HW 8

1.

$$有F(u,v)=\int_{-\infty}^{\infty}\int_{-\infty}^{\infty}f(x,y)e^{-j(ux+vy)}dxdy$$

$$证明f(x,y)=\frac{1}{4\pi^2}\int_{-\infty}^{\infty}\int_{-\infty}^{\infty}F(u,v)e^{j(ux+vy)}dudv$$

$$设\overline{x}=(x,y),\overline{\epsilon}=(\epsilon_1,\epsilon_2)$$

$$\overline{x}\cdot\overline{\epsilon}=x\epsilon_1+y\epsilon_2$$

$$e^{-2\pi j(\overline{x}\cdot\overline{\epsilon})}=e^{-2\pi j(x\epsilon_1+y\epsilon_2)}$$

$$对于一维傅里叶逆变换，有$$

$$f(x)=\frac{1}{2\pi}\int_{-\infty}^{\infty}F(u)e^{jux}du$$

$$f(\overline{x})=\frac{1}{2\pi}\int_{R^2}e^{j(\overline{x}\cdot\overline{\epsilon})}F(\overline{\epsilon})d\overline{\epsilon}$$

$$=\frac{1}{4\pi^2}\int_{-\infty}^{\infty}\int_{-\infty}^{\infty}F(u,v)e^{j(ux+vy)}dudv$$

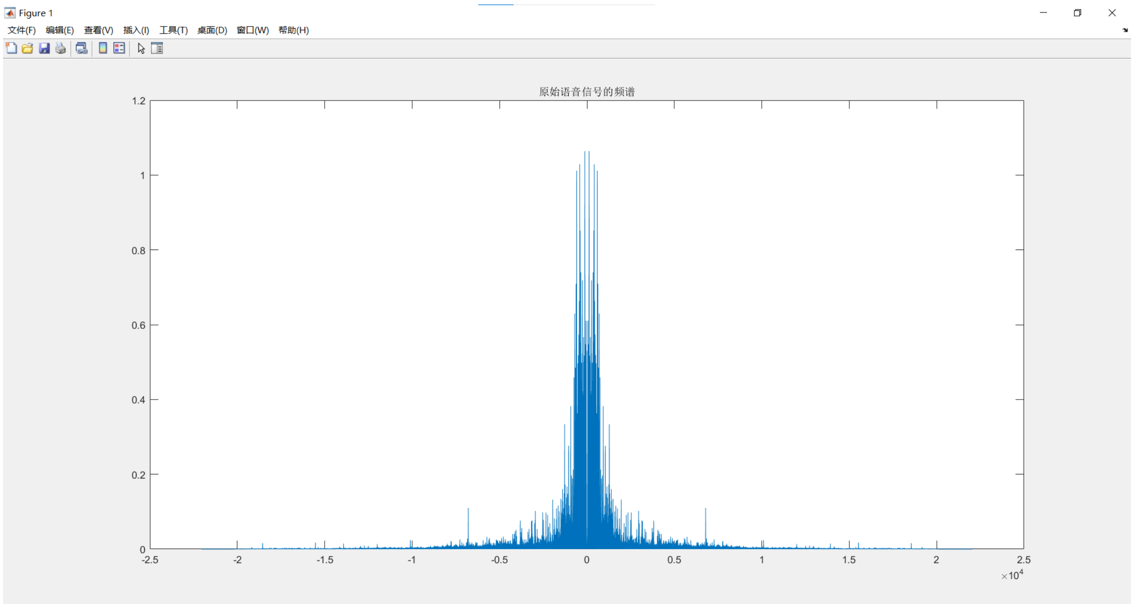
编程题

1. 使用如下代码打印声音的频谱

```
[x1,fs] = audioread('voice3.mp3');
x1 = x1(:,1);
figure(1);
y1 = fft(x1);
y1 = fftshift(y1);
N = length(x1);
derta_fs = fs/N;

N1 = floor(fs/(2*derta_fs));
w = [-N1*derta_fs:derta_fs:(N-N1-1)*derta_fs];
plot(w,abs(y1)/fs);
title('原始语音信号的频谱');
```

得到的图像如下:



2. 使用如下代码进行四个信号的调制与解调；首先读取音频文件

```
[x1,fs] = audioread('voice1.wma');
[x2,fs] = audioread('voice2.wma');
[x3,fs] = audioread('voice3.mp3');
[x4,fs] = audioread('voice4.mp3');

%只取左声道
x1 = x1(:,1);
x2 = x2(:,1);
x3 = x3(:,1);
x4 = x4(:,1);
```

