**实验三 连续时间系统的时域和s域分析**

**一、实验说明**

**1、实验类型：验证性**

**二、实验目的**

1. 学习使用冲激响应（impulse函数）；
2. 学习关于阶跃响应（step函数）；
3. 掌握零状态响应（lsim函数）；
4. 学习simulink，搭建相应的系统环境，并进行仿真与分析。

**三、实验原理与方法**

1. 通过impluse函数实现冲击响应
2. 通过step函数实现阶跃响应
3. 通过conv和lsim函数实现系统的零状态响应
4. 通过simulink搭建系统环境，用sine wave生成余弦函数，通过transfer fcn根据传递函数实现微分方程的构建，最后通过scope监测输出波形。
5. 响应分析可以通过峰值时间，超调量，上升时间。调节时间来进行。
6. dcgain可以求得系统的终值

**四、实验内容及步骤**

**1**、利用Matlab计算并绘制以下系统的单位冲激响应和单位阶跃响应

%%式一

%冲激响应

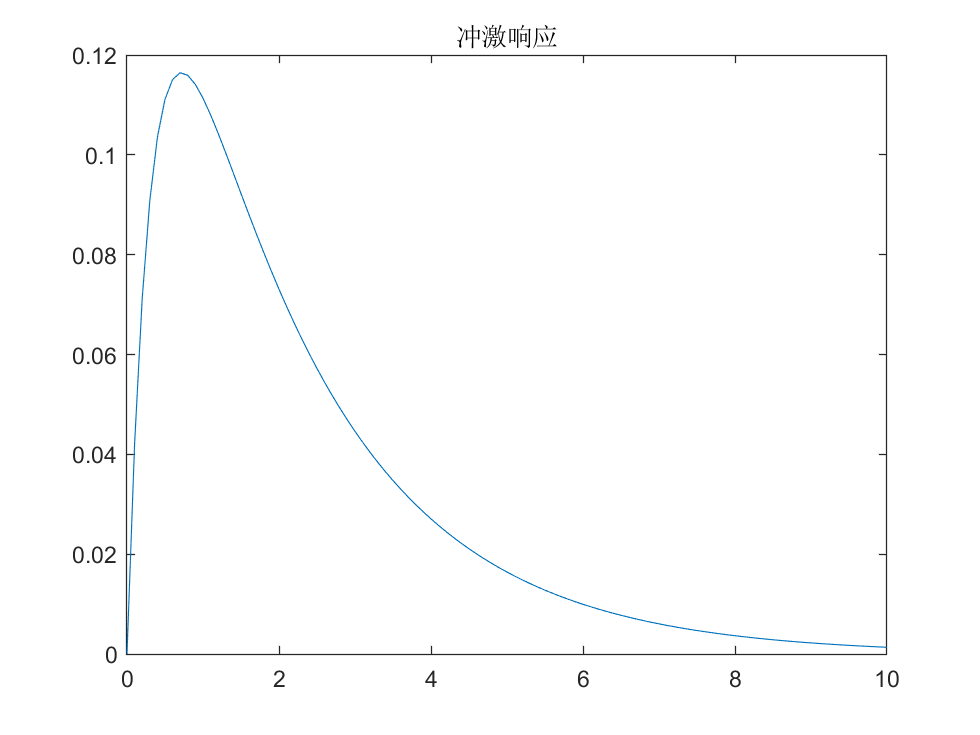
sys = tf([1],[2,7,3]);

t = 0:0.1:10;

y=impulse(sys,t);

plot(t,y)

title('冲激响应')



