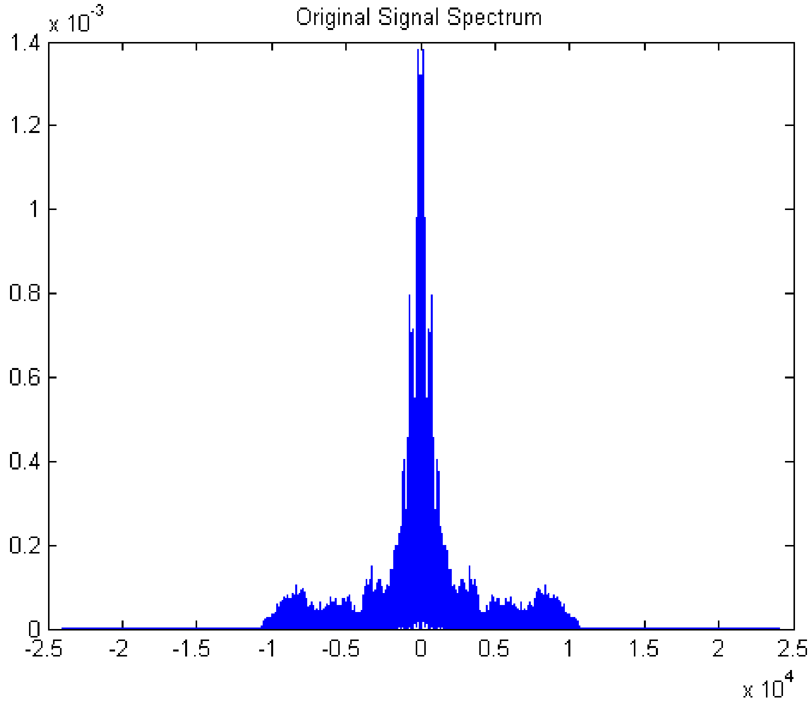
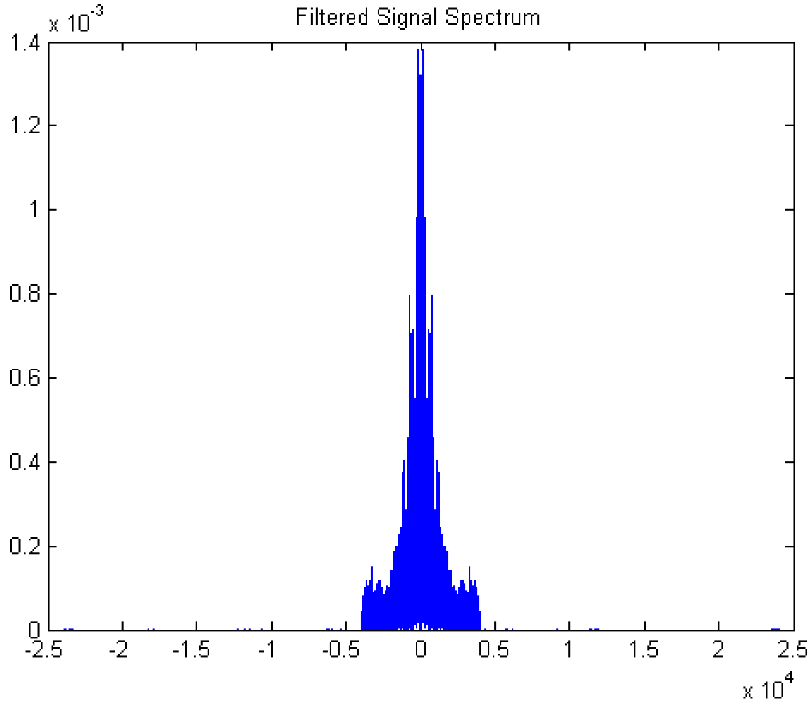
```
clear;
close all;

% Exp 3

[S, Fs] = audioread('eric.wav');

% Fourier transform
L = length(S);
F = fftshift(fft(S));
f = Fs/2*linspace(-1,1,L);

% Plot
figure; plot(f, abs(F)/L); title('Original Signal Spectrum');

% Filter as 4kHz
W = 4000;
F(f>=W | f<=-W) = 0;
y =ifft(ifftshift(F));

% Fourier Transform
L = length(y);
F = fftshift(fft(y));
f = Fs/2*linspace(-1,1,L);

% Plot after filter
figure; plot(f, abs(F)/L); title('Filtered Signal Spectrum');

%calculate time vector
tstart = 0;
tend = tstart + length(y) / Fs;
t1 = linspace(tstart, tend, length(y));
t1 = t1';

figure; plot(t1, y); title('Filtered Signal Time Domain');

% Hear the signal
sound(abs(y), Fs);

% Calculate constants
kf = 0.2/(2*pi*max(abs(cumsum(y)))./Fs);
fc=100000;
Ac = 1;

% resample at 5Fc
y = resample(y, 5*fc, Fs);
Fs = 5*fc;

%calculate time vector
tstart = 0;
tend = tstart + length(y) / Fs;
t1 = linspace(tstart, tend, length(y));
t1 = t1';

%FM modulated signal
y = Ac * cos(2*pi*fc*t1 + 2*pi*kf*cumsum(y)./Fs);

% Fourier transform
L = length(y);
F = fftshift(fft(y));
f = Fs/2*linspace(-1,1,L);
figure; plot(f, F); title('FM modulation in Frequency Domain');

% Discriminator
d_y = diff(y);
d_y = [0; d_y];
```

