**Final Report**

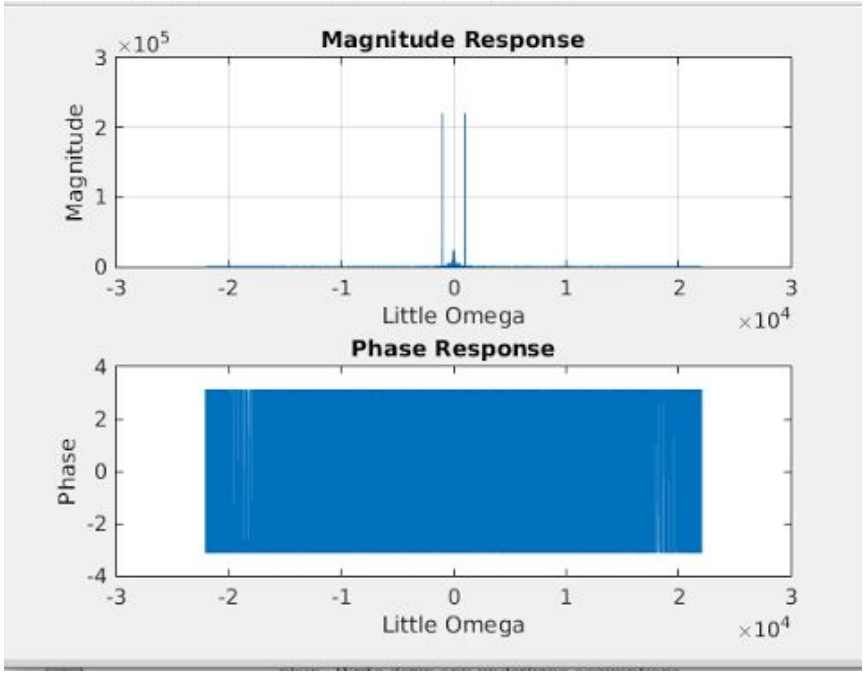
Name: Zhijie Jin

NetID: zhijiej2

1. Spectral Analysis

a)

```
%% 1. Spectral Analysis
load song1.mat
song1 = sigp;
Fs = 44100;

N = length(song1);
sigp_fft = fft(song1);
sigp_fftshift = fftshift(sigp_fft);
w = fftshift((0:N-1)/N*2*pi);
w(1:N/2) = w(1:N/2) - 2*pi;
Freq = Fs*w/(2*pi);

figure();
subplot(2,1,1);
plot(Freq, abs(sigp_fftshift));
xlabel('Little Omega');
ylabel('Magnitude');
title('Magnitude Response');
grid on;

subplot(2,1,2);
plot(Freq, angle(sigp_fftshift));
xlabel('Little Omega');
ylabel('Phase');
title('Phase Response');
```



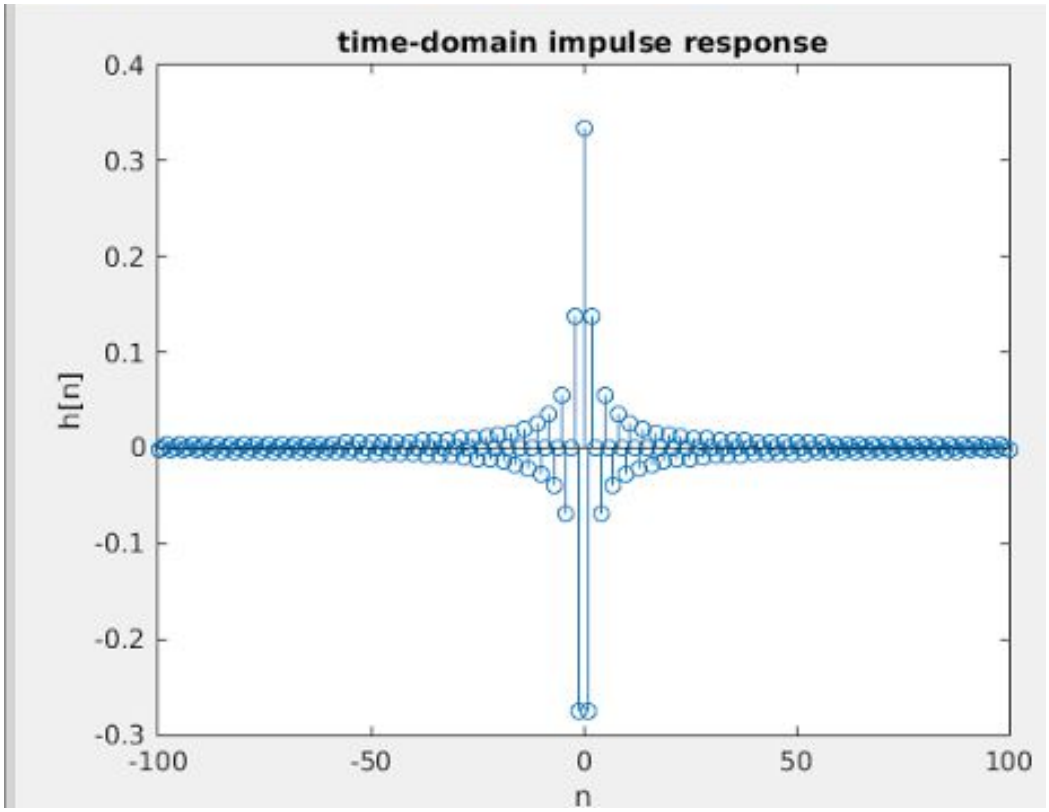
b) The contaminating pitch is at approximately 100 Hz, because there is an impulse at approximately 100 Hz.

