$$

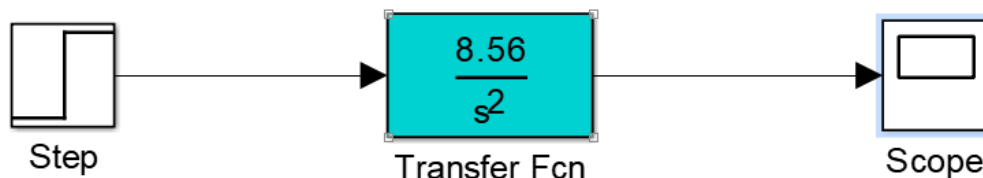
(四)稳定性分析（仅俯仰控制）

1.系统分析

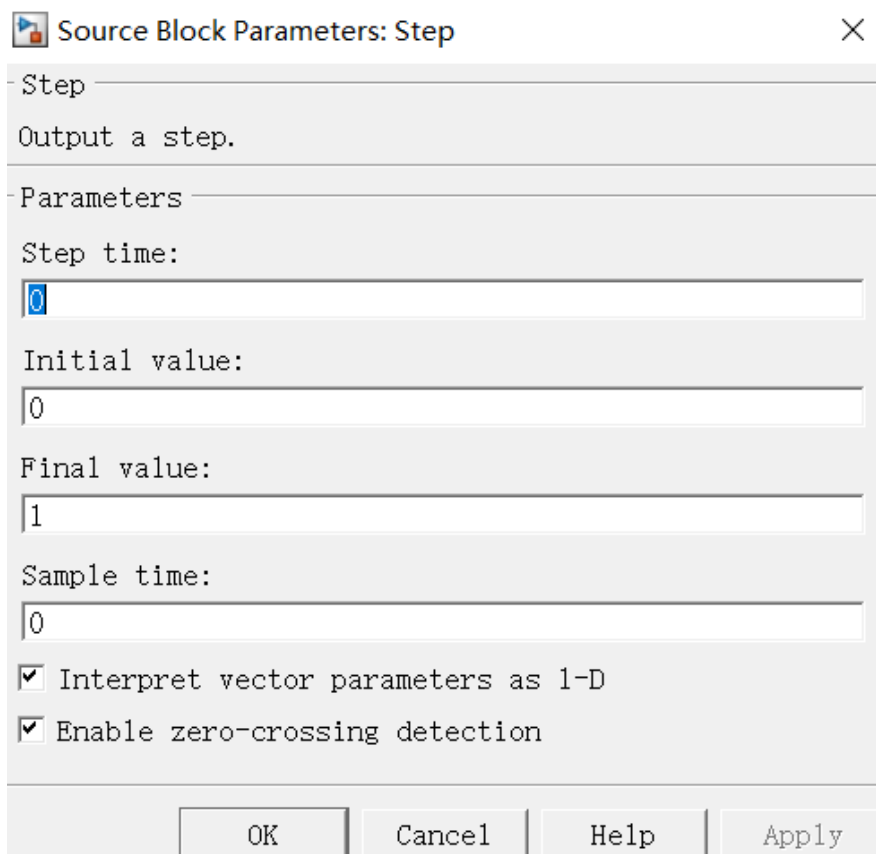
由第一章第一节中 3.可知系统数学模型为： $\frac{P}{V_f} = \frac{8.56}{s^2}$ ，由此可以看出系统两个极点均在[0]，只有极点均在左半平面时系统才能稳定，所以该系统不稳定


2.实验步骤

1) 打开 Matlab\Simulink 仿真环境，并建立如图 1.5 系统仿真结构图

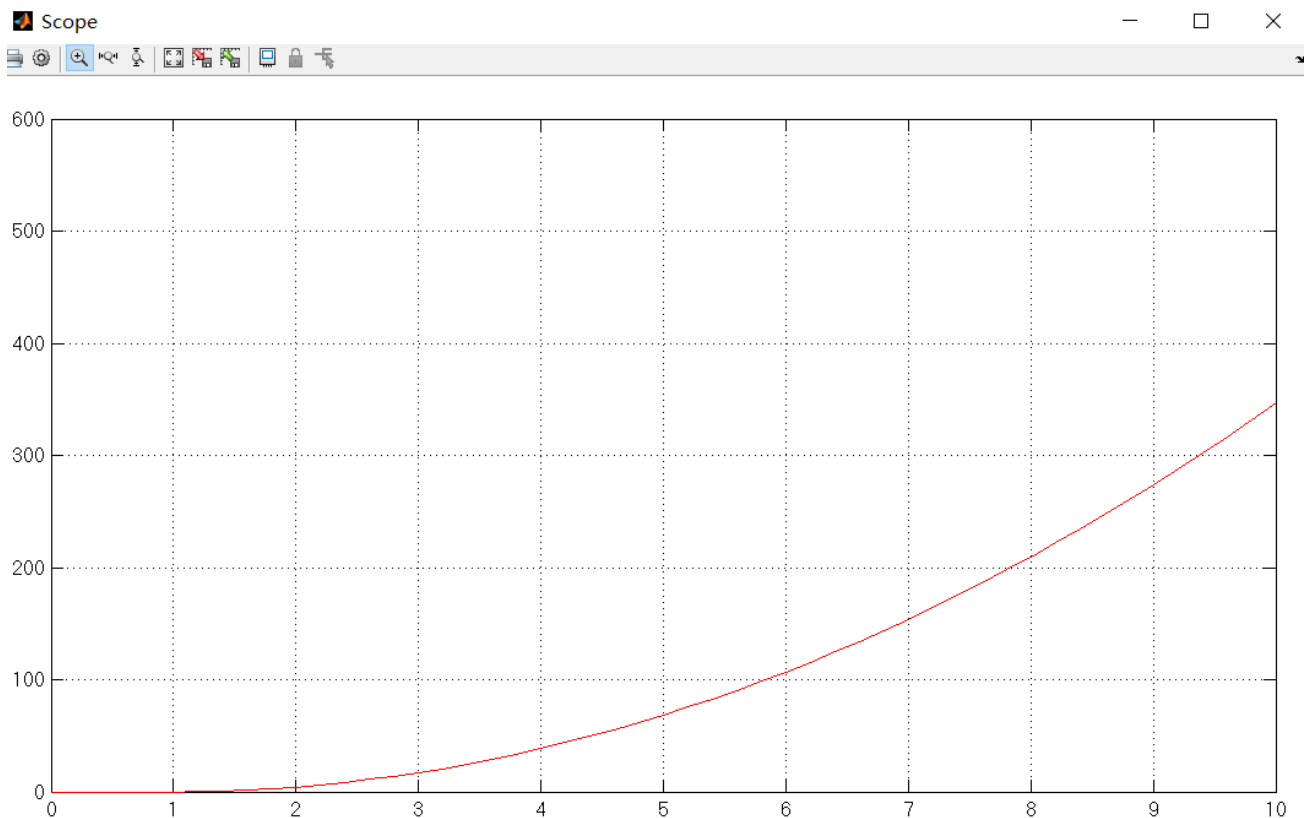


2) 双击“Step”模块，设置 Final value 值为 1



3) 点击“”，运行仿真程序

4) 双击 Scope 模块，观察仿真波形



(五)实验记录

内容	数据
开环系统传递函数	$\frac{P}{V_f} = \frac{8.56}{s^2}$
开环系统输入	1V
开环系统输出信号	开环系统不稳定

(六)实验分析及思考

- 1) 影响系统稳定的因素是系统的极点位置，如果极点位于 s 右半平面，则系统不稳定。测量系统稳定性的方法之一是加入适量大小的阶跃信号，根据其输出的阶跃响应分析系统的稳定性和其他性能指标
- 2) 根据四旋翼建模过程，总结机理法建模步骤。

