# 控制原理

# 实验指导书

时间安排：

第1次：Matlab和simulink基础；（无作业）

第2次：实验板和一阶环节搭建；（有作业）

第3次：二阶及高阶换件搭建及数据测量；（有作业）

第4次：系统频域分析——概念讲解（无作业）

第5次：系统频域分析——操作实践（有作业）

第6次：PID项目（一）（无作业）

第7次：PID项目（二）（有作业）

第8次：倒立摆编程设计（大作业，报告）

## 实验一 MATLAB及Simulink基础

### MATLAB基础

MATLAB（Matrix Laboratory）具有强大的矩阵计算功能和良好的图形可视化功能，在信号处理、图像处理、控制系统及神经网络等领域都有广泛的应用。

MATLAB提供了很多专用的工具箱，如控制系统工具箱（Control System Toolbox）、信号处理工具箱（Signal Procession Toolbox）。控制系统工具箱主要是运用经典控制理论处理线性时不变（LTI）系统的函数集合，为LTI定常系统的建模、分析和设计提供了完整的解决方案。为完成本课程的软件模拟请安装上述两个工具箱。

#### 变量

##### 创建变量

变量命名规则：

* 由字母、数字、下划线构成
* 以字母开头
* 大小写敏感
* 变量名应具有意义

scalar1 = 1 % 虽然是标量，查看工作区可以看到其大小为1x1的矩阵

scalar1 = 1

vector2 = [4,5,6] % 行向量：可用逗号或空格分隔行向量中的元素

vector2 = 1×3

4 5 6

vector3 = [4;5;6] % 列向量：使用分号分隔列向量中的元素

vector3 = 3×1

4

5

6

matrix4 = [10 20 30; 40 50 60;70 80 90] % 矩阵，包含行和列；使用空格或逗号分隔一行中的元素；使用分号分隔不同的行

matrix4 = 3×3

10 20 30

40 50 60

70 80 90

##### 变量及元素的调用

使用变量名或索引调用变量或变量中的元素。

Matlab中索引的使用，需注意：

* Matlab的索引是从1开始的；
* 关键字end可来表示最后一个索引；
* 索引值由小括号（）括起来。

vector2 % 变量的引用

vector2 = 1×3

4 5 6

vector2(1) % 元素的引用

ans = 4

matrix4 % 矩阵

matrix4 = 3×3

10 20 30

40 50 60

70 80 90

matrix4(2,2) % 行列索引，位于第2行，第2列的元素

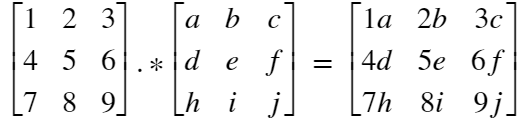
ans = 50

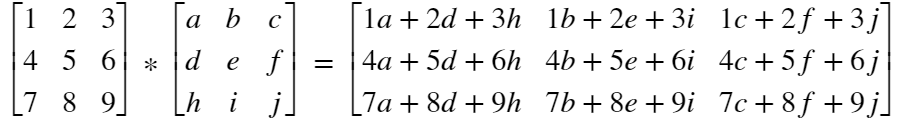
matrix4(4) % 元素索引，按列的方向计数

ans = 20

#### 矩阵运算







matrix1 = [1:3;4:6;7:9] % 生成矩阵

matrix1 = 3×3

1 2 3

4 5 6

7 8 9

matrix2 = ones(3,3) % 生成都是1的3x3矩阵

matrix2 = 3×3

1 1 1

1 1 1

1 1 1

matrix1.\*matrix2 % 点乘操作

ans = 3×3

1 2 3

4 5 6

7 8 9

matrix1\*matrix2 % 矩阵乘法

ans = 3×3

6 6 6

15 15 15

24 24 24

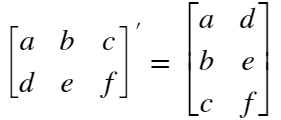
matrix2\*matrix1 % 矩阵乘法的两个矩阵变换位置后，结果会不同

ans = 3×3

12 15 18

12 15 18

12 15 18



matrix3 = [1 2 3; 4 5 6] % 实数矩阵

matrix3 = 2×3

1 2 3

4 5 6

matrix3' % 实数矩阵转置

ans = 3×2

1 4

2 5

3 6

matrix4 = matrix3+1j\*9 % 复数矩阵

matrix4 = 2×3 complex

1.0000 + 9.0000i 2.0000 + 9.0000i 3.0000 + 9.0000i

4.0000 + 9.0000i 5.0000 + 9.0000i 6.0000 + 9.0000i

matrix4' % 复数矩阵不带点转置，会同时执行共轭操作

ans = 3×2 complex

1.0000 - 9.0000i 4.0000 - 9.0000i

2.0000 - 9.0000i 5.0000 - 9.0000i

3.0000 - 9.0000i 6.0000 - 9.0000i

matrix4.' % 复数矩阵点转置

ans = 3×2 complex

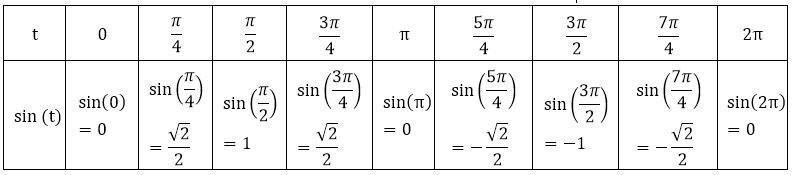
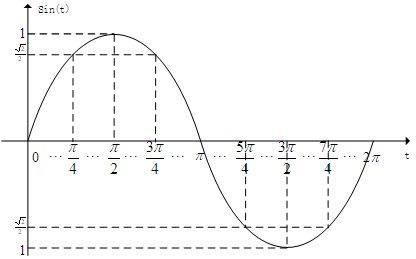
1.0000 + 9.0000i 4.0000 + 9.0000i

2.0000 + 9.0000i 5.0000 + 9.0000i

3.0000 + 9.0000i 6.0000 + 9.0000i

#### Creat Singnals（信号创建）

##### Create a signal with numeric method



clear;

t = 0:pi/4:2\*pi % 先定义时间轴（横轴）

t = 1×9

0 0.7854 1.5708 2.3562 3.1416 3.9270 4.7124 ⋯

sigt = sin(t) % 根据时间轴算出对应函数值。时间轴、函数值一一对应

sigt = 1×9

0 0.7071 1.0000 0.7071 0.0000 -0.7071 -1.0000 ⋯

##### Create a signal with symbolic method

clear;

syms x % 定义符号变量

y = sin(2\*pi\*x) % 写出符号表达式

y = IMG_256

#### 绘图

##### plot & stem

* plot用于绘制数值法创建的连续信号
* stem用于绘制数值法创建的时间离散信号

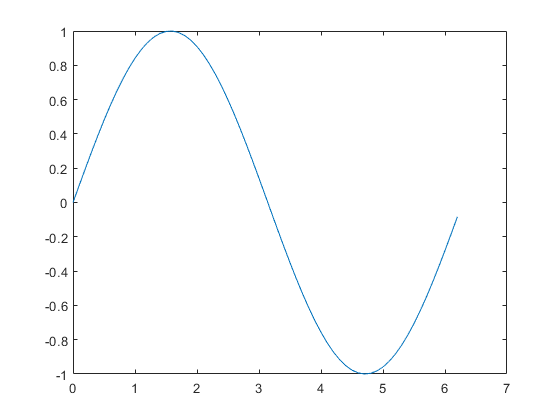
clear;

t = 0:0.1:2\*pi-0.01; % 先定义时间轴（横轴）

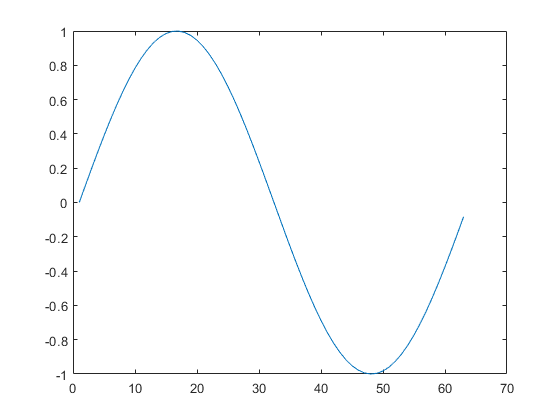
sigx = sin(t); % 根据时间轴算出函数值

sigy = cos(t);

