

## 实验八 数字音频信号分频与去噪实验

### 一、 实验目的

1. 复习巩固数字信号处理的基本理论知识；
2. 了解数字音频信号分频基本原理，掌握数字音频信号分频方法；
3. 了解数字音频信号去噪基本原理，掌握数字音频信号去噪方法；
4. 综合运用所学知识处理实际工程应用问题。

### 二、 实验原理与方法

#### 1. 数字音频信号分频

**原理：**扬声器是日常生活中的常见物品。人耳能听到的声音频率范围为20Hz~20000Hz，但由于很难用单线圈扬声器做到完美地覆盖大部分音频频率范围或某一特定的低频范围，因此为了得到良好的音质，扬声器一般有两个或多个喇叭。高保真音响系统通常都把全频段的音频信号按频率高低分隔成两个或多个频段，分别送往相应频段的用扬声器单元去播放，以求得到互调失真较小、音频较宽的音响效果。负责把音频信号分成若干频段的装置称为分频器。

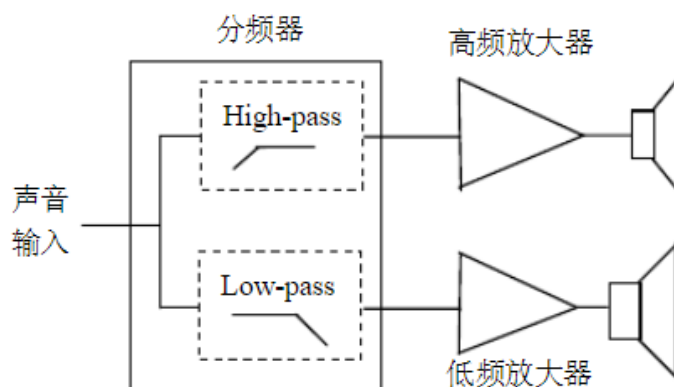


图 8-1 双频段数字音频分频系统示意图

图 8-1 中给出的是双频段数字音频分频系统，即将音频分成高频和低频两段。现在的音频播放系统中，大都采用数字分频器。数字音频信号通过高通系统后，得到的输出信号是输入信号的高频部分。而通过低通系统后，得到的输出信号是输入信号中的低频部分。高频部分和低频部分分别送往高频和低频扬声器进行播放。

**方法：**高通滤波器和低通滤波器可以用 IIR 或 FIR 滤波器设计实现。具体设

计与实现方法可以参考实验六和实验七。

## 2. 数字音频信号去噪

**原理：**在用固定电话或者是移动电话进行语音通信的时候，信号不好时经常会有各式各样的噪声，使得通话过程不顺畅。换句话说，噪声可能会严重污染音频信号的质量。同样，在听音乐的过程中，由于各种噪声的影响，也会使音乐休闲的效果大打折扣。音频信号的带宽比较小，并且其频率主要集中在低频段，通常是 2 kHz 以下，而噪声的带宽一般很大。为了提高音频信号的质量，往往需要滤除高频的噪声。

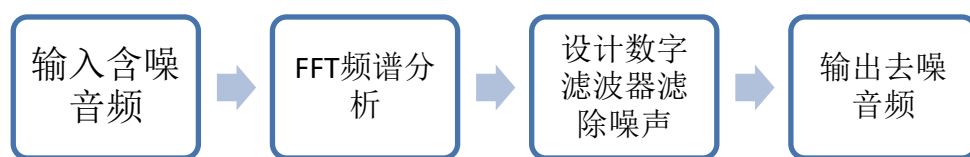


图 8-2 数字音频信号去噪基本过程

**方法：**MATLAB 中利用函数 `audioread` 和 `audiowrite` 可以读取和生成音频信号，函数 `sound` 可以播放音乐。数字滤波器的具体设计与实现方法可以参考实验六和实验七。其中音频处理函数 `audioread`、`audiowrite` 和 `sound` 的使用说明如下：

### 1) `audioread` 函数

**格式：**`[y,Fs] = audioread(filename)`

`[y,Fs] = audioread(filename,samples)`

`[y,Fs] = audioread(___,dataType)`

**说明：**`[y,Fs] = audioread(filename)` 从名为 `filename` 的文件中读取数据，并返回样本数据 `y` 以及该数据的采样率 `Fs`。

`[y,Fs] = audioread(filename,samples)` 读取文件中所选范围的音频样本，其中 `samples` 是 `[start,finish]` 格式的向量。也就是说你可以得到任意一段的音频数据。

`[y,Fs] = audioread(___,dataType)` 返回数据范围内与 `dataType`（‘native’或 ‘double’）对应的采样数据，可以包含先前语法中的任何输入参数。

### 2) `audiowrite` 函数

**格式：**`audiowrite(filename,y,Fs)`

`audiowrite(filename,y,Fs,Name,Value)`

**说明：**audiowrite(filename,y,Fs) 以采样率 Fs 将音频数据矩阵 y 写入名为 filename 的文件。filename 输入还指定了输出文件格式。输出数据类型取决于音频数据 y 的输出文件格式和数据类型。

audiowrite(filename,y,Fs,Name,Value) 使用一个或多个 Name,Value 对组参数指定的其他选项。

### 3) sound 函数

**格式：**

sound(y)

sound(y, Fs)

sound(y, Fs, nBits)

**说明：**

y 为音频信号矩阵，Fs 为采样率，即单位时间的样本个数 (Hz)，nBits 表示每个样本点用几个比特表示，即分辨率。第一种用法默认 Fs 为 8192Hz。

## 三、 实验内容

### 1. 数字音频信号分频实验

设计数字分频器，输入事先准备好的音频信号文件，要求绘制低通滤波器和高通滤波器的冲激响应和幅频响应图形，并绘制输入音频信号、低通滤波器输出信号和高通滤波器输出信号的时域和频域波形图。其中，数字分频器的采样频率为 44100Hz，低通滤波器和高通滤波器的截止频率为 1000Hz，过度带宽为 600~1400Hz。低通滤波器的通带起伏为 0.02dB，阻带衰减为 50dB，即 0~600Hz 范围内的起伏为 0.02dB，1400Hz 处的衰减为 50dB。高通滤波器的通带起伏和阻带衰减也为分别为 0.02dB 和 50dB，即 1.4~44.1kHz 范围的起伏为 0.02dB，600Hz 处的衰减为 50dB。