第四章 系统硬件检测

(一)实验目的

1. 了解并熟练使用系统软件接口
2. 掌握 Matlab\Simulink 仿真环境的使用

(二)实验设备

1. GHP3001 四旋翼飞行仿真器
2. PC (MATLAB 平台)

(三)实验原理

1. 硬件连接

硬件连接连接架构图如图 4.1 所示：

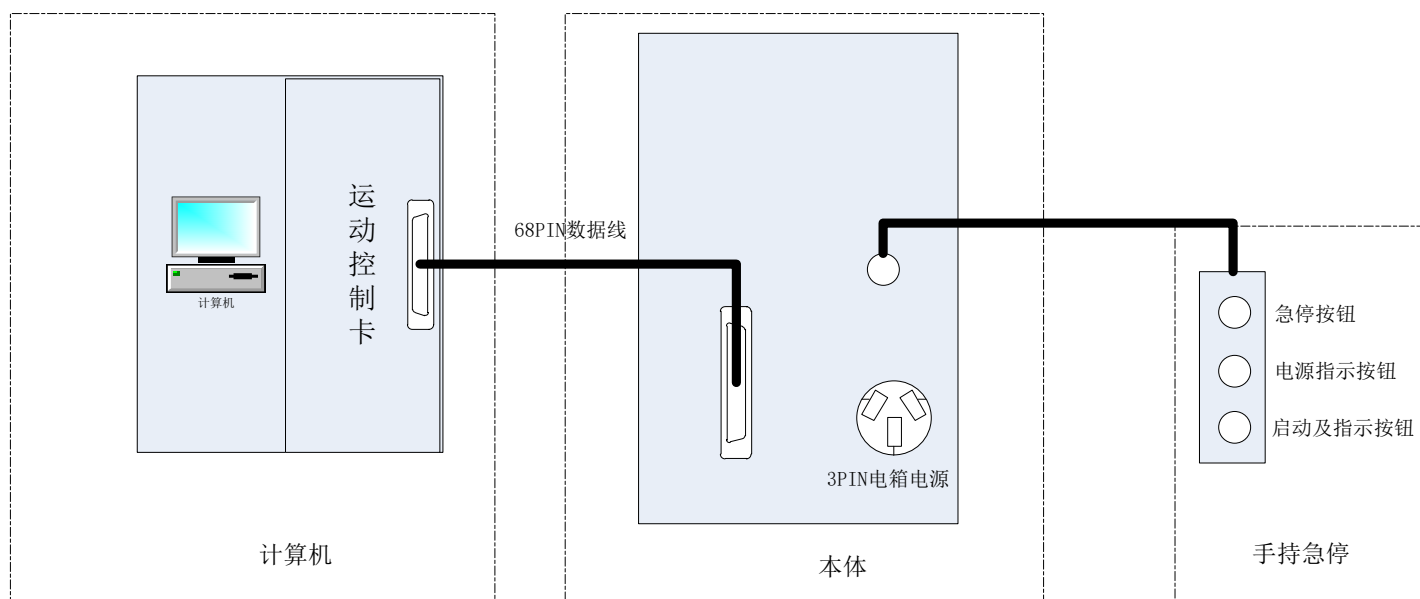


图 4.1 硬件连接连接架构图

1) 运动控制卡（PC 机中）连接四旋翼本体（信号传输）

把输出的电压信号发送到各轴电机驱动器，然后通过驱动器控制电机旋转速度及方向；也把俯仰角、滚动角及偏航角实反馈到上位机。

2) 连接手持急停至四旋翼本体（安全保护）

四旋翼系统出现非正常情况，保护设备及人身安全

3) 系统电源输入（220V）

2. 检测原理

1) 系统输入模拟电压检测

通过 matlab 让运动控制卡发出模拟电压信号，观察对应螺旋桨是否正常旋转（螺旋桨和轴相对应、旋转正方向是否正确）

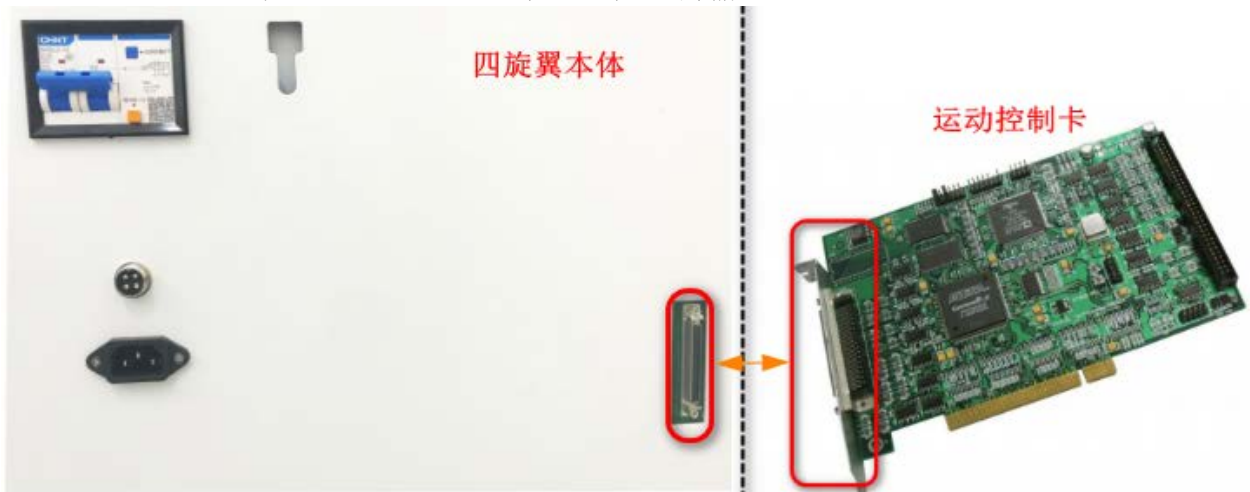
2) 系统角度反馈检测

手动变化俯仰角，检查编码器反馈是否正常：反馈反向是否正确及反馈轴通道是否正确（例如：俯仰角度编码器连接运动控制卡二轴通道，结果在一轴通道检测到了，说明硬件接线错了）反馈编码器方向详见第三章中实验原理第一节

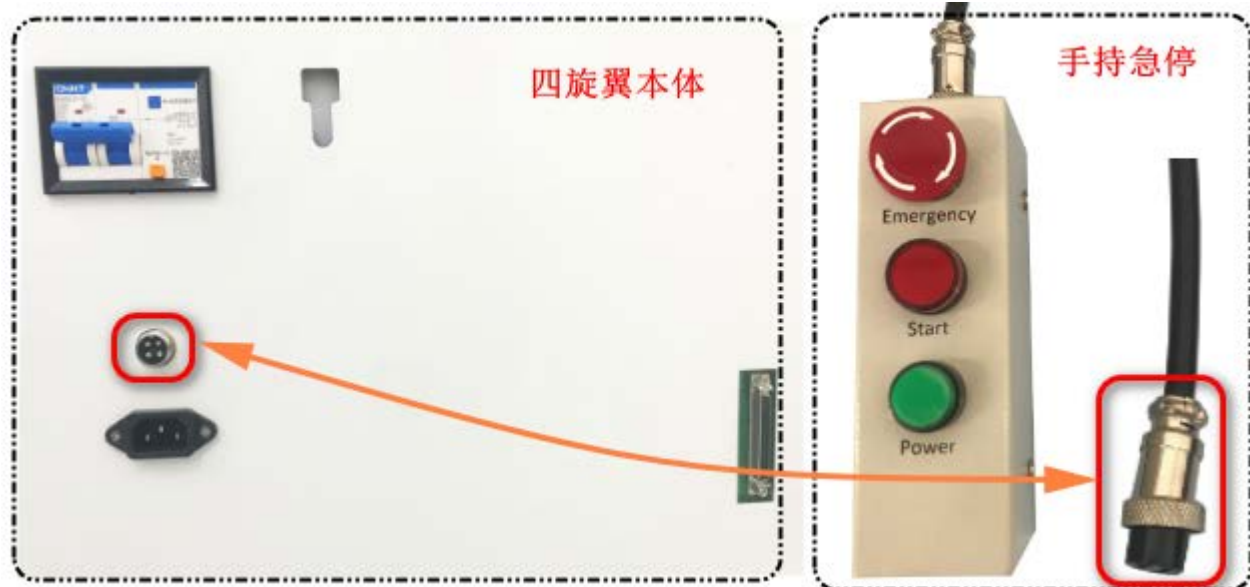
(四)实验步骤

1. 系统硬件连接

1) 运动控制卡（PC 机中）连接四旋翼本体（信号传输）



2) 连接手持急停至四旋翼本体（安全保护）



3) 系统电源输入（220V）



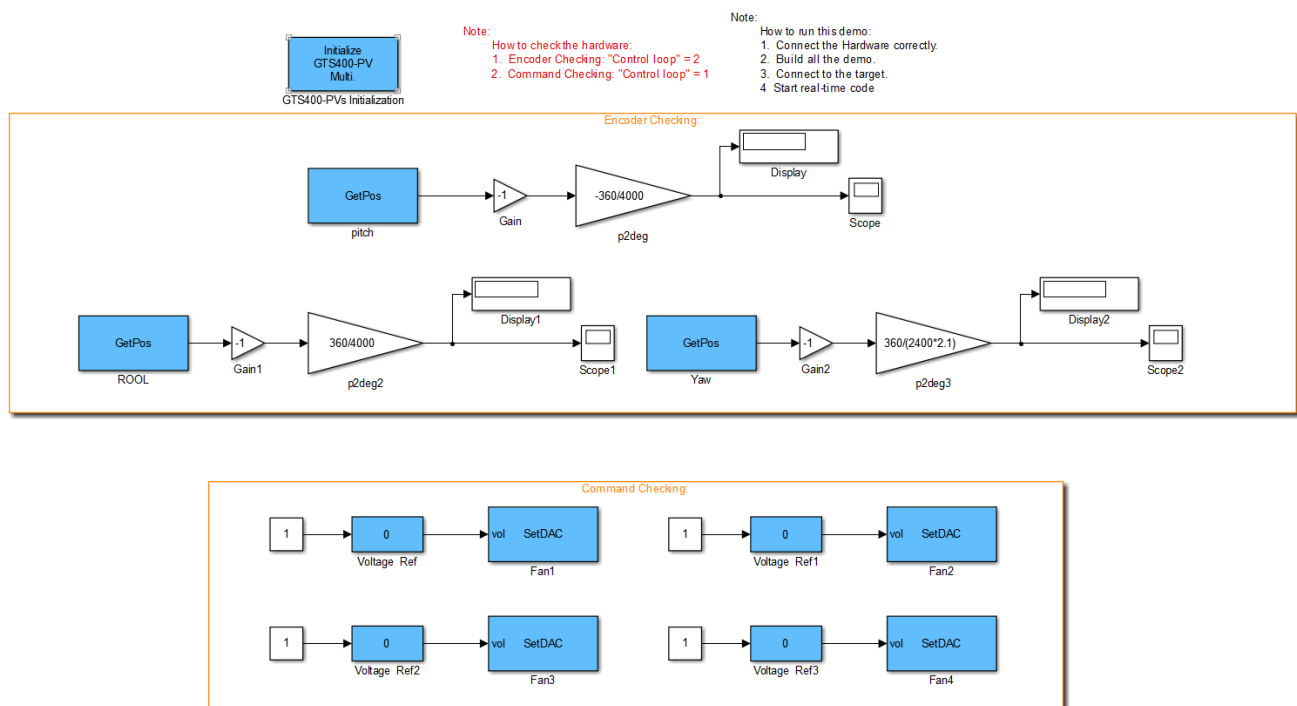
2. 系统硬件检测

1) 系统输入模拟电压检测

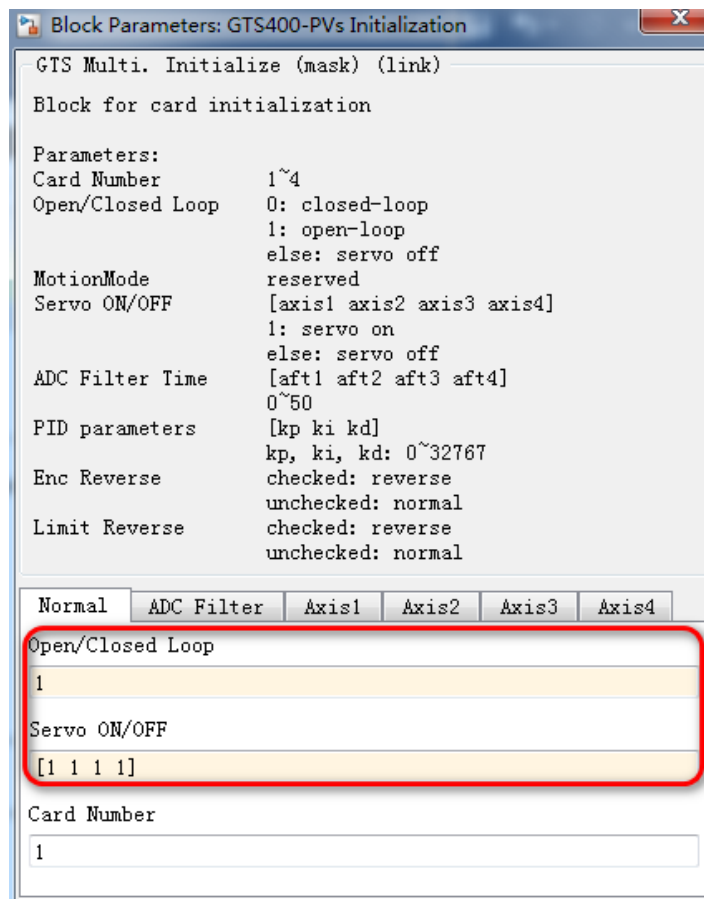
a) 打开四旋翼飞行仿真器的电源详见第一章第一节



b) 在 MATLAB 命令行中输入“gtbox”，打开 gtbox 工具箱中的“4 Hover”→“GTS”→“0-Hardware Testing”中的“Real Time Control”，弹出如图实时控制界面。