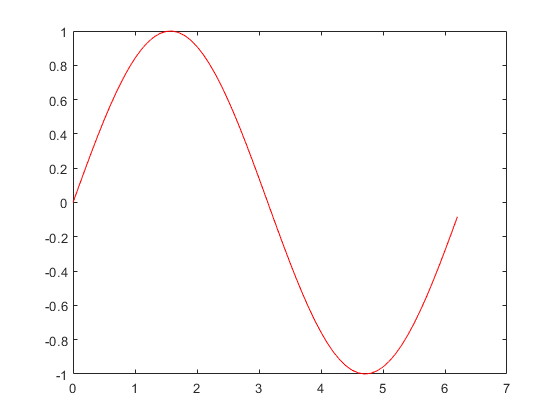
plot(t,sigx); % 绘制连续信号图形，横轴坐标为时间



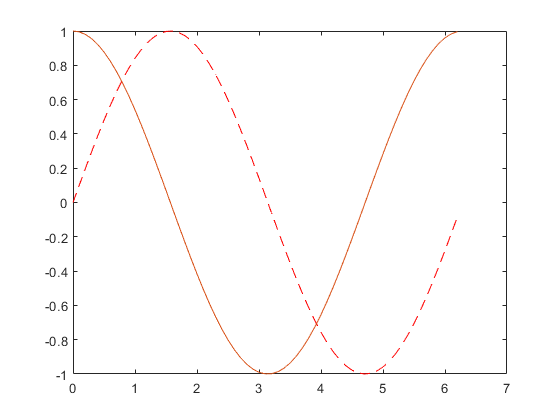
plot(sigx); % 未给出时间轴，横轴为索引



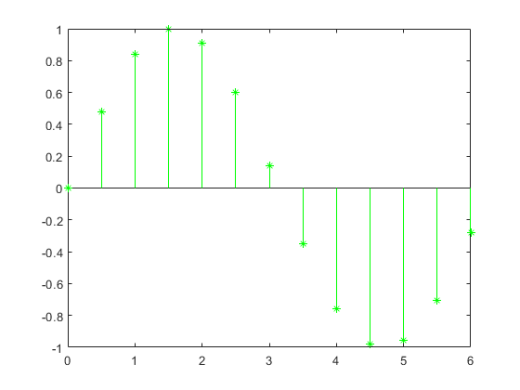
plot(t,sigx, '-r'); % 改变绘图的颜色（'r'）和线型（'-'）



plot(t,sigx,'--r',t,sigy); % 将2个信号绘制在同一幅图中



stem(t(1:5:end),sigx(1:5:end),'\* g') % 绘制时间离散信号



##### fplot

* 用于绘制符号法创建的信号

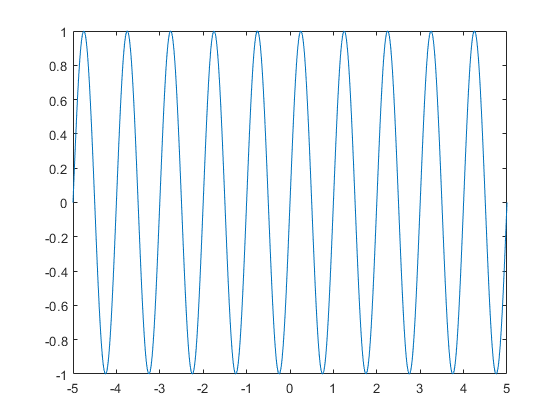
clf

syms x % 定义符号变量

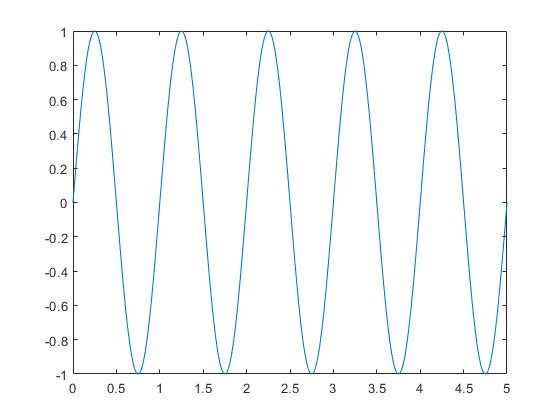
y = sin(2\*pi\*x)

y = IMG_256

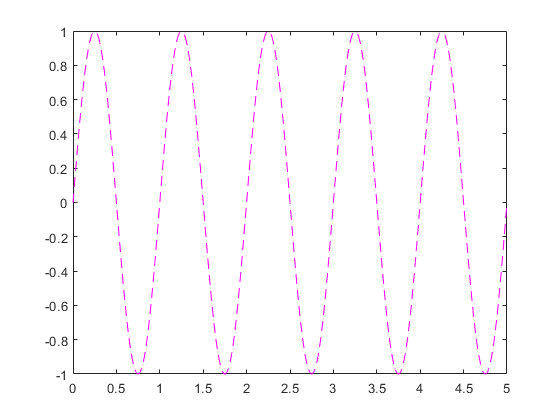
fplot(x,y);



fplot(y,[0 5]); % 第二个参数[0 5]限制了y的横轴范围，如省略，默认范围为[-5 5]



fplot(x,y,[0 5],"Color",'m',"LineStyle","--") % 修改图片颜色、线型等



##### subplot

* 创建子图，subplot(行数, 列数, 当前活动位置)

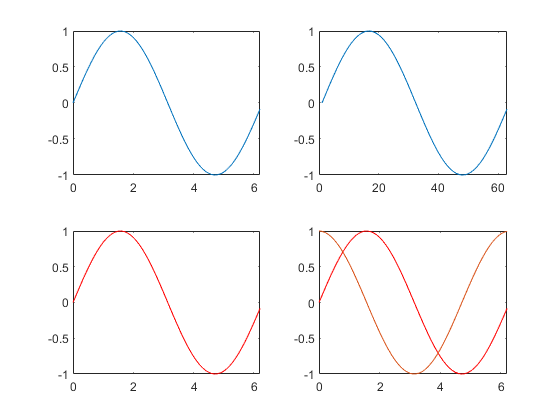
clf;

subplot(2,2,1); plot(t,sigx); % 创建2行2列的绘图区域，并在区域1上绘制(t,sigx)

subplot(2,2,2); plot(sigx);

subplot(2,2,3); plot(t,sigx,'-r');

subplot(2,2,4); plot(t,sigx,'-r',t,sigy);



##### 绘图标注

clf;

