

《 数字信号处理 》

实验报告本

|  |  |
| --- | --- |
| 班 级： | 信工211 |
| 学 号： | 21012909 |
| 姓 名： | 孟依然 |
| 指导教师： | 万永菁 |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **实验** | **实验一** | **实验二** | **实验三** | **实验四** | **综合实践项目** |
| **成绩** |  |  |  |  |  |

教师签名：\_\_\_\_\_\_\_\_\_\_\_

信息科学与工程学院

2023年12月

实验一 离散时间信号与系统

1. 实验目的

(1) 熟悉应用MATLAB表示离散时间信号。

(2) 掌握线性卷积求解系统输出的基本方法。

(3) 掌握求解离散时间系统输出的方法。

(4) 理解采样率变化对信号离散化产生的影响。

1. 实验内容
2. 画出幅度按指数衰减的有限长复指数序列的实部、虚部、幅度和相位。

提示：可以调用的函数有exp()、stem()、real()、imag()、abs()、angle()等

程序清单：

n=0:29;k=0.9.^n;

c=-0.2\*pi\*1i;

x=k.\*exp(c.\*n);

subplot(2,2,1);stem(n,real(x));

grid on;title('Real part');axis([0 30 -1 1]);

subplot(2,2,2);stem(n,imag(x));

grid on;title('Imaginary part');axis([0 30 -1 1]);

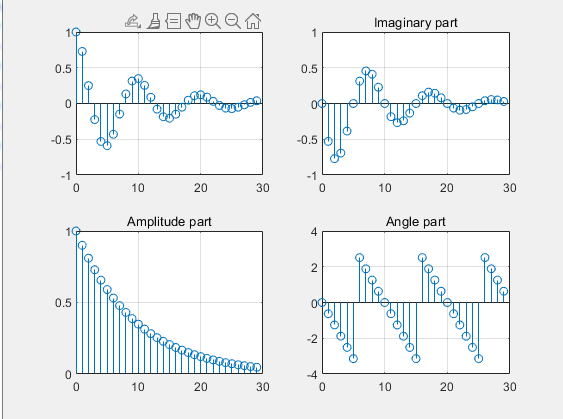
subplot(2,2,3);stem(n,abs(x));

grid on;title('Amplitude part');axis([0 30 0 1]);

subplot(2,2,4);stem(n,angle(x));

grid on;title('Angle part');axis([0 30 -4 4]);

实验结果：



1. 11阶滑动平均系统的输入/输出关系是，输入信号是，其中是一个在[-5, 5]之间均匀分布的随机序列。试求：
   * 1. 用plot函数在**0 ≤ *n* ≤ 100**之间画出输入信号和输出信号；
     2. 画出的2阶差分信号；
     3. 画出与的相关序列；
     4. 再产生一个随机序列，画出它与的相关序列。

（可以调用ones()、rand()、conv()、xcorr()-等函数）

程序清单：

①

w=-5+(5+5)\*rand(1,101);

n=0:100;

x=10\*cos(0.08\*pi.\*n)+w;

h=1/11.\*ones(1,11);

y=conv(x,h);

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

grid on;title('x(n)');

subplot(2,1,2);plot(y);

grid on;title('y(n)');

②

w=-5+(5+5)\*rand(1,101);

n=0:100;

x=10\*cos(0.08\*pi.\*n)+w;

h=1/11.\*ones(1,11);

y=conv(x,h);

b=[1,-2,1];a=1;

v=filter(b,a,x);

plot(v);

grid on;title('v(n)');

③

w=-5+(5+5)\*rand(1,101);

n=0:100;

x=10\*cos(0.08\*pi.\*n)+w;

h=1/11.\*ones(1,11);

y=conv(x,h);

b=[1,-2,1];a=1;

v=filter(b,a,x);

d=xcorr(v,w);

stem(d);

grid on;title('v(n)与w(n)的相关序列');

④

w=rand(1,101);

n=0:100;

x=10\*cos(0.08\*pi.\*n)+w;

h=1/11.\*ones(1,11);

y=conv(x,h);

b=[1,-2,1];a=1;

v=filter(b,a,x);

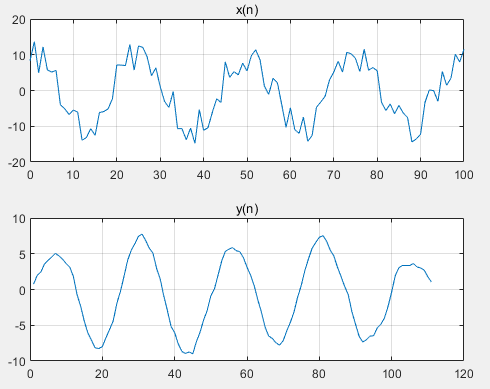
d=xcorr(v,w);

stem(d);

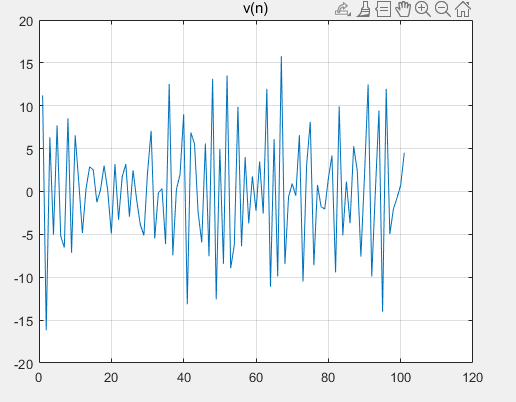
grid on;title('v(n)与随机序列的相关序列');

实验结果：

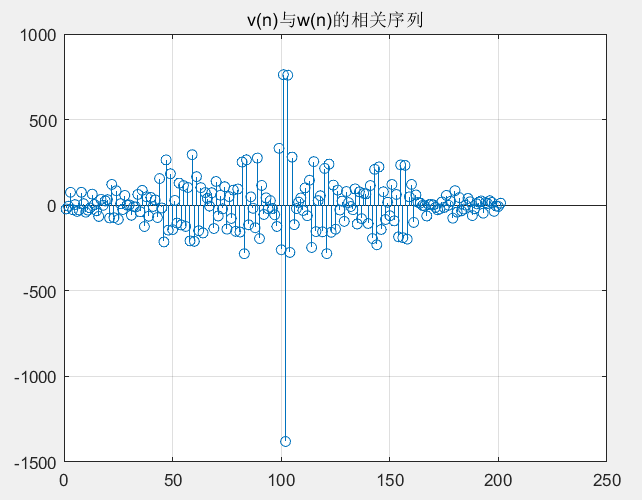
①



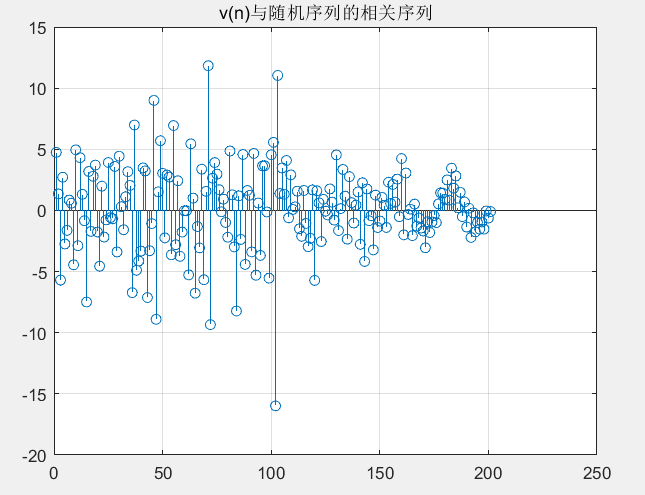
②



③



④



实验分析与讨论（请手写）：

1. 试分析11阶滑动平均系统的滤波特性。
2. 试分析2阶差分系统的滤波特性。
3. 由(C)和(D)的实验结果，你得出了什么结论？
4. 下面的差分方程可以产生声音的混响效果，请为音频文件good.wav合成混响的效果，并保存在new\_good.wav文件中。用耳机欣赏混响前的音乐与混响后的音乐有何区别。



程序清单：

alpha=0.3;

R=5000;

b=[1,alpha];a=[1,zeros(1,R-1),alpha];

signal='E:\good.wav';

[x,fs]=audioread(signal);

disp('Playing original audio...');

sound(x, fs);

pause(length(x) / fs + 1);

y=filter(b,a,x);

filname='E:\new\_good.wav';

audiowrite(filname,y,fs);

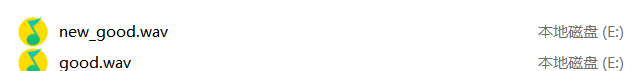
[y,Fs]=audioread(filname);

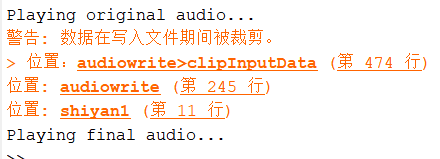
disp('Playing final audio...');

sound(y, Fs);

pause(length(y) / Fs + 1);

实验结果：





实验分析与讨论（请手写）：

1. 请问该系统是IIR系统还是FIR系统？
2. 请分析系统的因果性和稳定性在不同的***α***值和***R***值（***R* ≠ ∞**）下，系统的因果性和稳定性是否会有变化？
3. 请用文字简要描述不同***α***值和***R***值下的混响效果的区别。
4. 请编程实现实际音频信号经抽取系统、、后的音效。待处理的音频文件分别为钢琴乐曲卡农片段（canon.wav）和语音片段（dsp.wav）。展示你的程序设计方法并按照要求进行分析。