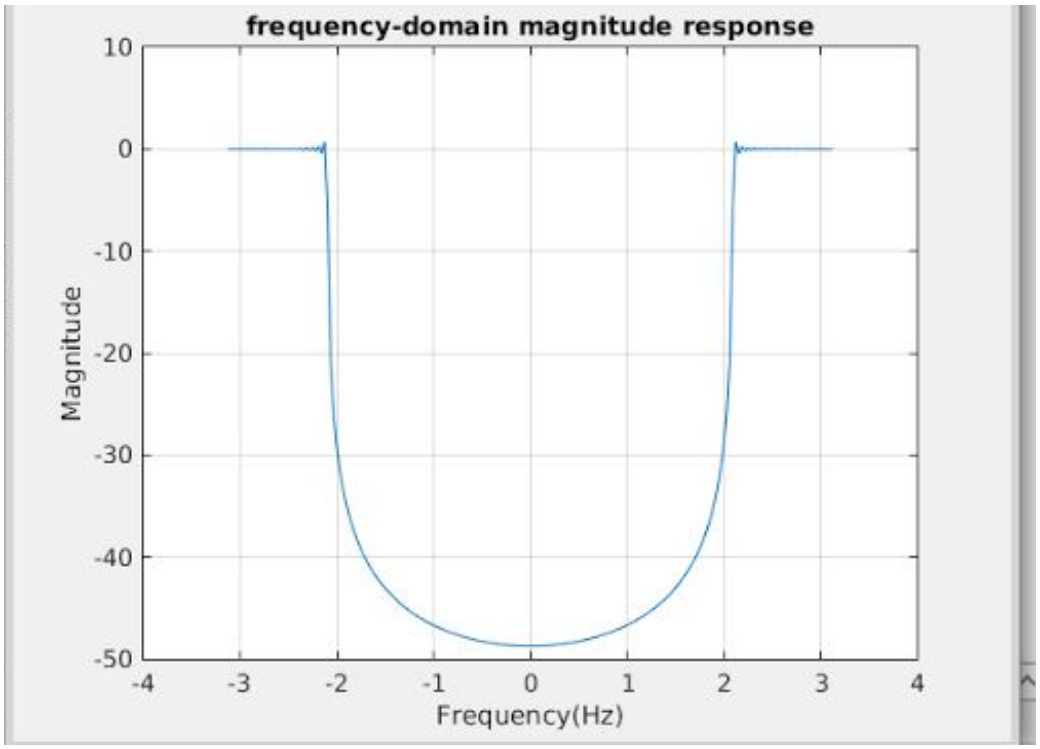
c) Yes, a low pass filter is reasonable to use to remove the pitch.
Beyond 100 Hz, the magnitude response of the signal is almost 0. Most of the information is contained within -100Hz and 100Hz.

2. Filter Design

a)

```
N = 100;
n = -N:N;
wc = 2/3*pi;

delta = zeros(1, length(n));
delta(N+1) = 1;
hn = delta - wc./pi.*sinc(wc./pi.*n);
figure();
stem(n, hn)
title('time-domain impulse response')
xlabel('n')
ylabel('h[n]')

N2 = length(hn);
hn_fft = fft(hn);
hn_fftshift = fftshift(hn_fft);
w = fftshift((0:N2-1)/N2*2*pi);
w(1:N2/2) = w(1:N2/2) - 2*pi;

figure();
plot(w, mag2db(abs(hn_fftshift)));
xlabel('Frequency(Hz)');
ylabel('Magnitude');
title('frequency-domain magnitude response');
grid on;
```