%阶跃响应

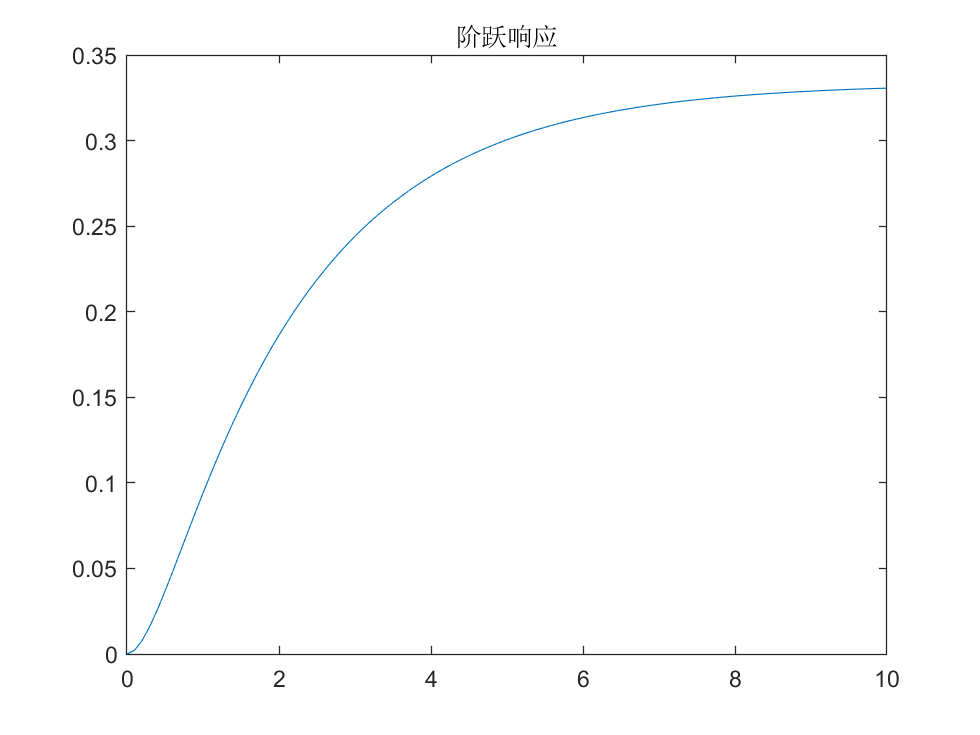
sys = tf([1],[2,7,3]);

t = 0:0.1:10;

y=step(sys,t);

plot(t,y)

title('阶跃响应')





%%式二

%冲激响应

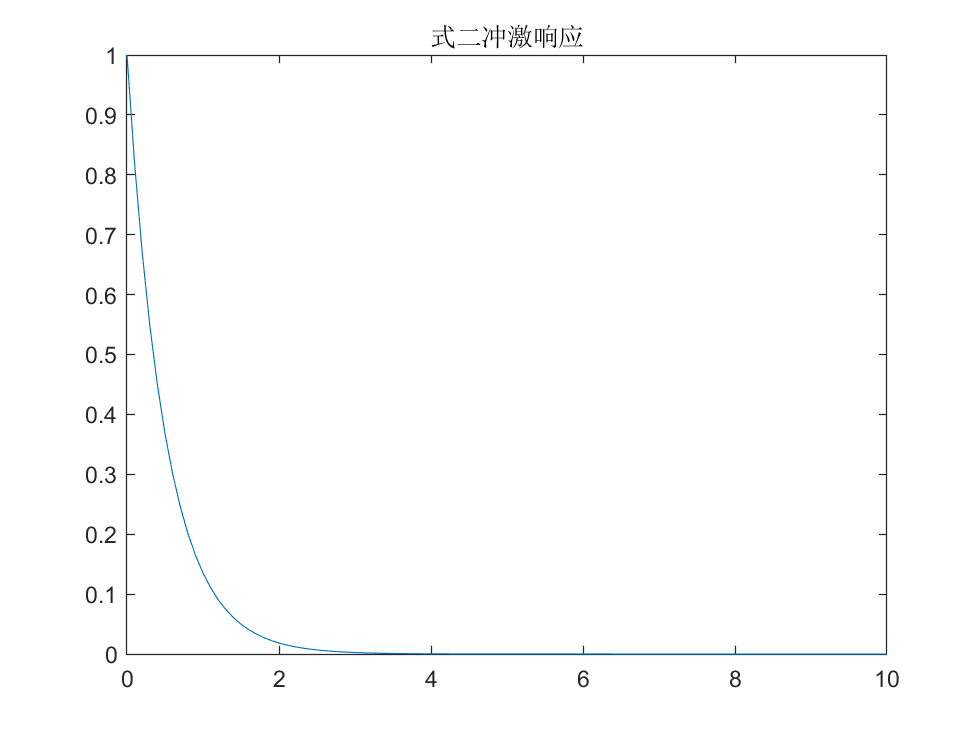
sys = tf([1,1],[1,3,2]);

t = 0:0.1:10;

y=impulse(sys,t);

plot(t,y)

title('式二冲激响应')



