

**Final Project 2015**

Tang Qi

M-B5-5515-2

*Department:*

Electrical and Computer Engineering

*Course Code:*

ELCE705

*Course Title:*

Digital Signal Processing

**Part I. (50%) Digital Filter for Guitar Wave**

In this part, the spectrum of wav file guitar1.wav is analyzed. A digital filter is designed and used to remove interference (a single tone added) from an audio signal.

**Signal Access and Exploration**

1. **Use MATLAB’s wavread command to load the guitar1.wav file:**
2. **Look at the guitar signal in the frequency domain by computing and plotting (in dB) the windowed DFT of various 8192-point blocks of the guitar signal. Please indicate in your report in what range the significant portion of the guitar signal’s spectrum lies?**

% load the guitar1.wav

[y,Fs]=wavread(( 'guitar1.wav'));

% sample

y\_rec= y.\*rectwin(length(y));

figure

subplot(3,1,1)

plot(0:length(y)-1,y);

% rectangular window

subplot(3,1,2)

plot(0:length(y)-1,y\_rec);

grid on

xlabel('length');

ylabel('amplitude');

title('Rectangular Window');

% hamming window

y\_ham=y.\*hamming(length(y));

subplot(3,1,3)

plot(0:length(y)-1,y\_ham);

grid on

xlabel('length');

ylabel('amplitude');

title('Hamming Window');

%-------------------------------------choose rectangular window

Y\_REC=fft(y\_rec,8192);

figure

subplot(2,1,1);

plot(0:8191,20.\*log10(abs(Y\_REC)));

% plot(0:8191,20.\*log10(abs(Y\_REC)));

grid on

ylabel('dB');

title('DFT after Rectangular Window Truncation');

%\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_DFT after rectangular window, in dB

subplot(2,1,2);

plot((0:8191)/8192\*2,abs(Y\_REC));

grid on

xlabel('w/pi');

%----------------------------------significant portion

sound(y,Fs);

%----------------------------------sound command

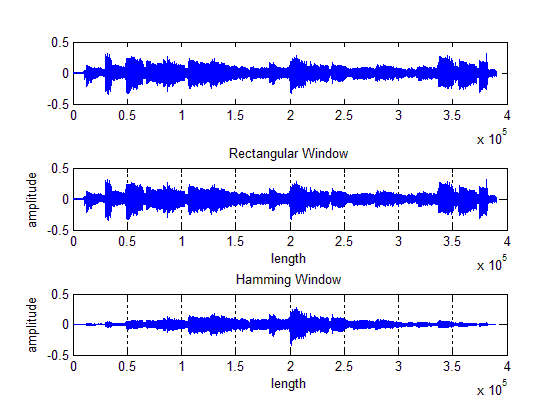


Fig.01 Original sample and truncated by rectangular & hamming window

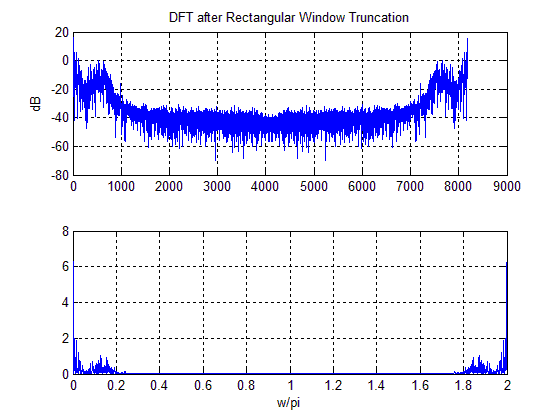


Fig.02 DFT after rectangular window truncation

Before doing the DFT, rectangular window and hamming window is applied to truncate the signal, as shown in Fig.01. Later, rectangular window is chosen to do further processing.

**Adding the Interference**

**1. Create a sinusoid whose frequency is 10kHz that is sampled at the same rate as the guitar signal and has the same length. The amplitude of this sinusoid should be one.**

**2. Add this signal to the guitar signal to create the simulated recorded signal that has the interference (call this signal y\_10 to indicate that it has an interference at 10 kHz).**

**3. Look at the new signal in the frequency domain. Comment on it.**

**4. Listen to the new signal.**

[y,Fs]=wavread('guitar1.wav');

n=0:length(y)-1;

y\_10=sin(2\*pi\*10000/Fs\*n);% angular frequency = 2\* pi \* f

figure

subplot(4,1,1);

plot(0:length(y)-1,y\_10);

set(gca,'xlim',[0 0.0001\*length(y)],'ylimmode','auto');

title ('noise of 10KHz');

%---------------------10K Hz sin interference of same length & rate

y\_tot=y\_10.'+y;

subplot(4,1,2);

plot(0:length(y)-1,y\_tot);

title('sample add noise');

%--------------------new signal in time domain

Y\_tot=fft(y\_tot,8192);

subplot(4,1,3);

plot(0:8191,20.\*log10((abs(Y\_tot))));

title('DFT of sample add noise, in dB');

grid on;

ylabel('dB');

subplot(4,1,4);