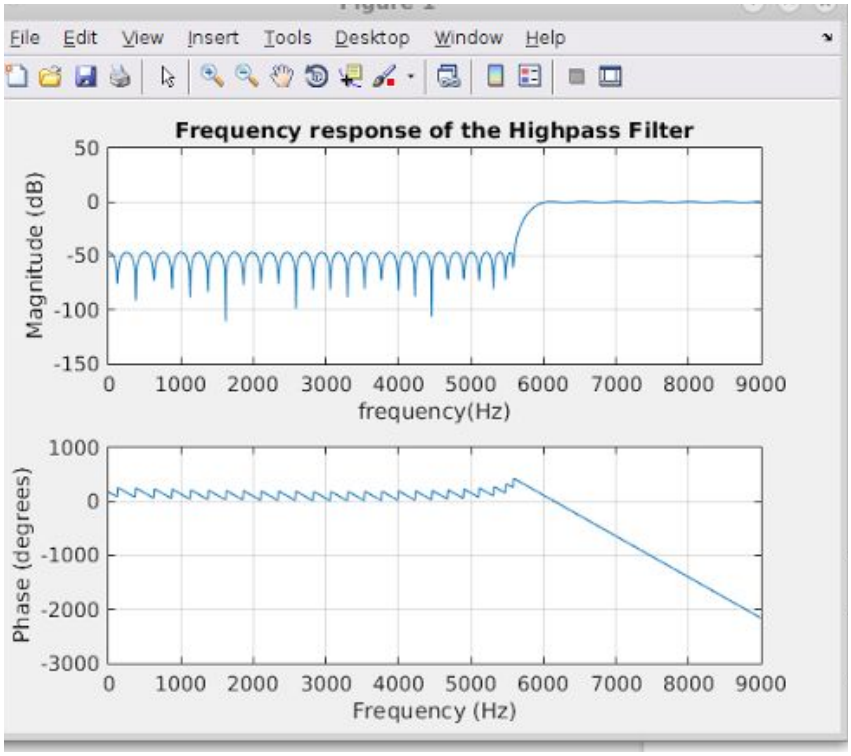

time-domain impulse response
0.4
0.3
0.2
0.1
0
-0.1
-0.2
-0.3
h[n]
-100
-50
0
50
100
n

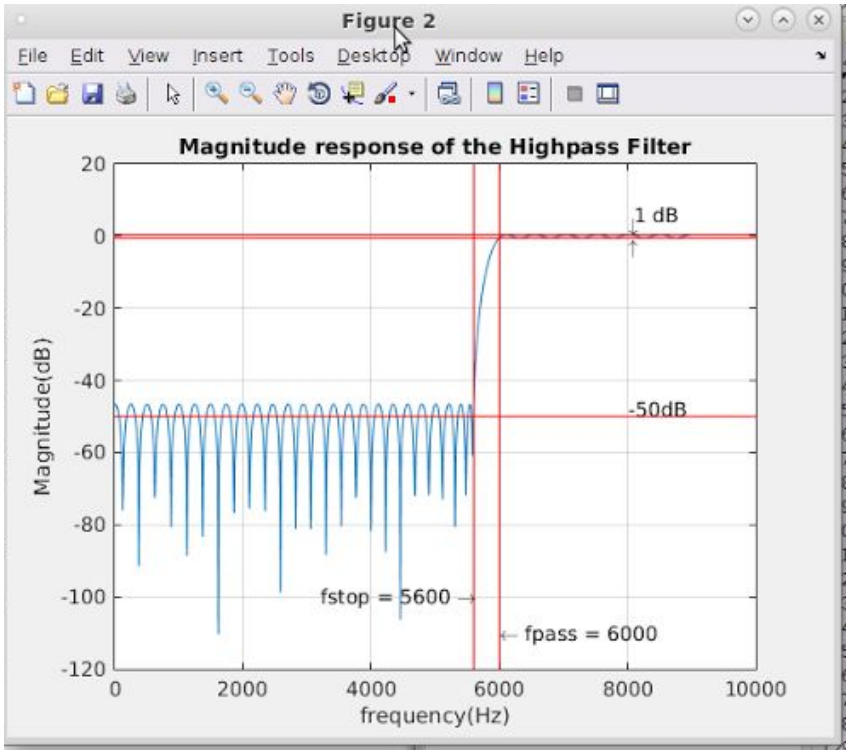

frequency-domain magnitude response
10
0
-10
-20
-30
-40
-50
Magnitude
-4
-3
-2
-1
0
1
2
3
4
Frequency(Hz)