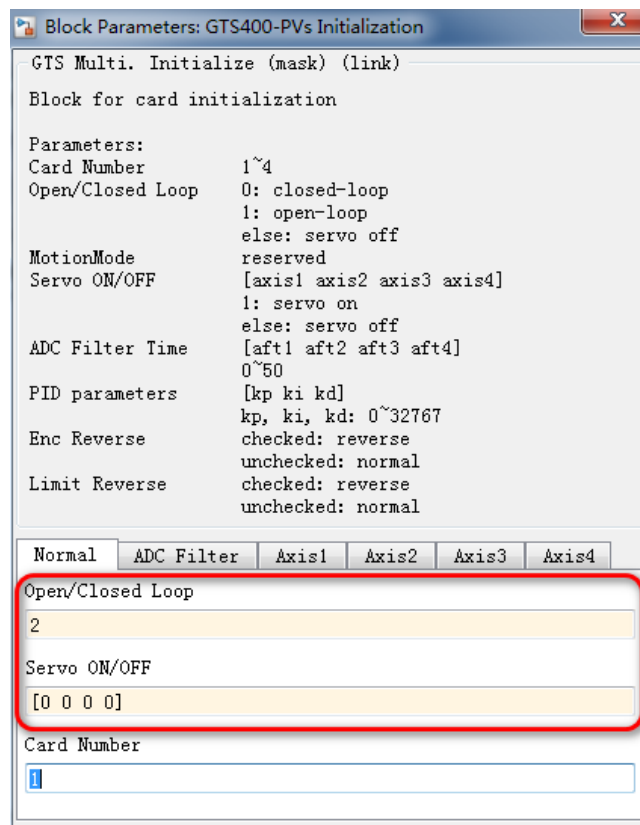
Googol 4 Hover Hardware Test Control Demo





c) 双击 GTS400-PVs Initialization 模块，打开设置“Open/Closed Loop”设置值为 1，“Servo On/OFF”设置值为[1 1 1 1]



- d) 点击“”运行程序，等待程序编译运行完成，随后“咚”的一声即程序运行
- e) 双击“Voltage Ref”模块，设置值为 1，则可以观察四旋翼一轴电机旋转方向，产生的力是否是向上的（风向下吹），确认完毕后，设置“Voltage Ref”模块值为 0，等待停转后设置“Voltage Ref”模块值为-1，确认一轴电机产生向下的力（风向上吹）。
- f) 双击“Voltage Ref1”模块，设置值为 1，则可以观察四旋翼二轴电机旋转方向，产生的力是否是向上的（风向下吹），确认完毕后，设置“Voltage Ref”模块值为 0，等待停转后设置“Voltage Ref”模块值为-1，确认二轴电机产生向下的力（风向上吹）。
- g) 双击“Voltage Ref2”模块，设置值为 1，则可以观察四旋翼三轴电机旋转方向，产生的力是否是向上的（风向下吹），确认完毕后，设置“Voltage Ref”模块值为 0，等待停转后设置“Voltage Ref”模块值为-1，确认三轴电机产生向下的力（风向上吹）。
- h) 双击“Voltage Ref3”模块，设置值为 1，则可以观察四旋翼四轴电机旋转方向，产生的力是否是向上的（风向下吹），确认完毕后，设置“Voltage Ref”模块值为 0，等待停转后设置“Voltage Ref”模块值为-1，确认四轴电机产生向下的力（风向上吹）。
- i) 确认四个螺旋桨均没有问题，点击“”停止程序运行。
- 2) 系统角度反馈检测
 - a) 紧接上述步骤，双击 GTS400-PVs Initialization 模块，打开设置“Open/Closed Loop”设置值为 2，“Servo On/OFF”设置值为[0 0 0 0]



- b) 点击“”运行程序，等待程序编译运行完成（点击运行程序后，菜单栏图标变灰，直到变回彩色即开始运行）
- c) 手动俯仰方向运动，向上“Display”模块显示值为正，向下“Display”模块显示值为负
- d) 手动滚动方向运动，向上“Display1”模块显示值为正，向下“Display1”模块显示值为负
- e) 手动俯仰方向运动，向上“Display2”模块显示值为正，向下“Display2”模块显示值为负
- f) 确认三个编码器均没有问题，点击“”停止程序运行

(五)实验记录

实验内容	实验结果
硬件连接	
硬件检测	

(六)实验分析及思考

该实验主要是深入了解四旋翼飞行仿真器的硬件结构及如何通过使用运动控制卡来检测系统硬件问题，对运动控制卡在 matlab 中的工具箱中的模块有更深层次的理解并使用

1. 程序中“p2deg”模块的作用是什么？
2. 为何要确定电机旋转方向及编码器的反馈方向？

第五章 PID 控制器设计

(一)实验目的

1. 会用 PID 法校正直流伺服电机系统；
2. 设计 PID 校正环节，控制四旋翼系统俯仰角度，系统开环传递函数为： $\frac{P}{V_f} = \frac{8.56}{s^2}$ ；

(二)实验设备

1. GHP3001 四旋翼飞行仿真器
2. PC（MATLAB 平台）

(三)实验原理

通过凑试法确定 PID 控制器中 k_p k_i k_d 的值，使四旋翼飞行器俯仰角度稳定在初始位置（程序运行前需要人为用手把四旋翼俯仰方向扶至水平位置左右，手扶的位置即为初始位置）

(四)实验步骤

1. 未校正系统 Simulink 仿真实验

- 1) 建立系统的 MATLAB 仿真模型，参考程序如图 5.1，并将文件保存名为“Pitch_sim”，默认类型为 slx

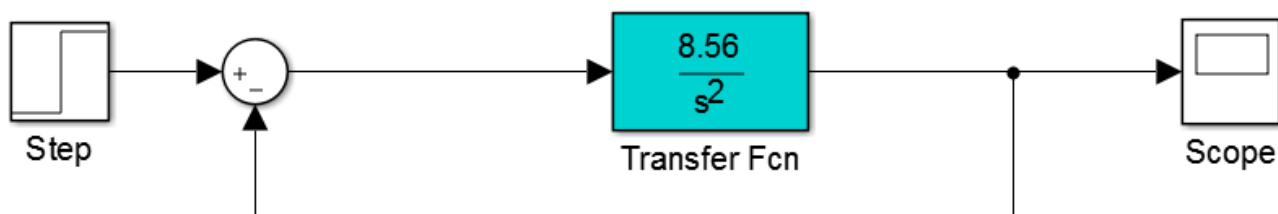
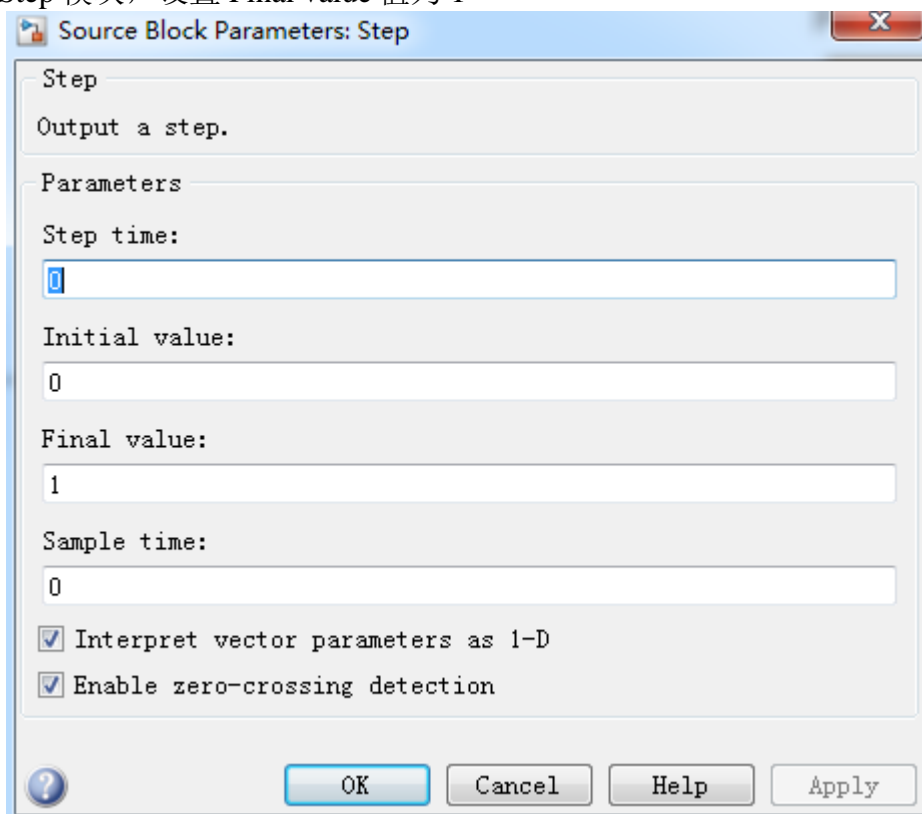
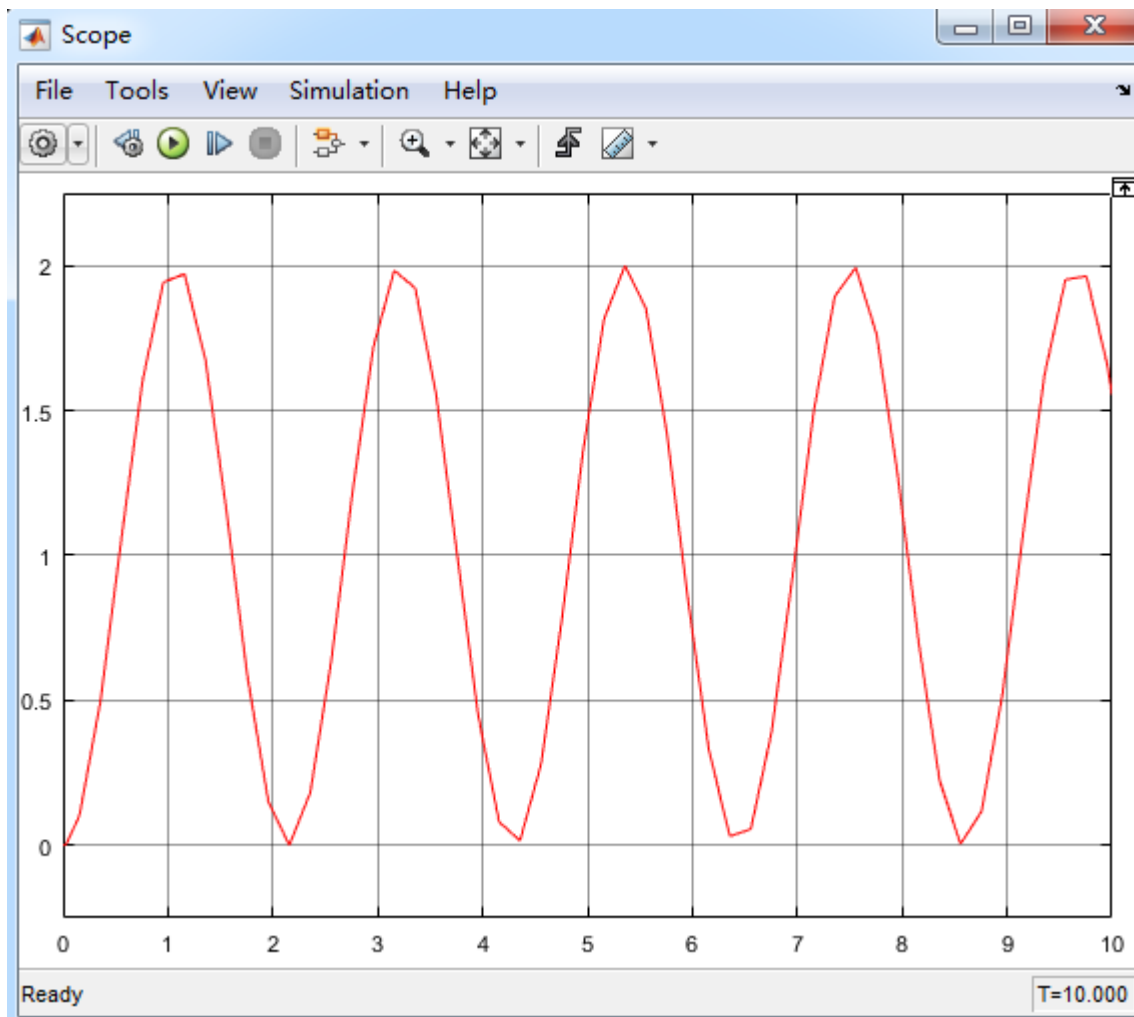


