

## 附录一 信号、系统和系统响应

### 1、理想采样信号序列

(1) 首先产生信号  $x(n), 0 \leq n \leq 50$

```
n=0:50; %定义序列的长度是 50
A=444.128; %设置信号有关的参数
a=50*sqrt(2.0)*pi;
T=0.001; %采样率
w0=50*sqrt(2.0)*pi;
x=A*exp(-a*n*T).*sin(w0*n*T); %pi 是 MATLAB 定义的  $\pi$ ，信号乘可采用“.*”
close all %清除已经绘制的 x(n)图形
subplot(3,1,1);stem(x); %绘制 x(n)的图形
title('理想采样信号序列');
```

(2) 绘制信号  $x(n)$  的幅度谱和相位谱

```
k=-25:25;
W=(pi/12.5)*k;
X=x*(exp(-j*pi/12.5)).^(n'*k);
magX=abs(X); %绘制 x(n)的幅度谱
subplot(3,1,2);stem(magX);title('理想采样信号序列的幅度谱');
angX=angle(X); %绘制 x(n)的相位谱
subplot(3,1,3);stem(angX); title('理想采样信号序列的相位谱')
```

(3) 改变参数为:  $A = 1, \alpha = 0.4, \Omega_0 = 2.0734, T = 1$

```
n=0:50; %定义序列的长度是 50
A=1; %设置信号有关的参数
a=0.4;
T=1; %采样率
w0=2.0734;
x=A*exp(-a*n*T).*sin(w0*n*T); %pi 是 MATLAB 定义的  $\pi$ ，信号乘可采用“.*”
close all %清除已经绘制的 x(n)图形
subplot(3,1,1);stem(x); %绘制 x(n)的图形
title('理想采样信号序列');
k=-25:25;
W=(pi/12.5)*k;
X=x*(exp(-j*pi/12.5)).^(n'*k);
magX=abs(X); %绘制 x(n)的幅度谱
subplot(3,1,2);stem(magX);title('理想采样信号序列的幅度谱');
angX=angle(X); %绘制 x(n)的相位谱
```

```
subplot(3,1,3);stem(angX) ; title ('理想采样信号序列的相位谱')
```

## 2、单位脉冲序列

在 MatLab 中，这一函数可以用 zeros 函数实现：

```
n=1:50; %定义序列的长度是 50
x=zeros(1,50); %注意：MATLAB 中数组下标从 1 开始
x(1)=1;
close all;
subplot(3,1,1);stem(x);title('单位冲击信号序列');
k=-25:25;
X=x*(exp(-j*pi/12.5)).^(n'*k);
magX=abs(X); %绘制 x(n)的幅度谱
subplot(3,1,2);stem(magX);title('单位冲击信号的幅度谱');
angX=angle(X); %绘制 x(n)的相位谱
subplot(3,1,3);stem(angX) ; title ('单位冲击信号的相位谱')
```

## 3、矩形序列

```
n=1:50
x=sign(sign(10-n)+1);
close all;
subplot(3,1,1);stem(x);title('单位冲击信号序列');
k=-25:25;
X=x*(exp(-j*pi/25)).^(n'*k);
magX=abs(X); %绘制 x(n)的幅度谱
subplot(3,1,2);stem(magX);title('单位冲击信号的幅度谱');
angX=angle(X); %绘制 x(n)的相位谱
subplot(3,1,3);stem(angX) ; title ('单位冲击信号的相位谱')
```

## 4、特定冲击串： $x(n) = \delta(n) + 2.5\delta(n-1) + 2.5\delta(n-2) + \delta(n-3)$

```
n=1:50; %定义序列的长度是 50
x=zeros(1,50); %注意：MATLAB 中数组下标从 1 开始
x(1)=1;x(2)=2.5;x(3)=2.5;x(4)=1;
close all;
subplot(3,1,1);stem(x);title('单位冲击信号序列');
k=-25:25;
X=x*(exp(-j*pi/12.5)).^(n'*k);
magX=abs(X); %绘制 x(n)的幅度谱
subplot(3,1,2);stem(magX);title('单位冲击信号的幅度谱');
angX=angle(X); %绘制 x(n)的相位谱
subplot(3,1,3);stem(angX) ; title ('单位冲击信号的相位谱')
```