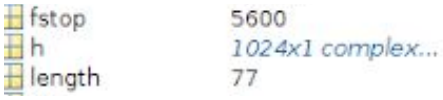
b)

```
fs = 18000;
fstop = 5600;
fpass = 6000;
rs= 45;
rp = 1;

d = designfilt('highpassfir','StopbandFrequency',fstop, ...
  'PassbandFrequency',fpass,'StopbandAttenuation',rs, ...
  'PassbandRipple',rp,'SampleRate',fs,'DesignMethod','equiripple');

figure()
freqz(d.Coefficients,1,1024,fs)
title('Frequency response of the Highpass Filter')
xlabel('frequency(Hz)')

% generate marked plot
[h,w] = freqz(d.Coefficients,1,1024,fs);
length = length(d.Coefficients);

% plot magnitude response
figure();
plot(w, mag2db(abs(h)))

% mark 50dBHigh pass filter design
fifty = -50;
fiftyline = refline([0, fifty]);
fiftyline.Color = 'r';
txt1 = '-50dB';
text(8000, -48, txt1)

% mark rp
rpl = -0.5;
fiftyline = refline([0, rpl]);
fiftyline.Color = 'r';
txt2 = '\uparrow';
text(8000, -3, txt2)
```

```
rph = 0.5;
fiftyline = refline([0, rph]);
fiftyline.Color = 'r';
y = get(gca, 'ylim');
txt3 = '\downarrow';
text(8000, 3, txt3)

txt4 = '1 dB';
text(8100, 6, txt4)

% mark fstop
hold on
plot([5600 5600], y, 'r')
txt5 = 'fstop = 5600 \rightarrow';
text(3200, -100, txt5)

% mark fpass
hold on
plot([6000 6000], y, 'r')
txt6 = '\leftarrow fpass = 6000';
text(6000, -110, txt6)

grid on;
title('Magnitude response of the Highpass Filter')
xlabel('frequency(Hz)')
ylabel('Magnitude(dB)')
```




Filter length is 77.

3. Image Processing

a). Using low pass filter, because it reduces the edge content in an image, which in return resulting a blur image.

b).

```
131 -    I = imread('Grumpy-Cat.jpg');
132 -    Iblure = imgaussfilt(I(:,:,1), 16);
133 -    figure();
134 -    subplot(1,2,1);
135 -    imshow(I)
136 -    title('Original image')
137 -    subplot(1,2,2);
138 -    imshow(Iblure)
139 -    title('blured cat')
```

imgaussfilt filters the image I with a 2-D Gaussian smoothing kernel with the standard deviation of 16. Therefore, the gaussian filter resembles a low pass filter and can be used to blur pictures.

c). Using high pass filter, because it increases the edge content in an image, which in return identifies a coin.

d).

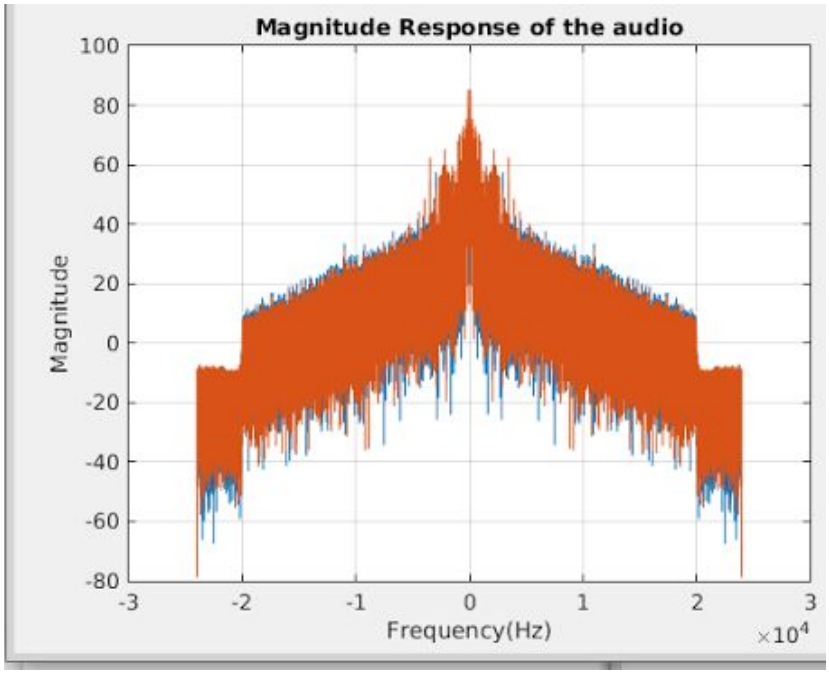
```
149 -    y = imread('coins.png');
150 -    figure();
151 -    imagesc(y)
152 -    colormap('gray')
153 -    title('Image2 Before Filtering')
154 -    h = [-1, -1, -1; -1, 8, -1; -1, -1, -1];
155 -    y2 = conv2(double(h), double(y(:,:,1)));
156 -    figure();
157 -    imagesc(y2)
158 -    colormap('gray')
159 -    title('Image2 After Filtering')
```

h is a high pass filter. By applying the high pass filter, we shows the deges in the picture and eliminates other effects.