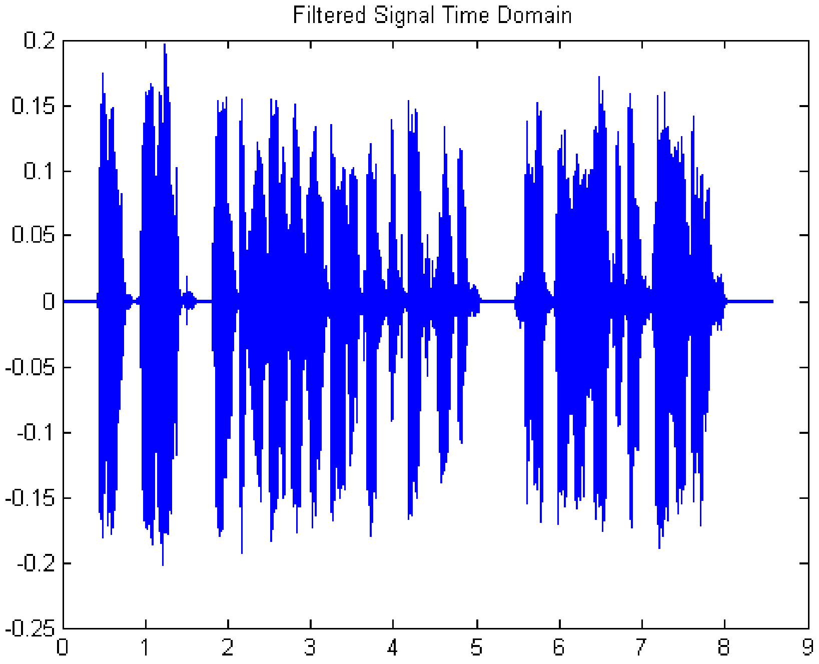
```
% envelope detector
envelopeFM = abs(hilbert(d_y)) -  mean(abs(hilbert(d_y)));

% plot demodulated signal in time domain
figure; plot(t1,envelopeFM); title('Demodulated FM in Time Domain using ED');
ylim([-2*10^-4 2*10^-4]);

% resample to sound the signal
envelopeFM = resample(envelopeFM, Fs/5, Fs);
sound(500.*abs(envelopeFM), Fs/5);
```