图 5.1 仿真参考程序

- 2) 双击打开 Step 模块，设置 Final value 值为 1



- 3) 点击“”，双击 Scope 模块，查看系统仿真曲线。



- 4) 闭环系统不稳定，设计 PID 校正环节，采用凑试法整定参数。
2. 加 PID 校正后 Simulink 仿真实验
- 1) 建立系统 MATLAB 仿真模型，参考程序如图 5.2，并将文件保存为“PID_sim”，默认类型为 slx

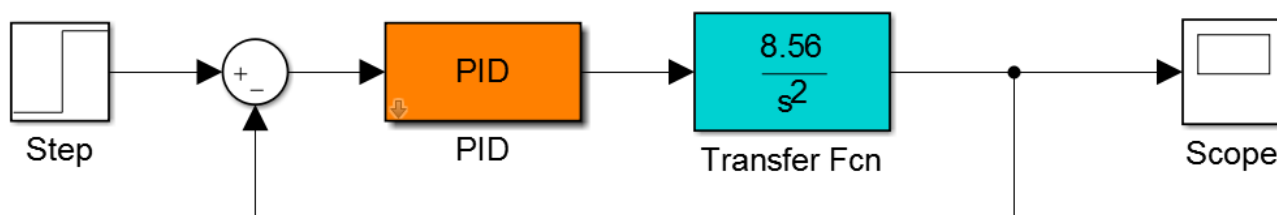
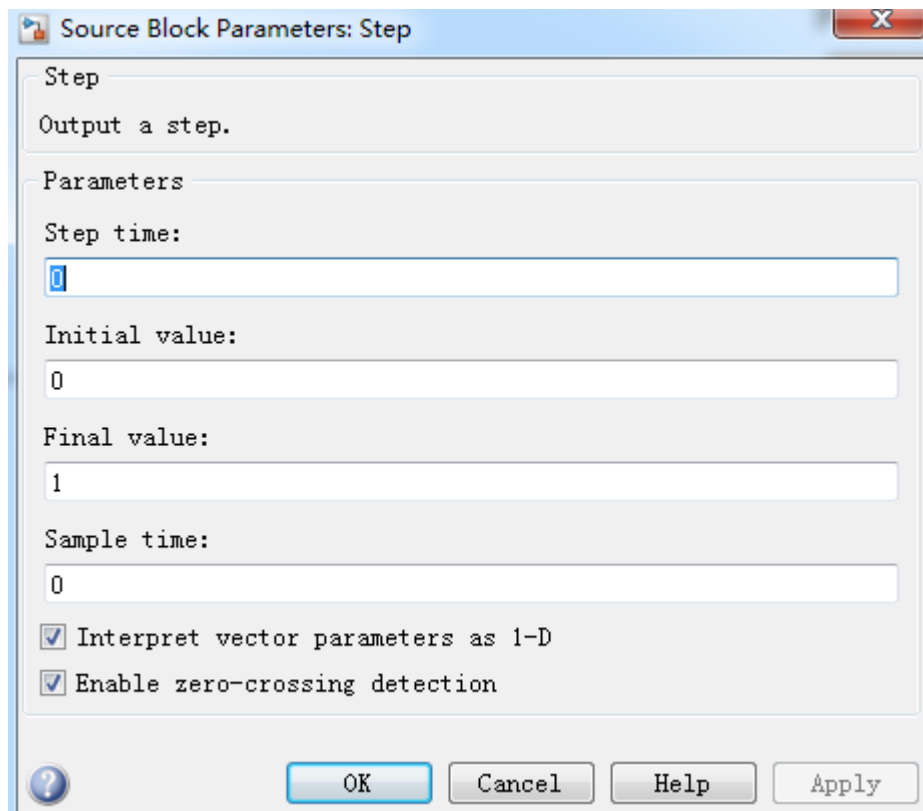


图 5.2 参考程序

- 2) 双击打开 Step 模块，设置 Final value 值为 1



3) 点击 PID 模块左下角向下的箭头，如图 5.3 所示

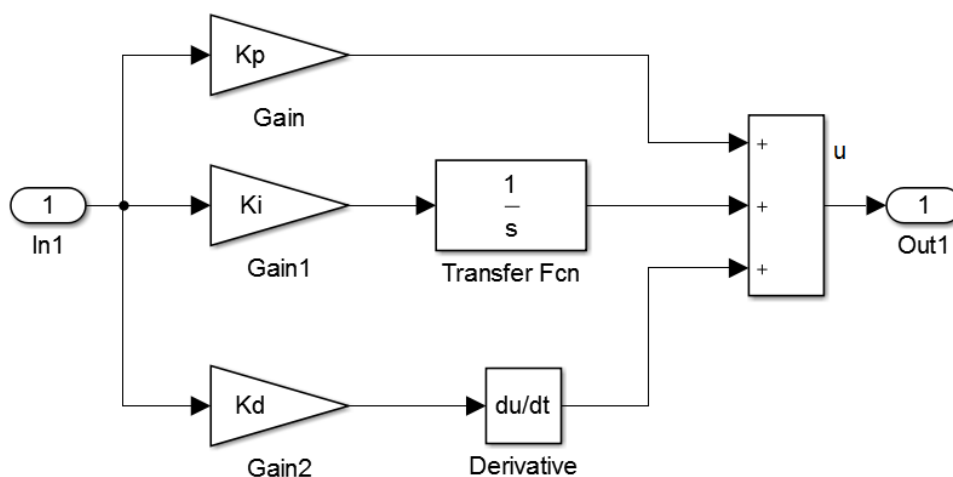
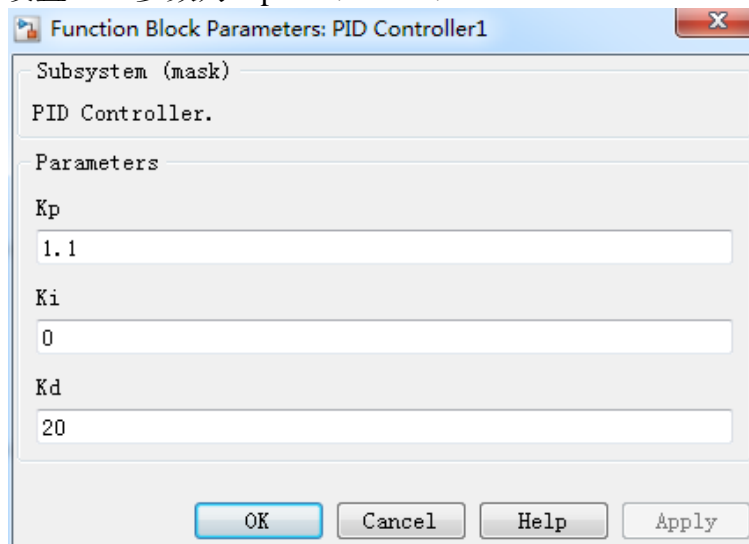
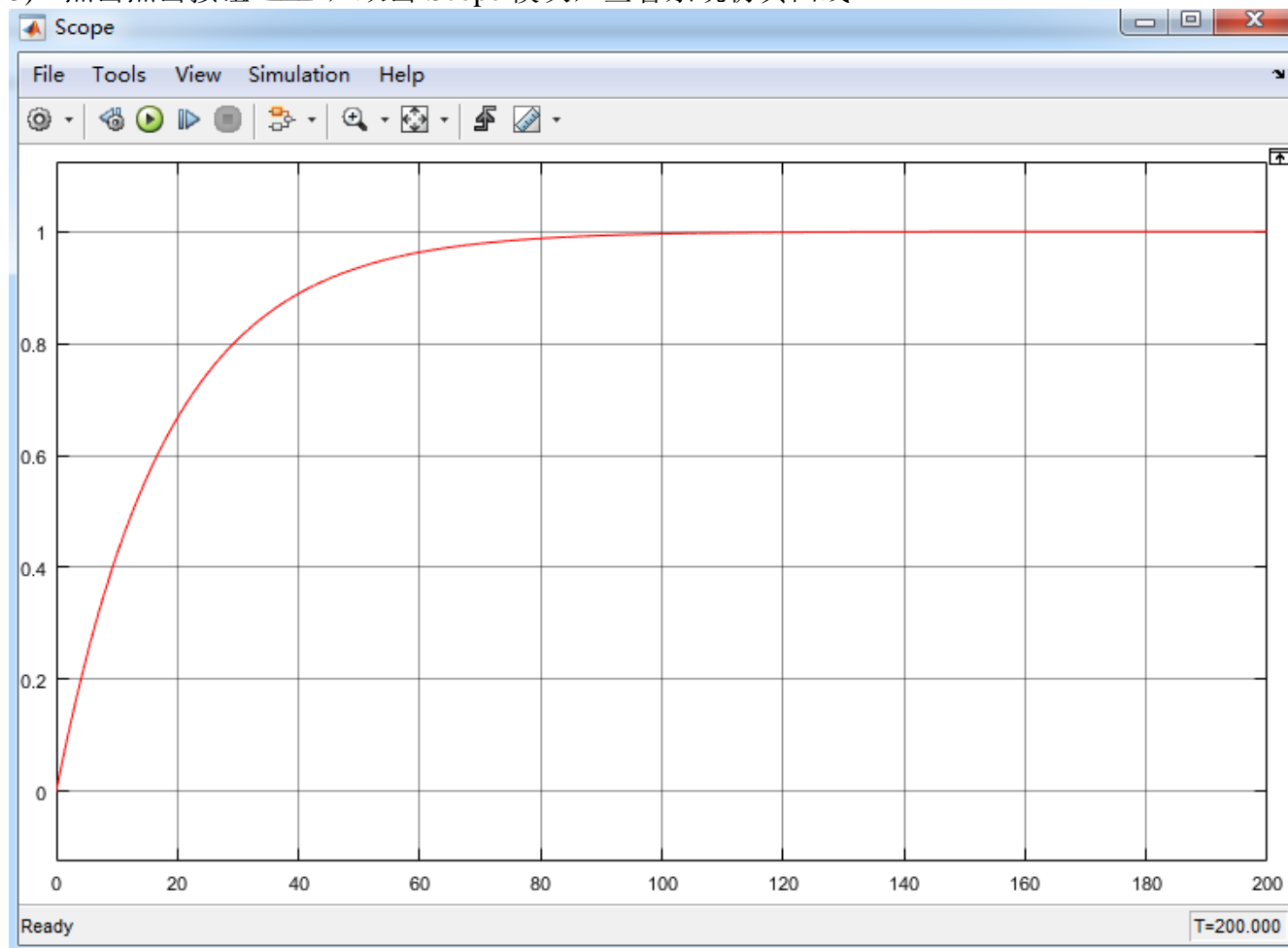