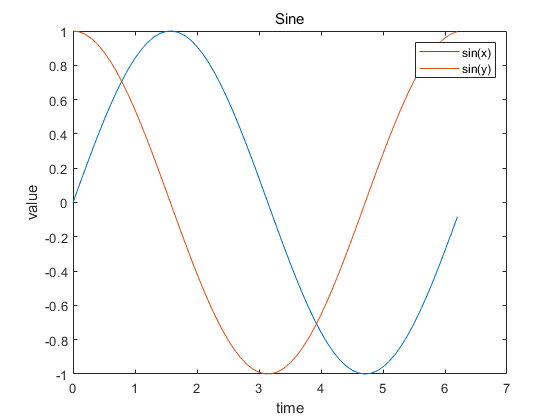
plot(t,sigx,t,sigy);

title('Sine'); % 图片标题

xlabel('time'); % 横轴性质，单位

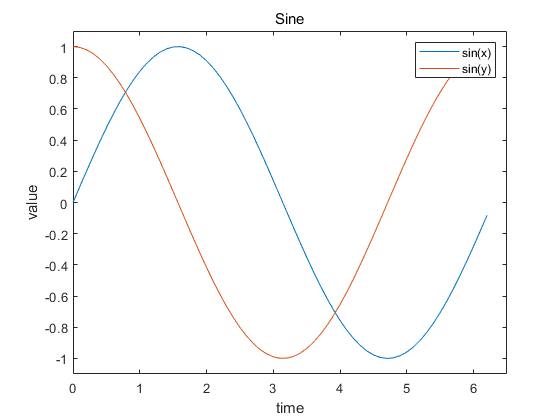
ylabel('value'); % 纵轴性质，单位

legend('sin(x)','sin(y)'); % 图片中有多个信号时，每个信号的说明



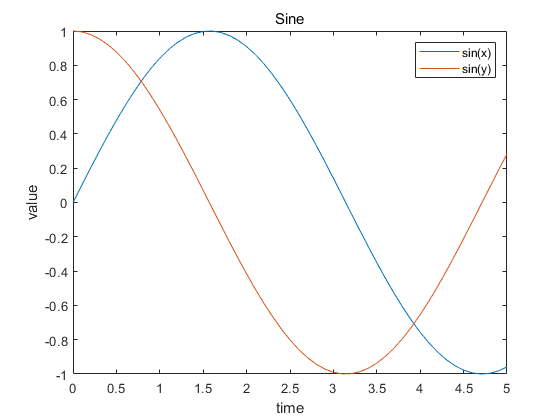
##### 显示范围设置

axis([0 6.5 -1.1 1.1]); % 使用数组的形式组织范围参数:[xmin xmax ymin ymax]



xlim([0 5]); % [xmin xmax] % 只限制x轴范围

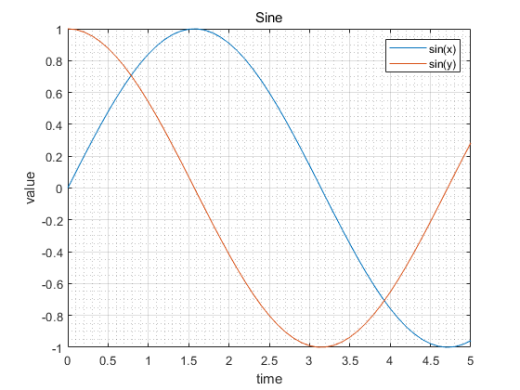
ylim([-1 1]); % [ymin ymax] % 只限制y轴范围



##### 网格

grid on; % 大网格

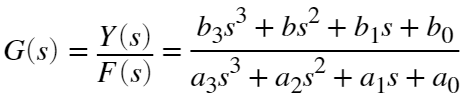
grid minor; % 更细致的网格



#### 自控相关函数

##### 传递函数的表示

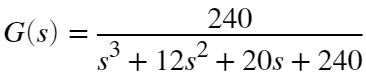




可使用函数来生成传递函数，其中





如：

clf;clear;

num = [200]; % 分子

den = [1,12,20,200]; % 分母

G = tf(num,den) % 生成传递函数

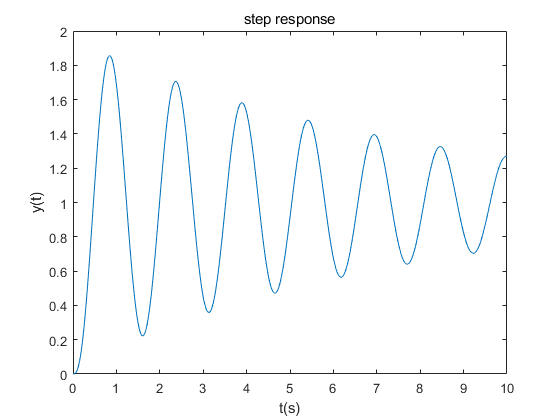
G = 200 ------------------------- s^3 + 12 s^2 + 20 s + 200 Continuous-time transfer function.

##### 单位阶跃响应：step

t = 0:0.01:10;

s = step(G,t);

plot(t,s); xlabel('t(s)');ylabel('y(t)');title('step response')

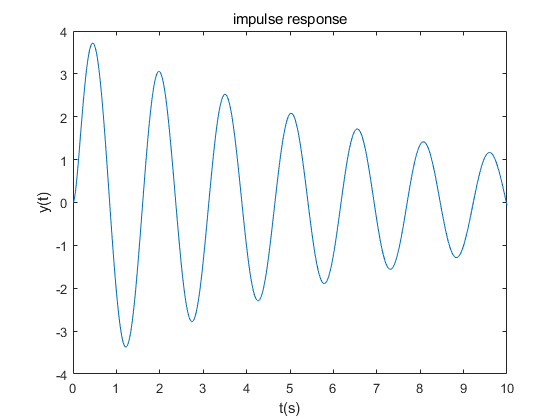


##### 冲击响应：impulse

t = 0:0.01:10;

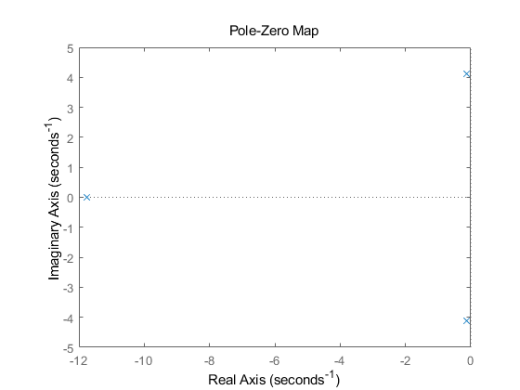
s = impulse(G,t);

plot(t,s); xlabel('t(s)');ylabel('y(t)');title('impulse response')



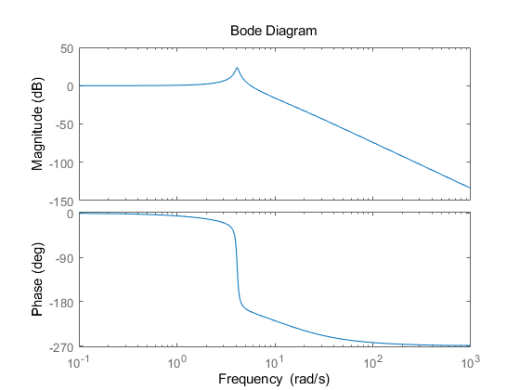
##### 零极点图

pzplot(G)