## 5、卷积计算： $y(n) = x(n) * h(n) = \sum_{m=-\infty}^{+\infty} x(m)h(n-m)$

在 MATLAB 中。提供了卷积函数 conv，即  $y=\text{conv}(x,h)$ ，调用十分方便。例如：

系统：  $h_b(n) = \delta(n) + 2.5\delta(n-1) + 2.5\delta(n-2) + \delta(n-3)$

信号：  $x_a(t) = Ae^{-\omega t} \sin(\Omega_0 nT), 0 \leq n < 50$

```
n=1:50; %定义序列的长度是 50
hb=zeros(1,50); %注意：MATLAB 中数组下标从 1 开始
hb(1)=1;hb(2)=2.5;hb(3)=2.5;hb(4)=1;
close all;
subplot(3,1,1);stem(hb);title('系统 hb[n]');
m=1:50; %定义序列的长度是 50
A=444.128; %设置信号有关的参数
a=50*sqrt(2.0)*pi;
T=0.001; %采样率
w0=50*sqrt(2.0)*pi;
x=A*exp(-a*m*T).*sin(w0*m*T); %pi 是 MATLAB 定义的  $\pi$ ，信号乘可采用“.*”
subplot(3,1,2);stem(x);title('输入信号 x[n]');
y=conv(x,hb);
subplot(3,1,3);stem(y);title('输出信号 y[n]');
```

## 6、卷积定律验证

```
k=-25:25;
X=x*(exp(-j*pi/12.5)).^(n'*k);
magX=abs(X); %绘制 x(n)的幅度谱
subplot(3,2,1);stem(magX);title('输入信号的幅度谱');
angX=angle(X); %绘制 x(n)的相位谱
subplot(3,2,2);stem(angX);title('输入信号的相位谱');

Hb=hb*(exp(-j*pi/12.5)).^(n'*k);
magHb=abs(Hb); %绘制 hb(n)的幅度谱
subplot(3,2,3);stem(magHb);title('系统响应的幅度谱');
angHb=angle(Hb); %绘制 hb(n)的相位谱
subplot(3,2,4);stem(angHb);title('系统响应的相位谱');

n=1:99;
k=1:99;
Y=y*(exp(-j*pi/12.5)).^(n'*k);
magY=abs(Y); %绘制 y(n)的幅度谱
subplot(3,2,5);stem(magY);title('输出信号的幅度谱');
angY=angle(Y); %绘制 y(n)的相位谱
subplot(3,2,6);stem(angY);title('输出信号的相位谱');
```

% 以下将验证的结果显示

```

XHb=X.*Hb;
Subplot(2,1,1);stem(abs(XHb));title('x(n)的幅度谱与 hb(n)幅度谱相乘');
Subplot(2,1,2);stem(abs(Y));title('y(n)的幅度谱'); axis([0,60,0,8000])

```

## 附录二 用 FFT 进行信号的频谱分析

1、高斯序列：

$$x_a(n) = \begin{cases} e^{-\frac{(n-p)^2}{q}}, & 0 \leq n \leq 15 \\ 0, & \text{else} \end{cases}$$

```

n=0:15; %定义序列的长度是 15
p=8;q=2;
x=exp(-1*(n-p).^2/q);
close all;
subplot(3,1,1);
stem(abs(fft(x)))
p=8;q=4;
x=exp(-1*(n-p).^2/q);
subplot(3,1,2);
stem(abs(fft(x)))
p=8;q=8;
x=exp(-1*(n-p).^2/q);
subplot(3,1,3);
stem(abs(fft(x)))

```

2、衰减正弦序列：

$$x_b(n) = \begin{cases} e^{-an} \sin 2\pi fn, & 0 \leq n \leq 15 \\ 0, & \text{else} \end{cases}$$

```

n=0:15; %定义序列的长度是 15
a=0.1;f=0.0625;
x=exp(-a*n).*sin(2*pi*f*n);
close all;
subplot(2,1,1);
stem(x);
subplot(2,1,2);
stem(abs(fft(x)))

```

3、三角波序列：

$$x_c(n) = \begin{cases} n+1, & 0 \leq n \leq 3 \\ 8-n, & 4 \leq n \leq 7 \\ 0, & \text{else} \end{cases}$$

```

for i=0:3
    x(i)=i+1;x(i+4)=8-(i+4);
end
for i=8:15
    x(i)=0;
end
close all;
subplot(2,1,1);
stem(x);
subplot(2,1,2);
stem(abs(fft(x,16)))

```

4、反三角序列：

$$x_d(n) = \begin{cases} 4-n, & 0 \leq n \leq 3 \\ n-3, & 4 \leq n \leq 7 \\ 0, & \text{else} \end{cases}$$

## 附录三 窗函数法设计 FIR 滤波器

一、在 MATLAB 中产生窗函数十分简单：

(1) 矩形窗 (Rectangle Window)