接下来由于各个音频的长度不同，需要统一音频的长度

```
len1 = length(x1);
len2 = length(x2);
len3 = length(x3);
len4 = length(x4);
if len1>len2 && len1>len3 && len1>len4
    x2(len2+1:len1) = 0;
    x3(len3+1:len1) = 0;
    x4(len4+1:len1) = 0;
elseif len2>len1 && len2>len3 && len2>len4
    x1(len1+1:len2) = 0;
    x3(len3+1:len2) = 0;
    x4(len4+1:len2) = 0;
elseif len3>len1 && len3>len2 && len3>len4
    x1(len1+1:len3) = 0;
    x2(len2+1:len3) = 0;
    x4(len4+1:len3) = 0;
elseif len4>len1 && len4>len2 && len4>len3
    x1(len1+1:len4) = 0;
    x2(len2+1:len4) = 0;
    x3(len3+1:len4) = 0;
end
```

接下来进行滤波操作；设置截止频率为4000Hz

```
%低通滤波（FIR滤波器），在《数字信号处理》中要专门讲滤波器设计。
fp = 4000;
N1 = 2*pi*0.9/(0.1*pi);
wc1 = 2*pi*fp/fs;
if rem(N1,2)
    N1 = N1+1;
end
Window = blackman(N1+1);
b1 = fir1(N1,wc1/pi,Window);%低通滤波器，b1只有19个数，精度不算高。
figure(1);
freqz(b1,1,512);
title('低通滤波器的频率响应');
x1_low = filter(b1,1,x1);%将x1低通滤波
x2_low = filter(b1,1,x2);%将x2低通滤波
x3_low = filter(b1,1,x3);%将x3低通滤波
x4_low = filter(b1,1,x4);%将x4低通滤波

audiowrite('./res/voice1AfterLowpassFilter.wav', x1_low, fs);
audiowrite('./res/voice2AfterLowpassFilter.wav', x2_low, fs);%把低通滤波结果保
存
audiowrite('./res/voice3AfterLowpassFilter.wav', x3_low, fs);
audiowrite('./res/voice4AfterLowpassFilter.wav', x4_low, fs);
```

