在MATLAB中，可以用函数[z，p，K]=tf2zp（num，den）求得有理分式形式的系统转移函数的零、极点，用函数zplane（z，p）绘出零、极点分布图；也可以用函数zplane（num，den）直接绘出有理分式形式的系统转移函数的零、极点分布图。使h=freqz(num,den,n)函数可求系统的频率响应，w是频率的计算点，如n=0:pi/255:pi, h是复数，abs(h)为幅度响应，angle(h)为相位响应。另外还可以使用[h,w] =freqz(num,den,n),w便是频域横坐标.另外，在MATLAB中，可以用函数 [r，p，k]=residuez（num，den）完成部分分式展开计算；可以用函数sos=zp2sos（z，p，K）完成将高阶系统分解为2阶系统的串联。

**例1** 求下列直接型系统函数的零、极点，并将它转换成二阶节形式 mhtml:file://D:\amily\教师\教学\数字信号处理\我的ppt\上机\MATLAB第五章数字信号处理系统的实现%20%20%20.mht!http://jwc.seu.edu.cn/zq/signal/new/MATLAB/MATLAB44.gif

num=[1 -0.1 -0.3 -0.3 -0.2];%零点

den=[1 0.1 0.2 0.2 0.5];%极点

[z,p,k]=tf2zp(num,den);% zpk分别是零极点增益系数

m=abs(p);

disp('零点');disp(z);

disp('极点');disp(p);

disp('增益系数');disp(k);

sos=zp2sos(z,p,k);%%zp2sos（z，p，K）完成将高阶系统分解为2阶系统的串联。

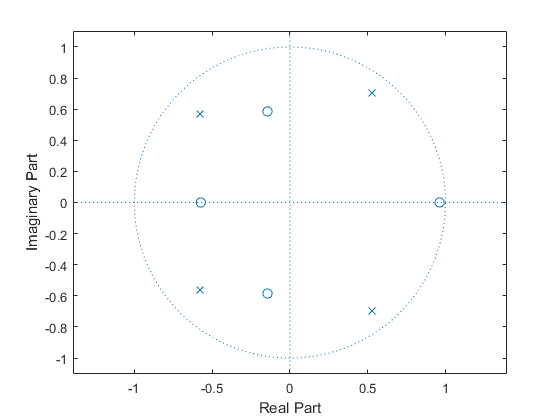
disp('二阶节');disp(real(sos));

zplane(num,den)%直接根据系统函数系数绘出有理分式形式的系统转移函数的零、极点分布图

%等价于 下边这句话

%figure

%zplane(z,p)



**例2** 差分方程

MATLAB42

所对应的系统的频率响应。

**解：**差分方程所对应的系统函数为



程序如下:

k=256;

num=[0.8 -0.44 0.36 0.02];%分子系数 (x系数)

den=[1 0.7 -0.45 -0.6];%分母系数 (y系数)

w=0:pi/k:pi;%数字频率

h=freqz(num,den,w);%求解系统函数的频率响应

%freqz(num,den,w) 直接输入这句话会显示幅度相应和相位响应

subplot(2,2,1);

plot(w/pi,real(h));grid

title('实部')

xlabel('\omega/\pi');ylabel('幅度')

subplot(2,2,2);

plot(w/pi,imag(h));grid

title('虚部')

xlabel('\omega/\pi');ylabel('Amplitude')

subplot(2,2,3);

plot(w/pi,abs(h));grid

title('幅度谱')

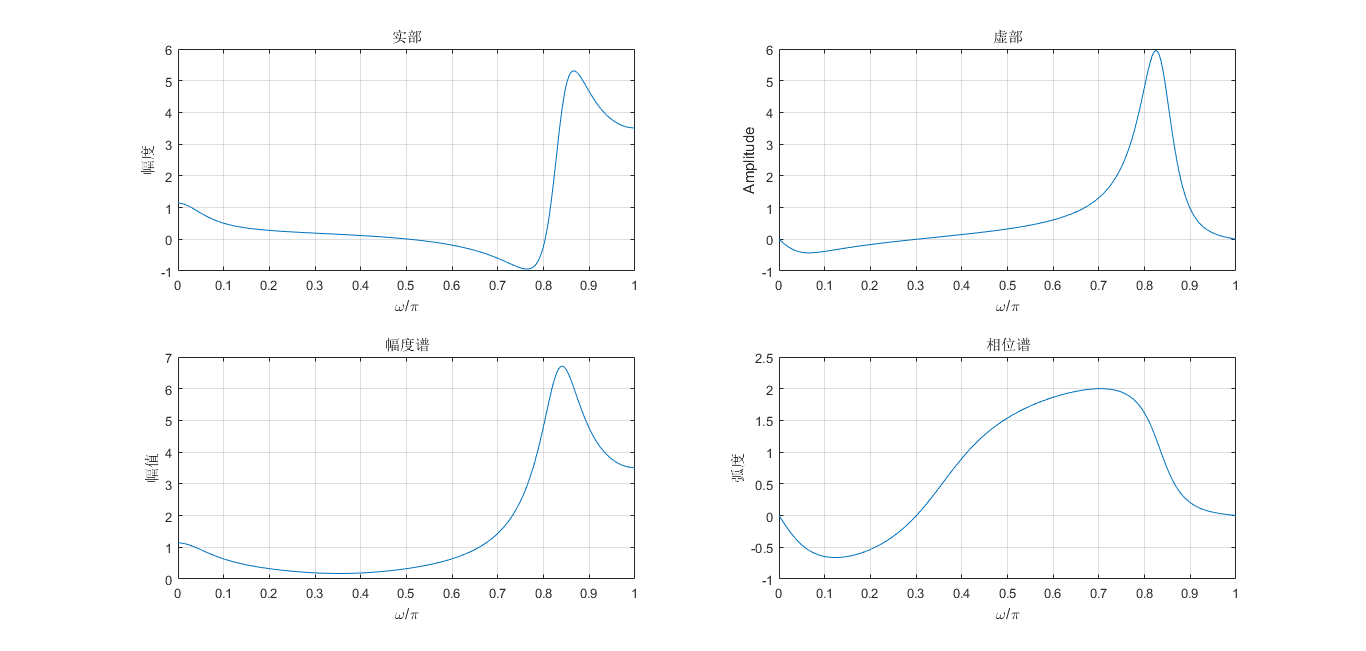
xlabel('\omega/\pi');ylabel('幅值')