4. Multi-rate DSP

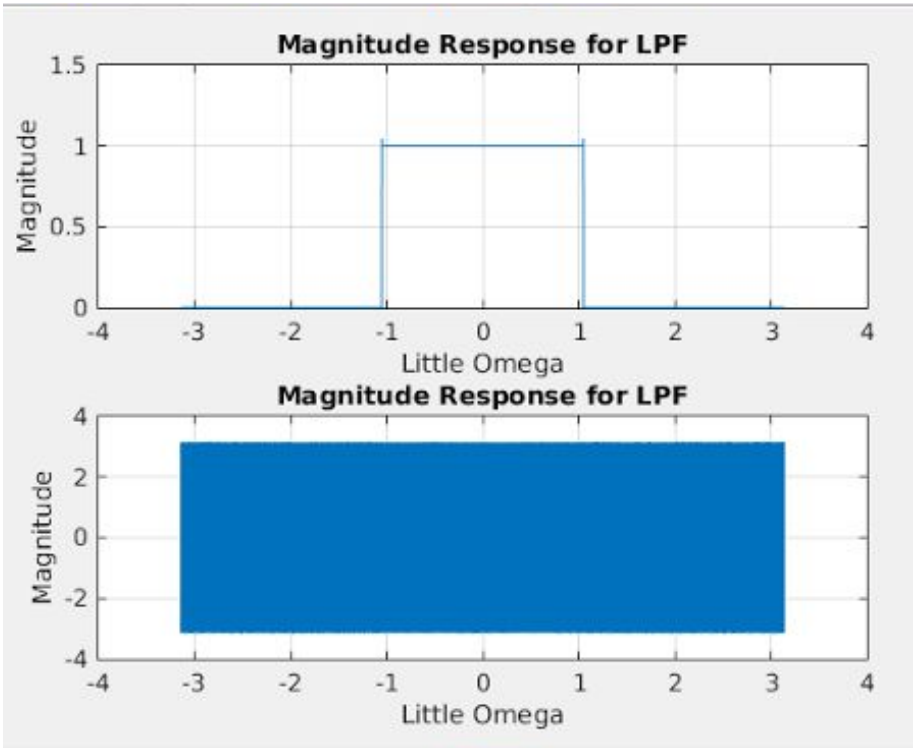
a).

```
163 -    [audio, Fs] = audioread('Radiohead_chopped.wav');
164      % sound(audio, Fs)
165 -    N = length(audio);
166
167 -    audio_fft = fft(audio);
168 -    audio_fftshift = fftshift(audio_fft);
169 -    w = fftshift((0:N-1)/N*2*pi);
170 -    w(1:N/2) = w(1:N/2) - 2*pi;
171 -    Freq = Fs*w/(2*pi);
172
173 -    figure();
174 -    plot(Freq, mag2db(abs(audio_fftshift)));
175 -    xlabel('Frequency(Hz)');
176 -    ylabel('Magnitude');
177 -    title('Magnitude Response of the audio');
178 -    grid on;
179
```



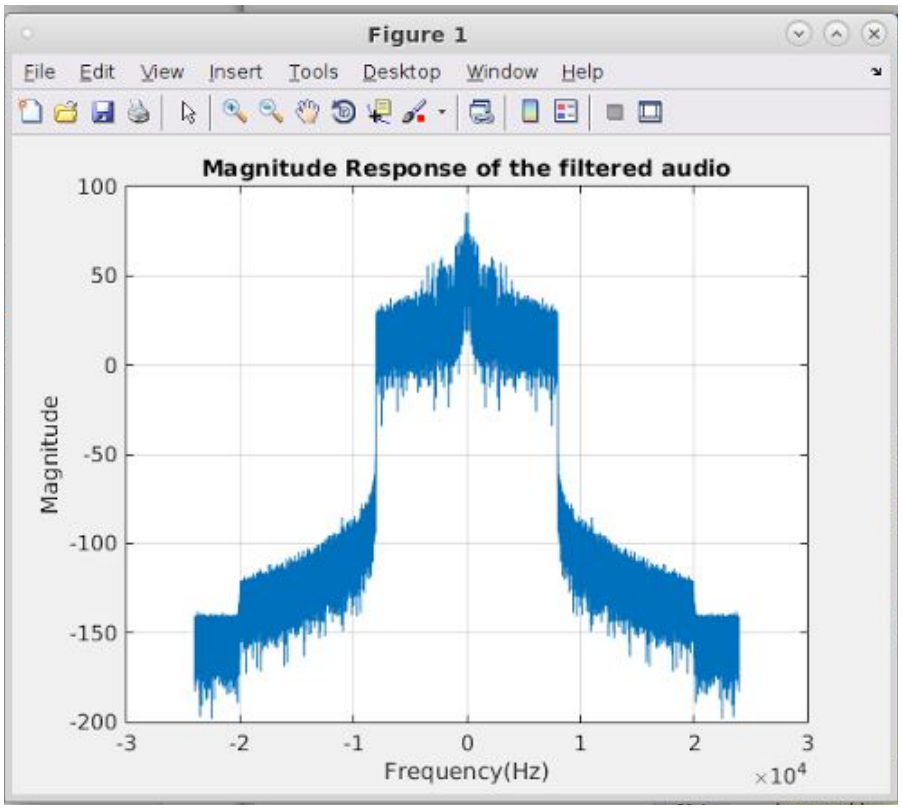
b).