调用格式：w=boxcar(n)，根据长度 n 产生一个矩形窗 w。

(2) 三角窗 (Triangular Window)

调用格式：w=triang(n)，根据长度 n 产生一个三角窗 w。

(3) 汉宁窗 (Hanning Window)

调用格式：w=hanning(n)，根据长度 n 产生一个汉宁窗 w。

(4) 海明窗 (Hamming Window)

调用格式：w=hamming(n)，根据长度 n 产生一个海明窗 w。

(5) 布拉克曼窗 (Blackman Window)

调用格式：w=blackman(n)，根据长度 n 产生一个布拉克曼窗 w。

(6) 恺撒窗 (Kaiser Window)

调用格式：w=kaiser(n,beta)，根据长度 n 和影响窗函数旁瓣的 β 参数产生一个恺撒窗 w。

二、基于窗函数的 FIR 滤波器设计

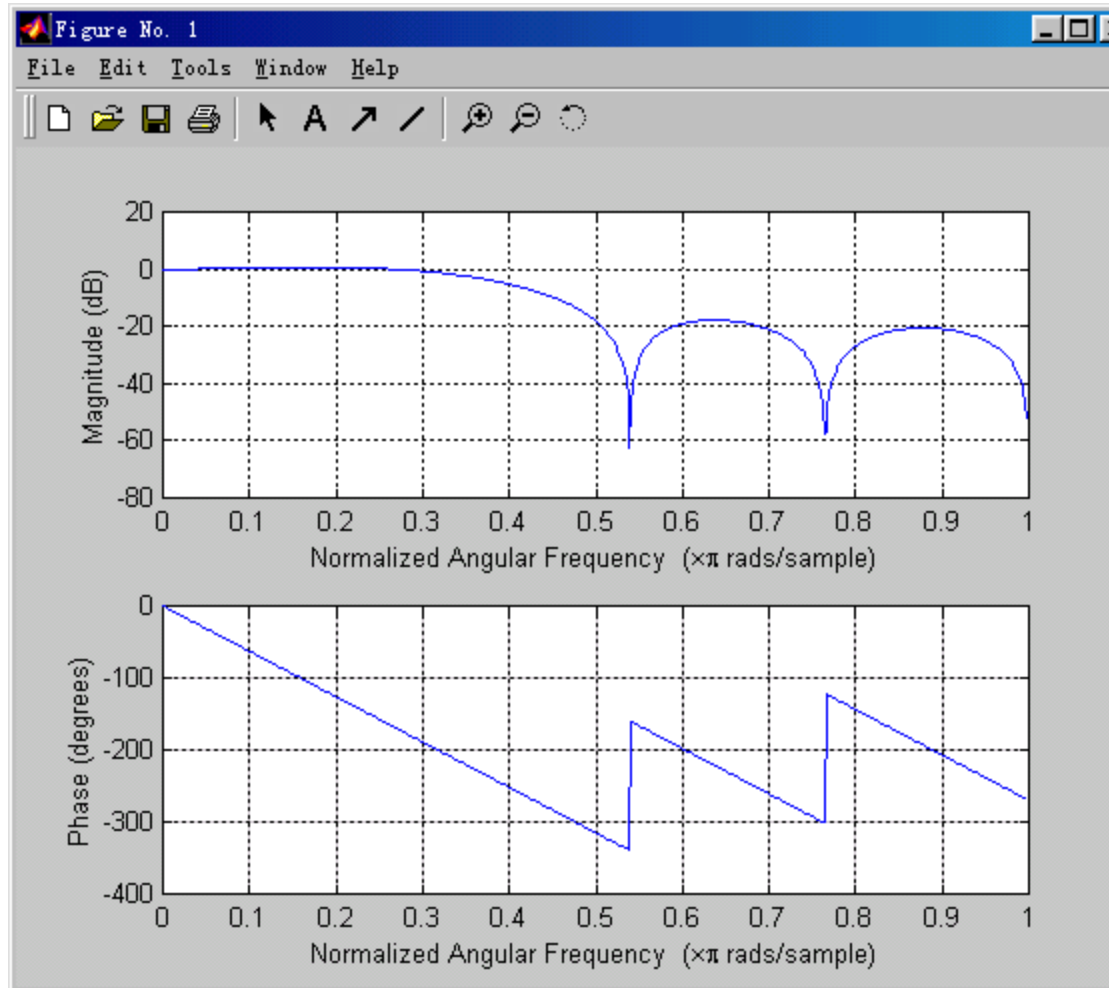
利用 MATLAB 提供的函数 firl 来实现

调用格式：firl(n,Wn,'ftype',Window)，n 为阶数、Wn 是截止频率（如果输入是形如[W1 W2] 的矢量时，本函数将设计带通滤波器，其通带为  $W1 < \omega < W2$ ）、ftype 是滤波器的类型（低通-省略该参数、高通-ftype=high、带阻-ftype=stop）、Window 是窗函数。