subplot(2,2,4);

plot(w/pi,angle(h));grid

title('相位谱')

xlabel('\omega/\pi');ylabel('弧度')



**实验内容**：求系统

**** 的零、极点和幅度频率响应和相位响应。

**实验要求**：编程实现系统参数输入，绘出幅度频率响应和相位响应曲线和零、极点分布图。

num=[0.0528 0.0797 0.1295 0.1295 0.797 0.0528];

den=[1 -1.8107 2.4947 -1.8801 0.9537 -0.2336];

[z,p,k]=tf2zp(num,den); %zpk分别为零极点增益系数

disp('零点');disp(z);

disp('极点');disp(p);

disp('增益系数');disp(k);

figure(1)

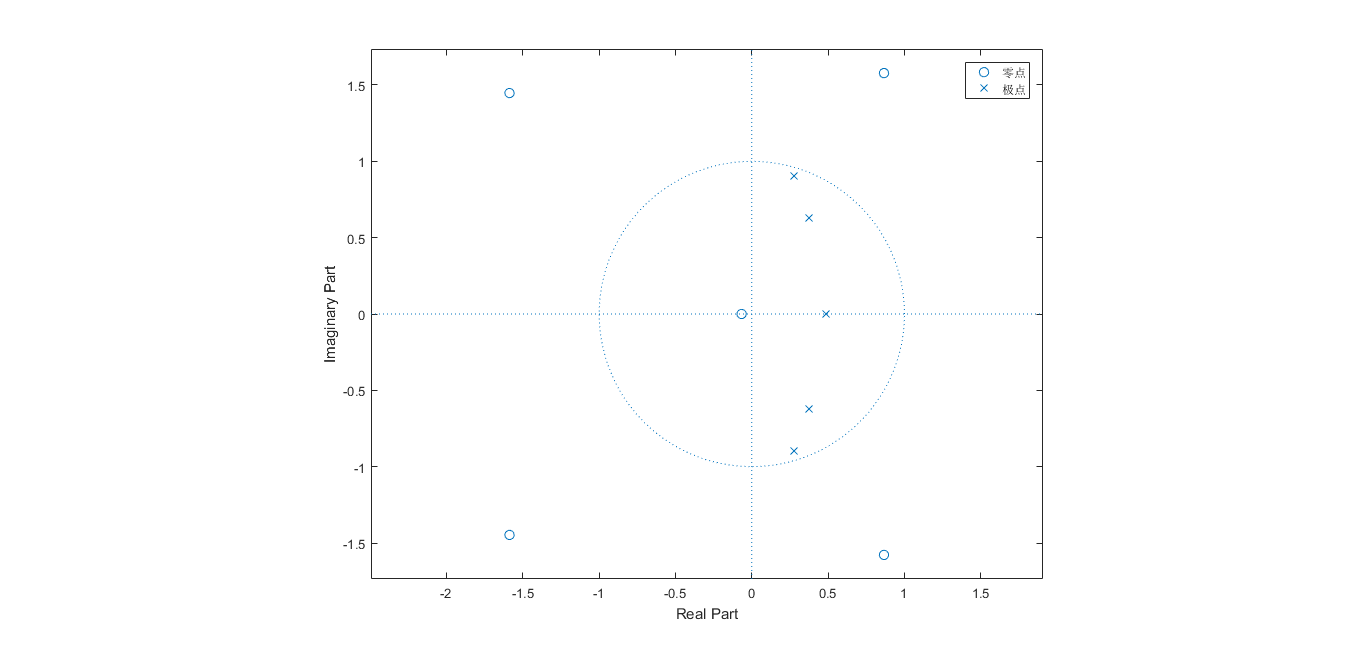
zplane(num,den)%直接根据系统函数系数绘制零极点

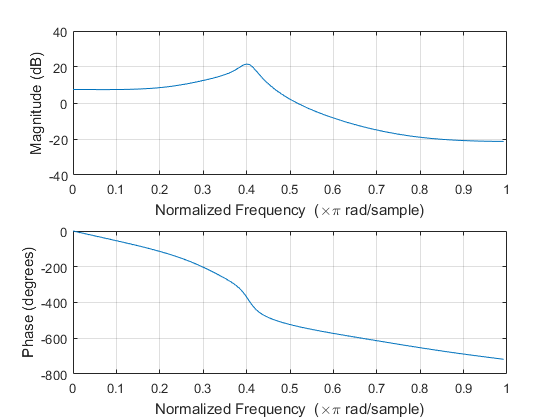
legend('零点','极点')