%冲激响应

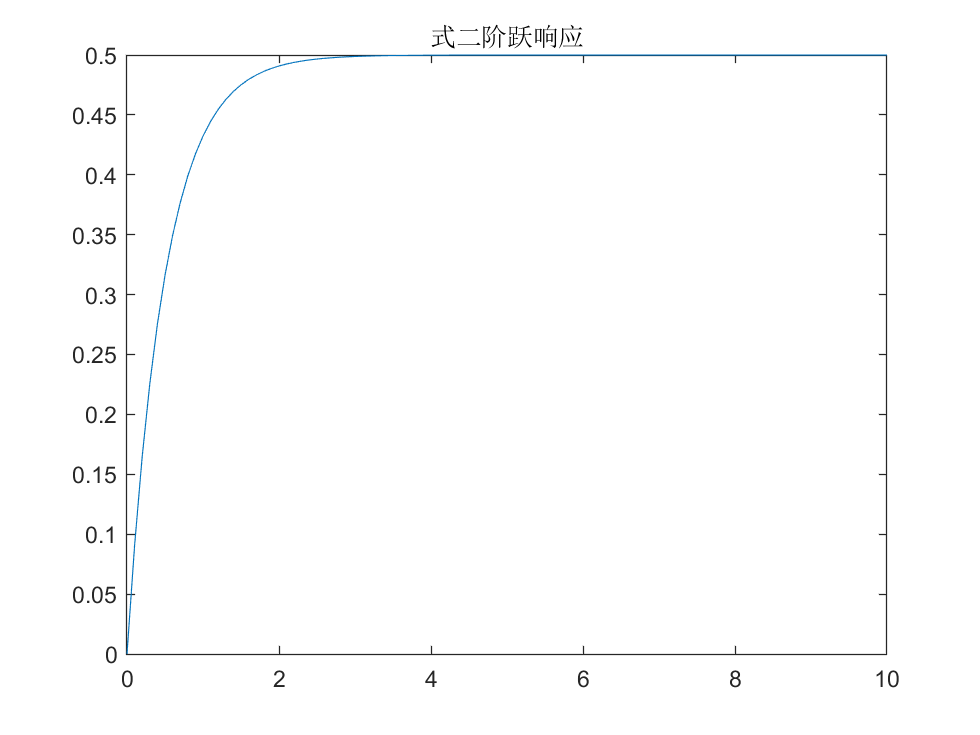
sys = tf([1,1],[1,3,2]);

t = 0:0.1:10;

y=step(sys,t);

plot(t,y)

title('式二阶跃响应')



**例1**.利用Matlab计算并绘制以下系统的单位冲激响应和单位阶跃响应。



% 定义传递函数

num = [1 1]; % 传递函数分子多项式系数，对应 s + 1

den = [1 2 4]; % 传递函数分母多项式系数，对应 s^2 + 2s + 4

sys = tf(num, den);

% 定义时间向量

t = 0:0.01:10; % 从0开始，步长0.01，到10结束，确定计算响应的时间范围

% 计算单位冲激响应

y\_impulse = impulse(sys, t);

% 使用impulse函数计算系统在时间向量t上的单位冲激响应

% 绘制单位冲激响应

figure; % 创建新的绘图窗口

plot(t, y\_impulse);

title('单位冲激响应');

xlabel('时间t');

ylabel('h(t)');

% 计算单位阶跃响应

y\_step = step(sys, t);