图 5.3 PID 模块结构

4) 双击 PID 模块，设置 PID 参数为 $K_p=1.1$, $K_i=0$, $K_d=20$ 。



5) 点击点击按钮“”，双击 Scope 模块，查看系统仿真曲线

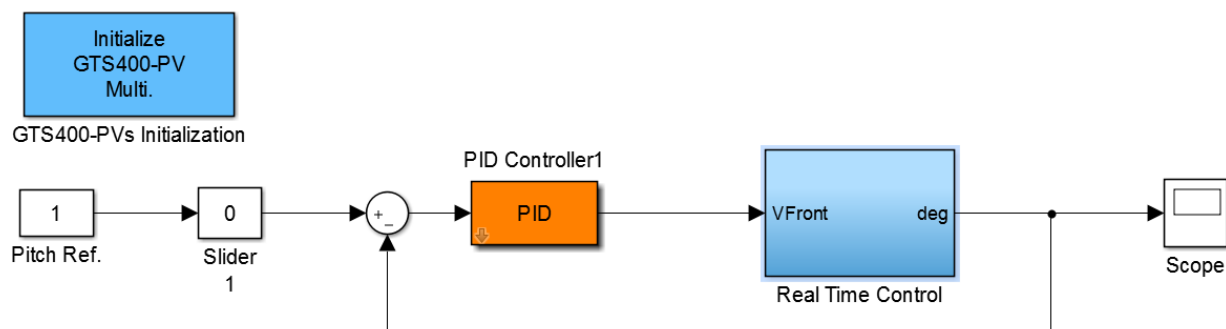


6) 系统稳定，调节时间 71.2s，稳态误差 0，超调量为 0

3. 加 PID 校正后实时控制实验

1) 四旋翼平台上电，详见第一章 1.1 节；

2) 打开 MATLAB，在 MATLAB 命令行中输入 gtbox，打开 gtbox 工具箱中的“4 Hover”→“GTS”→“1-6 States”下的“FH_LQR_State_6_P.slx”，弹出如下实时控制界面



a) 实时控制程序设计简介

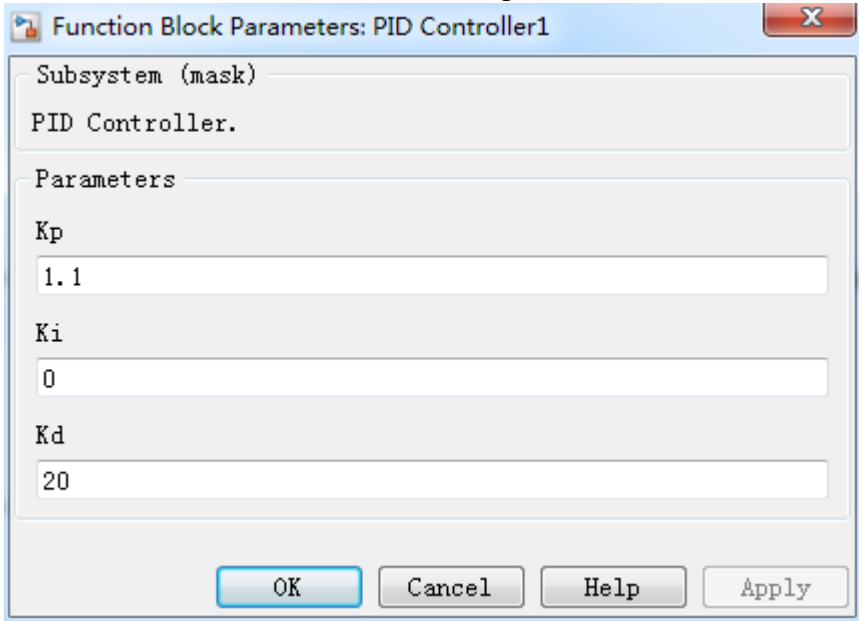
闭环负反馈系统，加 PID 校正环节，控制四旋翼俯仰角度。目标角度与实际角度比较后经过 PID 校正环节，来调节俯仰螺旋桨的旋转方向及旋转速度。



b) 程序模块说明

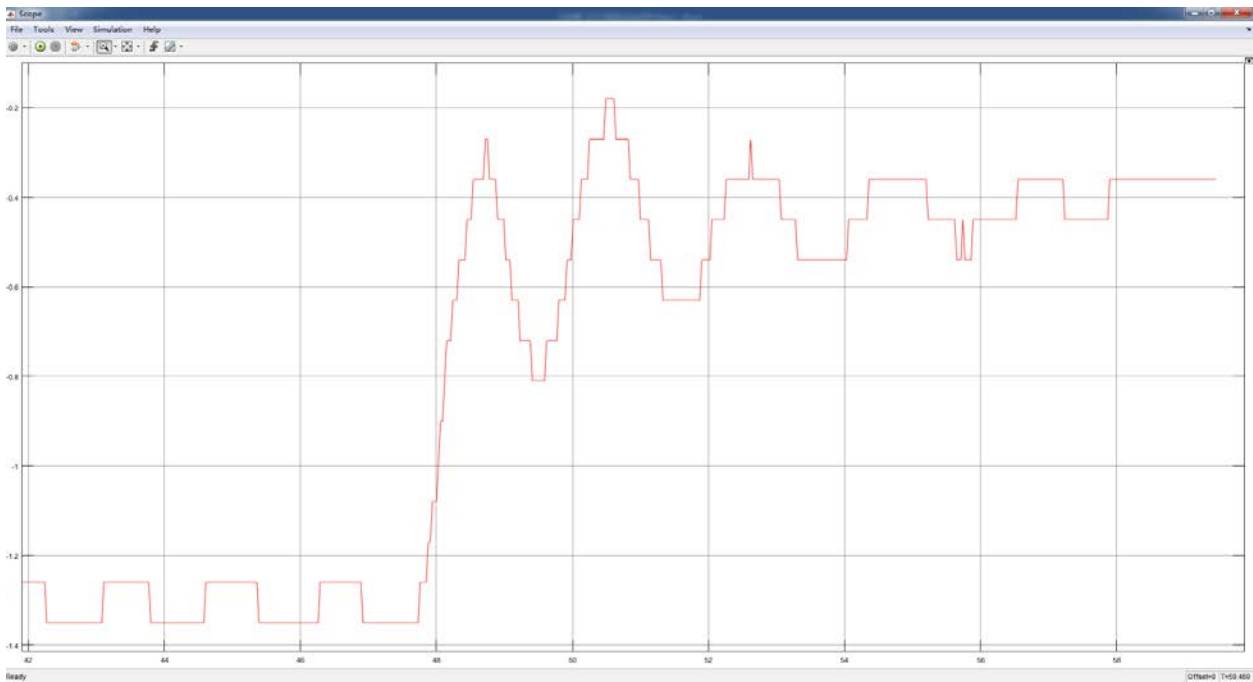
- ① Pitch Ref 及 Slider1 模块：设置目标俯仰角度；
- ② PID Controller1 模块：设计的 PID 校正环节；
- ③ Real Time Control 模块：详见第二章 2.4 节中的 1

④ 蓝色模块均为运动控制卡模块，详见第二章

3) 双击“PID Controller1”模块，设置 PID 参数为：Kp=1.1、Ki=0、Kd=20。



- 4) 点击“”运行程序，电机开始转动，控制俯仰角度在 0 度左右待其稳定后，设置 Slider1 模块值为 1，任其运行 100 秒钟左右，然后点击  停止程序
- 5) 双击 Scope，打开示波器观察四旋翼俯仰角度曲线。



(五)实验记录

项目	控制器参数	仿真系统	实时控制系统
未校正系统	/	不稳定	/
PID 控制器	Kp=200、Ki=10、Kd=10	系统稳定，调节时间 0.169s，稳态误差 0.002，超调量为 0	系统震荡
	Kp=1.1、Ki=0、Kd=20	系统稳定，调节时间 71.2s，稳态误差 0，超调量为 0	系统稳定，调节时间 10.1s，稳态误差 1.36，超

			调量为 50%
--	--	--	---------

(六)实验分析及思考

1. PID 环节从根轨迹角度讲，是给开环系统增加了两个位置可变的零点和一个位于原点的极点。可以尝试采用根轨迹分析法实现对系统的稳定控制
2. 从实测的阶跃响应曲线可以看出，系统的实际性能指标和仿真的性能指标有差异，产生误差的主要原因是什么？

答案：

- a) 输入电压和力之间不一定是线性关系；
- b) 螺旋桨正反转不对称，造成相同电压对应正反转的升力不同；
- c) 由于电机是开环控制，电机的摩擦力矩造成系统必定会有电压死区；
- d) 系统三个姿态角不可能处于绝对的水平零度，各个姿态角运动时会有干涉。

第六章 四旋翼仿真系统 LQR 控制

6.1 四旋翼 LQR 六状态控制

(一)实验目的

1. 掌握 LQR 控制器设计方法；
2. 设计 LQR 控制器控制四旋翼飞行仿真器姿态；

(二)实验设备

1. GHP3001 四旋翼飞行仿真器；
2. PC (MATLAB) 平台；

(三)实验原理

1. 建立六状态状态方程

选定俯仰角、俯仰角速度、滚动角、滚动角速度、偏航角、偏航角速度为系统六个状态，系统输入量为前、后、左、右四个电机的电压，则系统状态方程（详见第三章实验原理第二节）为：

$$\begin{bmatrix} \dot{P} \\ \dot{P} \\ \dot{R} \\ \dot{R} \\ \dot{Y} \\ \dot{Y} \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & - \end{bmatrix} \begin{bmatrix} P \\ \dot{P} \\ R \\ \dot{R} \\ Y \\ \dot{Y} \end{bmatrix} + \begin{bmatrix} 0 & 0 & 0 & 0 \\ -8.3848 & -4.1151 & -4.1151 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 16.6667 & -16.6667 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 5.9703 \end{bmatrix} \begin{bmatrix} V_f \\ V_l \\ V_r \\ V_b \end{bmatrix} \quad (6-1)$$

$$\begin{bmatrix} P \\ R \\ Y \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} P \\ \dot{P} \\ R \\ \dot{R} \\ Y \\ \dot{Y} \end{bmatrix} + \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} V_f \\ V_l \\ V_r \\ V_b \end{bmatrix} \quad (6-2)$$

2. 线性二次型最优控制器

线性二次型 (LQ—Linear Quadratic) 是指系统的状态方程是线性的，指标函数是状态变量和控制变量的二次型。线性二次型控制理论已成为反馈系统设计的一种重要工具。其特点是为多变量反馈系统的设计提供了一种有效的分析方法，可以适应于时变系统，可以处理扰动信号和测量噪声问题，可以处理有限和无限的时间区间，设计的闭环系统具有良好的稳定裕度。

1) LQR 线性二次型最优控制器原理

考虑系统状态方程式 (6-1) (6-2) 所示，找一状态反馈控制率

$$u = -Kx$$

使得如下性能指标最小化：

$$J = \frac{1}{2} \int_0^T f[x^T Qx + u^T Ru] dt + \frac{1}{2} x^T(t_f) Mx(t_f)$$