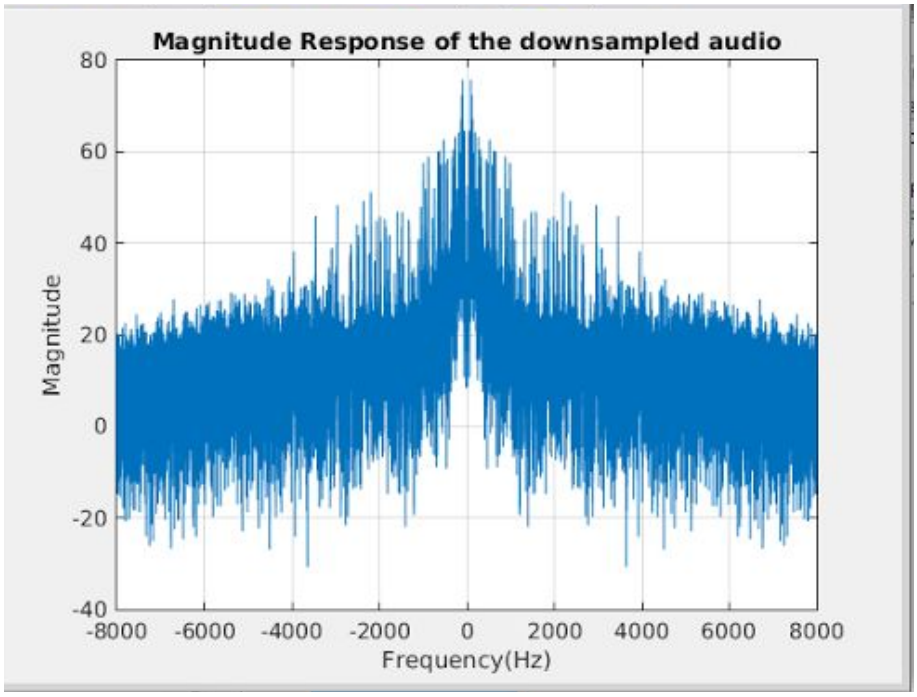
```
180 -   %%
181 -   N= length(audio);
182 -   n = -N/2:1:N/2-1;
183 -   wc = pi/3;
184 -   hl = wc./pi.*sinc(wc./pi.*n);
185 -   HL = fft(hl);
186 -   Hwl = fftshift(HL);
187 -   w = fftshift((0:length(n)-1)/length(n)*2*pi);
188 -   w(1:length(n)/2) = w(1:length(n)/2) - 2*pi;
189
190
191 -   figure();
192 -   subplot(2,1,1);
193 -   plot(w, abs(Hwl));
194 -   xlabel('Little Omega');
195 -   ylabel('Magnitude');
196 -   title('Magnitude Response for LPF');
197 -   grid on;
198
199 -   subplot(2,1,2);
200 -   plot(w, angle(Hwl));
201 -   xlabel('Little Omega');
202 -   ylabel('Magnitude');
203 -   title('Magnitude Response for LPF');
204 -   grid on;
```



c).