% 使用step函数计算系统在时间向量t上的单位阶跃响应

% 绘制单位阶跃响应

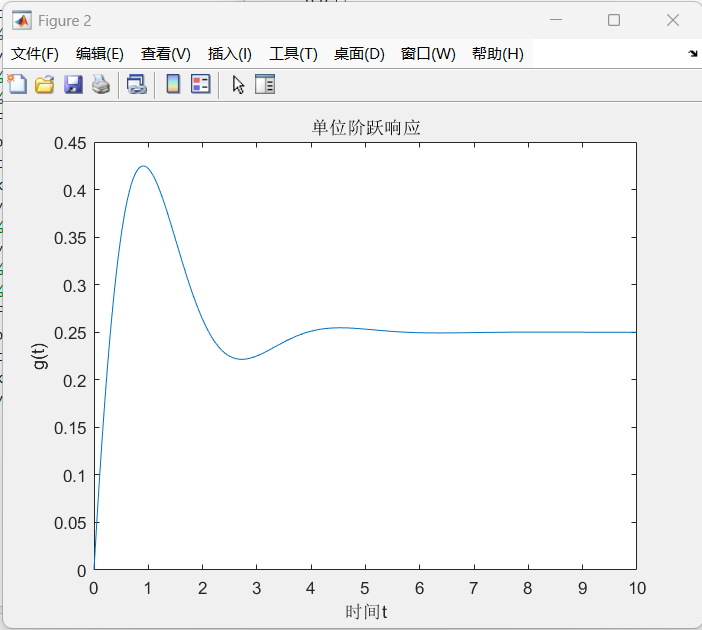
figure; % 再创建一个新的绘图窗口

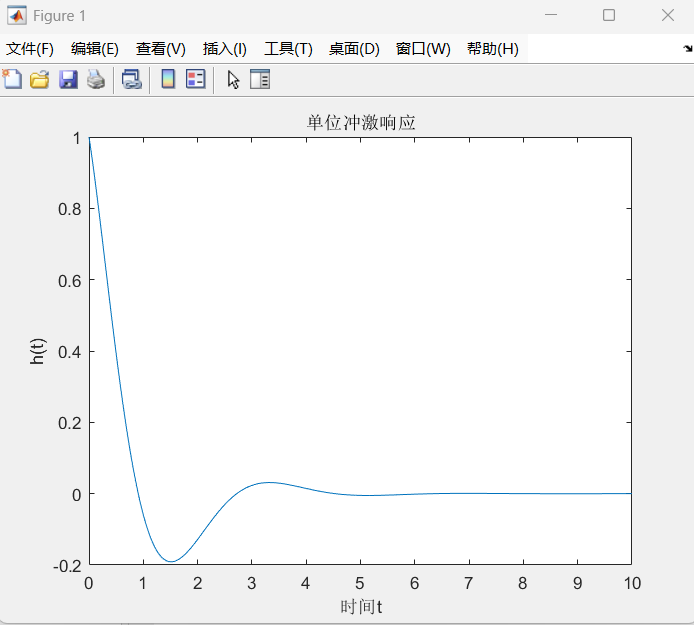
plot(t, y\_step);

title('单位阶跃响应');

xlabel('时间t');

ylabel('g(t)');





**2**、已知某LTI系统的单位冲激响应为，指定输入激励为，分别利用Matlab 中的lsim函数和conv函数计算系统的零状态响应。

clear

T=0.1

T = 0.1000

t=0:T:5;

t1=0:T:10

t1 = 1×101

0 0.1000 0.2000 0.3000 0.4000 0.5000 0.6000 ⋯

h=exp(-t)+exp(-2\*t);

f=exp(-t);

yf=conv(h,f)

yf = 1×101

2.0000 3.5332 4.6861 5.5298 6.1232 6.5149 6.7449 ⋯

yf=yf.\*T

yf = 1×101

0.2000 0.3533 0.4686 0.5530 0.6123 0.6515 0.6745 ⋯

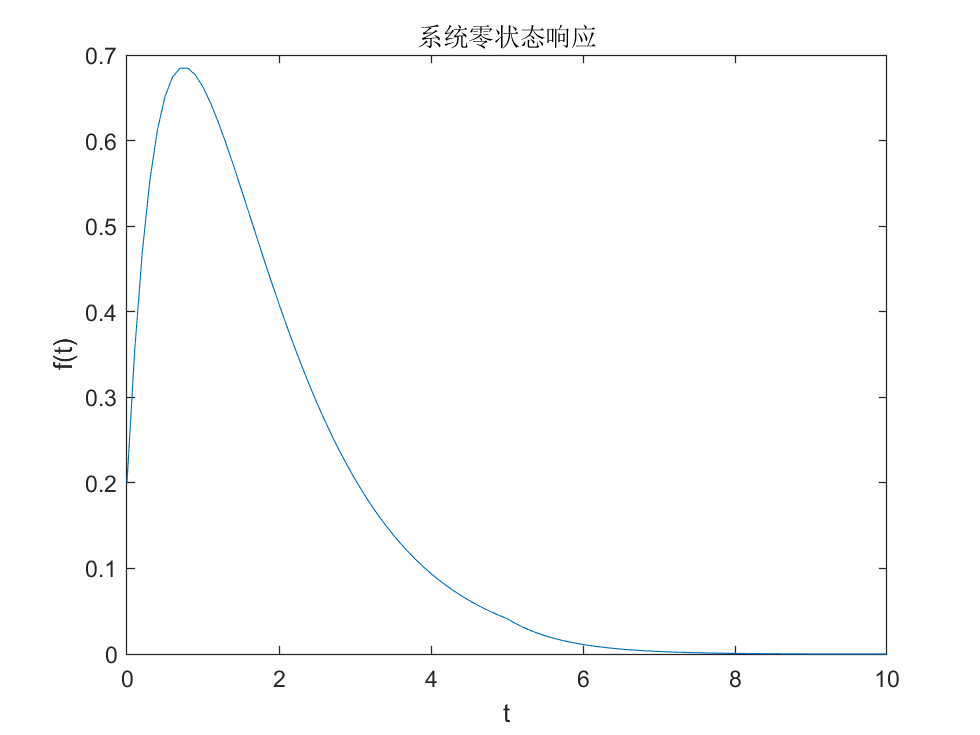
figure(1)