程序清单：

filename1 = 'E:\canon.wav'; % 文件名

filename2 ='E:\dsp.wav';

[x, Fs] = audioread(filename1); % 读取音频文件

[y, fs] = audioread(filename2);

% 抽取系统 T[x(2n)]

x\_2n = x(1:2:end); % 抽取 x(2n)

y\_2n = y(1:2:end);

% 抽取系统 T[x(4n)]

x\_4n = x(1:4:end); % 抽取 x(4n)

y\_4n = y(1:4:end);

% 抽取系统 T[x(8n)]

x\_8n = x(1:8:end); % 抽取 x(8n)

y\_8n = y(1:8:end);

% 播放原始音频

disp('Playing original audio...');

sound(x, Fs);

pause(length(x) / Fs + 1); % 暂停足够的时间以播放完整个音频

% 播放抽取后的音频

disp('Playing T[x(2n)] audio...');

sound(x\_2n, Fs);

pause(length(x\_2n) / Fs + 1);

disp('Playing T[x(4n)] audio...');

sound(x\_4n, Fs);

pause(length(x\_4n) / Fs + 1);

disp('Playing T[x(8n)] audio...');

sound(x\_8n, Fs);

pause(length(x\_8n) / Fs + 1);

disp('Playing original audio...');

sound(y, fs);

pause(length(y) / fs + 1); % 暂停足够的时间以播放完整个音频

% 播放抽取后的音频

disp('Playing T[y(2n)] audio...');

sound(y\_2n, fs);

pause(length(y\_2n) / fs + 1);

disp('Playing T[y(4n)] audio...');

sound(y\_4n, Fs);

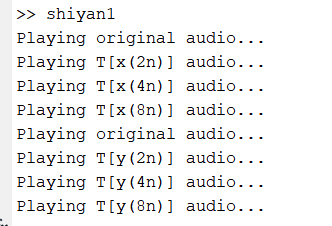
pause(length(y\_4n) / fs + 1);

disp('Playing T[y(8n)] audio...');

sound(y\_8n, fs);

pause(length(y\_8n) / fs + 1);

实验结果：



实验分析与讨论（请手写）：

1. 请比较分析两段乐曲在不同采样率下是否存在失真情况？
2. 请问随着采样率的不断下降，钢琴音频和语音哪一个失真效果更显著？请解释你的结论。

实验二 离散傅立叶变换与分析

1、实验目的

(1) 熟悉应用MATLAB求解信号频谱的方法。

(2) 掌握应用FFT的方法求解系统输出的步骤。

(3) 对比分析利用线性卷积求解系统输出和利用FFT方法求解系统输出这两种方法的不同之处。

(4) 掌握系统分析方法和简单滤波器的设计方法。

2、实验内容

(1) 设输入信号，某LSI系统的单位脉冲响应为，求：

① 利用线性卷积求输入信号通过系统后的输出。

② 利用FFT的方法，先求解输入信号的频谱以及单位脉冲响应的频谱，通过计算求解系统的输出。

（可以调用conv()、fft()、ifft()等函数）

程序清单：

n=0:199;

x=sin(0.1\*pi\*n)+cos(0.5\*pi\*n);

h=1/4.\*ones(1,4);

nh=0:3;nL=0:202;

yl=conv(x,h);

X=fft(x);

H=fft(h,200);

y=ifft(X.\*H);

yn=[y,0,0,0];

subplot(4,1,1);stem(nL,yl);

grid on;title('yl(n)');

subplot(4,1,2);stem(X);

grid on;title('X(k)');

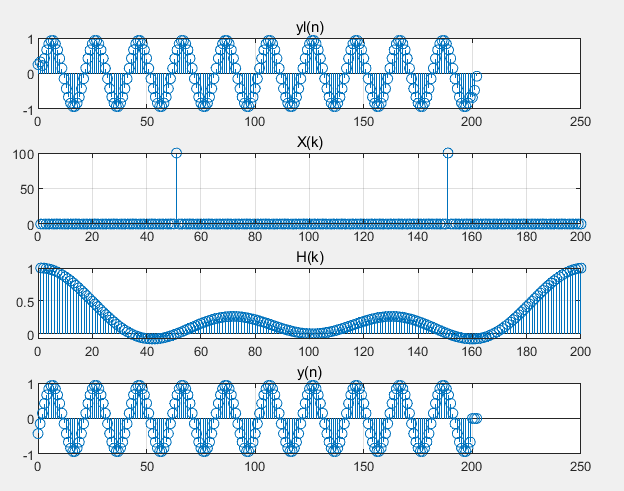
subplot(4,1,3);stem(H);

grid on;title('H(k)');

subplot(4,1,4);stem(nL,yn);

grid on;title('y(n)');

实验结果：



实验分析与讨论（请手写）：

1. 请比较两种不同方法求得的输出及其它们的频谱及；
2. 试分析该系统的滤波特性，并结合输出信号的频率成分进行分析。

(2) 已知某系统的系统函数为，且系统稳定，试求：

1. 求系统的零极点；（提示：可以用tf2zp()函数）
2. 画出系统的零极点图；（提示：可以用zplane()函数）
3. 画出系统的幅频响应、相频响应、群延迟。（提示：可以用freqz()、grpdelay()函数）

程序清单：

a=[1,0.9,0.81];

b=[1,1,1];

figure(1);

[x,p,k]=tf2zp(b,a);

disp(x);

disp(p);

[H,~]=freqz(b,a);

[gd,w]=grpdelay(b,a);

subplot(2,2,1);zplane(b,a);

grid on;title('零极点');

subplot(2,2,2);plot(w/pi,abs(H));

grid on;title('幅频响应');

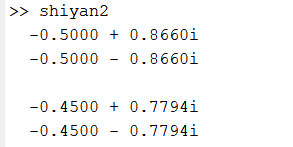
subplot(2,2,3);plot(w/pi,angle(H));

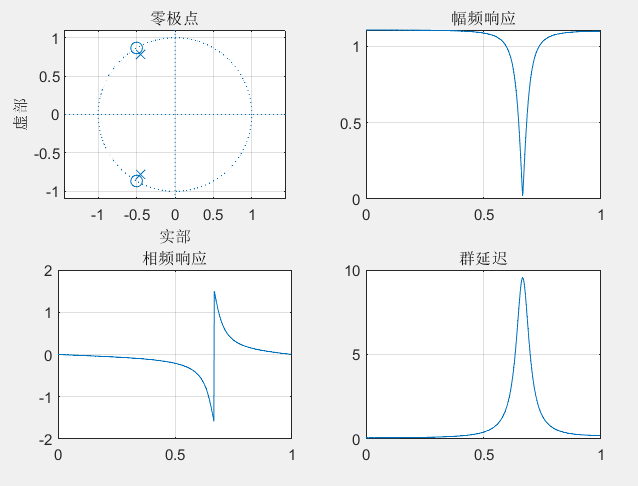
grid on;title('相频响应');

subplot(2,2,4);plot(w/pi,gd);

grid on;title('群延迟');

实验结果：





实验分析与讨论（请手写）：

(A) 试求该系统的ROC，并说明系统的因果性；

(B) 试分析该系统的滤波特性；

(C) 该滤波器是IIR滤波器还是FIR滤波器？该滤波器具有线性相位吗？

(3) 一个LSI系统由下面的差分方程描述：



1. 用filter函数计算并画出在内的系统单位脉冲响应，由画出的单位脉冲响应判断系统的稳定性。
2. 画出系统零极点图及系统的幅频和相频响应曲线。
3. 如果这个系统的输入是，利用filter函数求在内的系统输出。分析输出信号，观察中的直流分量和频率成份分量的通过情况。
4. 如果希望将中的直流分量完全滤除，而频率成份分量仍然保留，应该怎样修改该系统的差分方程，用实验的方法验证你的结论。

程序清单：

①②

a=0.3125;

b=[1,-0.8,0.64];

n=0:100;

x=zeros(1,101);

x(1)=1;

y=filter(b,a,x);

[H,w]=freqz(b,a);

subplot(2,2,1);plot(n,y);

grid on;title('单位脉冲响应');

subplot(2,2,2);zplane(b,a);

grid on;title('零极点');

subplot(2,2,3);plot(w/pi,abs(H));

grid on;title('幅频响应');

subplot(2,2,4);plot(w/pi,angle(H));

grid on;title('相频响应');

③④

a=[1,-0.3125];

b=[1,-0.8,0.64];

n=0:200;

x=(5+3\*cos(pi/3\*n));

y=filter(b,a,x);

y1=y-mean(y);

[Y,w]=freqz(y);

subplot(2,1,1);plot(w/pi,abs(Y));grid on;title('yn');

axis([0 1 0 500]);xlabel('w/pi');ylabel('y的幅频特性');