其中 Q、M 是半正定矩阵，R 是正定矩阵，Q、M 分别是对状态变量和输入向量的加权矩阵，x 是 n 维状态变量，u 是 m 维输入变量，终端时间 t_f ，终端状态 $x(t_f)$ 。要使性能函数 J 最小，则首先构造一个 Hamilton 函数：

$$H = -\frac{1}{2} [x^T Qx + u^T Ru] + \lambda^T [Ax + Bu]$$

当输入信号不受约束时，则可对 Hamilton 函数进行求导并令其值为 0，求出最小值。

$$\frac{\partial H}{\partial u} = -Ru + B^T \lambda = 0$$

从而得到最优控制信号

$$u^* = R^{-1}B^T \lambda$$

λ 可由下式求出

$$\lambda = -Px$$

P 可由 Riccati 方程求出

$$P = -PA - A^T P + PBR^{-1}B^T P - Q$$

当 $t_f \rightarrow \infty$ 时, P 趋近于一个常值矩阵, 且 $\dot{P}=0$, 因此上式给出的 Riccati 方程就化简为:

$$-PA - A^T P + PBR^{-1}B^T P - Q = 0$$

LQR 问题的解是一个全状态反馈调节器, 有非常好的鲁棒性能。他可以通过适当选取权重矩阵 Q 和 R, 在控制信号能量和输出性能之间进行调节。

利用 MATLAB 的 LQR 函数可以方便、准确的求出 LQR 的解-----K 阵。

2) LQR 线性二次型最优调节器的计算

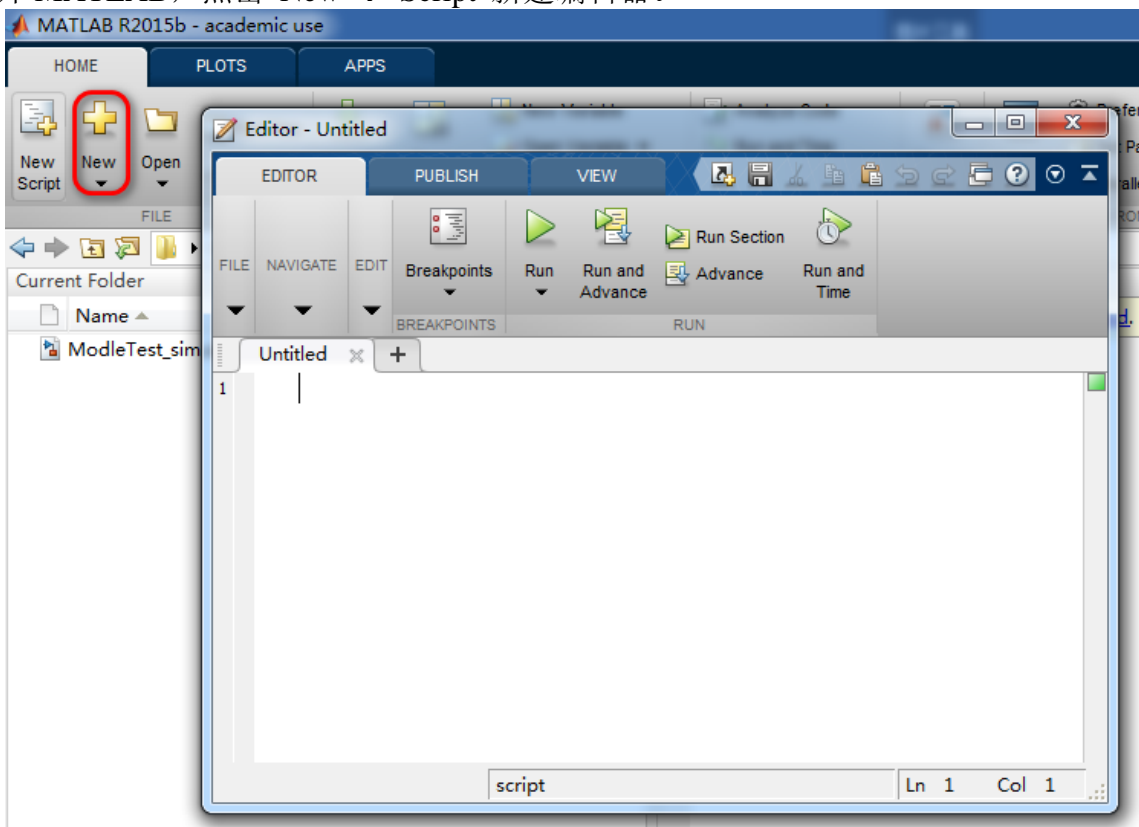
LQR 线性二次型最优调节器通过选取权矩阵 Q 和 R 求解的全状态反馈阵 K 与系统的性能指标联系起来。其实质是设计最优的全状态反馈阵。从而比全状态反馈调节器更易设计出性能鲁棒性更强的控制系统。

(四)实验步骤

1. 加 LQR 校正后系统仿真实验

在 Matlab 中构件下述 M 文件, 可以测量系统阶跃响应

1) 打开 MATLAB, 点击“New”→“Script”新建编辑器。



2) 在编辑器中输入如下代码, 并保存为 Simulink_State_6.m。

```
clear;
```

```
clc;
```

```
lf = 0.51;
```

```
lrp= 0.25
```

```
lp = 0.52 ;
```

```
lr= 0.45;
```

Ma=1;
Mp=0.984;

$$\begin{aligned} J_p &= 2 * Ma * l_f * l_f + 2 * Ma * l_{rp} * l_{rp} + M_p * l_p * l_p \\ J_r &= 2 * Ma * l_r * l_r \\ J_y &= 2 * Ma * l_f * l_f + 2 * Ma * l_{rp} * l_{rp} + M_p * l_p * l_p \end{aligned}$$

```

Kc=15;
A=[ 0 1 0 0 0 0 ;
    0 0 0 0 0 0 ;
    0 0 0 1 0 0 ;
    0 0 0 0 0 0 ;
    0 0 0 0 0 1 ;
    0 0 0 0 0 0 ;]
B=[ 0 0 0 0;
    -(Kc*lf)/Jp -(Kc*lrp)/(Jp) -(Kc*lrp)/(Jp) 0;
    0 0 0 0;
    0 (Kc*lr)/(Jr) -(Kc*lr)/(Jr) 0;
    0 0 0 0;
    0 0 0 (Kc*ly)/Jy;]

```


$$C = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$
$$D = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix};$$

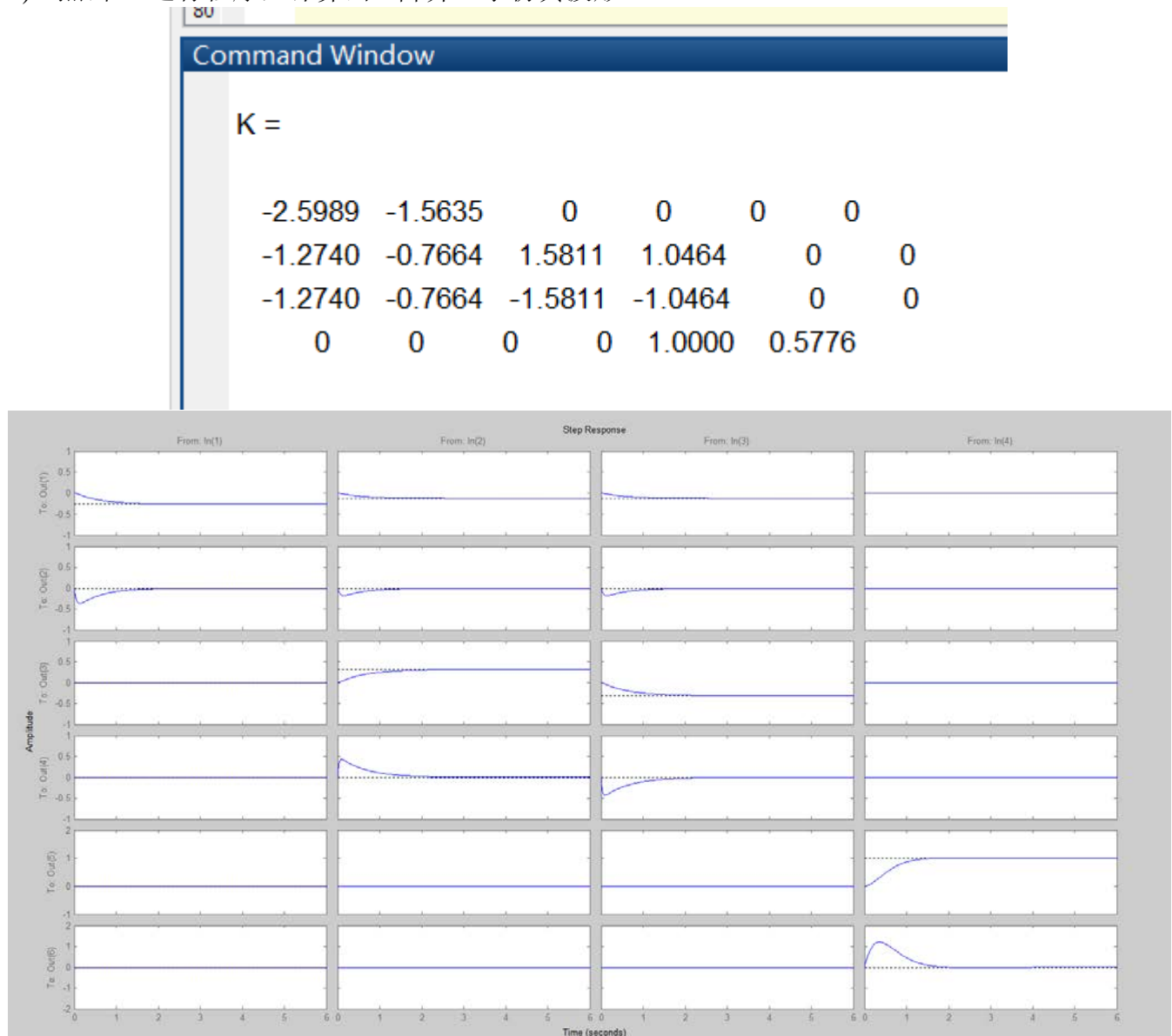
```
Q11=10;Q22=3;Q33=5; Q44=2;Q55=1;Q66=0.1;
Q=[Q11 0 0 0 0 0;
    0 Q22 0 0 0 0;
    0 0 Q33 0 0 0;
    0 0 0 Q44 0 0;
    0 0 0 0 Q55 0;
    0 0 0 0 0 Q66];
```

$$R = \begin{bmatrix} 1 & 0 & 0 & 0; \\ 0 & 1 & 0 & 0; \\ 0 & 0 & 1 & 0; \\ 0 & 0 & 0 & 1 \end{bmatrix};$$

```
K = lqr2(A,B,Q,R)
Ac=(A-B*K);
Bc=B;
Cc=C;
```

Dc=D;
 sys=ss(Ac,Bc,Cc,Dc);
 step(sys)