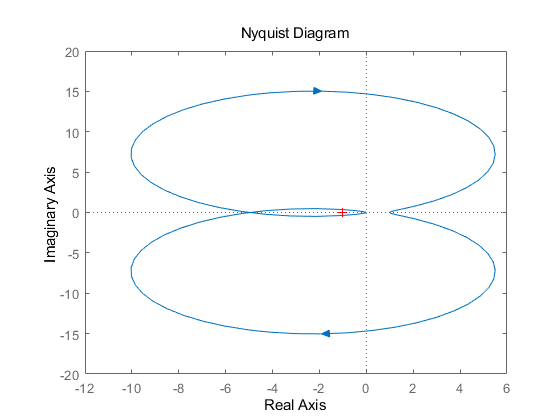
##### 伯德图

bode(G)



##### 奈奎斯特图

nyquist(G)

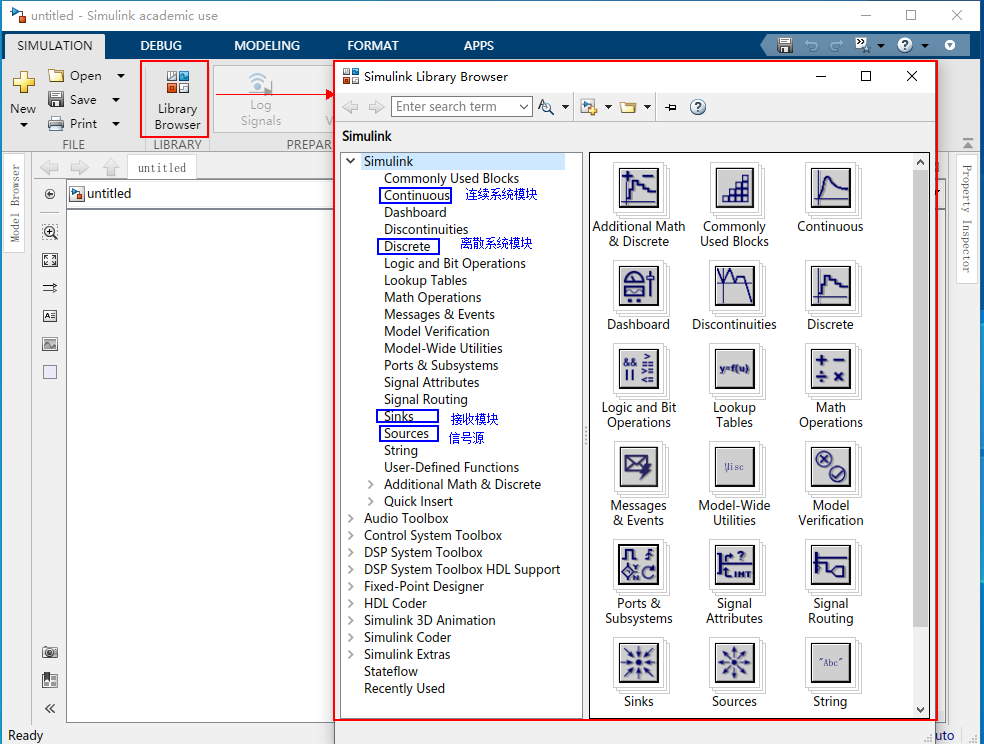


## Simulink基础

Simulink是MATLAB提供的一个动态仿真集成环境，是MATLAB的扩展。它为用户提供了一个用图形模块搭建系统动态模型的平台。

simulink % 启动simulink

本课程中涉及的常用模块如下，从模块库选定模块后将其拖拽至模型界面并进行设置及连线即可。



## 实验二 线性系统时域分析

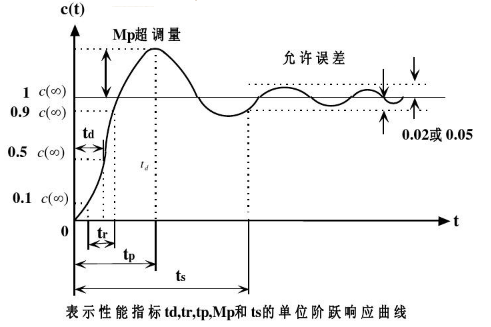
时域分析是指控制系统在一定输入下，根据输出量的时域表达式，分析系统的稳定性、瞬态（动态）性能、稳态性能等。系统输出量的时域表示可以由微分方程得到，也可以由传递函数得到（当初值为零时，通常利用传递函数进行研究）。

从对系统瞬态（动态）响应的分析中，可以了解到系统的稳定性、响应的速度、阻尼情况等信息。从对系统稳态响应的分析中，可以了解到系统最终输出量对输入量的跟随情况，以提供稳态误差信息。

为了求解系统的时间响应，必须了解输入信号的解析表达式。然而，控制系统的外加输入信号具有随机性，无法事先确定。因此需要选择若干典型信号作为输入信号，以研究线性控制系统在典型输入信号下的时间响应过程和性能指标。在控制系统中常用的典型输入信号有：阶跃信号、斜坡函数、抛物线、脉冲函数、正弦函数等。虽然不同的输入信号对应的输出响应是不同的，但对于线性系统而言它们表征的系统特性（传递函数）是一致的。而为了在一个统一的基础上对各类控制系统的性能进行比较，本事件将统一选用单位阶跃信号作为输入信号进行讲解，同学们在实际操作时可选用其它信号进行研究。

单位阶跃响应曲线性能指标如下图所示：

* 延迟时间：指响应曲线第一次达到其终值的一半所需的时间；
* 上升时间：指响应从终值的10%上升到90%所需要的时间；对于有振荡的系统，也可以定义为从零第一次上升到终值所需得时间；
* 峰值时间：指响应超过其终值到达第一个峰值所需要的时间；
* 调节时间：指响应到达并保持在（或）允许误差内所需要的时间；
* 超调量Mp（）：指响应的最大偏离量与终值的差与终值的百分比，即：



### 一阶系统时域分析

典型一阶系统可以是：比例（P）、积分（I）、比例积分（PI）、比例微分（PD）、惯性、比例积分微分（PID）等环节。

本实验中，我们将使用运算发达器、电阻、电容等原件来构建各类典型环节的模拟电路，写出各环节的传递函数，绘制结构图，观察并记录各种典型环节的阶跃响应曲线，测量各环节的特性参数。

#### 比例环节（P）

比例环节传递函数：

比例环节模拟电路图如下图所示：





图2-1 比例环节模拟电路

由运算放大器的虚短虚断性质可知，运放反向输入端电流为0；反向输入端电压同同向输入端，即。令参考电流方向如图中箭头所示。由基尔霍夫电流定律，可求得U101的传递函数为：

可以看出U101是一个反向比例放大器。

同理可求得U108的传递函数为：

可以看出U108是一个反相器。

因此图2-1所示的比例环节传递函数为：

使用MATLAB对比例环节进行仿真，获取时域的输入输出图形。

|  |  |
| --- | --- |
| MATLAB代码 | 时域输入输出仿真结果 |
| clf; clear;  syms H(s) ui(t)  K =2;  H(s) = K;  ui(t) = heaviside(t);  Ui(s) = laplace(ui);  Uo(s) = Ui(s)\*H(s);  uo(t) = ilaplace(Uo);  fplot(ui(t)); hold on;  fplot(uo(t));  axis([0 5 0 2.5]);  legend('ui(t)','uo(t)') xlabel('t(s)'); ylabel('y(t)');  title('阶跃响应（比例）'); | C:\Users\sist\AppData\Local\Temp\ConnectorClipboard1601726145969399181\image16606247877950.png |