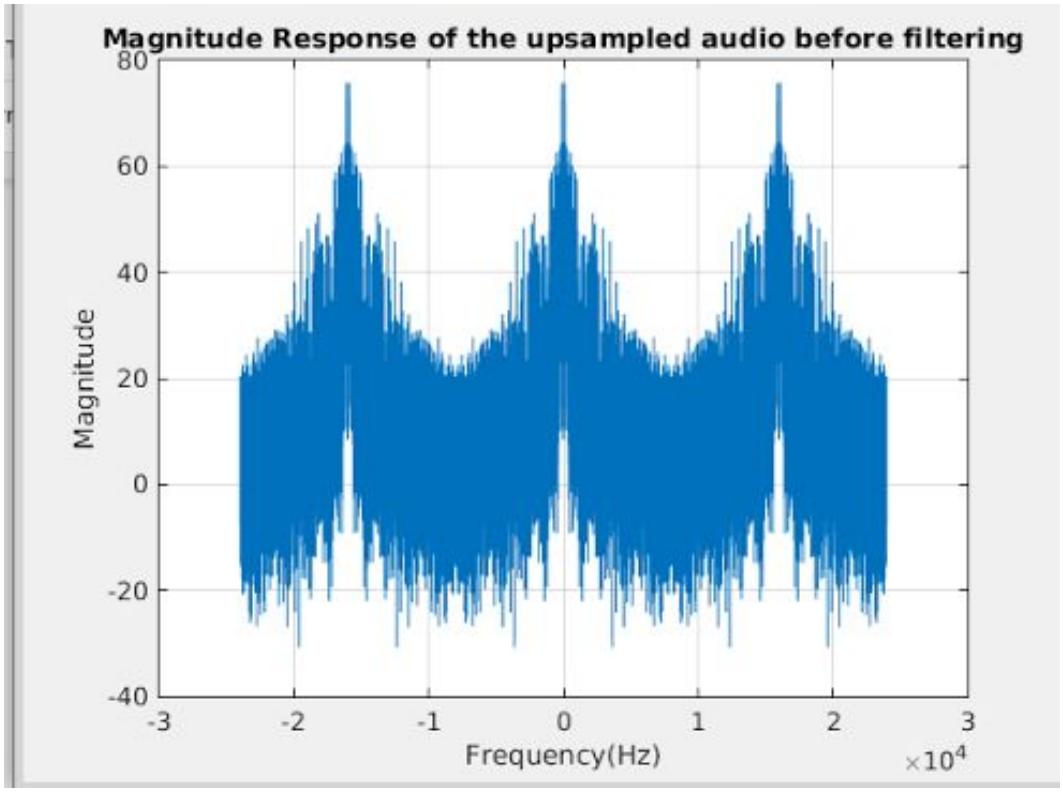
```
210
211 -   audioLPF = audio_fftshift(:,1).*(Hwl.');
212 -   audiodownsamples = real(ifft(ifftshift(audioLPF)));
213
214     % down sampling
215 -   D = 3;
216 -   audiodown = downsample(audiodownsamples, D);
217 -   N = length(audiodown);
218 -   audiodown_fft = fft(audiodown);
219 -   audiodown_fftshift = fftshift(audiodown_fft);
220 -   w = fftshift((0:N-1)/N*2*pi);
221 -   w(1:N/2) = w(1:N/2) - 2*pi;
222 -   Freq = 1/D*Fs*w/(2*pi);
223
224 -   figure();
225 -   plot(Freq, mag2db(abs(audiodown_fftshift)));
226 -   xlabel('Frequency(Hz)');
227 -   ylabel('Magnitude');
228 -   title('Magnitude Response of the downsampled audio');
229 -   grid on;
230
```