plot(t1,yf)

title('系统零状态响应')

xlabel('t')

ylabel('f(t)')



clear

syms s h f

%s=0:0.1:5;%此处不能先赋值

h=exp(-s)+exp(-2.\*s);

f=exp(-s);

H=laplace(h,s);%变换后不能直接使用

F=laplace(f,s)

F =



Hs=F\*H;

hs=ilaplace(Hs);

sys2=tf([2,3],[1,4,5,2]);

sys1=tf([2,3],[1,3,2]);

%sys2=tf([1],[1,2])

s=0:0.1:10;

%yf2=step(sys2,s);

f=exp(-s);

F=tf([1],[1,1])

F =

1

-----

s + 1

Continuous-time transfer function.

yf=lsim(sys1,f,s,0);

%yf1=lsim(sys2,f,s,0)

%yf=conv(h,f);

%yf=yf1+yf

%画图

t=0:0.1:10;

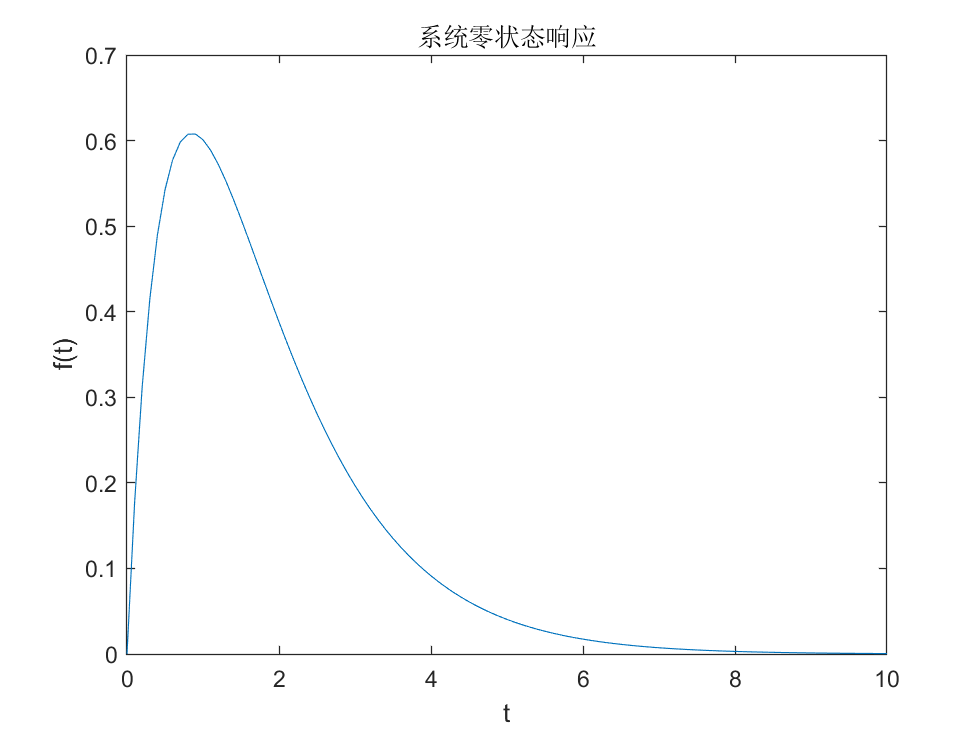
figure(1)

plot(t,yf)

title('系统零状态响应')

xlabel('t')

ylabel('f(t)')