#### 积分环节（I）

积分环节传递函数：

积分环节模拟电路图如下图所示：





图2-2 积分环节模拟电路

U201的传递函数为：

由拉普拉斯变换的性质可知，U201电路在时域具有积分的性质。

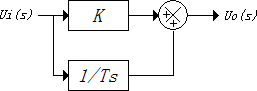
已知U108是一个反相器，其传递函数为

因此图2-2所示积分环节传递函数为

使用MATLAB对比例环节进行仿真，获取时域的输入输出图形。

|  |  |
| --- | --- |
| MATLAB代码 | 时域输入输出仿真结果 |
| clf; clear;  syms H(s) ui(t)  RC =1;  H(s) = 1/(RC\*s);  ui(t) = heaviside(t);  Ui(s) = laplace(ui);  Uo(s) = Ui(s)\*H(s);  uo(t) = ilaplace(Uo);  fplot(ui(t)); hold on;  fplot(uo(t));  axis([0 2.5 0 2.5]);  legend('ui(t)','uo(t)')  xlabel('t(s)'); ylabel('y(t)');  title('阶跃响应（积分）') | C:\Users\sist\AppData\Local\Temp\ConnectorClipboard1601726145969399181\image16606270968210.png |

#### 比例积分（PI）

比例积分环节传递函数：

比例积分环节模拟电路图如下图所示：





图2-3 比例积分环节模拟电路

U301的传递函数为：

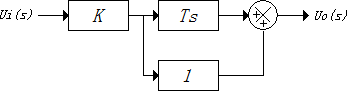
已知U108是一个反相器，其传递函数为

因此图2-2所示积分环节传递函数为

使用MATLAB对比例环节进行仿真，获取时域的输入输出图形。

|  |  |
| --- | --- |
| MATLAB代码 | 时域输入输出仿真结果 |
| clf; clear;  syms H(s) ui(t)  K = 0.5;  RC =1;  H(s) = K+1/(RC\*s);  ui(t) = heaviside(t);  Ui(s) = laplace(ui);  Uo(s) = Ui(s)\*H(s);  uo(t) = ilaplace(Uo);  fplot(ui); hold on;  fplot(uo);  axis([0 2.5 0 2.5])  legend('ui(t)','uo(t)')  xlabel('t(s)'); ylabel('y(t)');  title('阶跃响应（比例积分）') | C:\Users\sist\AppData\Local\Temp\ConnectorClipboard1601726145969399181\image16606287347440.png |

#### 比例微分（PD）

比例微分环节传递函数：

比例微分环节模拟电路图如下图所示：





图2-4 比例微分环节模拟电路

U401的传递函数计算如下：

取，即取，传递函数可简化为：

使用MATLAB对比例环节进行仿真，获取时域的输入输出图形。

|  |  |
| --- | --- |
| MATLAB代码 | 时域输入输出仿真结果 |
| clf; clear;  syms H(s) ui(t)  K = 2;  RC =0.05;  H(s) = K\*(1+RC\*s);  ui(t) = heaviside(t);  Ui(s) = laplace(ui);  Uo(s) = Ui(s)\*H(s);  uo(t) = ilaplace(Uo);  fplot(ui); hold on;  fplot(uo);  axis([0 5 0 2.5])  legend('ui(t)','uo(t)')  xlabel('t(s)'); ylabel('y(t)');  title('阶跃响应（比例微分）') | C:\Users\sist\AppData\Local\Temp\ConnectorClipboard1601726145969399181\image16606336576480.png |

#### 惯性环节

惯性环节传递函数：

惯性环节模拟电路图如下图所示：





图2-5 惯性环节模拟电路

U501的传递函数为：

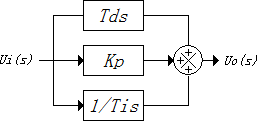
已知U108是一个反相器，其传递函数为

因此图2-2所示积分环节传递函数为

使用MATLAB对比例环节进行仿真，获取时域的输入输出图形。

|  |  |
| --- | --- |
| MATLAB代码 | 时域输入输出仿真结果 |
| clf; clear;  syms H(s) ui(t)  K = 1;  RC =0.2;  H(s) = K/(1+RC\*s);  ui(t) = heaviside(t);  Ui(s) = laplace(ui);  Uo(s) = Ui(s)\*H(s);  uo(t) = ilaplace(Uo);  fplot(ui); hold on;  fplot(uo);  axis([0 5 0 2.5])  legend('ui(t)','uo(t)')  xlabel('t(s)'); ylabel('y(t)');  title('阶跃响应（惯性）' | C:\Users\sist\AppData\Local\Temp\ConnectorClipboard1601726145969399181\image16606350873740.png |

#### 比例积分微分（PID）

比例微分积分环节传递函数：

比例微分积分环节模拟电路图如下图所示：





图2-6 比例微分积分环节模拟电路

U701的传递函数计算如下：

当，即取，传递函数可简化为：

使用MATLAB对比例环节进行仿真，获取时域的输入输出图形。

|  |  |
| --- | --- |
| MATLAB代码 | 时域输入输出仿真结果 |
| clf; clear;  syms H(s) ui(t)  Kp = 0.5; % 比例  Ti =0.2; % 积分  Td =0.05; % 微分  H(s) = Kp+1/(Ti\*s)+Td\*s;  ui(t) = heaviside(t);  Ui(s) = laplace(ui);  Uo(s) = H(s)\*Ui(s);  uo(t) = ilaplace(Uo);  fplot(ui); hold on;  fplot(uo);  axis([0 2.5 0 2.5])  legend('ui(t)','uo(t)')  xlabel('t(s)'); ylabel('y(t)'); title('PID') | C:\Users\sist\AppData\Local\Temp\ConnectorClipboard1601726145969399181\image16606388468410.png |

### 练习：

1. 实验目的和任务：
2. 熟悉实验板、虚拟仪器的使用；
3. 熟练搭建一阶环节的模拟电路；
4. 熟悉一阶环节的阶跃响应，能将表达式对应至实测图形，初步判断图形对错。
5. 实验仪器、设备及材料：
6. 安装有虚拟仪器的PC机；
7. 控制原理实验板；
8. 导线若干。
9. 实测数据记录：

打开电源控制界面，设置正电源12V，设置负电源-12V，电流限制100mA。。

打开信号源界面，设置S1为方波，频率0.5Hz，峰峰值3000mV，直流1500mV。

打开示波器，使用CH1通道观察输入信号，使用CH2通道观察输出信号。

搭建实验原理中各个一阶环节，使用虚拟仪器观测一阶环节的阶跃响应并记录实测图。

|  |  |
| --- | --- |
| 一阶环节 | 实测单位阶跃响应 |
| 比例 |  |
| 积分 |  |
| 比例积分 |  |
| 比例微分 |  |
| 惯性 |  |
| 比例积分微分 |  |

### 二、二阶系统时域分析

二阶系统在控制工程中的应用极为广泛，如RLC网络，电动机、物体运动等都可以使用二阶微分方程进行描述。二阶系统在数学上容易分析，而且在一定条件下可近似地代替高阶系统进行研究。