%等价这句话 zplane(z,p)

figure(2)

freqz(num,den,128)





**P105 4.12求滤波器输出的前十个采样值**；

clear;

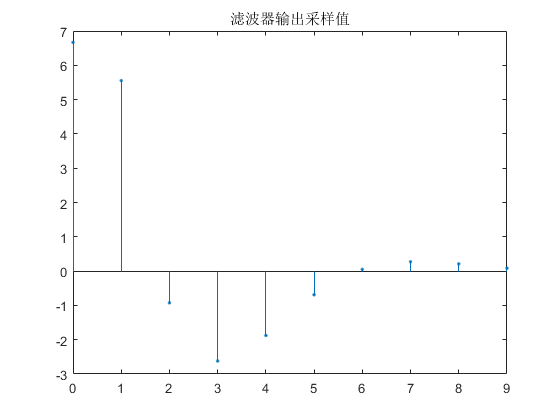
B=[0.3 -0.25 0.1];

A=1; n=0:9;

X=[2\*(n==0)-(n==2)];

Y=filter(A,B,X); stem(n,Y,'.');

title('滤波器输出采样值')



**6.22． 对于传输函数**



**求脉冲响应和阶跃相应**

num=[5];

den=[1 -1 0.09];

subplot(2,1,1)

%前40个点的冲激响应

impz(num,den,40);

[h,t]=impz(num,den,40);

title('脉冲响应')

step=ones(1,100);

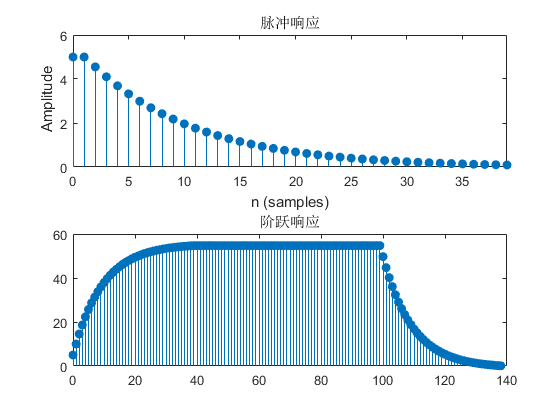
subplot(2,1,2)

y=conv(h,step);

n=0:138;

stem(n,y,'filled')

title('阶跃响应')



**7.26 滤波器的传输函数为：**



**不计算，画出幅度响应曲线**。

%将系统函数化为分子分母形式

k=256;

num=[0,0,1,1];

den=[1,-1,0,0];

w=0:pi/k:pi;

h=freqz(num,den,w);

subplot(2,2,1);

plot(w/pi,real(h));grid

title('实部')

xlabel('\omega/\pi');ylabel('幅度')

subplot(2,2,2);

plot(w/pi,imag(h));grid

title('虚部')

xlabel('\omega/\pi');ylabel('Amplitude')

subplot(2,2,3);

plot(w/pi,abs(h));grid

title('幅度谱')

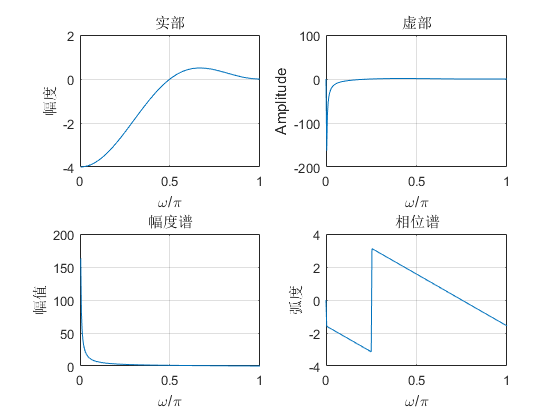
xlabel('\omega/\pi');ylabel('幅值')

subplot(2,2,4);

plot(w/pi,angle(h));grid

title('相位谱')

xlabel('\omega/\pi');ylabel('弧度')



**7.28 由七项滑动平均滤波器的零极点推断滤波器形状**

解: 由七项滑动平均滤波器特点，写出其对应传输函数

=

%先写出7项滑动平均系统函数

num=[1 1 1 1 1 1 1];

den=[7 0 0 0 0 0 0];

[z,p,k]=tf2zp(num,den);

disp('零点');disp(z);

disp('极点');disp(p);

disp('增益系数');disp(k);

figure(1)

zplane(num,den)

figure(2)

freqz(num,den,128)