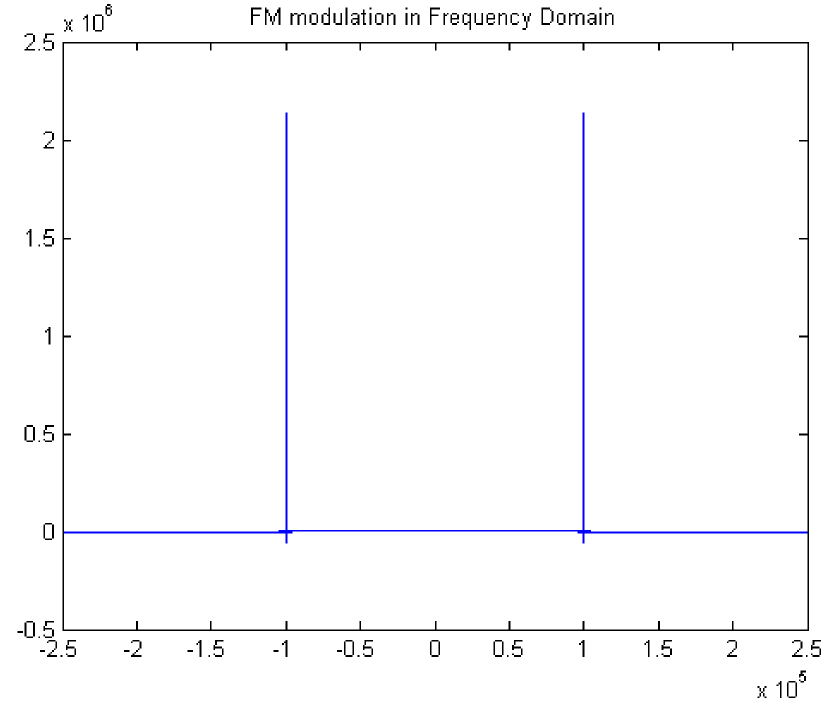


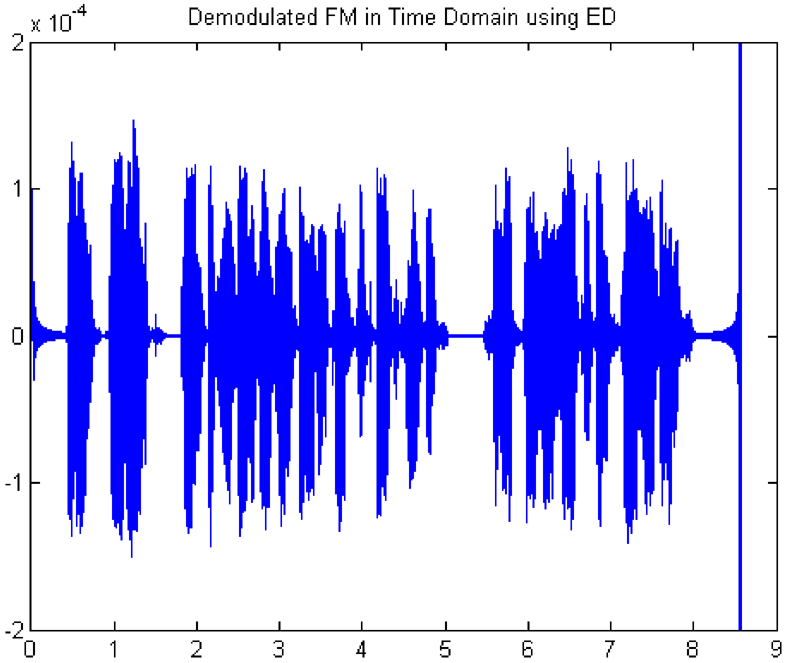
Original Signal Spectrum



Filtered Signal Spectrum


Filtered Signal Time Domain
0.2
0.15
0.1
0.05
0
-0.05
-0.1
-0.15
-0.2
-0.25
0
1
2
3
4
5
6
7
8
9


x 10^6
FM modulation in Frequency Domain
2.5
2
1.5
1
0.5
0
-0.5
-2.5
-2
-1.5
-1
-0.5
0
0.5
1
1.5
2
2.5
x 10^5



Published with MATLAB® R2013a