[例] 设计一个长度为 8 的线性相位 FIR 滤波器。

其理想幅频特性满足  $|H_d(e^{j\omega})| = \begin{cases} 1, 0 \leq \omega \leq 0.4\pi \\ 0, \text{else} \end{cases}$

```
Window=boxcar(8);
b=fir1(7,0.4,Window);
freqz(b,1)
```



```
Window=blackman(8);
b=fir1(7,0.4,Window);
freqz(b,1)
```

[例] 设计线性相位带通滤波器，其长度  $N=15$ ，上下边带截止频率分别为  $\omega_1=0.3\pi$ ， $\omega_2=0.5\pi$

```
Window=blackman(16);
b=fir1(15,[0.3 0.5],Window);
freqz(b,1)
```

设计指标为：  $\omega_p=0.2\pi$        $R_p=0.25\text{dB}$        $\omega_a=0.3\pi$        $A_s=50\text{dB}$   
 的低通数字 FIR 滤波器

```
wp=0.2*pi;ws=0.3*pi;
tr_width=ws-wp;
M=ceil(6.6*pi/tr_width)+1;
N=[0:1:M-1];
wc=(ws+wp)/2;
hd=ideal_lp(wc,M);
w_ham=(boxcar(M))';
h=hd.*w_ham;
[db,mag,pha,grd,w]=freqz_m(h,[1]);
delta_w=2*pi/1000;
Rp=-(min(db(1:1:wp/delta_w+1)));
As=-round(max(db(ws/delta_w+1:1:501)));
Close all;
subplot(2,2,1);stem(hd);title('理想冲击响应')
axis([0 M-1 -0.1 0.3]);ylabel('hd[n]');
subplot(2,2,2);stem(w_ham);title('汉明窗');
axis([0 M-1 0 1.1]);ylabel('w[n]');
subplot(2,2,3);stem(h);title('实际冲击响应');
axis([0 M-1 -0.1 0.3]);ylabel('h[n]');
subplot(2,2,4);plot(w/pi,db); title('衰减幅度');
axis([0 1 -100 10]);ylabel('Decibels');
```

## 附录四 IIR 滤波器的实现

MATLAB 中滤波器的分析和实现

1、freqs 函数：模拟滤波器的频率响应

[例] 系统传递函数为  $H(s) = \frac{0.2s^2 + 0.3s + 1}{s^2 + 0.4s + 1}$  的模拟滤波器，在 MATLAB 中可以用以下