接下来进行调制操作，4个声音的载波频率分别是fc1=9000Hz，fc2=18000Hz，fc3=27000Hz，fc4=36000Hz

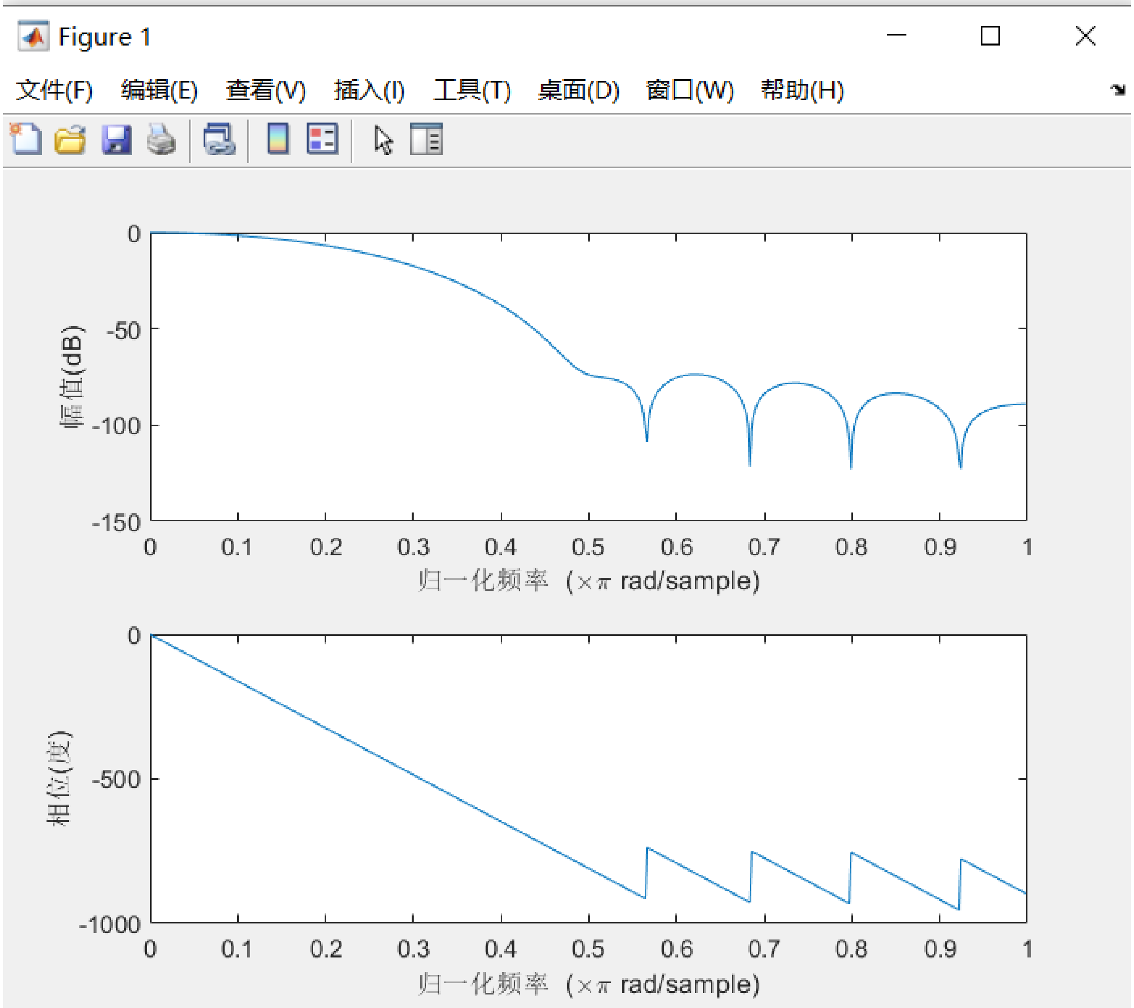
```
x5 = zeros(1,len1);
fc1 = 9000;
fc2 = 18000;
fc3 = 27000;
fc4 = 36000;
for i =1:length(x3)
    x5(i) = x1_low(i)*cos(2*pi*fc1*(i-1)/fs)+x2_low(i)*cos(2*pi*fc2*(i-
1)/fs)+x3_low(i)*cos(2*pi*fc3*(i-1)/fs)+x4_low(i)*cos(2*pi*fc4*(i-1)/fs);%两
个加起来
end
audiowrite('./res/voice1and2andVoice3andVoice4AfterModulation.wav', x5, fs);
```

解调操作

```
x1_afterModulation = zeros(1,len1);
x2_afterModulation = zeros(1,len2);
x3_afterModulation = zeros(1,len3);
x4_afterModulation = zeros(1,len4);
for i =1:length(x3)
    x1_afterModulation(i) = x3(i)*cos(2*pi*fc1*(i-1)/fs);
    x2_afterModulation(i) = x3(i)*cos(2*pi*fc2*(i-1)/fs);%两个信号各自乘以相应的
载波
    x3_afterModulation(i) = x3(i)*cos(2*pi*fc3*(i-1)/fs);
    x4_afterModulation(i) = x3(i)*cos(2*pi*fc4*(i-1)/fs);
end
x1_afterModulation = filter(b1,1,x1_afterModulation);
x2_afterModulation = filter(b1,1,x2_afterModulation);%低通滤波。
x3_afterModulation = filter(b1,1,x3_afterModulation);
x4_afterModulation = filter(b1,1,x4_afterModulation);
audiowrite('./res/voice1AfterDemodulation.wav', x1_afterModulation, fs);
```

```
audiowrite('./res/voice2AfterDemodulation.wav', x2_afterModulation, fs);
audiowrite('./res/voice3AfterDemodulation.wav', x3_afterModulation, fs);
audiowrite('./res/voice4AfterDemodulation.wav', x4_afterModulation, fs);
```

运行代码得到的结果如下



当调低载波频率时，声音信号间的相互干扰增大

附录

第一题的完整代码如下

```
[x1,fs] = audioread('voice3.mp3');
x1 = x1(:,1);
figure(1);
y1 = fft(x1);
y1 = fftshift(y1);
N = length(x1);
derta_fs = fs/N;

N1 = floor(fs/(2*derta_fs));
w = [-N1*derta_fs:derta_fs:(N-N1-1)*derta_fs];
plot(w,abs(y1)/fs);
title('原始语音信号的频谱');
```