**代码：**% 数字音频信号分频实验。

```
fs = 44100; %采样频率
```

```
wc=2*pi*1000/fs; %截止频率
```

```
N=183; %滤波器长度
```

```
hL = fir1(N-1,wc/pi); %计算低通滤波器系数
```

```
hH = fir1(N-1,wc/pi,'high',hamming(N));%计算高通滤波器系数
```

```

figure;%绘图，分频器低通和高通系统单位冲激响应
subplot(2,1,1);stem(hL);grid;xlabel('n');ylabel('x(n)');title('低通滤波器单位冲激响应');
subplot(2,1,2);stem(hH);grid;xlabel('n');ylabel('|X(f)|');title('高通滤波器单位冲激响应');

[HL f]=freqz(hL,1,44100,fs);%计算分频器低通系统频率响应
[HH f]=freqz(hH,1,44100,fs);%计算分频器高通系统频率响应
figure;%绘图，分频器低通和高通系统幅频响应
subplot(2,1,1);semilogx(f,20*log10(abs(HL)));grid
axis([1 45000 -200 20]);
xlabel('Hz');ylabel('dB');
title('低通滤波器幅频响应');
subplot(2,1,2);semilogx(f,20*log10(abs(HH)));grid
axis([1 45000 -200 20]);
xlabel('Hz');ylabel('dB');
title('高通滤波器幅频响应');

xt = audioread('we.wav');%读取音频信号
x = xt(:,1);%提取其中的一个声道
N=length(x);
f1=[0:N/2]*fs/N;
magX=2*abs(fft(x))/N;magX(1)=magX(1)/2;%分析输入音频信号的频谱
figure;%绘图，输入音频信号的时域和频域波形
subplot(2,1,1);plot(x);
axis([1 N -0.1 0.15]);
xlabel('n');ylabel('x(n)');
title('输入音频信号的时域波形');grid;
subplot(2,1,2);semilogx(f1,magX(1:N/2+1));
axis([1 N/2+1 0 0.003]);
xlabel('Hz');ylabel('|X(f)|');

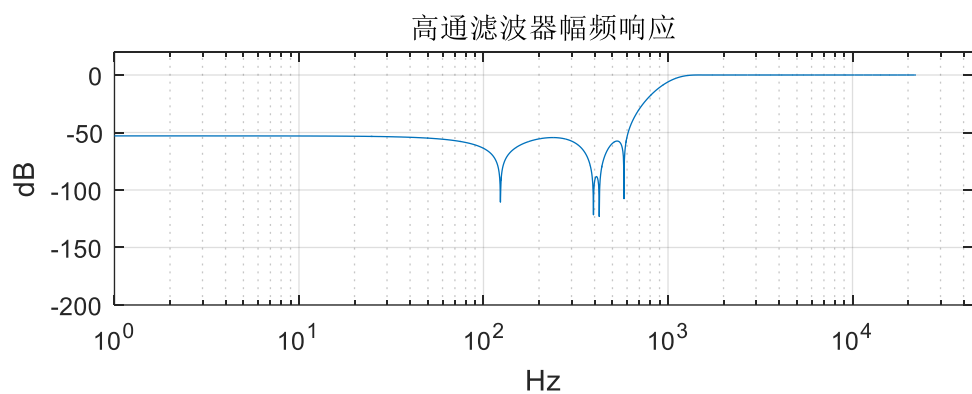
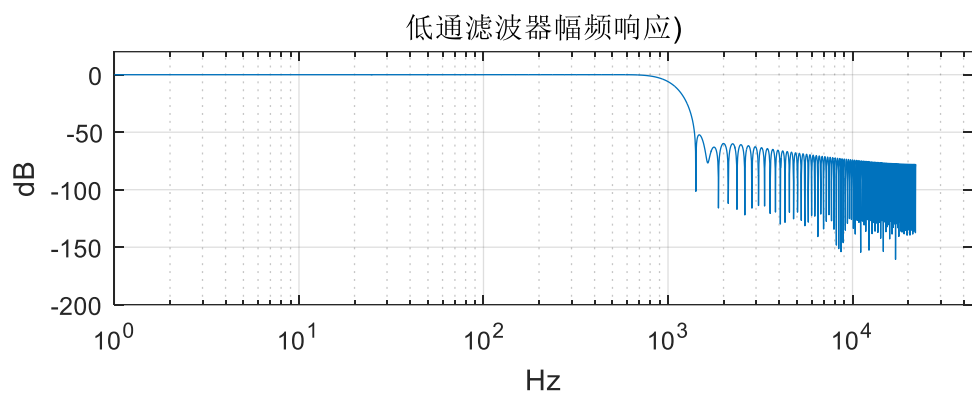
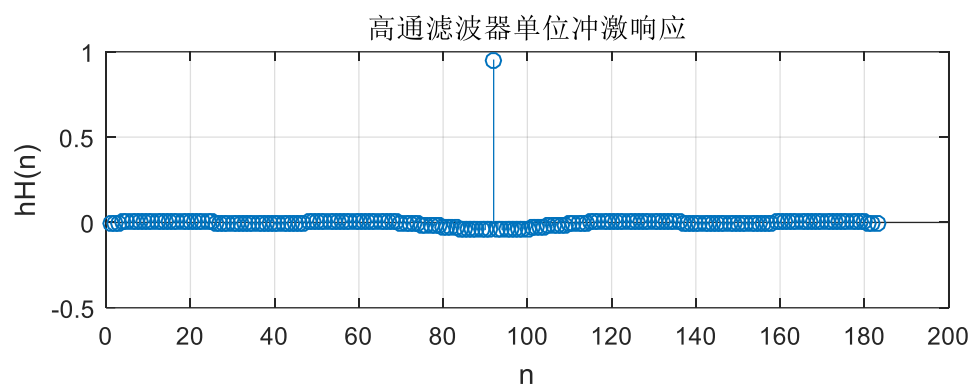
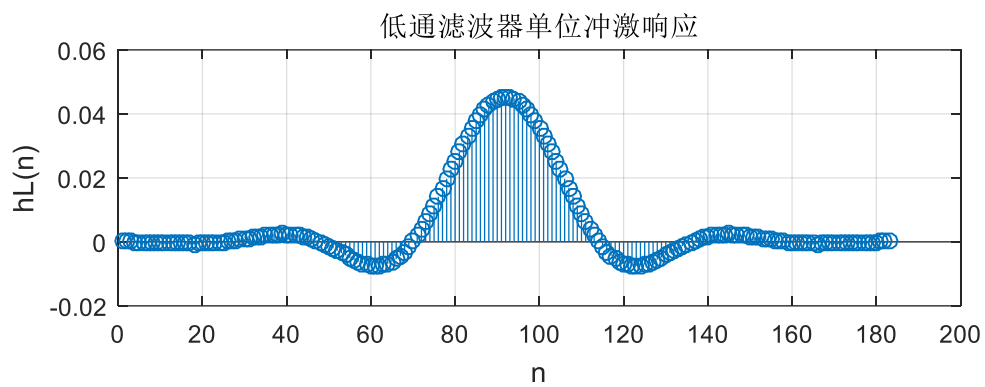
```