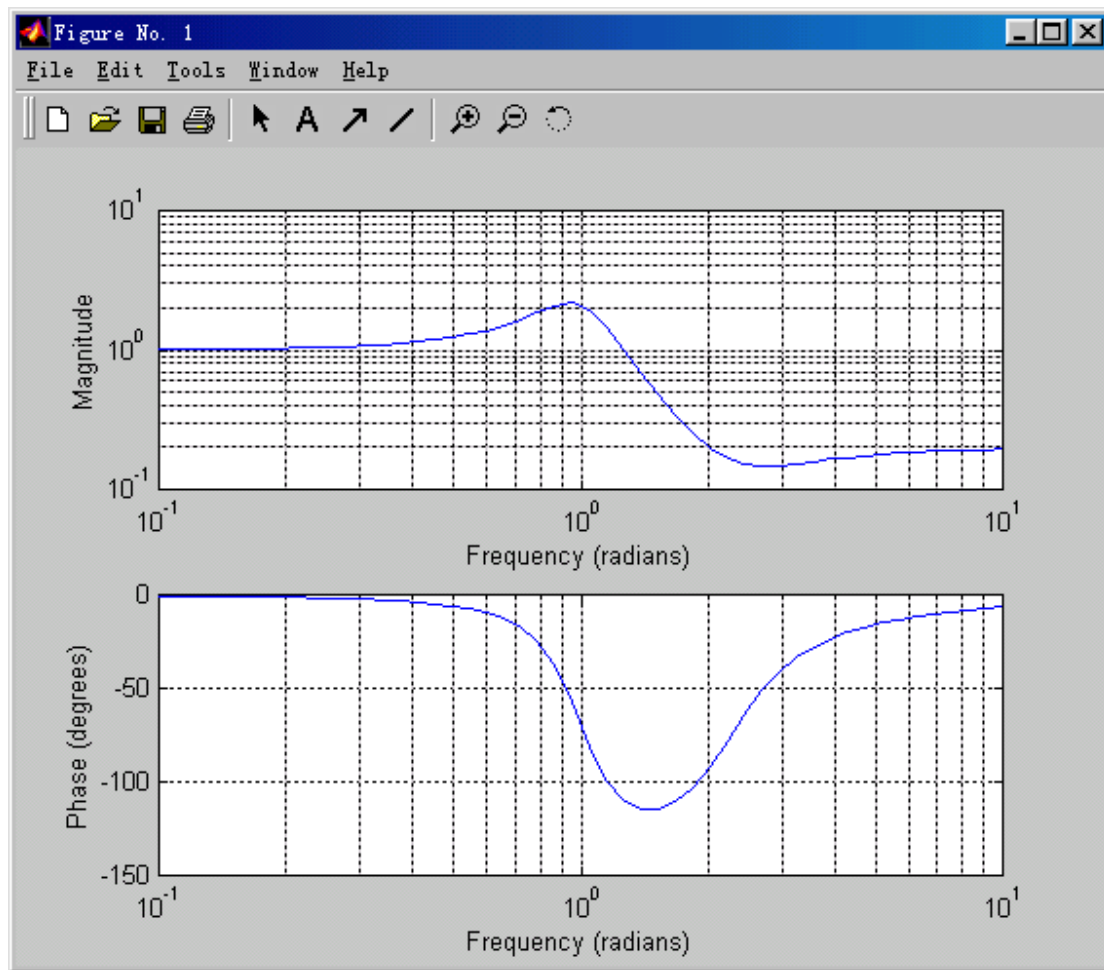
程序来实现：

```
a=[1 0.4 1];
```

```
b=[0.2 0.3 1];
```

```
%w=logspace(-1,1);                      %产生从  $10^{-1}$  到  $10^1$  之间地 0 个等间距点，即 50 个频率点
```

```
freqs(b,a,w)                              %根据输入的参数绘制幅度谱和相位谱
```



## 2、freqz 函数：数字滤波器的频率响应

[例] 系统传递函数为  $H(z) = \frac{0.2 + 0.3z + z^{-2}}{1 + 0.4z + z^{-2}}$  的模拟滤波器，在 MATLAB 中可以用以下

程序来实现：

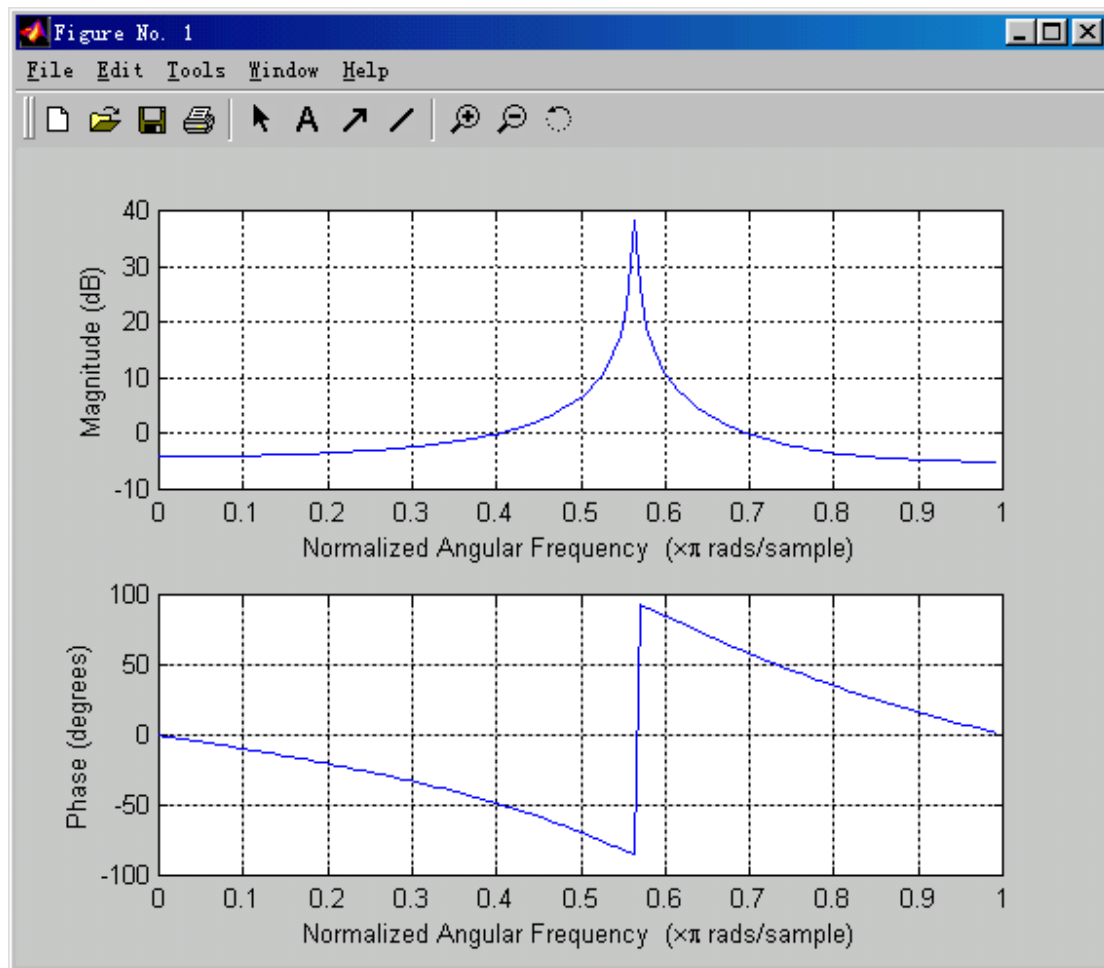
```
a=[1 0.4 1];
```

```
b=[0.2 0.3 1];
```