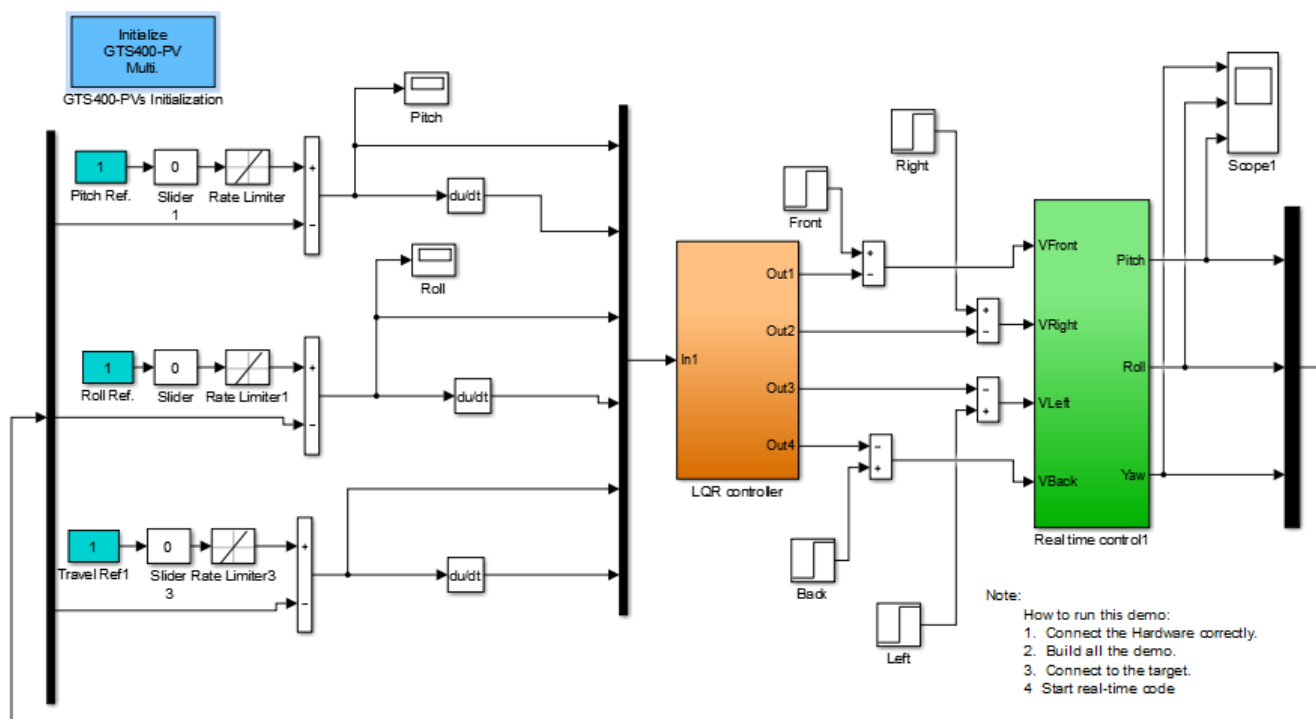
3) 点击  运行程序，计算出 k 阵并显示仿真波形



2. 加 LQR 校正实时控制

- 1) 打开四旋翼飞行平台电源（空开上电、松开手持急停及按下绿色启动按钮，此时手持急停指示红绿灯均亮起）；
- 2) 打开 MATLAB，在 MATLAB 命令行中输入 gtbox，打开 gtbox 工具箱中的“4 Hover”→“GTS”→“1-6 States”下的“FH_LQR_State_6_P.slx”，弹出如下实时控制界面

Googol 6 state 4RotorHover LQR Control Demo Position mode of Yaw axis




a) 实时控制程序设计简介

输入状态方程的六个状态变量到 LQR 控制器，控制器输出各轴电压，来控制四个螺旋桨的旋转方向及旋转速度，确保四旋翼姿态和目标姿态一致。

b) 程序模块说明

- ① Pitch Ref、Slider1 及 Rate Limiter 模块为设定目标俯仰角；
- ② Roll Ref、Slider 及 Rate Limiter1 模块为设定目标滚动角；
- ③ Travel Ref1、Slider3 及 Rate Limiter3 模块为设定目标偏航角；
- ④ LQR controller 模块：LQR 控制器；
- ⑤ Front、Right、Back 及 Left 模块配合后面 Sum 模块使控制器输出反向；
- ⑥ Real time control1 模块：详见第二章 2.4 节中 2。

3) 用手把四旋翼扶至水平位置左右（注意：小心螺旋桨旋转打到手，随时准备摀下手持急

停，以防发生人身伤害），电机“”运行程序，等待程序编译运行，程序运行后（运行程序 matlab 菜单栏所有图标均会变灰色，等待变回彩色即程序运行），松开扶着的手，此时，四旋翼控制在初始位置，双击“Scope1”可查看三个姿态（俯仰、翻滚及偏航）的实时波形

4) 测试完成后，点击“”停止运行程序。

5) 系统稳定，俯仰角稳态误差为，滚动角稳态误差为，偏航角稳态误差为。

(五)实验记录

项目	控制器参数	仿真系统稳定性	实时控制系统稳定性
未校正系统	/	不稳定	/
LQR 控制器	$K =$ -2.5989 -1.5635 0 0 0 0 -1.2740 -0.7664 1.5811 1.0464 0 0 -1.2740 -0.7664 -1.5811 -1.0464 0 0 0 0 0 0 1.0000 0.5776	稳定	稳定

(六)实验分析及思考

4. 从实测的阶跃响应曲线可以看出，系统的实际性能指标和仿真的性能指标有差异，产生误差的原因主要原因是什么？

答案：

- e) 输入电压和力之间不一定是线性关系；
- f) 螺旋桨正反转不对称，造成相同电压对应正反转的升力不同；
- g) 由于电机是开环控制，电机的摩擦力矩造成系统必定会有电压死区；
- h) 系统三个姿态角不可能处于绝对的水平零度，各个姿态角运动时会有干涉。

6.2 四旋翼 LQR 九状态控制

(一)实验目的

1. 掌握 LQR 控制器设计方法；
2. 设计 LQR 控制器控制四旋翼飞行仿真器姿态；

(二)实验设备

1. GHP3001 四旋翼飞行仿真器；
2. PC (MATLAB) 平台；

(三)实验原理

1. 建立九状态状态方程

选定俯仰角、俯仰角速度、俯仰角度的积分、滚动角、滚动角速度、滚动角度的积分、偏航角、偏航角速度、偏航角度的积分为系统九个状态，系统输入量为：前、后、左、右四个电机的电压，则状态方程为：

$$\begin{bmatrix} \dot{P} \\ \dot{P} \\ P \\ \dot{R} \\ \ddot{R} \\ R \\ \dot{Y} \\ \ddot{Y} \\ Y \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \end{bmatrix} \begin{bmatrix} P \\ \dot{P} \\ P_a \\ R \\ \dot{R} \\ R_a \\ Y \\ \dot{Y} \\ Y_a \end{bmatrix} + \begin{bmatrix} 0 & 0 & 0 & 0 \\ -8.3948 & -4.1151 & -4.1151 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 16.6667 & -16.6667 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 5.9703 \\ 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} V_f \\ V_l \\ V_r \\ V_b \end{bmatrix}$$

$$\begin{bmatrix} P \\ R \\ Y \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \end{bmatrix} \begin{bmatrix} P \\ \dot{P} \\ P_a \\ R \\ \dot{R} \\ R_a \\ Y \\ \dot{Y} \\ Y_a \end{bmatrix} + \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} V_f \\ V_l \\ V_r \\ V_b \end{bmatrix}$$

2. 线性二次型最优控制器

线性二次型 (LQ—Linear Quadratic) 是指系统的状态方程是线性的，指标函数是状态变量和控制变量的二次型。线性二次型控制理论已成为反馈系统设计的一种重要工具。其特点是为多变量反馈系统的设计提供了一种有效的分析方法，可以适应于时变系统，可以处理扰动信号和测量噪声问题，可以处理有限和无限的时间区间，设计的闭环系统具有良好的稳定裕度。

1) LQR 线性二次型最优控制器原理

考虑系统状态方程式 (6-1) (6-2) 所示，找一状态反馈控制率

$$u = -Kx$$

使得如下性能指标最小化：

$$J = \frac{1}{2} \int_0^T f[x^T Q x + u^T R u] dt + \frac{1}{2} x^T(t_f) M x(t_f)$$

其中 Q、M 是半正定矩阵，R 是正定矩阵，Q、M 分别是对状态变量和输入向量的加权矩阵，x 是 n 维状态变量，u 是 m 维输入变量，终端时间 t_f ，终端状态 $x(t_f)$ 。要使性能函数 J