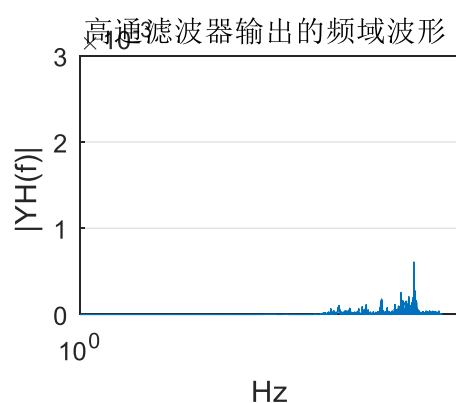
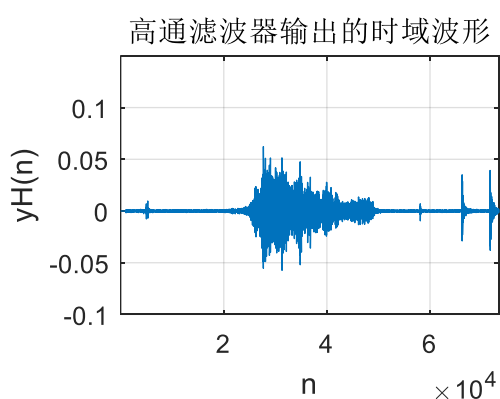
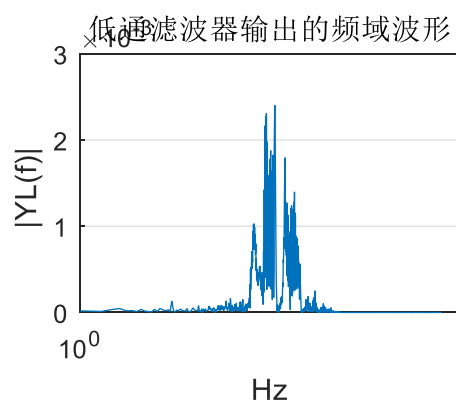
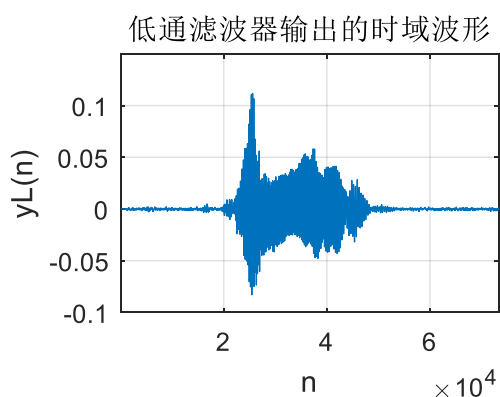
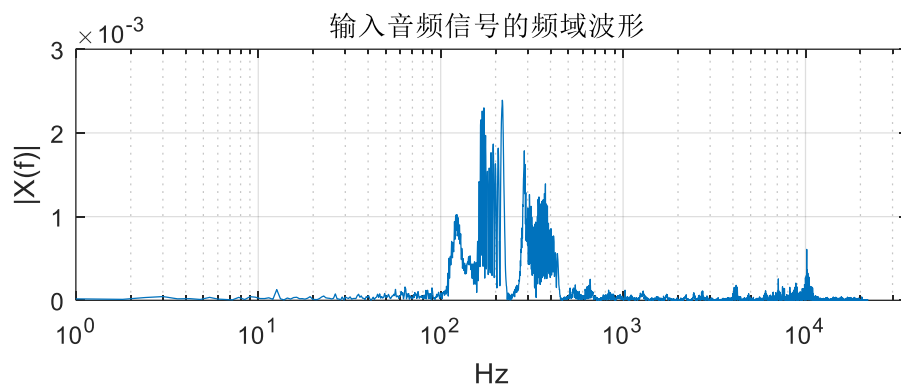
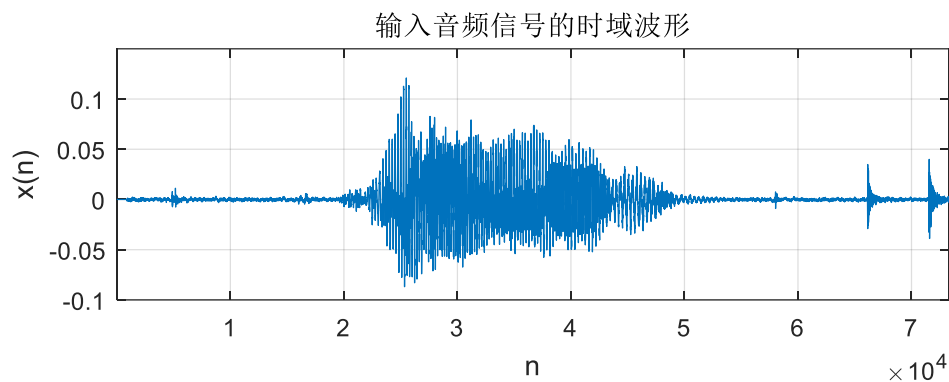
```

title('输入音频信号的频域波形 ');grid;
yH=filter(hH,1,x);%音频信号通过分频器高通系统
yL=filter(hL,1,x);%音频信号通过分频器低通系统
magYH=2*abs(fft(yH))/N;magYH(1)=magYH(1)/2;%分析高通系统输出信号
的频谱
magYL=2*abs(fft(yL))/N;magYL(1)=magYL(1)/2;%分析低通系统输出信号的
频谱
figure;%绘图，低通系统输出的时域和频域波形，高通系统输出的时域和频
域波形
subplot(2,2,1); plot(yL);
axis([1 N -0.1 0.15]);
xlabel('n');ylabel('yL(n)');
title('低通滤波器输出的时域波形');grid;
subplot(2,2,2);semilogx(f1,magYL(1:N/2+1));
axis([1 N/2+1 0 0.003]);
xlabel('Hz'); ylabel('|YL(f)| ');
title('低通滤波器输出的频域波形');grid;
subplot(2,2,3); plot(yH);
axis([1 N -0.1 0.15]);
xlabel('n');ylabel('yH(n)');
title('高通滤波器输出的时域波形');grid;
subplot(2,2,4);semilogx(f1,magYH(1:N/2+1));
axis([1 N/2+1 0 0.003]);
xlabel('Hz'); ylabel('|YH(f)| ');
title('高通滤波器输出的频域波形 ');grid;