%根据输入的参数绘制幅度谱和相位谱，得到 0 到  $\pi$  之间 128 个点处的频率响应

```
freqz(b,a,128)
```





### 3、ButterWorth 模拟和数字滤波器

(1) `butterd` 函数：ButterWorth 滤波器阶数的选择。

调用格式：`[n,Wn]=butterd(Wp,Ws,Rp,Rs)`，在给定滤波器性能的情况下（通带临界频率  $W_p$ 、阻带临界频率  $W_s$ 、通带内最大衰减  $R_p$  和阻带内最小衰减  $R_s$ ），计算 ButterWorth 滤波器的阶数  $n$  和截止频率  $W_n$ 。

相同参数条件下的模拟滤波器则调用格式为：`[n,Wn]=butterd(Wp,Ws,Rp,Rs,'s')`

(2) `butter` 函数：ButterWorth 滤波器设计。

调用格式：`[b,a]=butter(n,Wn)`，根据阶数  $n$  和截止频率  $W_n$  计算 ButterWorth 滤波器分子分母系数（ $b$  为分子系数的矢量形式， $a$  为分母系数的矢量形式）。

相同参数条件下的模拟滤波器则调用格式为：`[b,a]=butter(n,Wn,'s')`

[例] 采样频率为 1Hz，通带临界频率  $f_p = 0.2\text{Hz}$ ，通带内衰减小于 1dB ( $\alpha_p=1$ )；阻带临界频率  $f_s=0.3\text{Hz}$ ，阻带内衰减大于 25dB ( $\alpha_s=25$ )。设计一个数字滤波器满足以上参数。

```
[n,Wn]=butterd(0.2,0.3,1,25);
```

```
[b,a]=butter(n,Wn);
```

```
freqz(b,a,512,1);
```

### 4、Chebyshev 模拟和数字滤波器

(1) **cheb1ord** 函数: Chebyshev I 型 II 滤波器阶数计算。

调用格式: **[n,Wn]=cheb1ord(Wp,Ws,Rp,Rs)**, 在给定滤波器性能的情况下 (通带临界频率 **Wp**、阻带临界频率 **Ws**、通带内波纹 **Rp** 和阻带内衰减 **Rs**), 选择 Chebyshev I 型滤波器的最小阶 **n** 和截止频率 **Wn**。

(2) **cheby1** 函数: Chebyshev I 型滤波器设计。

调用格式: **[b,a]=butter(n,Rp,Wn)**, 根据阶数 **n**、通带内波纹 **Rp** 和截止频率 **Wn** 计算 ButterWorth 滤波器分子分母系数 (**b** 为分子系数的矢量形式, **a** 为分母系数的矢量形式)。

注: Chebyshev II 型滤波器所用函数和 I 型类似, 分别是 **cheb2ord**、**cheby2**。

[例] 实现上例中的滤波器

```
[n,Wn]=cheb1ord(0.2,0.3,1,25);
```

```
[b,a]=cheby1(n,1,Wn);
```

```
freqz(b,a,512,1);
```