**例2** 已知某LTI系统的单位冲激响应为输入激励为 

，用Matlab 中的conv函数计算系统的零状态响应。

t = 0:0.1:5;

h = exp(-2\*t)+exp(-3\*t);

f = exp(-t).\* (t >= 0); % 修正输入激励，确保符合u(t)特性

yf = conv(h,f);

dt = 0.1; % 时间步长

t\_new = (0:length(yf)-1)\*dt; % 重新生成时间轴

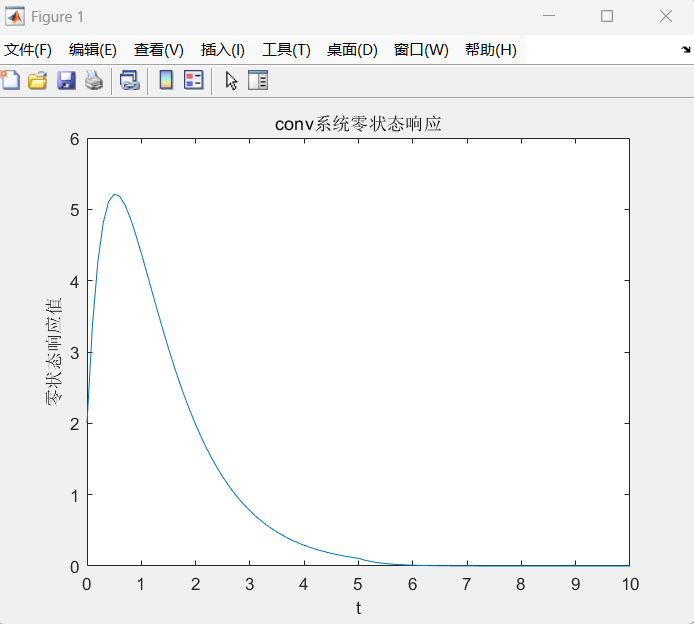
figure(1);

plot(t\_new,yf);

title('conv系统零状态响应');

xlabel('t');

ylabel('零状态响应值');





% 假设系统函数分子分母系数

num = [1]; % 分子系数，这里假设分子为1

den = [1 1.5 0.5]; % 分母系数，对应s^2 + 1.5s + 0.5，可根据实际系统修改

% 创建传递函数对象

sys = tf(num, den);

% 绘制零极点图

pzmap(sys);

title('零极点分布图');

xlabel('实轴 (秒⁻¹)');

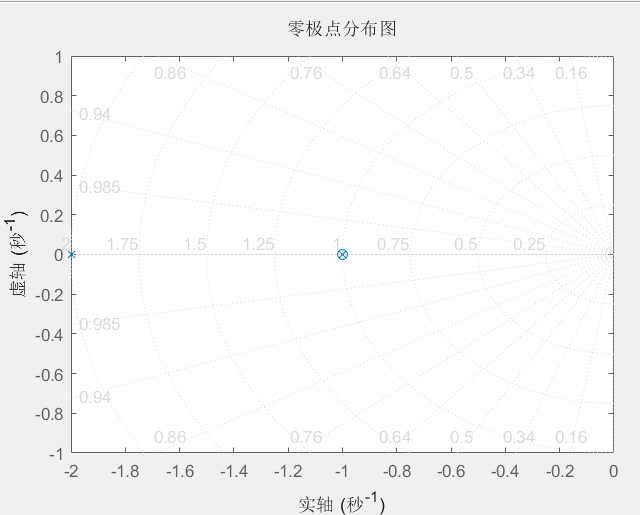
ylabel('虚轴 (秒⁻¹)');

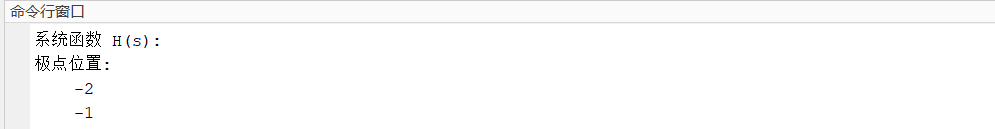
% 设置坐标轴范围

axis([-2 0 -1 1]);