#### 1、二阶系统数学模型

二阶系统的微分方程可表述为：

为系统输出量；为系统输入量；为二阶系统时间常数；为系统的阻尼系数。

当和均为正时，系统稳定。阻尼系数为无阻尼状态；为欠阻尼状态；为临界阻尼状态；为过阻尼状态。

二阶系统的微分方程也可以表述为：

其中称为系统的自然频率。

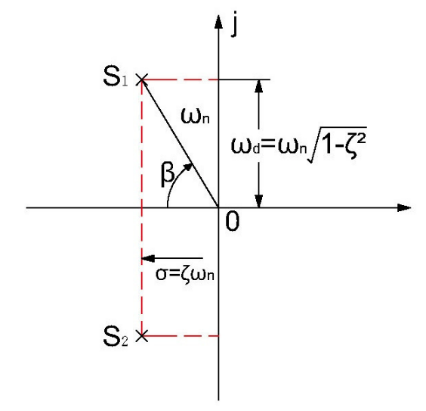
由二阶系统微分方程可得，二阶系统闭环传递函数的形式如下：

特征方程为：

特征根：

|  |  |  |
| --- | --- | --- |
| 取值  （） | 特征根 | 极点位置及阶跃响应 |
|  | ,  两个不相等的正实根，无震荡  ,  共轭复根，实部大于0，有震荡  ,极点位于s平面右侧，阶跃响应不收敛，系统不稳定 | C:\Users\sist\AppData\Local\Temp\ConnectorClipboard1601726145969399181\image16607884810170.png |
| 无阻尼 | ，共轭纯虚根；  极点位于虚轴上，阶跃响应等幅震荡，系统临界稳定。 | C:\Users\sist\AppData\Local\Temp\ConnectorClipboard1601726145969399181\image16607885350700.png |
| 欠阻尼 | ，共轭复根；  ,极点位于s平面左半平面；，有震荡；  阶跃响应震荡收敛，系统稳定。 | C:\Users\sist\AppData\Local\Temp\ConnectorClipboard1601726145969399181\image16607885767840.png |
| 临界阻尼 | ，相等负实根；  ，极点位于s平面左半平面；无虚部，无震荡；  阶跃响应不发散，系统稳定。 | C:\Users\sist\AppData\Local\Temp\ConnectorClipboard1601726145969399181\image16607886093820.png |
| 过阻尼 | ，不相等负实根；  , 极点位于s平面左半平面；无虚部，无震荡；  阶跃响应不发散，系统稳定。 | C:\Users\sist\AppData\Local\Temp\ConnectorClipboard1601726145969399181\image16607889370700.png |

##### 2.1 欠阻尼系统阶跃响应及性能指标

对于欠阻尼二阶系统（即），其阶跃响应的拉氏变换为：

其中，称为阻尼震荡频率。

对进行拉氏反变换可以得到系统的阶跃响应表达式为：

其中，称为二阶系统阻尼角。

* 终值：
* 上升时间：从零第一次上升到终值所需得时间

使的第一个第t值，即，因为，所以，即，

* 峰值时间：响应超过其终值到达第一个峰值所需要的时间
* 超调量：峰值时间处的函数值
* 调节时间：

阶跃响应偏差为：

，忽略正弦函数的影响，得

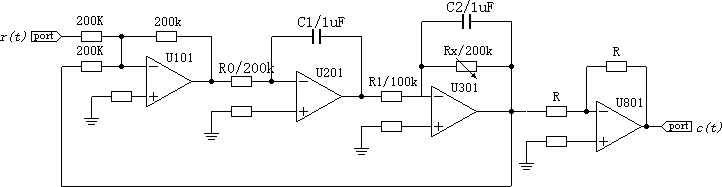
##### 2.2 稳定系统性能指标

稳定二阶系统在欠阻尼、临界阻尼、过阻尼状态下的阶跃响应性能指标：

|  |  |  |  |
| --- | --- | --- | --- |
| 性能指标 |  |  |  |
| 终值： | 1 | | |
| 峰值： |  | | |
| 超调量： |  | 不存在 | 不存在 |
| 峰值时间： |  | 不存在 | 不存在 |
| 调节时间： |  | 不存在 | 不存在 |

#### 2、典型二阶系统分析

典型二阶系统模拟电路图：





其结构图可以表示为：

其开环传递函数为：

其闭环传递函数为：

与标准形式对齐后：

### 练习：

1. 实验目的和任务：
2. 熟悉二阶系统各项性能指标的定义及作用；
3. 理解二阶系统各参数对系统的影响；
4. 观察欠阻尼、临界阻尼、过阻尼状态下系统的响应曲线；
5. 实验仪器、设备及材料：
6. 安装有虚拟仪器的PC机；
7. 控制原理实验板；
8. 导线若干。
9. 实测数据记录：

打开电源控制界面，设置正电源12V，设置负电源-12V，电流限制100mA。。

打开信号源界面，设置S1为方波，频率0.5Hz，峰峰值3000mV，直流1500mV。

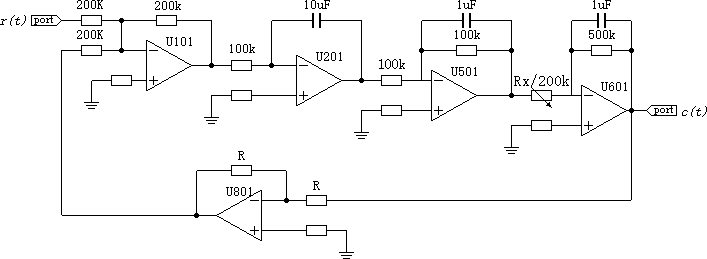
打开示波器，使用CH1通道观察输入信号，使用CH2通道观察输出信号。

搭建实验原理中所示的二阶系统，使用虚拟仪器观测三种阻尼状态，记录实测图并计算相关参数。

|  |  |  |
| --- | --- | --- |
| 状态 | 实测单位阶跃响应 | 性能指标 |
| 欠阻尼 |  |  |
| 临界阻尼 |  |  |
| 过阻尼 |  |  |

### 三阶系统时域分

典型二阶系统模拟电路图：





其结构图可以表示为：



其开环传递函数为：

系统的闭环传递函数传递函数为：

系统的特征方程为：

### 练习：

1. 实验目的和任务：
2. 熟悉高阶系统稳定性判断方法；
3. 观测高阶系统在稳定、临界稳定、不稳定状态下的阶跃响应；
4. 理解系统增益K对系统稳定性的影响；
5. 实验仪器、设备及材料：
6. 安装有虚拟仪器的PC机；
7. 控制原理实验板；
8. 导线若干。
9. 实测数据记录：

打开电源控制界面，设置正电源12V，设置负电源-12V，电流限制100mA。。

打开信号源界面，设置S1为方波，频率0.5Hz，峰峰值3000mV，直流1500mV。

打开示波器，使用CH1通道观察输入信号，使用CH2通道观察输出信号。

搭建实验原理中所示的高阶系统。判断K在不同取值下，系统的稳定性，并记录各状态下系统实测图。

|  |  |  |
| --- | --- | --- |
| 状态 | K的取值 | 实测单位阶跃响应图形 |
| 稳定 |  |  |
| 临界稳定 |  |  |
| 不稳定 |  |  |

## 实验三 线性系统频域分析

时域分析法是利用系统微分方程求解出系统时间响应，再根据时间响应的表达式或响应曲线来分析系统，对于高阶或较为复杂的系统则难以求解和定量分析。频域法只在频域内研究系统的控制规律，是利用频率特性进行控制系统分析和设计的图解法，不必求解微分方程就可以预示出系统的性能，同时也能指出如何调整系统参数来得到预期的性能技术指标。

频域测试的方法是将不同频率的正弦信号作用于被测系统，测取在不同频率下被测系统稳态输出信号与正弦输入信号的幅值比和相位差，从而确定被测系统的频率特性。对于线性定常系统，由输入产生的输出稳态分量是一个与输入信号有着相同频率，但幅值和相位不同的信号。并且其幅值和相位的变化是频率的函数，因此系统的频率特性可表示为：