第二题的完整代码如下

```
[x1,fs] = audioread('voice1.wma');
```

```
[x2,fs] = audioread('voice2.wma');
[x3,fs] = audioread('voice3.mp3');
[x4,fs] = audioread('voice4.mp3');

%只取左声道
x1 = x1(:,1);
x2 = x2(:,1);
x3 = x3(:,1);
x4 = x4(:,1);

%统一两个信号的长度。
len1 = length(x1);
len2 = length(x2);
len3 = length(x3);
len4 = length(x4);
if len1>len2 && len1>len3 && len1>len4
    x2(len2+1:len1) = 0;
    x3(len3+1:len1) = 0;
    x4(len4+1:len1) = 0;
elseif len2>len1 && len2>len3 && len2>len4
    x1(len1+1:len2) = 0;
    x3(len3+1:len2) = 0;
    x4(len4+1:len2) = 0;
elseif len3>len1 && len3>len2 && len3>len4
    x1(len1+1:len3) = 0;
    x2(len2+1:len3) = 0;
    x4(len4+1:len3) = 0;
elseif len4>len1 && len4>len2 && len4>len3
    x1(len1+1:len4) = 0;
    x2(len2+1:len4) = 0;
    x3(len3+1:len4) = 0;
end

derta_fs = fs/length(x1);

%低通滤波（FIR滤波器），在《数字信号处理》中要专门讲滤波器设计。
fp = 4000;
N1 = 2*pi*0.9/(0.1*pi);
wc1 = 2*pi*fp/fs;
if rem(N1,2)
    N1 = N1+1;
end
Window = blackman(N1+1);
b1 = fir1(N1,wc1/pi,Window);%低通滤波器，b1只有19个数，精度不算高。
figure(1);
freqz(b1,1,512);
title('低通滤波器的频率响应');
x1_low = filter(b1,1,x1);%将x1低通滤波
x2_low = filter(b1,1,x2);%将x2低通滤波
x3_low = filter(b1,1,x3);%将x3低通滤波
x4_low = filter(b1,1,x4);%将x4低通滤波

audiowrite('./res/voice1AfterLowpassFilter.wav', x1_low, fs);
audiowrite('./res/voice2AfterLowpassFilter.wav', x2_low, fs);%把低通滤波结果保存
audiowrite('./res/voice3AfterLowpassFilter.wav', x3_low, fs);
audiowrite('./res/voice4AfterLowpassFilter.wav', x4_low, fs);

%调制
```

```
x5 = zeros(1,len1);
fc1 = 9000;
fc2 = 18000;
fc3 = 27000;
fc4 = 36000;
for i =1:length(x3)
    x5(i) = x1_low(i)*cos(2*pi*fc1*(i-1)/fs)+x2_low(i)*cos(2*pi*fc2*(i-
1)/fs)+x3_low(i)*cos(2*pi*fc3*(i-1)/fs)+x4_low(i)*cos(2*pi*fc4*(i-1)/fs);%两个加起
来
end
audiowrite('./res/voice1and2andVoice3andVoice4AfterModulation.wav', x5, fs);

%解调
x1_afterModulation = zeros(1,len1);
x2_afterModulation = zeros(1,len1);
x3_afterModulation = zeros(1,len1);
x4_afterModulation = zeros(1,len1);
for i =1:length(x3)
    x1_afterModulation(i) = x3(i)*cos(2*pi*fc1*(i-1)/fs);
    x2_afterModulation(i) = x3(i)*cos(2*pi*fc2*(i-1)/fs);%两个信号各自乘以相应的载波
    x3_afterModulation(i) = x3(i)*cos(2*pi*fc3*(i-1)/fs);
    x4_afterModulation(i) = x3(i)*cos(2*pi*fc4*(i-1)/fs);
end
x1_afterModulation = filter(b1,1,x1_afterModulation);
x2_afterModulation = filter(b1,1,x2_afterModulation);%低通滤波。
x3_afterModulation = filter(b1,1,x3_afterModulation);
x4_afterModulation = filter(b1,1,x4_afterModulation);
audiowrite('./res/voice1AfterDemodulation.wav', x1_afterModulation, fs);
audiowrite('./res/voice2AfterDemodulation.wav', x2_afterModulation, fs);
audiowrite('./res/voice3AfterDemodulation.wav', x3_afterModulation, fs);
audiowrite('./res/voice4AfterDemodulation.wav', x4_afterModulation, fs);
```