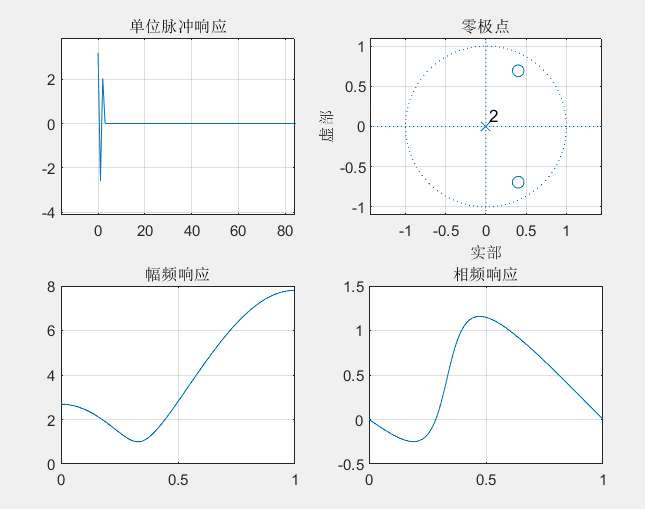
[Y1,w1]=freqz(y1);

subplot(2,1,2);plot(w1/pi,abs(Y1));grid on;title('y1n');

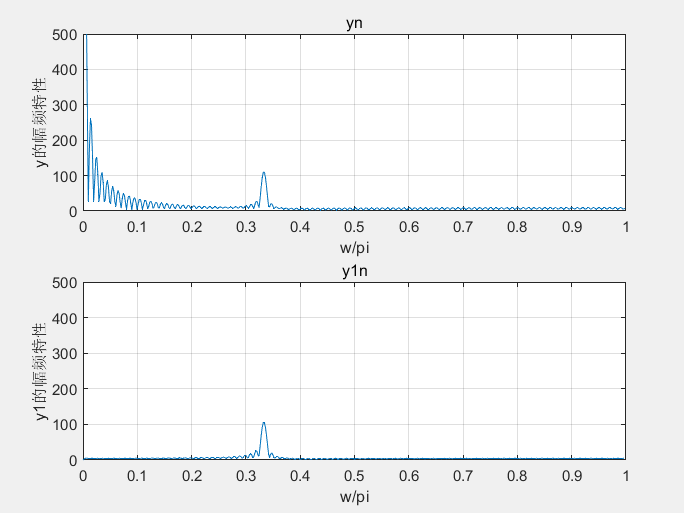
axis([0 1 0 500]);xlabel('w/pi');ylabel('y1的幅频特性');

实验结果：

①②



③④



实验分析与讨论（请手写）：

1. 请通过实验分析该系统的稳定性；
2. 请分析解释第③问中输入信号不同频率成份的通过情况，并解释原因；
3. 请描述第④问的设计思路。

(4) 音频文件test1.wav ~ test4.wav中录制了钢琴上的一些按键音，其中，请用FFT的方法识别出每段音频文件中含有那几个音符。

（注： (la)=220Hz (ci)=246.94Hz (do)=261.63Hz 2(rui)=293.66Hz

3 (mi)=329.63Hz 4(fa)=349.23Hz 5(so)=392Hz ）

1. 请读入test1.wav ~ test4.wav音频文件，画出音频信号的时域图。
2. 请识别test1.wav和test2.wav两个音频文件中弹奏的是哪个音符（请对音频信号中的有效音符区间进行谱分析，建议尝试使用不同的窗函数进行谱分析，以缓解频谱泄漏现象）。
3. 请识别test3.wav和test4.wav两个音频文件中弹奏了哪些音符；
4. （拓展部分）test3.wav和test4.wav两个音频文件有多个音符，且在最后有三音和弦，请尝试切分这些音符，并对其中的单音或和弦进行识别。

可以使用的窗函数有：boxcar()、triang()、hanning()、hamming()、blackman()等

程序清单：

①

% 读取音频文件

[audio1, fs1] = audioread('D:\dsp\test1.wav');

[audio2, fs2] = audioread('D:\dsp\test2.wav');

[audio3, fs3] = audioread('D:\dsp\test3.wav');

[audio4, fs4] = audioread('D:\dsp\test4.wav');

% 绘制音频信号的时域图

t1 = (0:length(audio1)-1) / fs1;

t2 = (0:length(audio2)-1) / fs2;

t3 = (0:length(audio3)-1) / fs3;

t4 = (0:length(audio4)-1) / fs4;

subplot(2,2,1);

plot(t1, audio1);

title('test1.wav - 时域图');

xlabel('时间 (秒)');

ylabel('幅度');

subplot(2,2, 2);

plot(t2, audio2);

title('test2.wav - 时域图');

xlabel('时间 (秒)');

ylabel('幅度');

subplot(2,2, 3);

plot(t3, audio3);

title('test3.wav - 时域图');

xlabel('时间 (秒)');

ylabel('幅度');

subplot(2,2, 4);

plot(t4, audio4);

title('test4.wav - 时域图');

xlabel('时间 (秒)');

ylabel('幅度');

②（更换文件为test2.wav即可）

[x, Fs] = audioread('D:\dsp\test1.wav');

% 读取音频文件

% 计算音频长度和时间

len = length(x);

duration = len/Fs;

time = linspace(0, duration, len);

% 进行快速傅里叶变换（FFT）

N = 2^nextpow2(len); % 使用2的幂作为FFT长度，以提高计算效率

w1 = boxcar(len);

X1 = fft(x.\*w1, N);

w2 = triang(len);

X2 = fft(x.\*w2, N);

w3 = hanning(len);

X3 = fft(x.\*w3, N);

w4 = hamming(len);

X4 = fft(x.\*w4, N);

w5 = blackman(len);

X5 = fft(x.\*w5, N);

%X = fft(x, N); % X是频域表示

% 计算频率轴

f = Fs\*(0:(N/2))/N; % 单边频谱

% 绘制频谱图

subplot(3,2,1);

plot(f, 2\*abs(X1(1:N/2+1))/N);

title('频谱分析');

xlabel('频率 (Hz)');

ylabel('振幅');

xlim([0, Fs/2]);

subplot(3,2,2);

plot(f, 2\*abs(X2(1:N/2+1))/N);

title('频谱分析');

xlabel('频率 (Hz)');

ylabel('振幅');

xlim([0, Fs/2]);

subplot(3,2,3);

plot(f, 2\*abs(X3(1:N/2+1))/N);

title('频谱分析');

xlabel('频率 (Hz)');

ylabel('振幅');

xlim([0, Fs/2]);

subplot(3,2,4);

plot(f, 2\*abs(X4(1:N/2+1))/N);

title('频谱分析');

xlabel('频率 (Hz)');

ylabel('振幅');

xlim([0, Fs/2]);

subplot(3,2,5);

plot(f, 2\*abs(X5(1:N/2+1))/N);

title('频谱分析');

xlabel('频率 (Hz)');

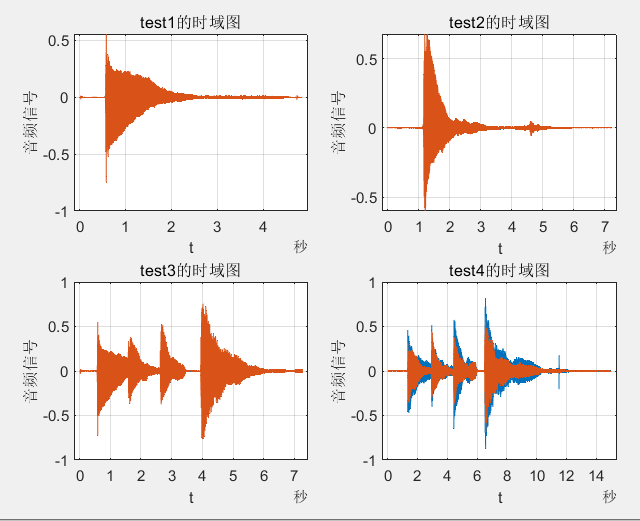
ylabel('振幅');