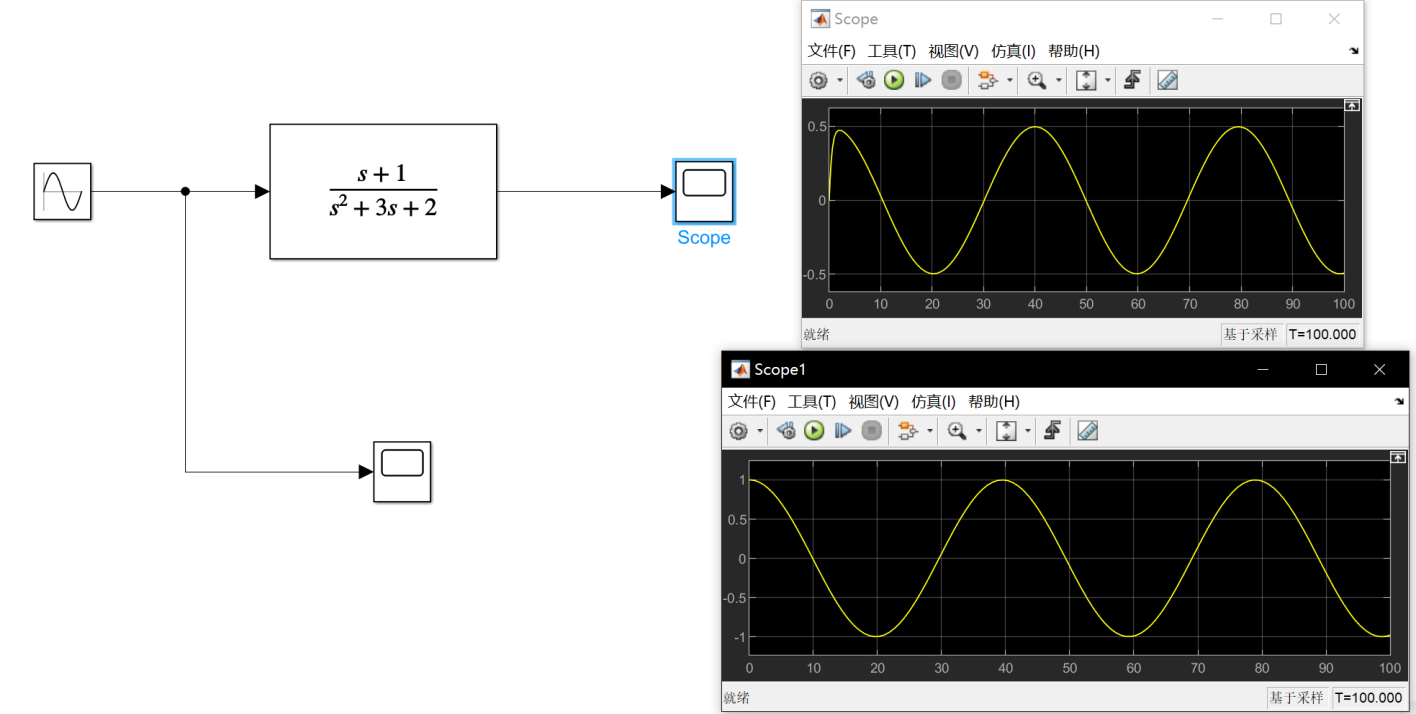
% 绘制网格线

grid on;





**4**、通过微分方程，进行系统函数的simulink系统搭建及响应分析。正弦函数做输入



%simulink

clear

%峰值时间tp

sys=tf([1,1],[1,3,2])

sys =

s + 1

-------------

s^2 + 3 s + 2

Continuous-time transfer function.

[y,t]=step(sys);

[Y,k ]=max(y);

tp=t(k)

tp = 4.4900

%超调量

C=dcgain(sys)

C = 0.5000

[Y,k]=max(y);

sita=100\*(Y-C)/C

sita = -0.0126

%上升时间tr

C=dcgain(sys);

n=1;

while y(n )<0.1\*C

n=n+1;

end

m=1;

while y(m)<0.9\*C

m=m+1;

end

tr=t(m)-t(n)

tr = 1.1052

%调节时间

[y,t]=step(sys);

C=dcgain(sys)

C = 0.5000

i=length(t)

i = 196

while(y(i)>0.98\*C)&(y(i)<1.02\*C)

i=i-1;

end

ts=t(i)

ts = 1.9342