另外，系统的频率特性与传递函数之间有着非常简单的转换关系，对于一般线性系统有：

系统的频率特性经常使用奈奎斯特Nyquist图（极坐标图）、伯德Bode图（对数坐标图）、尼克尔斯Nichols图（对数幅相图）等表示。本实验中我们将使用奈奎斯特图和伯德图。

### 一、一阶环节频率特性曲线测试

#### 1、奈奎斯特图

频率特性是频率的复变函数，在复平面上可以用向量或其端点坐标来表示。当从时，向量的端点在复平面上运动的轨迹即是频率特性的极坐标图或奈奎斯特图，简称奈氏图。并且规定极坐标实轴正方向为相角0度线，逆时针转过的角度为正，顺时针转过的角度为负。

以一阶惯性环节为例，其传递函数为，频率特性=

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |

可知一阶惯性环节的奈氏图具有如下形态：

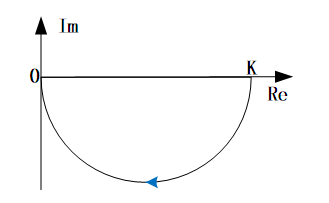


图 1 奈奎斯特图

#### 2、伯德图

伯德图由对数幅频特性和对数相频特性两张图组成，横坐标都是，其单位是rad/s。对数幅频特性图的纵坐标为，常用来表示，单位为分贝；对数相频特性图的纵坐标为，等分刻度，单位为角度的度（°）。



图 2 对数坐标系

以一阶惯性环节为例，有：

其图形为：



图 3 惯性环节伯德图

#### 3、典型一阶环节及其频率特性曲线测试

一阶系统电路图如下，可求的其传递函数为。

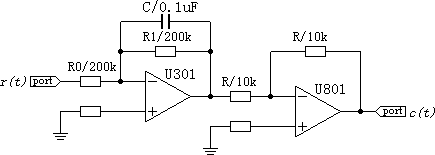


图 4 惯性环节模拟电路图

使用MATLAB进行模拟：

|  |  |
| --- | --- |
| clf; clear;  K = 1;T =0.02;  num = [K];  den = [T 1];  nyquist(num,den);grid on; | C:\Users\sist\AppData\Local\Temp\ConnectorClipboard291874174717233743\image16611561836850.png |
| bode(num,den); | C:\Users\sist\AppData\Local\Temp\ConnectorClipboard291874174717233743\image16611561985640.png |

### 二、二阶系统开环频率特性曲线测试

典型二阶系统电路图如下，可求的其传递函数为



图5 二阶系统模拟电路

使用MATLAB进行模拟：

|  |  |
| --- | --- |
| clf; clear;  K = 1;T1 =0.1;T2 =0.2;  num = [K];  den = conv([T1 1],[T2 1]);  nyquist(num,den); | C:\Users\sist\AppData\Local\Temp\ConnectorClipboard291874174717233743\image16611585412730.png |
| bode(num,den); | C:\Users\sist\AppData\Local\Temp\ConnectorClipboard291874174717233743\image16611585624250.png |

### 三、伯德图分析

系统的开环传递函数通常可以写成典型环节串联的形式，即

可知其频率特性为

因此在绘制系统开环对数频率特性曲线时，可先将开环传递函数化成典型环节的乘积；然后画出各典型环节的对数幅频和对数相频特性曲线；最后将各环节的对数幅频、相频曲线相加。

如有系统的开环传递函数为：



### 练习：

1. 实验目的和任务：
2. 掌握系统频域分析方法，如奈奎死他图（MATLAB模拟）及伯德图（MATLAB模拟及实测）；
3. 会根据伯德图推断系统的传递函数；
4. 理解系统增益K对系统稳定性的影响；
5. 实验仪器、设备及材料：
6. 安装有虚拟仪器的PC机；
7. 控制原理实验板；
8. 导线若干。
9. 实测数据记录：

打开电源控制界面，设置正电源12V，设置负电源-12V，电流限制100mA。。

打开信号源界面，设置S1为方波，频率0.5Hz，峰峰值3000mV，直流1500mV。

打开示波器，使用CH1通道观察输入信号，使用CH2通道观察输出信号。

1. 搭建图4、图5所示模拟电路，获取其伯德图：

|  |  |
| --- | --- |
| 系统 | 实测图形 |
| 一阶惯性环节 |  |
| 二阶系统 |  |

1. 根据以下伯德图推导系统传递函数，并补充模拟电路中R1、R2、R6及C1、C2的值。





|  |  |
| --- | --- |
| 项目 | 理论值 |
| R1 |  |
| R2 |  |
| R6 |  |
| C1 |  |
| C2 |  |

如下设置伯德图测试参数

|  |  |
| --- | --- |
| 设备：EPI-AN302 | 起始频率：1Hz |
| 终止频率：100Hz | 信号源幅度：3kmVpp |
| 信号源直流偏置：0mV | Y轴标尺：对数 |

实测伯德图：

|  |
| --- |
|  |

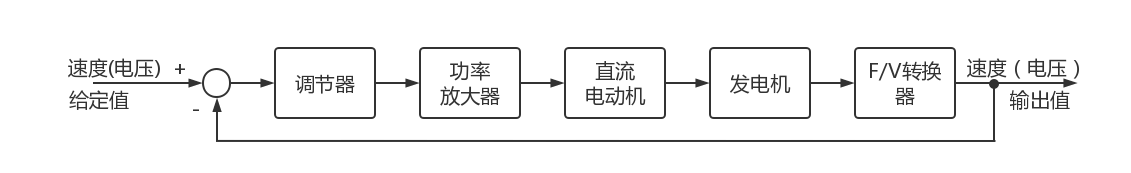
## 实验四 PID控制实验

具有比例（P）、积分（I）、微分（D）功能的控制器，称为PID控制器。当被控对象的传递函数难以描述时，它提供了一种应用广泛、行之有效的控制方法。PID控制器是一种线性控制器，它根据给定与实际输出构成控制偏差，将骗差按比例、积分、微分通过线性组合构成控制量，对被控对象进行控制。PID控制器的输出和输入的关系可描述为：

其中为控制器的输出信号；为控制器的偏差信号；为比例系数；为积分时间常数；为微分时间常数。控制系统的开环传递函数为：

直流电机能将直流电能转换为旋转运动的机械能，在机器人操纵系统、传送带系统、机床及伺服阀驱动器等时机控制系统中，直流电机都得到了相当广泛的应用。

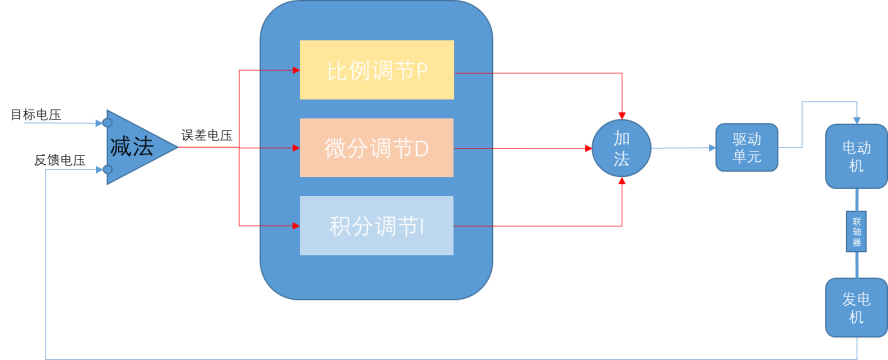
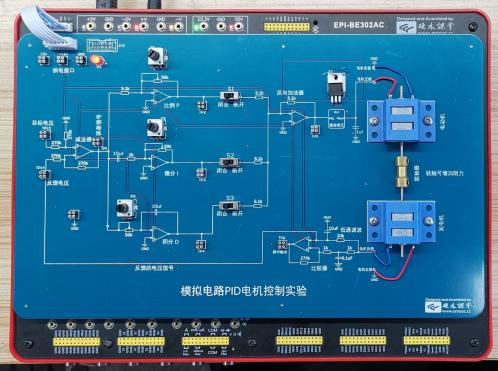
在本次实验中，我们将使用直流电机去驱动一个小型的发电机，通过PID来调节直流电机的转速，使发电机的输出电压达到目的电压。后续我们也可以利用该结构达到其它应用目的，比如通过调节内燃机的转速使小型汽油发电机达到用户设定电压；如将发电机改为气压传感器，电动机改为气泵，就形成了医院里病房氧气管路的气压控制系统；而如果将发电机改为速度计，就可以实现车辆定速巡航。本实验中使用到的直流电机调速控制系统结构框图如下：