% 设置坐标轴范围

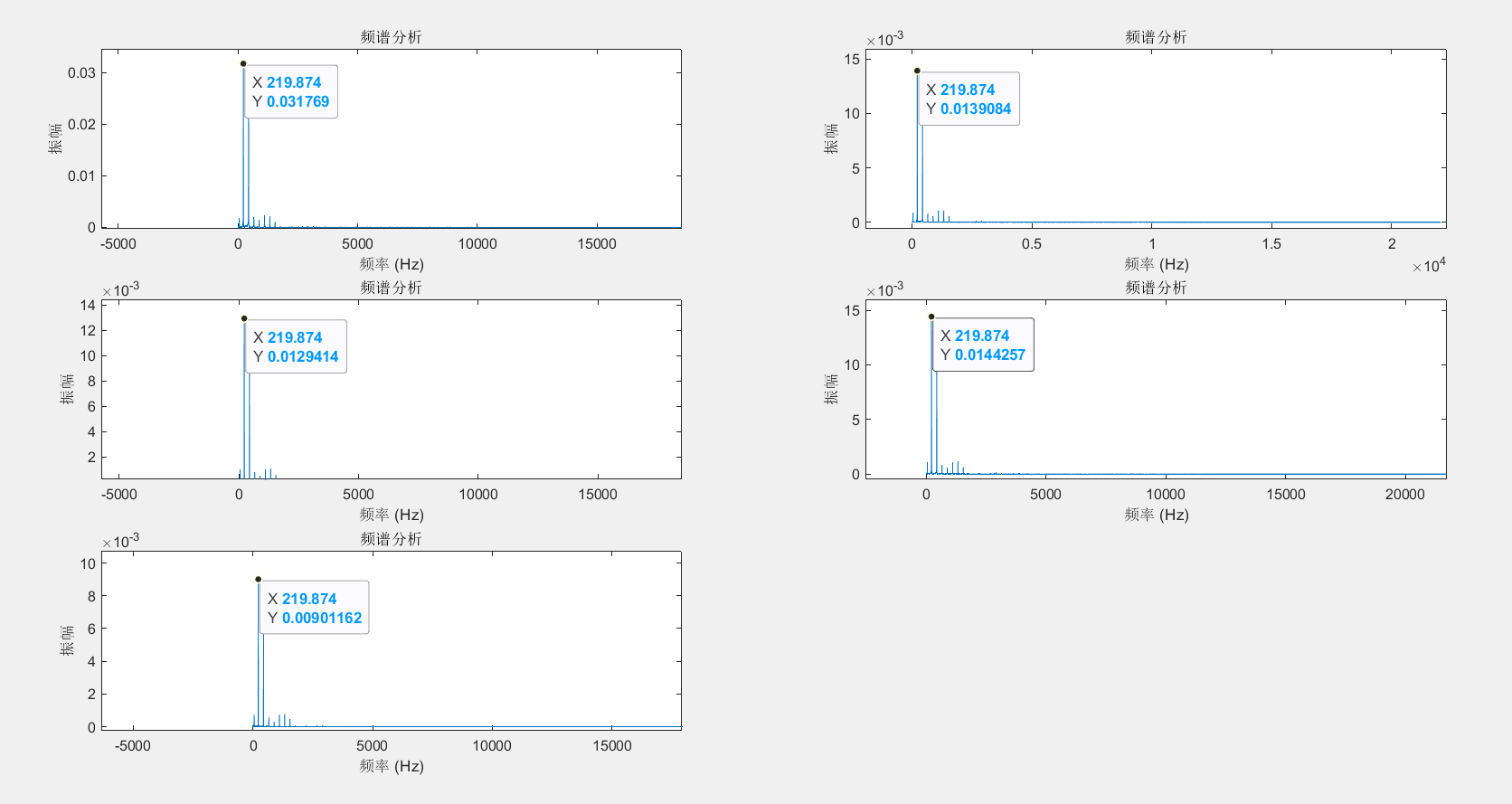
xlim([0, Fs/2]);