```

结果：





## 2. 数字音频信号去噪

读入事先准备好的音频信号文件，在音频信号中加入随机噪声，对含噪声信

号进行频谱分析，并根据分析结果设计数字滤波器，要求绘制原始音频信号、加噪音频信号和滤波后信号的时域和频域波形图，并绘制滤波器的冲激响应、幅频响应和相位响应图形。其中，滤波器的通带边沿频率为 2000Hz，过度带宽为 1500~2500Hz，通带起伏为 0.02dB，阻带衰减为 50dB。

**代码：**% 音频信号的去噪。

```
[s,fs] = audioread('music.wav'); % 读取音频信号
Ts = 1/fs; %采样周期
M = length(s); %音频信号长度
m = 0:M-1;
sigma = mean(s.*s); %噪声方差
v = sqrt(sigma)*randn([M,1]); %噪声生成
x = s+v; %受宽带噪声干扰的音频信号
f=[0:M/2]*fs/M;
X = fft(x);
magX = abs(X); %受噪声污染音频信号的频谱
S = fft(s);
magS = abs(S); %受噪声污染音频信号的频谱
figure;%绘图，原始输入音频信号的时域和频域波形
subplot(2,1,1);plot(m,s);
xlabel('n');ylabel('s(n)');title('原始输入音频信号的时域波形')
axis([0 M+100 -1.5 1.5]);grid;
subplot(2,1,2); semilogx(f,magS(1:M/2+1));
xlabel('Hz'); ylabel('|S(f)|');title('原始输入音频信号的频域波形')
axis([0 max(f)+100*fs/M 0 2500]);grid;
figure;%绘图，受噪声污染信号的时域和频域波形
subplot(2,1,1);plot(m,x);
xlabel('n');ylabel('x(n)');title('受噪声污染信号的时域波形')
axis([0 M+100 -1.5 1.5]);grid;
subplot(2,1,2); semilogx(f,magX(1:M/2+1));
```



```
xlabel('Hz'); ylabel('|X(f)|');title('受噪声污染信号的频域波形')
axis([0 max(f)+100*fs/M 0 2500]);grid;
wc=2*pi*2000/fs;%截止频率
Nwin = 147;%滤波器长度
hwin = fir1(Nwin-1,wc/pi);%计算低通滤波器系数;
[Hwin f1]=freqz(hwin,1,44100,fs);%计算低通滤波器频率响应
magHwin = abs(Hwin);%幅频响应
phaHwin = angle(Hwin);%相频响应
phaHwin = 180*unwrap(phaHwin)/pi;
nwin = 0:Nwin-1;
figure;%绘图，低通滤波器特性
subplot(3,1,1);stem(nwin,hwin);grid;
xlabel('n');ylabel('hwin(n)');
title('低通滤波器单位冲激响应');
subplot(3,1,2);semilogx(f1,20*log10(magHwin));grid;
xlabel('Hz');ylabel('dB');
title('低通滤波器幅频响应');
subplot(3,1,3);plot(f1,phaHwin);grid;
xlabel('Hz');ylabel('degree');
title('低通滤波器相位响应');
y = filter(hwin,1,x); %滤波
Y = fft(y);
magY = abs(Y);%滤波后信号的频谱
figure;%绘图，滤波后信号的时域和频域波形
subplot(2,1,1); plot(m,y);
xlabel('n');ylabel('y(n)');
title('滤波后信号的时域波形');grid;
subplot(2,1,2); semilogx(f,magY(1:M/2+1));
xlabel('Hz'); ylabel('|Y(f)|');
```