最小，则首先构造一个 Hamilton 函数：

$$H = -\frac{1}{2}[x^T Q x + u^T R u] + \lambda^T [Ax + Bu]$$

当输入信号不受约束时，则可对 Hamilton 函数进行求导并令其值为 0，求出最小值。

$$\frac{\partial H}{\partial u} = -Ru + B^T \lambda = 0$$

从而得到最优控制信号

$$u^* = R^{-1} B^T \lambda$$

λ 可由下式求出

$$\lambda = -Px$$

P 可由 Riccati 方程求出

$$P = -PA - A^T P + PBR^{-1}B^T P - Q$$

当 $t_f \rightarrow \infty$ 时，P 趋近于一个常值矩阵，且 $\dot{P}=0$ ，因此上式给出的 Riccati 方程就简化为：

$$-PA - A^T P + PBR^{-1}B^T P - Q = 0$$

LQR 问题的解是一个全状态反馈调节器，有非常好的鲁棒性能。他可以通过适当选取权重矩阵 Q 和 R，在控制信号能量和输出性能之间进行调节。

利用 MATLAB 的 LQR 函数可以方便、准确的求出 LQR 的解-----K 阵。

2) LQR 线性二次型最优调节器的计算

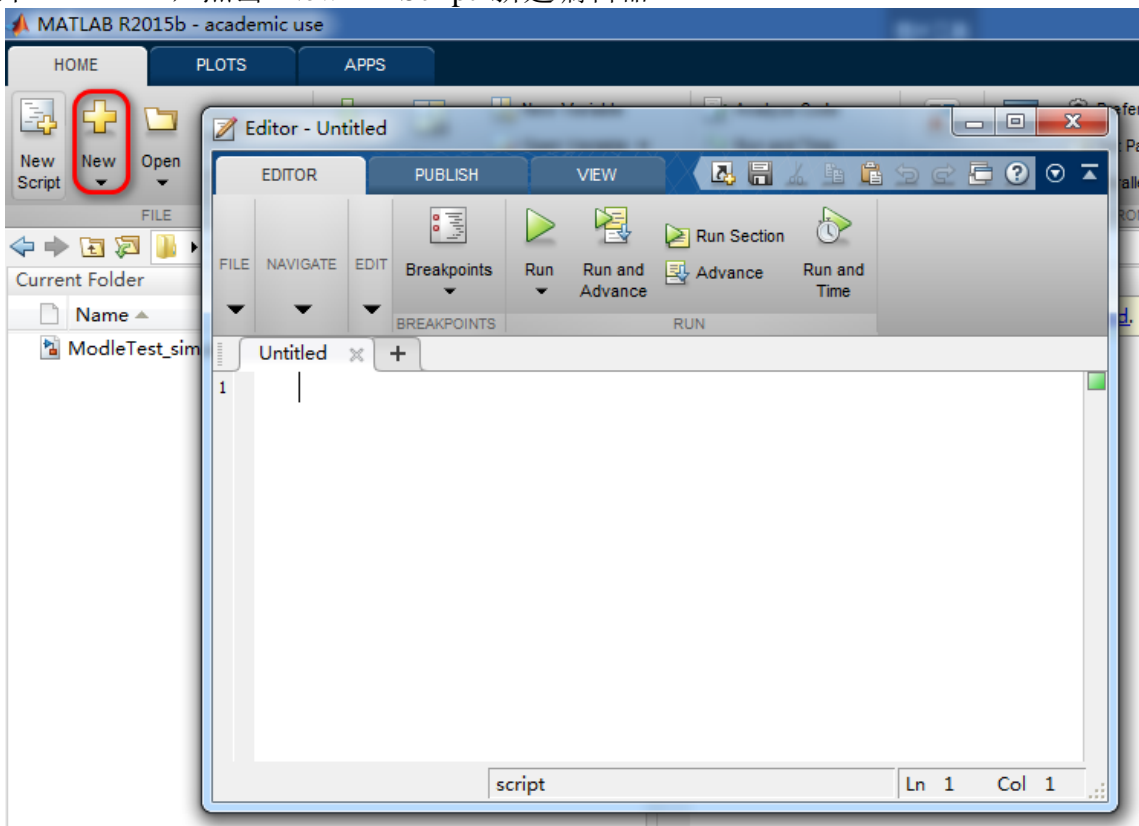
LQR 线性二次型最优调节器通过选取权矩阵 Q 和 R 求解的全状态反馈阵 K 与系统的性能指标联系起来。其实质是设计最优的全状态反馈阵。从而比全状态反馈调节器更易设计出性能鲁棒性更强的控制系统。

(四)实验步骤

1. 加 LQR 校正后系统仿真

在 Matlab 中构件下述 M 文件，可以测量系统阶跃响应

1) 打开 MATLAB，点击“New”→“Script”新建编辑器。



2) 在编辑器中输入如下代码，并保存为 Simulink_State_9.m

```
clear;
clc;

If = 0.51;
lrp = 0.25
lp = 0.52;
lr= 0.45;
ly = 0.52;

Ma=1;
Mp=0.984;

Jp=2*Ma*lf*lf+2*Ma*lrp*lrp+Mp*lp*lp
Jr=2*Ma*lr*lr
Jy=2*Ma*lf*lf+2*Ma*lrp*lrp+Mp*lp*lp

Kc =15;
A=[ 0 1 0 0 0 0 0 0 0;
    0 0 0 0 0 0 0 0 0;
    1 0 0 0 0 0 0 0 0;
    0 0 0 0 1 0 0 0 0;
    0 0 0 0 0 0 0 0 0;
    0 0 0 1 0 0 0 0 0;
    0 0 0 0 0 0 0 1 0;
    0 0 0 0 0 0 0 0 0;
    0 0 0 0 0 0 1 0 0]
B=[ 0 0 0 0;
    -(Kc*lf)/Jp -(Kc*lrp)/(Jp) -(Kc*lr)/(Jp) 0;
    0 0 0 0;
    0 0 0 0;
    0 (Kc*lr)/(Jr) -(Kc*lr)/(Jr) 0;
    0 0 0 0;
    0 0 0 (Kc*ly)/Jy;
    0 0 0 0;
    0 0 0 0]

C=[ 1 0 0 0 0 0 0 0 0;
    0 1 0 0 0 0 0 0 0;
    0 0 1 0 0 0 0 0 0;
    0 0 0 1 0 0 0 0 0;
    0 0 0 0 1 0 0 0 0;
    0 0 0 0 0 1 0 0 0;
    0 0 0 0 0 0 1 0 0;
    0 0 0 0 0 0 0 1 0;
    0 0 0 0 0 0 0 0 1]

D=[0 0 0 0;
    0 0 0 0;
    0 0 0 0;
    0 0 0 0;
    0 0 0 0;
    0 0 0 0;
    0 0 0 0;
    0 0 0 0;
    0 0 0 0];

Q11=10;Q22=2;Q33=0.2; Q44=10;Q55=0.5;Q66=0.1;Q77=0.8;Q88=0.6;Q99=0.01;
Q=[Q11 0 0 0 0 0 0 0 0;
    0 Q22 0 0 0 0 0 0 0;
    0 0 Q33 0 0 0 0 0 0;
    0 0 0 Q44 0 0 0 0 0;
    0 0 0 0 Q55 0 0 0 0;
```

```

00000 Q66000;
000000 Q7700;
0000000 Q880;
00000000 Q99];

```

```

R = [ 1 0 0 0;
      0 1 0 0;
      0 0 1 0;
      0 0 0 1];

```

```

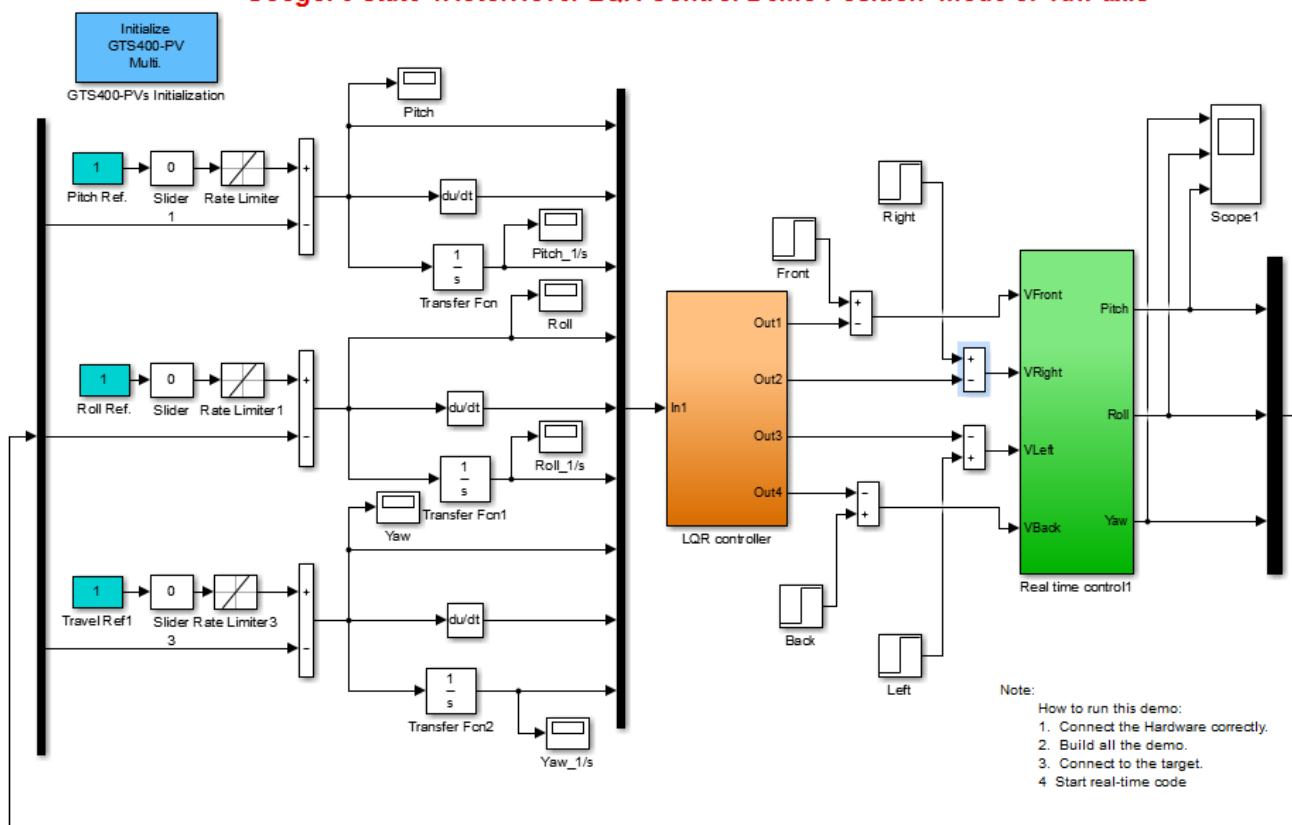
K = lqr2(A,B,Q,R)
Ac=(A-B*K);
Bc=B;
Cc=C;
Dc=D;
sys=ss(Ac,Bc,Cc,Dc);
step(sys)

```

2. 加 LQR 校正实时控制

- 1) 打开四旋翼飞行平台电源（空开上电、松开手持急停及按下绿色启动按钮，此时手持急停指示红绿灯均亮起）；
- 2) 打开 MATLAB，在 MATLAB 命令行中输入 gtbox，打开 gtbox 工具箱中的“4 Hover”→“GTS”→“1-9 States”下的“FH_LQR_State_9_P.slx”，弹出如下实时控制界面

Googol 9 state 4RotorHover LQR Control Demo Position mode of Yaw axis




a) 实时控制程序设计简介


输入状态方程的九个状态变量到 LQR 控制器，控制器输出各轴电压，来控制四个螺旋桨的旋转方向及旋转速度，确保四旋翼姿态和目标姿态一致。

b) 程序模块说明

- ① Pitch Ref、Slider1 及 Rate Limiter 模块为设定目标俯仰角；
- ② Roll Ref、Slider 及 Rate Limiter1 模块为设定目标滚动角；
- ③ Travel Ref1、Slider3 及 Rate Limiter3 模块为设定目标偏航角；
- ④ LQR controller 模块：LQR 控制器；

- ⑤ Front、Right、Back 及 Left 模块配合后面 Sum 模块使控制器输出反向；
 ⑥ Real time control1 模块：详见第二章 2.4 节中 2。
 3) 用手把四旋翼扶至水平位置左右（**注意：小心螺旋桨旋转打到手，随时准备摀下手持急**

停，以防发生人身伤害），电机“”运行程序，等待程序编译运行，程序运行后（运行程序 matlab 菜单栏所有图标均会变灰色，等待变回彩色即程序运行），松开扶着的手，此时，四旋翼控制在初始位置，双击“Scope1”可查看三个姿态（俯仰、翻滚及偏航）的实时波形

4) 测试完成后，点击“”停止运行程序。

5) 系统稳定，俯仰角稳态误差为，滚动角稳态误差为，偏航角稳态误差为。

(五)实验记录

项目	控制器参数	仿真系统稳定性	实时控制系统稳定性
未校正系统	/	不稳定	/
LQR 控制器	$K =$ -2.5989 -1.5635 0 0 0 0 -1.2740 -0.7664 1.5811 1.0464 0 0 -1.2740 -0.7664 -1.5811 -1.0464 0 0 0 0 0 0 1.0000 0.5776	稳定	稳定

(六)实验分析及思考

1. 从实测的阶跃响应曲线可以看出，系统的实际性能指标和仿真的性能指标有差异，产生误差的原因主要原因是什么？

答案：

- 输入电压和力之间不一定是线性关系；
- 螺旋桨正反转不对称，造成相同电压对应正反转的升力不同；
- 由于电机是开环控制，电机的摩擦力矩造成系统必定会有电压死区；
- 系统三个姿态角不可能处于绝对的水平零度，各个姿态角运动时会有干涉。