实验结果：

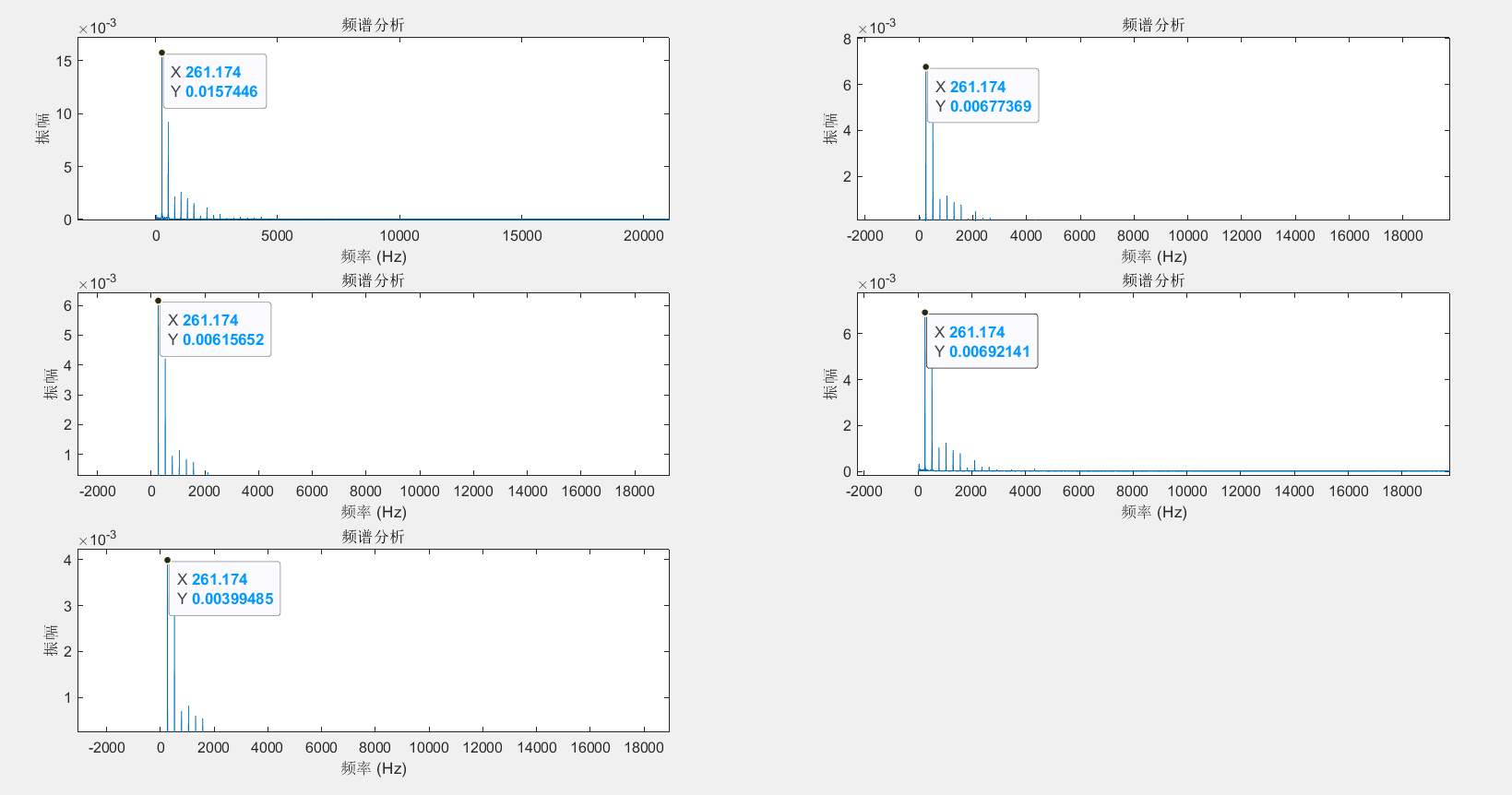
①



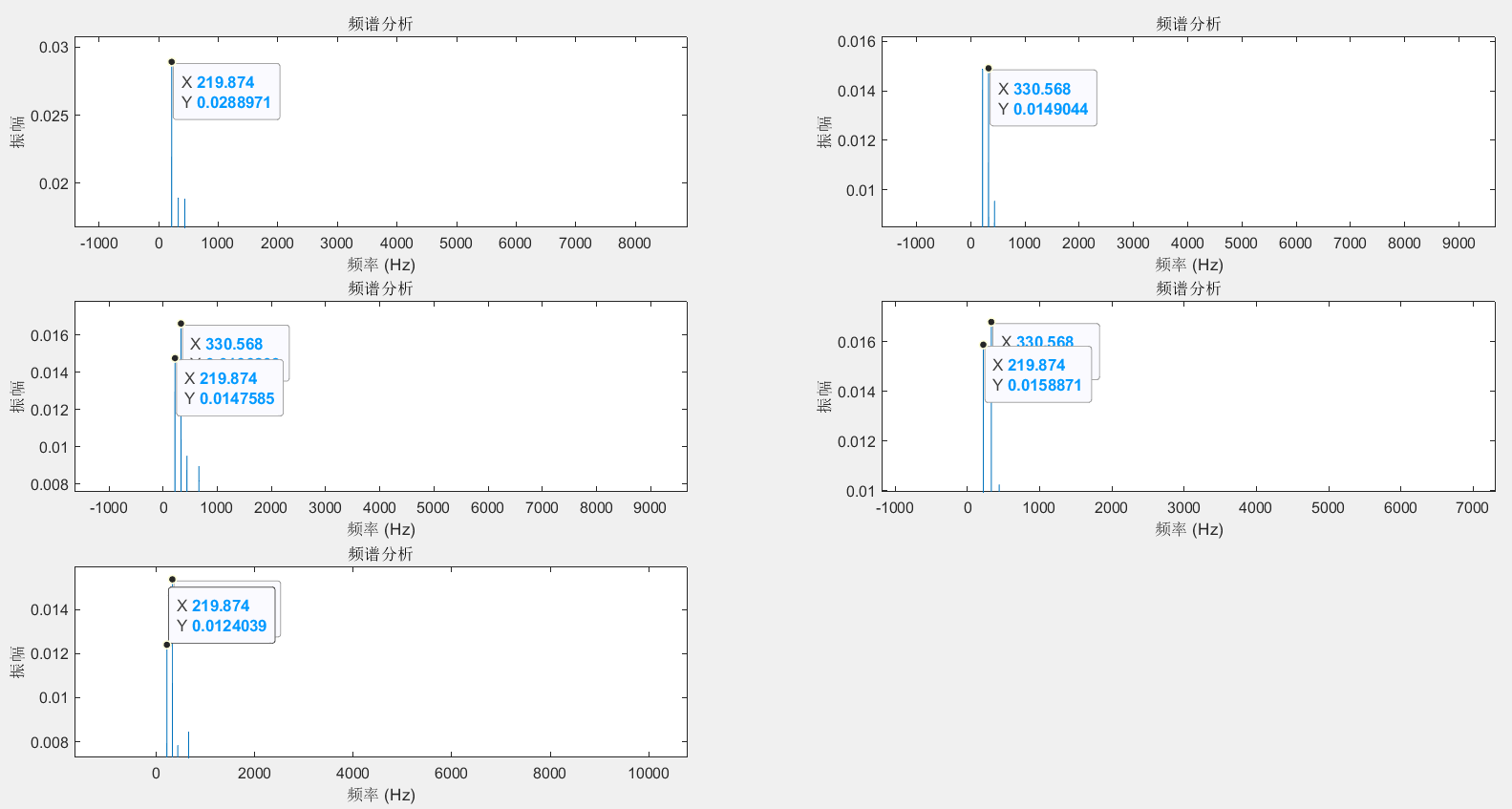
②test1.wav，弹奏的是la



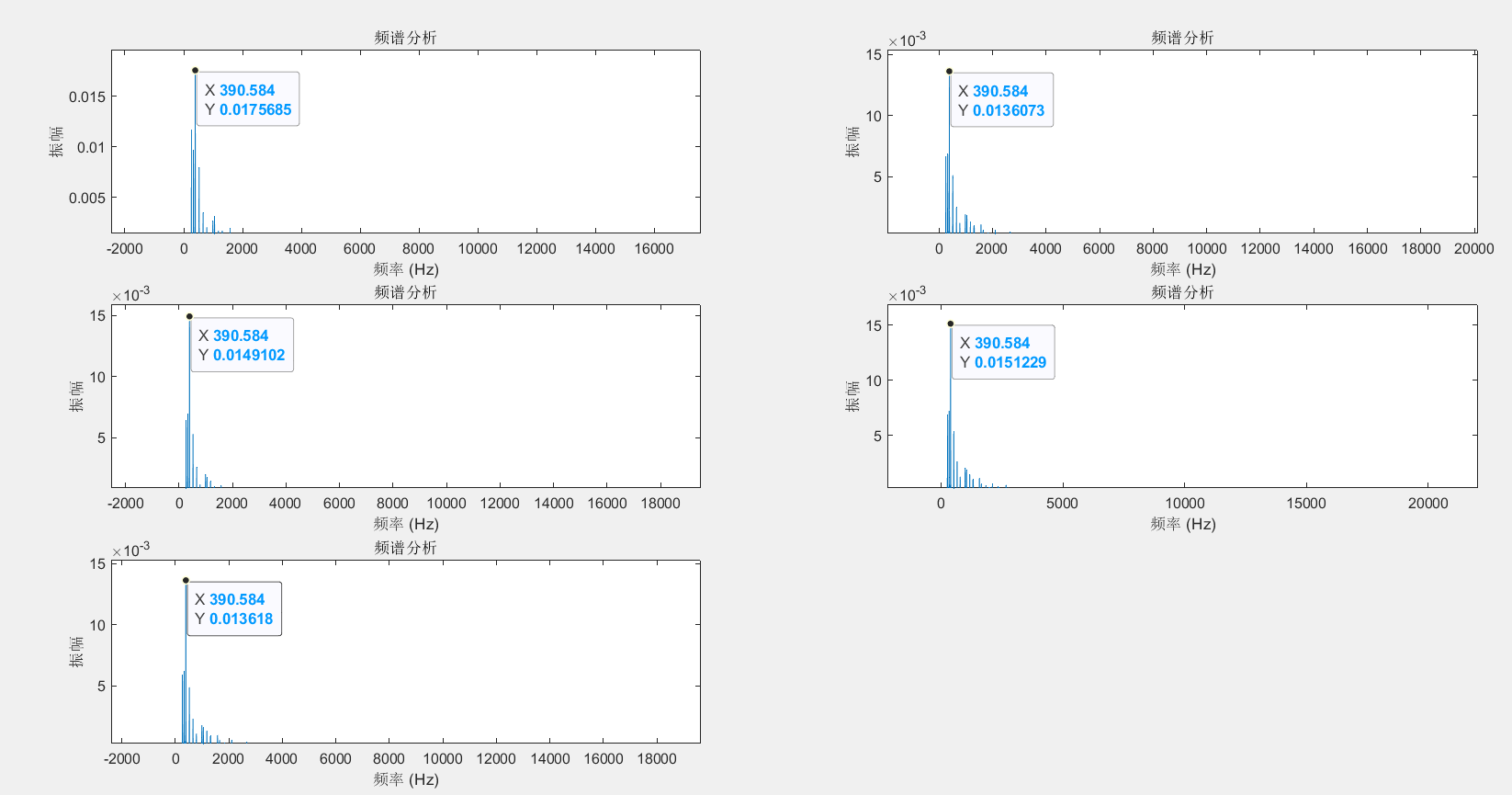
test2.wav,弹奏的是do



③test3.wav 弹奏了do,la和mi



test4.wav,弹奏的音符是do,mi,so



实验分析与讨论（请手写）：

1. 请分析使用不同窗函数时谱分析的结果；
2. 根据你的程序实验结果，请回答test1.wav ~ test4.wav音频文件中包含了哪些音符？

实验三 IIR数字滤波器的设计

1、实验目的

(1) 掌握应用MATLAB进行IIR数字滤波器设计的基本方法。

(2) 灵活运用MATLAB提供的函数设计各种类型的IIR数字滤波器。

2、实验内容

(1) 已知数字带通滤波器的指标为：

通带指标为：

阻带指标为：

分别设计满足以上指标的巴特沃斯、切比雪夫I型、切比雪夫II型、以及椭圆数字滤波器，并画出滤波器的频率响应。

程序清单：

rp=0.5;rs=50;

wp=[0.3,0.7];

ws=[0.2,0.8];

[n1,wn1]=buttord(wp,ws,rp,rs);

[b1,a1]=butter(n1,wn1);

[H1,w1]=freqz(b1,a1);

[n2,wn2]=cheb1ord(wp,ws,rp,rs);

[b2,a2]=cheby1(n2,rp,wn2);

[H2,w2]=freqz(b2,a2);

[n3,wn3]=cheb2ord(wp,ws,rp,rs);

[b3,a3]=cheby2(n3,rs,wn3);

[H3,w3]=freqz(b3,a3);

[n4,wn4]=ellipord(wp,ws,rp,rs);

[b4,a4]=ellip(n4,rp,rs,wn4);

[H4,w4]=freqz(b4,a4);

subplot(2,2,1);

plot(w1/2\*pi,abs(H1));grid on;

xlabel('f/fs');ylabel('幅值');

title('巴特沃斯滤波器频率响应');

subplot(2,2,2);

plot(w2/2\*pi,abs(H2));grid on;

xlabel('f/fs');ylabel('幅值');

title('切比雪夫1型滤波器频率响应');

subplot(2,2,3);

plot(w3/2\*pi,abs(H3));grid on;

xlabel('f/fs');ylabel('幅值');

title('切比雪夫2型滤波器频率响应');

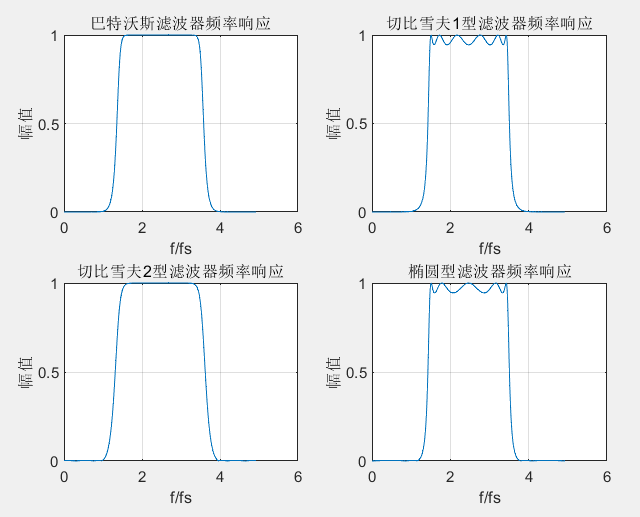
subplot(2,2,4);

plot(w4/2\*pi,abs(H4));grid on;

xlabel('f/fs');ylabel('幅值');

title('椭圆型滤波器频率响应');

实验结果：



实验分析与讨论（请手写）：

请比较和分析不同类型滤波器在相同技术指标情况下设计出的滤波器阶数、以及滤波器幅频响应的特点。

(2) 设计一个数字高通滤波器*H*(*z*)，它用在下面的结构中，

# A/D

***H*(*z*)**

# D/A



满足下列要求：

1. 采样频率为10kHz；
2. 阻带边缘频率为1.5kHz，；
3. 通带边缘频率为2kHz，；

④ 滤波器的幅频响应具有单调的通带和阻带特性。

画出该数字高通滤波器的频率响应。请自行设计一个输入信号（至少应含有两种及以上的频率成分）通过该滤波器，观察并解释滤波器的输出结果。

程序清单：

rp=3;rs=40;

fs=10000;

wp=0.4;

ws=0.3;

n=0:100;

x=sin(0.2\*pi\*n)+cos(0.5\*pi\*n);

[n,wn]=buttord(wp,ws,rp,rs);

[b,a]=butter(n,wn,'high');

[H,w]=freqz(b,a);

y=filter(b,a,x);

subplot(3,1,1);

stem(x);

title('输入信号');

subplot(3,1,2);

plot(w\*fs/2\*pi,abs(H));grid on;

xlabel('频率（HZ）');ylabel('幅值');