(1) 脉冲响应不变法设计数字 ButterWorth 滤波器

调用格式: **[bz,az]=impinvar(b,a,Fs)**, 再给定模拟滤波器参数 **b**, **a** 和取样频率 **Fs** 的前提下, 计算数字滤波器的参数。两者的冲激响应不变, 即模拟滤波器的冲激响应按 **Fs** 取样后等同于数字滤波器的冲激响应。

(2) 利用双线性变换法设计数字 ButterWorth 滤波器

调用格式: **[bz,az]=bilinear[b,a,Fs]**, 根据给定的分子 **b**、分母系数 **a** 和取样频率 **Fs**, 根据双线性变换将模拟滤波器变换成离散滤波器, 具有分子系数向量 **bz** 和分母系数向量 **az**。

模拟域的 **butter** 函数说明与数字域的函数说明相同

**[b,a]=butter(n,Wn,'s')**可以得到模拟域的 Butterworth 滤波器

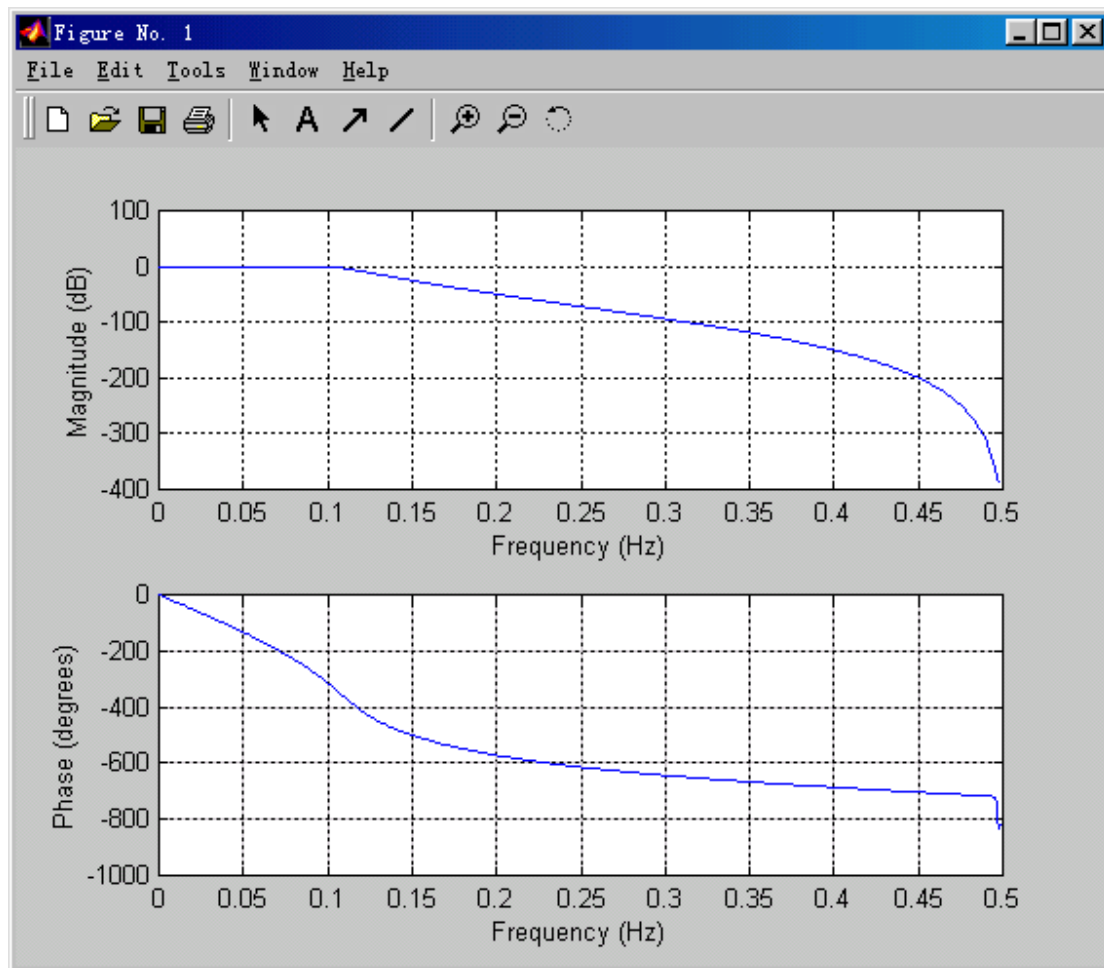
[例] 采样频率为 1Hz, 通带临界频率  $f_p = 0.2\text{Hz}$ , 通带内衰减小于 1dB ( $\alpha_p=1$ ); 阻带临界频率  $f_s=0.3\text{Hz}$ , 阻带内衰减大于 25dB ( $\alpha_s=25$ )。设计一个数字滤波器满足以上参数。