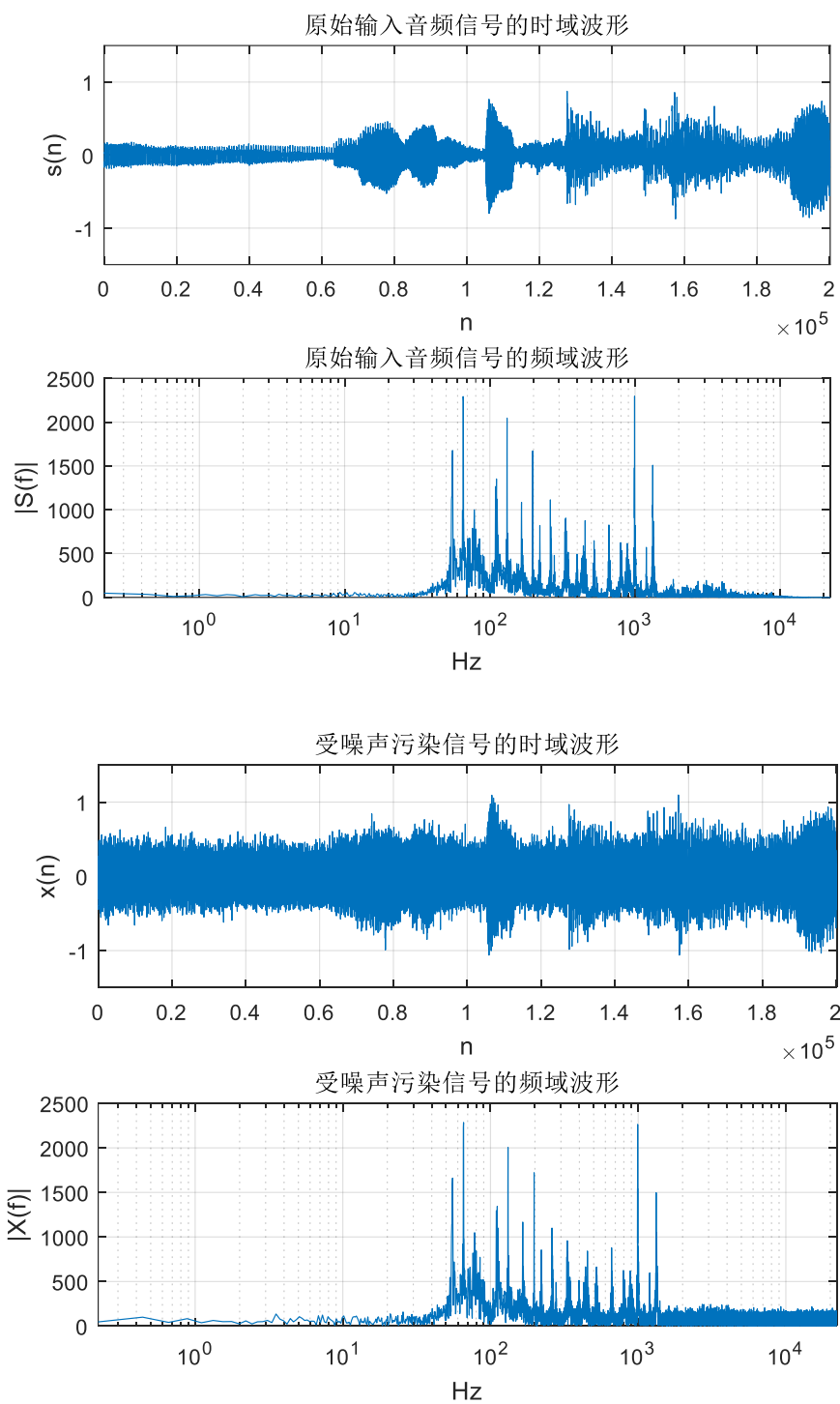
```
title('滤波后信号的频域波形');grid;
```

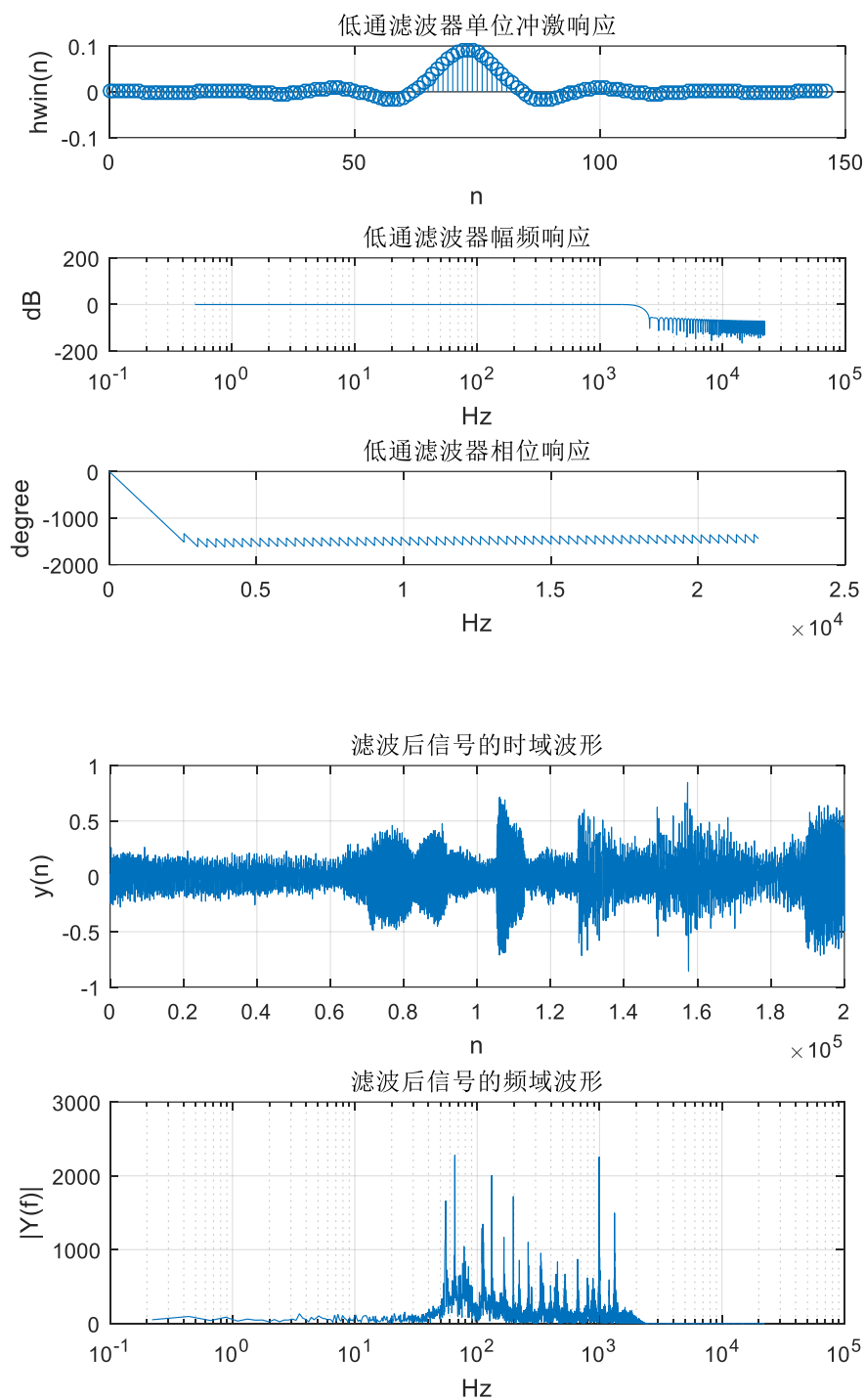
```
sound(s,fs) %播放原始音乐
```

```
sound(x,fs) %播放加噪音乐
```

```
sound(y,fs) %播放滤波后音乐
```