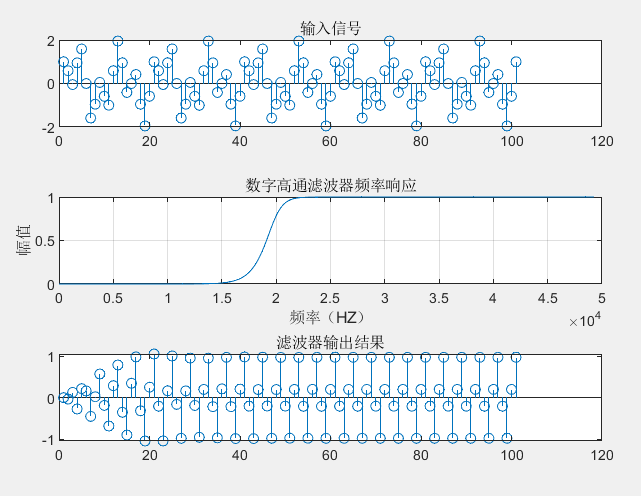
title('数字高通滤波器频率响应');

subplot(3,1,3);

stem(y);

title('滤波器输出结果');

实验结果：



实验分析与讨论（请手写）：

请写出该滤波器的技术指标，说明设计的输入信号以及实验获得的滤波结果。请预测一下，如果该滤波器用椭圆滤波器来设计，所得到的滤波器阶数应该会变多还是变少？为什么？

(3) 已知某系统如图所示：



系统的输入*x*(*t*)为：

若现在用数字滤波器模拟上述系统的输入输出效果，试求：

1. 请选择系统的采样频率*fs*，画出*x*(*n*)；

(B) 画出系统的幅频响应；

(C) 画出系统的输出*y*(*n*)。

程序清单：

Hs= tf(1,[1 1]);

bode(Hs);grid;

t=linspace(0,10,100);

x=zeros(size(t));

n=linspace(0,10,100);

x(t<=5)=2;

b=0.02;

a=[1,-0.9802];

[H,w]=freqz(b,a);

y=filter(b,a,x);

subplot(3,1,1);

stem(x);

xlabel('n');ylabel('x(n)');

title('输入x(n)的图像');

subplot(3,1,2);

plot(w,abs(H));grid on;

xlabel('w(rad）');ylabel('|H(e^jw)|');

title('数字低通滤波器频率响应');

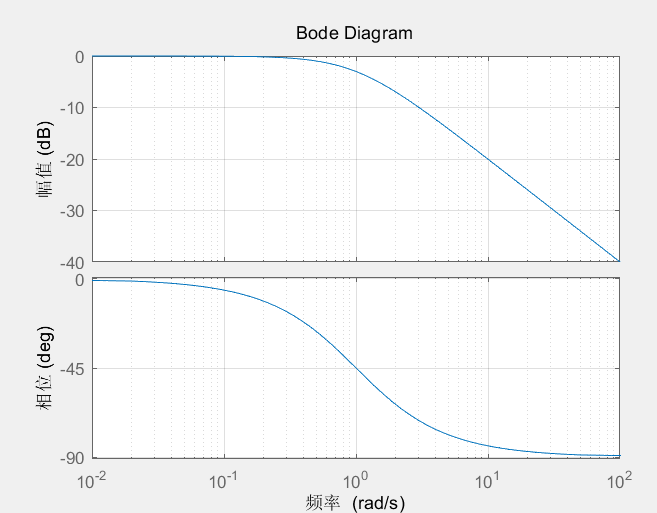
subplot(3,1,3);

stem(y);

xlabel('n');ylabel('y(n)');

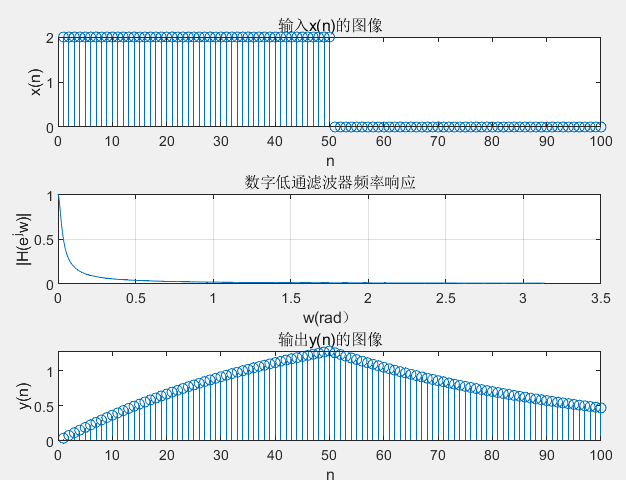
title('输出y(n)的图像');

实验结果：



由图可得Ωst=100rad/s,fst=16HZ;

Ωst<Ωs/2得Ωs=200rad/s,fs=32HZ



实验分析与讨论（请手写）：

请写出必要的系统分析步骤，分析该滤波器的滤波特性，并解释滤波结果。

实验四 FIR数字滤波器的设计

1、实验目的

(1) 掌握应用MATLAB实现FIR数字滤波器的窗函数设计方法。

(2) 掌握应用MATLAB实现FIR数字滤波器的频率采样设计方法。

(3) 理解FIR数字滤波器的线性相位约束条件。

2、实验内容

(1) 选择适当的窗函数设计一个数字带阻滤波器，技术指标为：

通带指标为：

阻带指标为：

请画出设计的滤波器的脉冲响应和幅频响应。

程序清单：

wp1=0.3\*pi;wp2=0.7\*pi;

ws1=0.4\*pi;ws2=0.6\*pi;

tr\_width=ws1-wp1;

M=ceil(6.6\*pi/tr\_width);

k=0:M-1;

wc1=(wp1+ws1)/2;wc2=(wp2+ws2)/2;

wc=[wc1,wc2];

h=fir1(M-1,wc/pi,"stop");

[H,w]=freqz(h,1);

subplot(211);

stem(h);

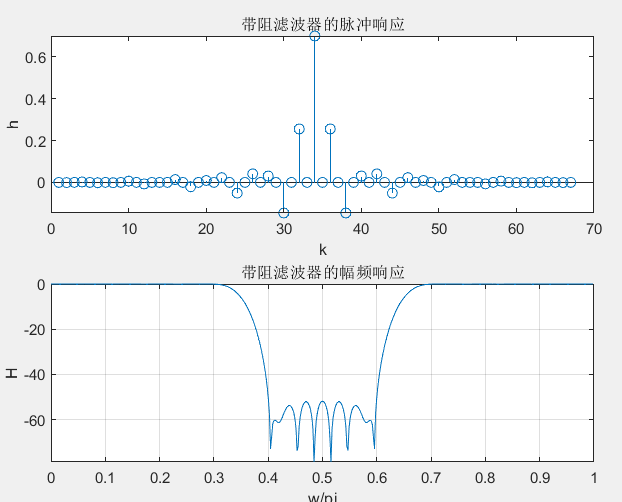
title('带阻滤波器的脉冲响应');xlabel('k');ylabel('h');

subplot(212);

plot(w/pi,20\*log10(abs(H)));grid on;

title('带阻滤波器的幅频响应');xlabel('w/pi');ylabel('H');

实验结果：



实验分析与讨论（请手写）：

请写出此题选窗的依据，以及滤波器长度的计算方法。另外，此题中对滤波器单位脉冲响应长度的奇偶性是否有要求？为什么？

(2) 用频率采样法设计题(1)中的带阻滤波器，选择适当的滤波器阶数，为确保滤波效果，要求过渡带中有两个样本（按照经验值，过渡带两个样本的幅度可以取为0.109和0.594）。请画出设计的滤波器的脉冲响应和幅频响应。

程序清单：M = 60;

alpha=(M-1)/2;

k=0:M-1;

wk= (2\*pi/M)\*k;

Hrs = [ones(1,10),0.594,0.109,zeros(1,7),0.109,0.594,ones(1,19),0.594,0.109,zeros(1,7),0.109,0.594,ones(1,9)];

Hdr = [1,1,0,1,0,1];

wdl = [0,0.35,0.35,0.65,0.65,1];

k1 = 0:floor((M-1)/2);

k2 = floor((M-1)/2)+1:M-1;

angH = [-alpha\*(2\*pi)/M\*k1, alpha\*(2\*pi)/M\*(M-k2)];

H = Hrs.\*exp(1i\*angH);

h = real(ifft(H,M));

[Ha,w]=freqz(h,1);

subplot(2,1,1); stem(k,h);

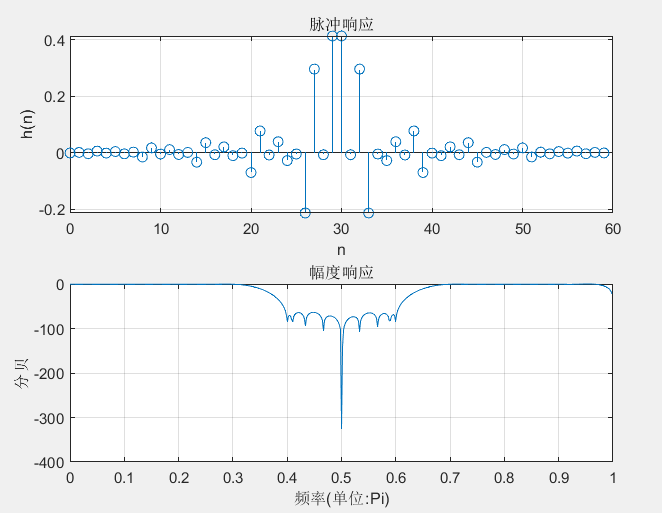
title('脉冲响应'); xlabel('n'); ylabel('h(n)');grid

subplot(2,1,2);plot(w/pi,20\*log10(abs(Ha)));

grid

title('幅度响应'); xlabel('频率(单位:Pi)');ylabel('分贝');