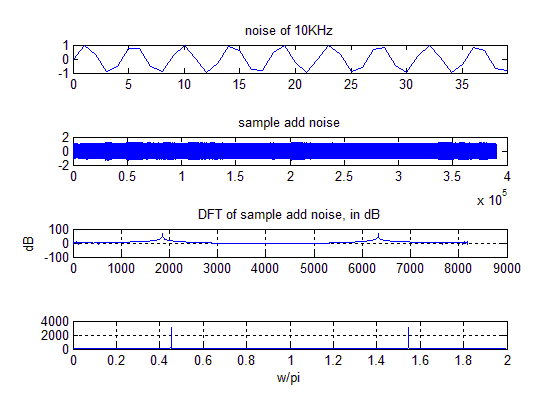
plot((0:8191)/8192\*2,(abs(Y\_tot)));

grid;

xlabel('w/pi');

%----------------------freqency domain

sound(y\_tot,Fs);

****

**Filter Design**

Design the digital filter satisfying the following requirement:

• passband edge frequency: 7 kHz

• stopband edge frequency: 9 kHz

• passband ripple: 1 dB

• minimum stopband attenuation: 40 dB.

**1) Design digital IIR low-pass filters of three types (*Butterworth, Type I Chebyshev, Elliptic*).**

**• Write down the exact expression for the system function of the filter in your report.**

Solution:

For *Butterworth:*

Building the *Butterworth* filter:

%Butterworth filter order and cutoff frequency

Fs=44100;

wp=7000/(Fs/2);

ws=9000/(Fs/2);

rp=1;

rs=40;

format long

[n,Wn]=buttord(wp,ws,rp,rs) %1/2 of sampling frequency

[num,den]=butter(n,Wn)

n =

17

Wn =

0.329436991237154

num =

Columns 1 through 4

0.000000177046223 0.000003009785796 0.000024078286369 0.000120391431844

Columns 5 through 8

0.000421370011454 0.001095562029779 0.002191124059559 0.003443194950735

Columns 9 through 12

0.004303993688419 0.004303993688419 0.003443194950735 0.002191124059559

Columns 13 through 16

0.001095562029779 0.000421370011454 0.000120391431844 0.000024078286369

Columns 17 through 18

0.000003009785796 0.000000177046223

den =

Columns 1 through 4

1.000000000000000 -5.792404676059926 17.575282394679920 -35.799760278837383

Columns 5 through 8

54.022332289886066 -63.455910435183497 59.674849064779366 -45.658899861310601

Columns 9 through 12

28.662604785635907 -14.799938437242101 6.269443046639585 -2.161725624944414

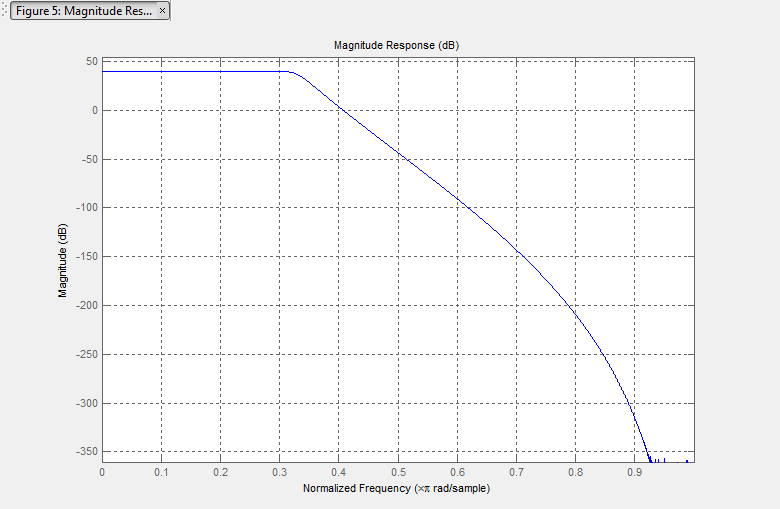
Columns 13 through 16

0.598160983196085 -0.129830902464762 0.021316856935195 -0.002491796735785

Columns 17 through 18

0.000184950364189 -0.000006556757495

**• Compute and plot the filter’s magnitude and phase response. Does your design meet the specifications?**



**• Also show the pole-zero plot for the designed filter in your report.**

**• Compare the three filters and discuss in your report. Matlab function to be used: buttord, cheb1ord, ellipord, butter, cheby1, ellip, etc.**

**• Use the designed filter to filter the interfered with guitar signal and test their effectiveness. (Use matlab function filter). Compare the two signals in frequency domain. Listen to the filtered signal. Comment on your observations.**

2) Design digital FIR filters using two different windows.

• Provide the shape and length of the selected window, explain why?

• Compute and plot the filter’s magnitude and phase response. Does your design

meet the specifications?

• Use the designed filter to filter the interfered with guitar signal and test their

effectiveness. (Use matlab function filter). Compare the two signals in

frequency domain. Listen to the filtered signal. Comment on your

observations.