结果:





#### 四、 学生作业

1. 自己上网下载或自己录制一段简短的音频信号文件，对音频信号进行频谱分析，并根据分析结果设计数字分频器，要求绘制低通滤波器和高通滤波器的冲激响应和幅频响应图形，并绘制输入音频信号、低通滤波器输出信号和高通滤波器输出信号的时域和频域波形图。其中，数字分频器的具体参数可以参考实验

内容 1 数字音频信号分频实验中的参数设置。

**程序代码:**

```
% 数字音频信号分频实验。

fs = 44100; %采样频率
wc=2*pi*1000/fs; %截止频率
N=183; %滤波器长度
hL = fir1(N-1,wc/pi); %计算低通滤波器系数
hH = fir1(N-1,wc/pi,'high',hamming(N));%计算高通滤波器系数
figure;%绘图，分频器低通和高通系统单位冲激响应
subplot(2,1,1);stem(hL);grid;xlabel('n');ylabel('x(n)');title('低通滤波器单位冲激响应');
subplot(2,1,2);stem(hH);grid;xlabel('n');ylabel('|X(f)|');title('高通滤波器单位冲激响应');

[HL f]=freqz(hL,1,44100,fs);%计算分频器低通系统频率响应
[HH f]=freqz(hH,1,44100,fs);%计算分频器高通系统频率响应
figure;%绘图，分频器低通和高通系统幅频响应
subplot(2,1,1);semilogx(f,20*log10(abs(HL)));grid
axis([1 45000 -200 20]);
xlabel('Hz');ylabel('dB');
title('低通滤波器幅频响应');
subplot(2,1,2);semilogx(f,20*log10(abs(HH)));grid
axis([1 45000 -200 20]);
xlabel('Hz');ylabel('dB');
title('高通滤波器幅频响应');

xt = audioread('Test.wav');%读取音频信号
x = xt(:,1);%提取其中的一个声道
N=length(x);
f1=[0:N/2]*fs/N;
magX=2*abs(fft(x))/N;magX(1)=magX(1)/2;%分析输入音频信号的频谱
```