实验结果：



实验分析与讨论（请手写）：

请写出应用频域采样法设计该滤波器的方法。

(3) 用等波纹最佳逼近法设计(1)中的滤波器

（注：可以使用remezord()函数和remez()函数，参考第七章讲义例程）。

程序清单：

f=[0.3,0.4,0.6,0.7];

m=[1,0,1];

rp=0.5;rs=50;

dat1=(10^(rp/20)-1)/(10^(rp/20)+1);dat2=10^(-rs/20);

rip=[dat1,dat2,dat1];

[M,fo,mo,wo]=remezord(f,m,rip);

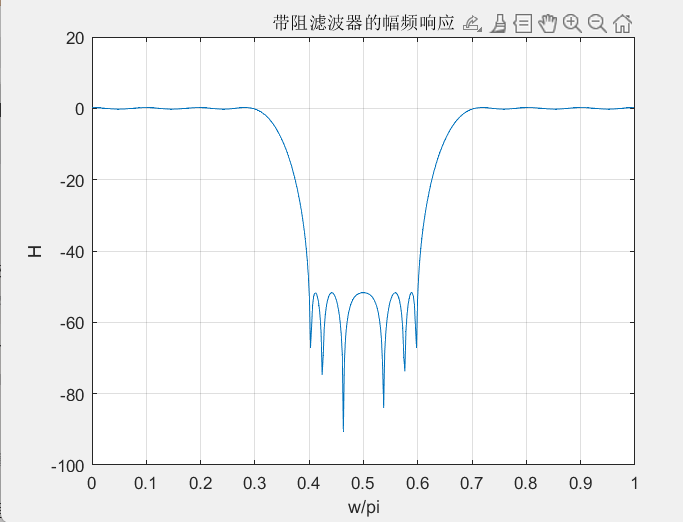
hn2=remez(M,fo,mo,wo);

[H,w]=freqz(hn2);

plot(w/pi,20\*log10(abs(H)));grid on;

title('带阻滤波器的幅频响应');xlabel('w/pi');ylabel('H');

实验结果：



实验分析与讨论（请手写）：

请比较窗函数法和等波纹最佳逼近法的滤波器阶数和滤波效果。

综合实践项目

钢琴乐音识别技术研究

1、实验目的

通过钢琴乐音识别技术的研究，掌握满足复杂工程问题需求的离散时间系统的基本设计方法与分析技术，了解影响设计目标和技术方案的各种因素，并得出有效结论。

2、实验内容

钢琴乐音识别技术对钢琴乐音信号进行基频估计，然后根据基频大小来区分音高，从而实现对乐曲的识别。

课题针对钢琴音频信号进行乐音识别技术的研究。在对音频信号其进行分帧、分音程检测后，对各音程段信号进行离散傅里叶变换，分析频谱中所蕴含的音符信息，并与乐谱进行比对，并得出结论。

请查阅提供的文献资料，并结合自己查阅的资料，完成以下实验内容：

(1) 基础要求（必做）：实现不同组别的钢琴音阶识别；

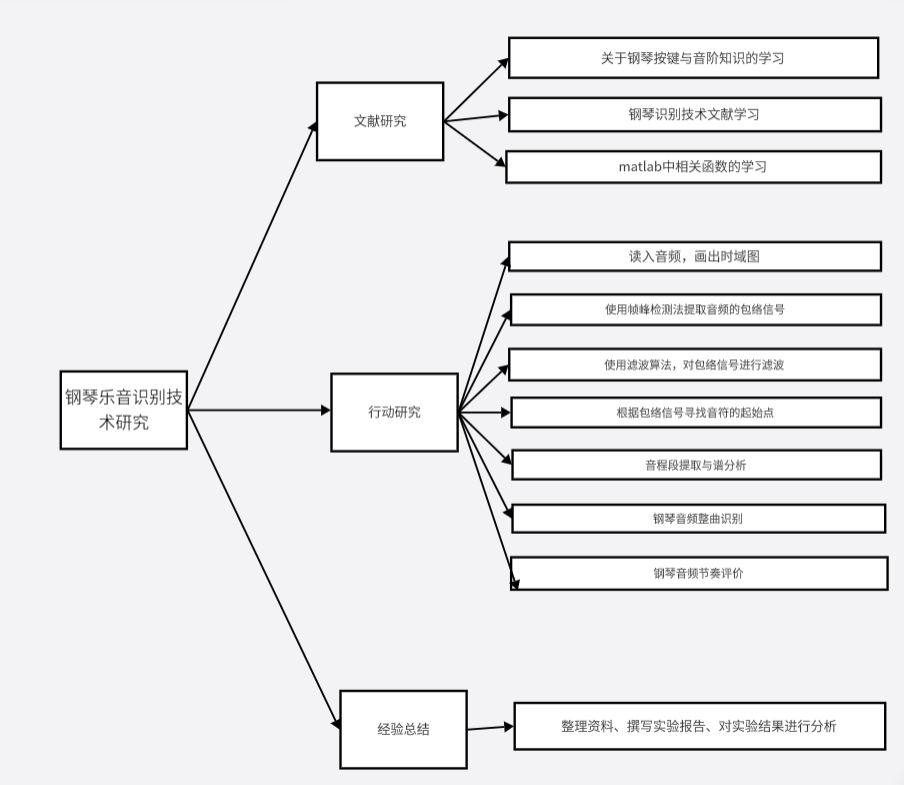
(2) 进阶要求（选做）：实现钢琴乐曲《小星星》的整曲识别。

(3) 拓展要求（选做）：对钢琴乐曲《小星星》的整曲节奏进行分析评价。

注：(2)和(3)至少选做1个。

3、实验报告要求

(1) 请写出钢琴乐音识别的技术路线；



1. 请阐述钢琴乐音识别的实现步骤；
2. 读入音频，画出时域图
3. 使用帧峰检测法提取音频的包络信号
4. 使用滤波算法，对包络信号进行滤波
5. 根据包络信号寻找音符的起始点
6. 音程段提取与谱分析
7. 钢琴音频整曲识别
8. 钢琴音频节奏评价
9. 请对实验结果进行分析，并得出自己的结论；

|  |  |  |
| --- | --- | --- |
| 时域包络图 | 音程的幅频响应 | 短时傅里叶谱图 |
| 峰值 | 音程提取与谱分析 | 短时傅里叶谱图 |
| 不同帧长下的谱泄漏现象对比 | 键号对应的图 | 实际节拍图 |
| 音程提取与谱分析2 | 键号对应的图 | 实际节拍 |

此实验参考老师的实验讲解PPT完成，代码中只包含了对69拍/min的音频的分析，实验途中还遇到了很多问题，例如帧峰点的横坐标与原来的时域图对不上，我利用了locs\*100/Fs作为横坐标画图才完成。对实际节拍图的分析得知汉明窗可以更好的处理谱泄漏问题。在绘制键号对应的图时，我对代码的理解有问题，对KeyEnergy\_MS的表示不太准确，导致绘图多次报错，请教同学后，对fdown和fup的表示有了新的理解。在实际节拍图，我发现节拍的识别还有误差，仍需要对代码进行优化。

此次实验将理论与实际相结合，让我对数字信号处理有了更具象化的学习，实验期间遇到了很多问题，也让我学到了新的知识。

(4) 请附上带注释的程序清单。

% 第一步：读入音频

[x,Fs]=audioread('D:\dsp\star\_69.m4a');%提取音频信号

music=x(:,1)./max(x(:,1)); %归一化

% 第二步：使用帧峰检测法提取音频的包络信号

Frame\_Num=100;%帧数

frames = buffer(music, Frame\_Num); % 分帧

envelope1 = max(frames) - min(frames);% 计算每帧的峰峰值

envelope1 = envelope1(:); % 转换为列向量

% 第三步：滤波处理包络信号

envelope2=medfilt1(envelope1,11); % 使用中值滤波对包络信号进行滤波

envelope2 = medfilt1(envelope2, 11);

envelope2 = medfilt1(envelope2, 11);

envelope2 = medfilt1(envelope2, 11);

envelope2 = medfilt1(envelope2, 11);

envelope2 = medfilt1(envelope2, 11);

% 绘制时域图和包络图

t\_music = (0:length(music)-1) / Fs; % 时域图的时间轴

t\_envelope = (0:length(envelope2)-1) / Fs; % 包络图的时间轴

figure(1); % 创建新的图形窗口

plot(t\_music, music); % 绘制时域图

hold on; % 保持绘图状态

plot(t\_envelope\*100, envelope2, 'r'); % 绘制包络图

hold off; % 结束绘图状态

xlabel('时间 (s)');

ylabel('幅度');

title('时域图和包络图');

% 添加图例

legend('时域图', '包络图');

% 第四步：寻找音符的起始点

peak\_Th = 0.5; % 阈值，小于该阈值的峰将被去除

[peaks, locs] = findpeaks(envelope2); % 寻找峰值点

valid\_peaks = peaks > peak\_Th; % 保留大于阈值的峰值

peaks = peaks(valid\_peaks);

locs = locs(valid\_peaks);

% 去除冗余峰值点，仅记录间隔超过最小峰间隔的点

min\_peak\_distance = 100; % 最小峰间隔，可根据需要进行调整

diff\_locs = diff(locs);

redundant\_locs = find(diff\_locs < min\_peak\_distance);

locs(redundant\_locs+1) = [];

peaks(redundant\_locs+1) = [];

hold on;

plot(locs\*100/Fs,peaks,'ro');

hold off;