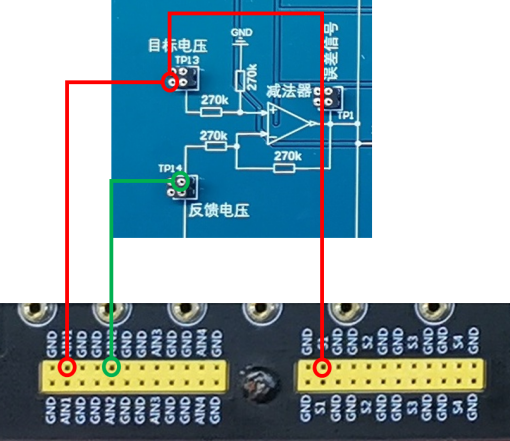


### 环境介绍

#### 1、实验电路板及接口

实验板整体外观及其电路框图如下所示，其中P、I、D三个环节可分别开关，其系数可单独调节。





连线时：

由信号源S1通道提供目标电压。

使用示波器的AIN1通道观察目标电压。

使用示波器的AIN2通道观察反馈电压。

调节P、I、D环节并观察反馈电压对目标电压的跟踪情况。

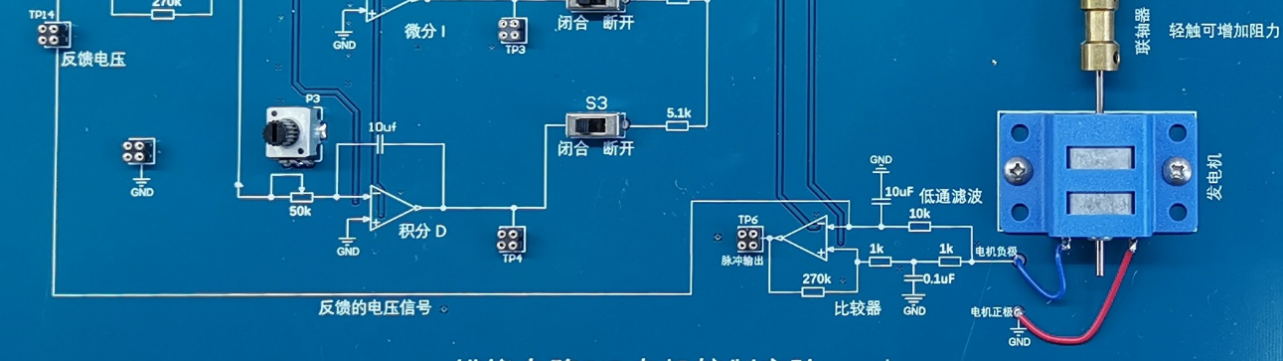
#### 2、PID调节单元

PID调节单元由独立的反向比例放大环节P，反向微分环节D和反向积分环节I构成。可使用三个环节中的电位器调节比例系数，微分系数以及积分系数（各环节模拟电路原理请参见实验二）。S1 、S2、 S3开关用于控制三个环节是否接入系统。

PID单元的输入为误差信号，输出进入反向加法器，将P、 I 、D 三路调节信号求和后输送给驱动单元。另外，各环节都有4孔输出测试点，可用示波器来进行观察各环节的输出情况。

#### 3、反馈电压单元

由于发电机是有刷电机，电刷的切换会带来较大噪声。而微分器对噪声敏感，因此反馈信号处使用了RC低通滤波器来滤除低频噪声，以获取干净的电压信号。电压信号的大小则跟转速成正比。

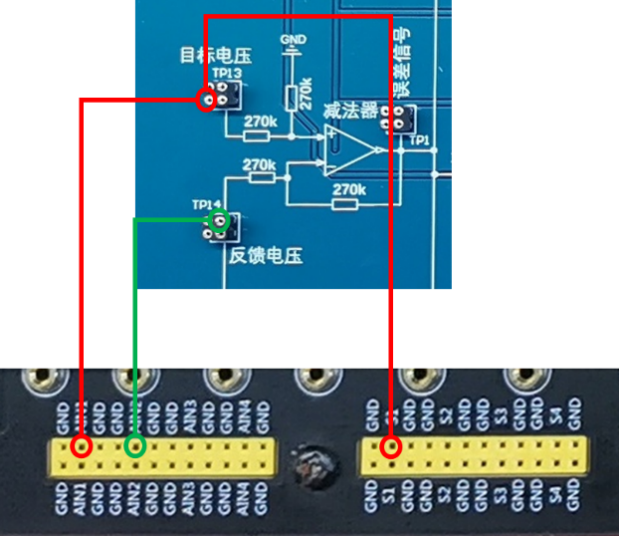
另外反馈电路还利用了比较器将发电机转动产生的电压波动装换成了方波信号。可使用示波器观察TP6处输出脉冲信号频率，电机每转动以前产生将产生6个脉冲，通过计量每分钟的脉冲个数就可以知道电机的转速。因此从脉冲频率的变化中，就可以观察到转速的稳定性及跟踪情况。

#### 4、实验连线

单用一个USB线时无法带动电机，需要使用配件盒里的5V适配器和电源线提供辅助供电。插接方法如下图所示。

与示波器及信号源的连接如下图所示，用于提供对目标电压及反馈电压的观测。



### 练习：

1. 实验目的和任务：
2. 熟悉PID控制的原理；
3. 知道P、I、D参数各自对系统响应产生的效果；
4. 调节PID参数，使系统输出达到预期。
5. 实验仪器、设备及材料：
6. 安装有虚拟仪器的PC机；
7. PID控制实验板，配套电源线；
8. 导线若干。
9. 实测数据记录：

打开电源控制界面，设置正电源12V，电流限制700mA。设置负电源-12V。

打开信号源界面，设置S1为方波，频率0.5Hz，直流2000mV，峰峰值500mV。

1. 分别观察P、I、D各自对系统输出的影响（观察单个环节时，请关闭其余两个环节）。

|  |  |  |  |
| --- | --- | --- | --- |
| 环节 | 参数值 | 实测图形 | 作用总结 |
| 比例（P） |  |  |  |
|  |  |
| 微分（D） |  |  |  |
|  |  |
| 积分（I） |  |  |  |
|  |  |

1. 叠加效果观察效果

|  |  |  |
| --- | --- | --- |
| 模式 | 最佳效果实测图 | 效果总结 |
| P+D |  |  |
| P+I |  |  |
| D+I |  |  |
| P+I+D |  |  |