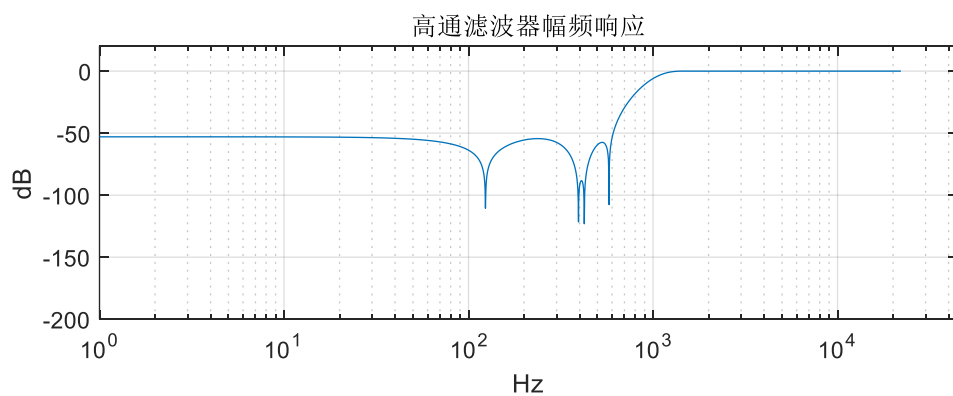
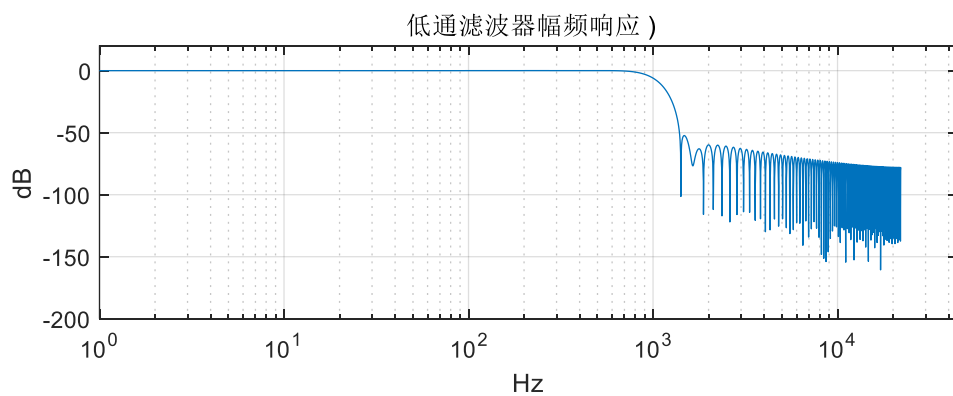
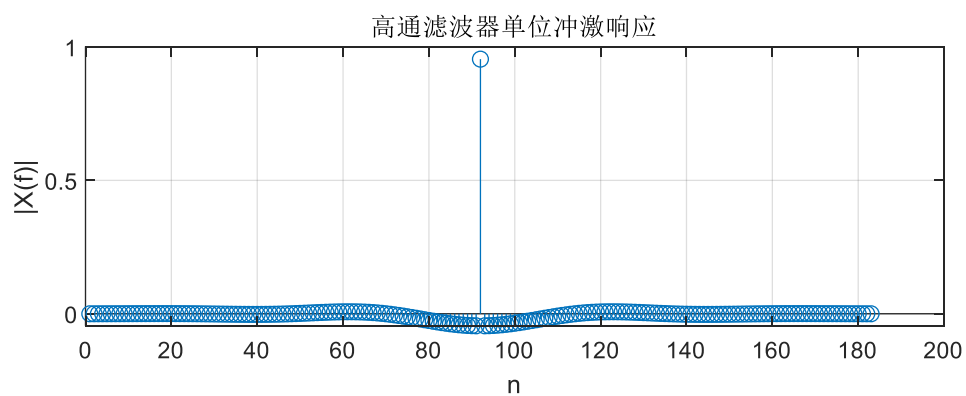
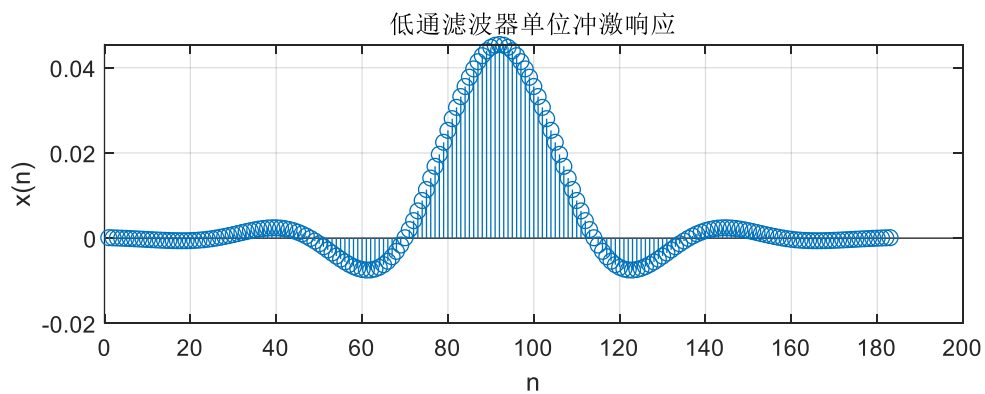
```

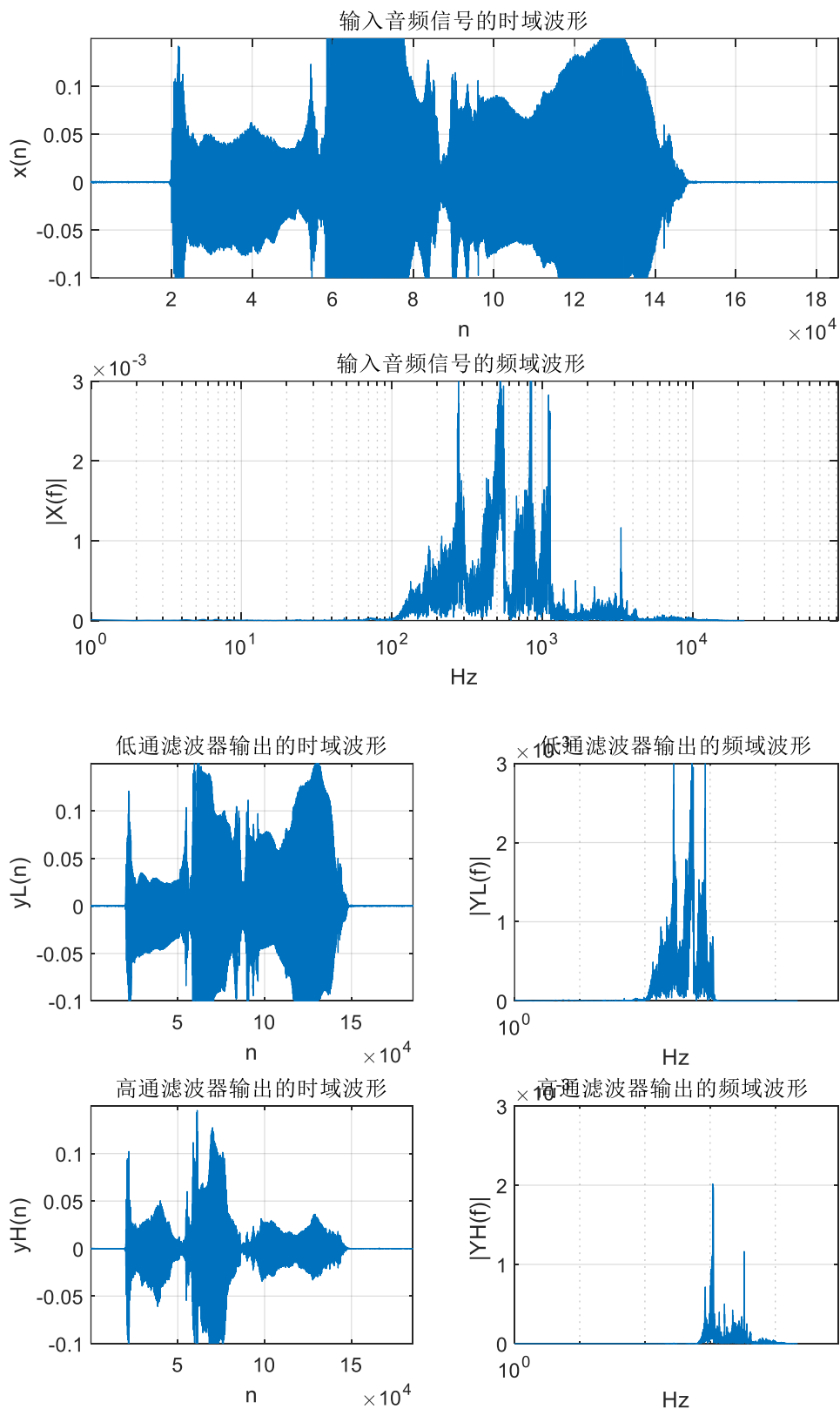
figure;%绘图，输入音频信号的时域和频域波形
subplot(2,1,1);plot(x);
axis([1 N -0.1 0.15]);
xlabel('n');ylabel('x(n)');
title('输入音频信号的时域波形');grid;
subplot(2,1,2); semilogx(f1,magX(1:N/2+1));
axis([1 N/2+1 0 0.003]);
xlabel('Hz'); ylabel('|X(f)|');
title('输入音频信号的频域波形 ');grid;
yH=filter(hH,1,x);%音频信号通过分频器高通系统
yL=filter(hL,1,x);%音频信号通过分频器低通系统
magYH=2*abs(fft(yH))/N;magYH(1)=magYH(1)/2;
magYL=2*abs(fft(yL))/N;magYL(1)=magYL(1)/2;
figure;
subplot(2,2,1); plot(yL);
axis([1 N -0.1 0.15]);
xlabel('n');ylabel('yL(n)');
title('低通滤波器输出的时域波形');grid;
subplot(2,2,2);semilogx(f1,magYL(1:N/2+1));
axis([1 N/2+1 0 0.003]);
xlabel('Hz'); ylabel('|YL(f)| ');
title('低通滤波器输出的频域波形');grid;
subplot(2,2,3); plot(yH);
axis([1 N -0.1 0.15]);
xlabel('n');ylabel('yH(n)');
title('高通滤波器输出的时域波形');grid;
subplot(2,2,4);semilogx(f1,magYH(1:N/2+1));
axis([1 N/2+1 0 0.003]);
xlabel('Hz'); ylabel('|YH(f)| ');