%直接设计数字滤波器

```
[n,Wn]=buttord(0.2,0.3,1,25);
```

```
[b,a]=butter(n,Wn);
```

```
freqz(b,a,512,1);
```



%脉冲响应不变法设计数字滤波器

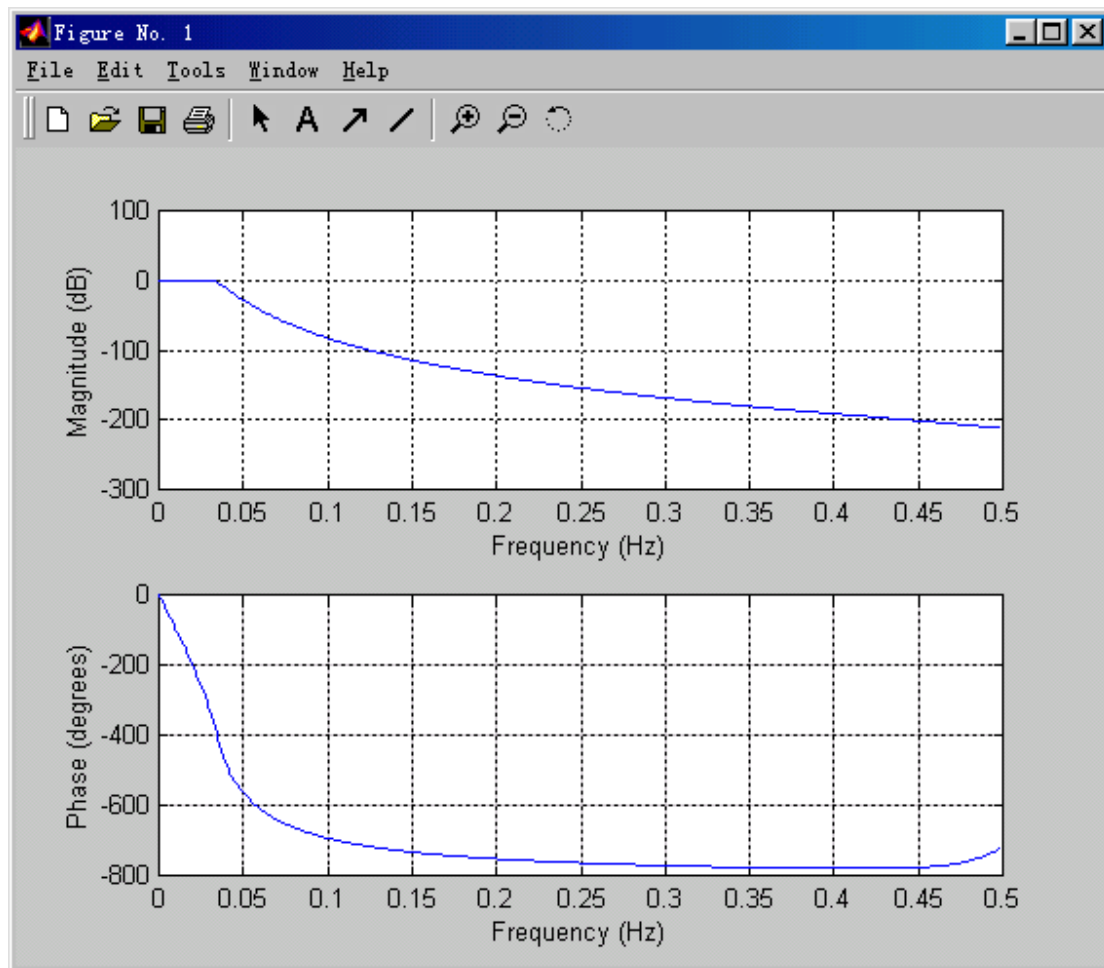
```
[n,Wn]=buttord(0.2,0.3,1,25,'s');
```

```
[b,a]=butter(n,Wn,'s');
```

```
freqs(b,a)
```

```
[bz,az]=impinvar(b,a,1);
```

```
freqz(bz,az,512,1)
```



% 双线性变换法设计 ButterWorth 数字滤波器

```
[n,Wn]=buttord(0.2,0.3,1,25,'s');
```

```
[b,a]=butter(n,Wn,'s');
```

```
freqs(b,a)
```

```
[bz,az]=bilinear(b,a,1);
```

```
freqz(bz,az,512,1)
```