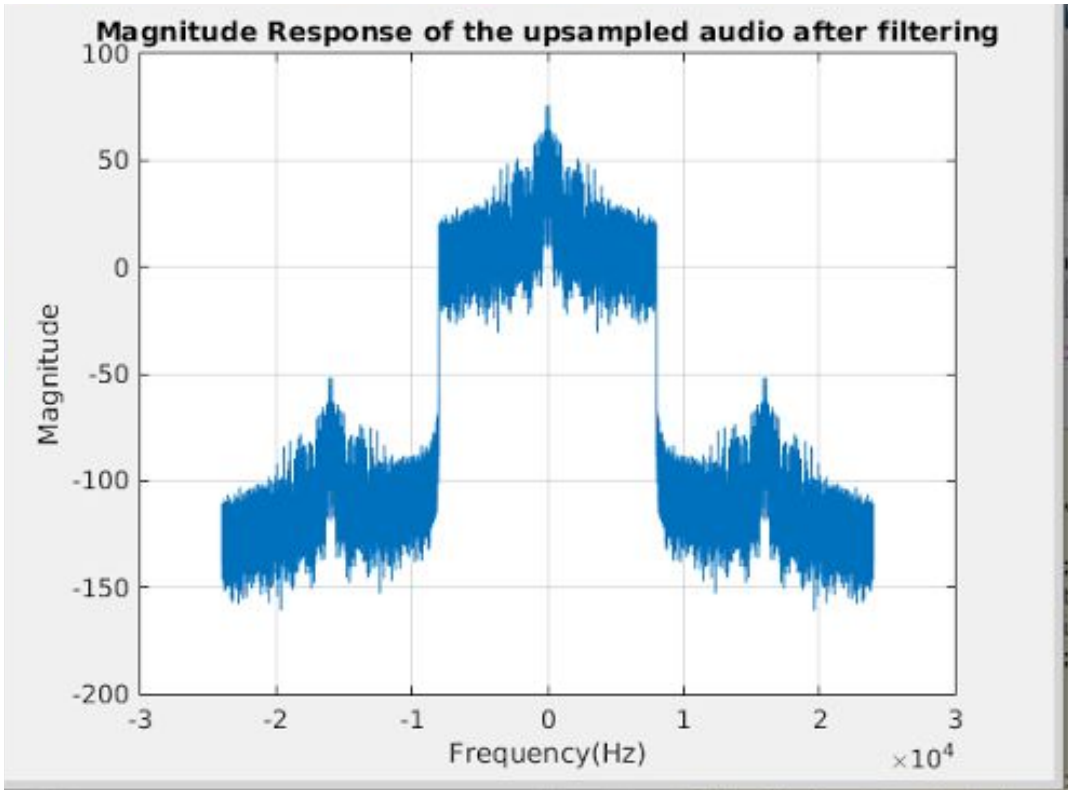





d).

```
237
238     %%
239 -   U = 3;
240 -   audioup = upsample(audiodown, U);
241 -   N = length(audioup);
242 -   audioup_fft = fft(audioup);
243 -   audioup_fftshift = fftshift(audioup_fft);
244 -   w = fftshift((0:N-1)/N*2*pi);
245 -   w(1:N/2) = w(1:N/2) - 2*pi;
246 -   Freq = U/D*Fs*w/(2*pi);
247
248 -   figure();
249 -   plot(Freq, mag2db(abs(audioup_fftshift)));
250 -   xlabel('Frequency(Hz)');
251 -   ylabel('Magnitude');
252 -   title('Magnitude Response of the upsampled audio before filtering');
253 -   grid on;
254
255 -   Hwl = [Hwl 0];
256 -   audioupLPF2 = audioup_fftshift(:,1).*(Hwl.');
257
258 -   N = length(audio)+1;
259 -   w = fftshift((0:N-1)/N*2*pi);
260 -   w(1:N/2) = w(1:N/2) - 2*pi;
261 -   Freq = Fs*w/(2*pi);
262
263 -   figure();
264 -   plot(Freq, mag2db(abs(audioupLPF2)));
265 -   xlabel('Frequency(Hz)');
266 -   ylabel('Magnitude');
267 -   title('Magnitude Response of the upsampled audio after filtering');
268 -   grid on;
```


Magnitude Response of the upsampled audio before filtering
Magnitude
Frequency(Hz)
×10^4
80
60
40
20
0
-20
-40
-3
-2
-1
0
1
2
3


Magnitude Response of the upsampled audio after filtering
Magnitude
Frequency(Hz)
×10^4
100
50
0
-50
-100
-150
-200
-3
-2
-1
0
1
2
3

5. Spectrograms and Chirps

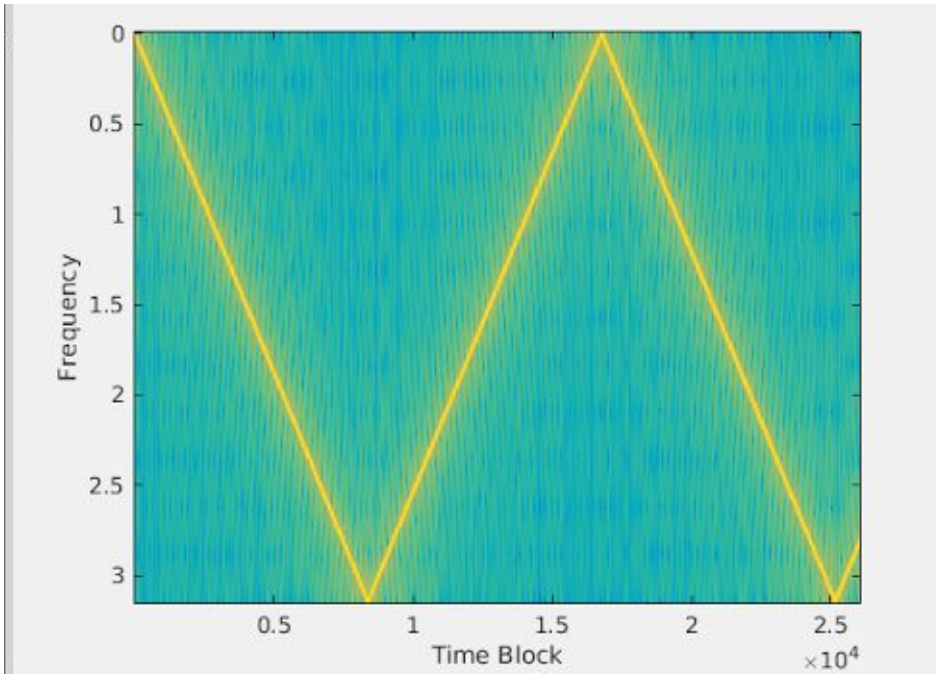
a). 8192*pi = inst = 4000t, t = 6.43s

Therefore, the aliasing appears at 6.43s

At aliasing, the sound is distorted and does not sound smoothly.

b).

```
268 -    load('waveform1.mat')
269 -    Fs = 8192;
270 -    sound(sig, Fs)
271
272 -    N = 500;
273 -    w_function = hamming(500);
274
275 -    [s, w, t] = spectrogram(sig, w_function ,250, 512);
276 -    figure()
277 -    imagesc(t,w,log(abs(s)));
278 -    xlabel('Time Block')
279 -    ylabel('Frequency')
280
```




c). There is a zig-zag. It represents the pitch of the signal. When listen to the signal, the pitch first decreases and then increases and then decreases again. Then after a short time of increases in pitch, the signal ends. Therefore, the zig-zag shape represents the pitch that I heard by playing the signal.