% 第五步：音程段提取与谱分析

% 在这一步中，您需要根据具体需求选择窗函数和FFT长度，并进行相应的处理和分析

Amp\_MS = []; F0 = 0.1; %观察的谱间隔暂定为0.1Hz

N = Fs/F0; %FFT点数

% 1. 取音程段

for i=1:length(locs)

if i<length(locs)

%若不是最后一段则取当前段与后一段的前2/3的信号

X\_temp = music( locs(i)\*Frame\_Num: floor(locs(i)+(locs(i+1)-locs(i))\*2/3)\*Frame\_Num );

else

%若是最后一段，则按照前一段的长度取（因为最后一段后面没有下一段的标记了）

X\_temp = music( locs(i)\*Frame\_Num: floor(locs(i)+(locs(i)-locs(i-1))\*2/3)\*Frame\_Num );

end

X\_temp = X\_temp.\*hanning(length(X\_temp)); % 2. 乘以窗函数（暂定汉宁窗）

Amp\_temp = fft(X\_temp,N); % 3. 做FFT（N比X\_temp的长度要长，相当于添零了）

Amp\_MS = [ Amp\_MS abs(Amp\_temp) ]; %保存本段音程的谱分析结果

end

figure(2);

k=0:N-1;

for i=1:3

for j=1:5

if( (i-1)\*5+j <= length(locs) ) %小于现有音程的段数

subplot(3,5,(i-1)\*5+j);

plot(k\*Fs/N, Amp\_MS(:,(i-1)\*5+j)/max(Amp\_MS(:,(i-1)\*5+j)) );

axis([0 2000 0 1]);

end

end

end

% 画谱图信息（按照固定帧画）

test\_Num = 400; %画谱图的临时帧长（非前面计算包络的帧长Frame\_Num） F0=1; %观察谱间隔暂定1 Hz

frame\_N = floor(length(music)/test\_Num); %帧数

frame\_t = (0:frame\_N-1)\*test\_Num/Fs; %谱图横坐标单位调为时间(s)

F0 = 1;%依据81行代码修改帧长

fft\_N = Fs/F0; %每一帧FFT的点数，暂设为44100点

frame\_f = (0:floor((fft\_N-1)))\*F0; %谱图纵坐标单位为Hz

frame\_A = zeros(fft\_N,frame\_N); % frame\_A记录某一帧的某一频率处的值，后面用颜色显示

for i=1:frame\_N-1

%每一帧400点，均乘400点哈明窗后，做44100点FFT

frame\_A(:,i)= abs( fft( music( (i-1)\*test\_Num+1 : i\*test\_Num ).\*hamming(test\_Num), fft\_N) );

end

figure(3); subplot(211);

imagesc(frame\_t,frame\_f,frame\_A);

M\_T = ceil(length(music)/44100);

axis xy; % axis xy ：笛卡尔轴模式，原点在左下角，y轴是竖直的，由底至顶标数，x轴是水平的，从左往右标数

axis([0 M\_T 0 2000]); % M\_T是音乐总的时间，M\_T = ceil(length(音频点数)/44100);

figure(4);

starti=locs(1)\*Frame\_Num;

test\_N = 400;

i=starti:starti+test\_N;

test\_music=music(i);

subplot(331)

plot(test\_music);title('实际长度：400点');

test\_music\_FFT=fft(test\_music,44100);

subplot(334)

stem(abs(test\_music\_FFT)/max(abs(test\_music\_FFT)));

title('矩形窗谱图，44100点FFT');axis([0 1000 0 1]);

test\_music\_FFT=fft(test\_music.\*hamming(length(test\_music)),44100);

subplot(337)

stem(abs(test\_music\_FFT)/max(abs(test\_music\_FFT)));

title('哈明窗谱图，44100点FFT');axis([0 1000 0 1]);

test\_N = 4000;

i=starti:starti+test\_N;

test\_music=music(i);

subplot(332)

plot(test\_music);title('实际长度：4000点');

test\_music\_FFT=fft(test\_music,44100);

subplot(335)

stem(abs(test\_music\_FFT)/max(abs(test\_music\_FFT)));

title('矩形窗谱图，44100点FFT');axis([0 1000 0 1]);

test\_music\_FFT=fft(test\_music.\*hamming(length(test\_music)),44100);

subplot(338)

stem(abs(test\_music\_FFT)/max(abs(test\_music\_FFT)));

title('哈明窗谱图，44100点FFT');axis([0 1000 0 1]);

test\_N = round((locs(2)\*Frame\_Num-locs(1)\*Frame\_Num+1)\*2/3);

i=starti:starti+test\_N;

test\_music=music(i);

subplot(333)

plot(test\_music);title('实际长度：一个音程\*2/3');

test\_music\_FFT=fft(test\_music,44100);

subplot(336)

stem(abs(test\_music\_FFT)/max(abs(test\_music\_FFT)));

title('矩形窗谱图，44100点FFT');axis([0 1000 0 1]);

test\_music\_FFT=fft(test\_music.\*hamming(length(test\_music)),44100);

subplot(339)

stem(abs(test\_music\_FFT)/max(abs(test\_music\_FFT)));

title('哈明窗谱图，44100点FFT');axis([0 1000 0 1]);

frame\_t\_ms = locs\*Frame\_Num/Fs; %谱图横坐标

frame\_N = length(locs);

fft\_N = Fs/F0;

frame\_f\_ms = (0:floor((fft\_N-1)))\*F0;

frame\_A\_ms = zeros(fft\_N,frame\_N);

for i=2:frame\_N

test\_ms = music( locs(i-1)\*Frame\_Num+1 : locs(i)\*Frame\_Num );

frame\_A\_ms(:,i-1)= abs( fft( test\_ms.\*hamming(length(test\_ms)), fft\_N) );

end

%最后一个音程的处理

test\_ms = music( locs(i)\*Frame\_Num+1 : locs(i)\*Frame\_Num+length(test\_ms));

frame\_A\_ms(:,i)= abs( fft( test\_ms.\*hamming(length(test\_ms)), fft\_N) );

figure(3);subplot(212)

imagesc(frame\_t\_ms,frame\_f\_ms,frame\_A\_ms);axis xy; axis([0 M\_T 0 2000]);

% 第六步：钢琴音频整曲识别

% 根据键号与频率的对应关系表进行音频识别，并显示识别结果

% 读入钢琴按键与频率的对应表

Piano\_F=xlsread('D:\dsp\Piano\_Key\_F.xlsx');

Piano\_F=Piano\_F(1:88);

key=1:88;

delter\_fq=zeros(1,length(Piano\_F)-1);

fdown=zeros(1,length(Piano\_F));

fup=zeros(1,length(Piano\_F));

for i=2:length(Piano\_F)

delter\_fq(1,i-1)=Piano\_F(i)-Piano\_F(i-1);

end

KeyEnergy\_MS = zeros(88, length(locs)); % 初始化 KeyEnergy\_MS 为一个 88x音程段数 的矩阵

for i=1:length(Piano\_F)

if(i==1)

fdown(:,i)=Piano\_F(i,:)-delter\_fq(:,i)/2;

fup(:,i)=Piano\_F(i,:)+delter\_fq(:,i)/2;

elseif(i==length(Piano\_F))

fdown(:,i)=Piano\_F(i,:)-delter\_fq(:,i-1)/2;

fup(:,i)=Piano\_F(i,:)+delter\_fq(:,i-1)/2;

else

fdown(:,i)=Piano\_F(i,:)-delter\_fq(:,i-1)/2;

fup(:,i)=Piano\_F(i,:)+delter\_fq(:,i)/2;

end

end

for i=1:length(locs)

for j=1:88

KeyEnergy\_MS(j,i) = mean(Amp\_MS(round((fdown(:,j)/F0):round(fup(:,j)/F0)),i));

end

KeyEnergy\_MS(:,i)=KeyEnergy\_MS(:,i)/max(KeyEnergy\_MS(:,i));

end

figure(5);

for i=1:6

for j=1:7

if( (i-1)\*7+j <= length(locs) ) %有13个音程

subplot(6,7,(i-1)\*7+j);

plot(key,KeyEnergy\_MS(:,(i-1)\*7+j));

axis([0 88 0 1]);

end

end

end

%钢琴音频节奏评

%标准节拍

ideal\_Beat\_T=[1 1 1 1 1 1 2 1 1 1 1 1 1 2 1 1 1 1 1 1 2 1 1 1 1 1 1 2 1 1 1 1 1 1 2 1 1 1 1 1 1 ];

%实际节拍

real\_Beat\_T=zeros(1,length(locs)-1);

error\_beat\_T=zeros(1,length(locs)-1);

for i=1:length(locs)-1

real\_Beat\_T(i)=(69/60)\*((locs(i+1)-locs(i))/Fs);

error\_beat\_T(i)=(real\_Beat\_T(i)-ideal\_Beat\_T(i))/ideal\_Beat\_T(i);

end

figure(6);

subplot(2,1,1);

plot(1:length(locs)-1,real\_Beat\_T,'bo');

hold on;

plot(1:length(locs)-1,ideal\_Beat\_T,'r.');

axis([0,length(locs)-1,0,2.2]);

title('实际节拍与标准节拍对比');

subplot(2,1,2);

plot(1:length(locs)-1,error\_beat\_T,'rx-');

axis([0,length(locs)-1,-0.5,0.5]);

title('误拍率');