```

`title('高通滤波器输出的频域波形 ');grid;`

运行截图：





2. 自己上网下载或自己录制一段简短的音频信号文件，在音频信号中加入随机噪声，对含噪声信号进行频谱分析，并根据分析结果设计数字滤波器，要求绘制原始音频信号、加噪音频信号和滤波后信号的时域和频域波形图，并绘制滤

波器的冲激响应、幅频响应和相位响应图形。数字滤波器的具体参数可以参考实验内容 2 数字音频信号去噪实验中的参数设置。

**程序代码：**

```
% 音频信号的去噪。

[s,fs] = audioread('BGM.wav'); % 读取音频信号

Ts = 1/fs; %采样周期

M = length(s); %音频信号长度

m = 0:M-1;

sigma = mean(s.*s); %噪声方差

v = sqrt(sigma).*randn([M,1]); %噪声生成

x = s+v; %受宽带噪声干扰的音频信号

f=[0:M/2]*fs/M;

X = fft(x);

magX = abs(X); %受噪声污染音频信号的频谱

S = fft(s);

magS = abs(S); %受噪声污染音频信号的频谱

figure;%绘图，原始输入音频信号的时域和频域波形

subplot(2,1,1);plot(m,s);

xlabel('n');ylabel('s(n)');title('原始输入音频信号的时域波形')

axis([0 M+100 -1.5 1.5]);grid;

subplot(2,1,2); semilogx(f,magS(1:M/2+1));

xlabel('Hz'); ylabel('|S(f)|');title('原始输入音频信号的频域波形')

axis([0 max(f)+100*fs/M 0 2500]);grid;

figure;%绘图，受噪声污染信号的时域和频域波形

subplot(2,1,1);plot(m,x);

xlabel('n');ylabel('x(n)');title('受噪声污染信号的时域波形')

axis([0 M+100 -1.5 1.5]);grid;

subplot(2,1,2); semilogx(f,magX(1:M/2+1));

xlabel('Hz'); ylabel('|X(f)|');title('受噪声污染信号的频域波形')
```



```
axis([0 max(f)+100*fs/M 0 2500]);grid;
wc=2*pi*2000/fs;%截止频率
Nwin = 147;%滤波器长度
hwin = fir1(Nwin-1,wc/pi);%计算低通滤波器系数;
[Hwin f1]=freqz(hwin,1,44100,fs);%计算低通滤波器频率响应
magHwin = abs(Hwin);%幅频响应
phaHwin = angle(Hwin);%相频响应
phaHwin = 180*unwrap(phaHwin)/pi;
nwin = 0:Nwin-1;
figure;%绘图，低通滤波器特性
subplot(3,1,1);stem(nwin,hwin);grid;
xlabel('n');ylabel('hwin(n)');
title('低通滤波器单位冲激响应');
subplot(3,1,2);semilogx(f1,20*log10(magHwin));grid;
xlabel('Hz');ylabel('dB');
title('低通滤波器幅频响应');
subplot(3,1,3);plot(f1,phaHwin);grid;
xlabel('Hz');ylabel('degree');
title('低通滤波器相位响应');
y = filter(hwin,1,x); %滤波
Y = fft(y);
magY = abs(Y);%滤波后信号的频谱
figure;%绘图，滤波后信号的时域和频域波形
subplot(2,1,1); plot(m,y);
xlabel('n');ylabel('y(n)');
title('滤波后信号的时域波形');grid;
subplot(2,1,2); semilogx(f,magY(1:M/2+1));
xlabel('Hz'); ylabel('|Y(f)|');
title('滤波后信号的频域波形');grid;
```

`%sound(s,fs) %播放原始音乐`

`%sound(x,fs) %播放加噪音乐`

`sound(y,fs) %播放滤波后音乐`

运行截图：

