The University of Texas at Austin
Dept. of Electrical and Computer Engineering
Midterm #1

Date: October 20, 2017 Course: EE 445S Evans

Name: **Chase,** **Annabeth**
Last, First

- The exam is scheduled to last 50 minutes.
- Open books and open notes. You may refer to your homework assignments and the homework solution sets.
- Calculators are allowed.
- You may use any standalone computer system, i.e. one that is not connected to a network. ***Please disable all wireless connections on your computer system(s).***
- Please turn off all cell phones.
- No headphones allowed.
- All work should be performed on the quiz itself. If more space is needed, then use the backs of the pages.
- **Fully justify your answers.** If you decide to quote text from a source, please give the quote, page number and source citation**.**

Problem	*Point Value*	*Your score*	*Topic*
1	28		Filter Analysis
2	24		Mixers
3	24		Filter Design
4	24		Potpourri
Total	100		

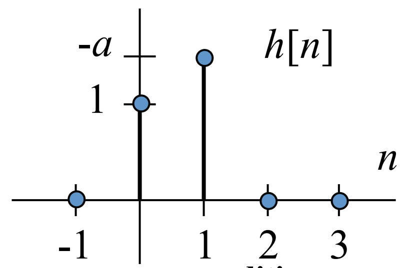
Lectures 3 & 5

HW 0.4 1.1 2.1 2.3 | Lab #3

Problem 1.1 *Filter Analysis*. 28 points.

Consider the following causal finite impulse response (FIR) linear time-invariant (LTI) filter with input $x[n]$ and output $y[n]$ described by

$$y[n] = x[n] - a\, x[n-1]$$



(a) Give a formula for the impulse response $h[n]$. Plot $h[n]$. *3 points*.

Input an impulse signal: $x[n] = \delta[n]$.
Impulse response is $h[n] = \delta[n] - a\, \delta[n-1]$.

(b) What are the initial conditions? What are their values? *3 points*.

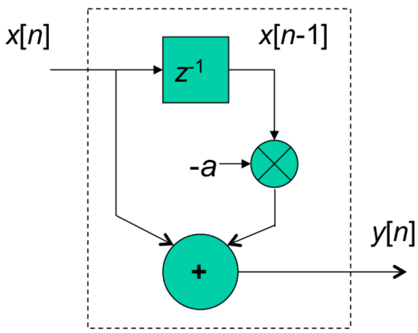
Compute output samples at $n = 0$, 1, etc., to reveal the initial conditions

$y[0] = x[0] - a\, x[-1]$ which gives an initial condition of $x[-1]$.

$y[1] = x[1] - a\, x[0]$ which does not reveal any additional initial conditions.

System must be at rest at $n = 0$ for LTI properties to hold. Hence, $x[-1] = 0$.

(c) Draw the block diagram of the FIR filter relating input $x[n]$ and output $y[n]$. *6 points*.



(d) Give a formula for the discrete-time frequency response of the FIR filter. *4 points*.

With $h[n] = \delta[n] - a\, \delta[n-1]$, $H(z) = 1 - a\, z^{-1}$ for $z \neq 0$. Since the region of convergence $z \neq 0$ includes the unit circle, we can compute $H_{freq}(\omega) = H(e^{j\omega}) = 1 - a e^{-j\omega}$.

(e) Does the FIR filter have linear phase? If yes, then give the conditions on the coefficient a for the filter to have linear phase. If no, then show that the coefficients cannot meet the conditions for linear phase. *6 points*

In order for an FIR filter to have linear phase, its impulse response would need to be either even or odd symmetric about the midpoint. This would mean $a = -1$ or $a = 1$, respectively. Although not expected as an answer, $a = 0$ would give a zero phase response.

(f) If parameter a were real-valued, what are all of the possible frequency selectivities that the FIR filter could provide: lowpass, highpass, bandpass, bandstop, allpass, notch? *6 points*

Transfer function $H(z) = 1 - a\, z^{-1}$ has a zero at $z = a$. Location of zero indicates stopband.

We can rewrite $H(z) = \frac{z-a}{z}$, which has an artificial pole at $z = 0$.

Allpass response when $a \approx 0$. Highpass response $a > 0.5$. Lowpass response $a < -0.5$.

Note: Lowpass averaging filter when $a = -1$. Highpass first-order difference when $a = 1$.

Lectures 1 4 5 & 6

HW 0.2 0.3 1.3 2.2 2.3 3.1 3.3 | Labs #2 and #3

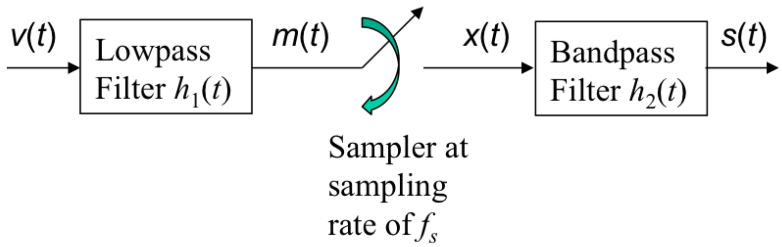
Problem 1.2 *Mixers.* 24 points.

Mixing provides an efficient implementation in analog continuous-time circuits for sinusoidal amplitude modulation of the form

$$s(t) = m(t)\cos(2\,\pi\, f_c\, t)$$

where $m(t)$ is the baseband message signal with bandwidth W, and

f_c is the carrier frequency such that $f_c > W$



We discussed mixers for about 30 minutes during the Q&A session for homework #3 on Friday, Oct. 13th.

(a) Give the passband and stopband frequencies for the lowpass filter. *3 points.*

$f_{pass} = W$.

f_{stop} could be *1.1 W* or ½ f_s. Using 1.1W would give 10% rolloff from passband to stopband.

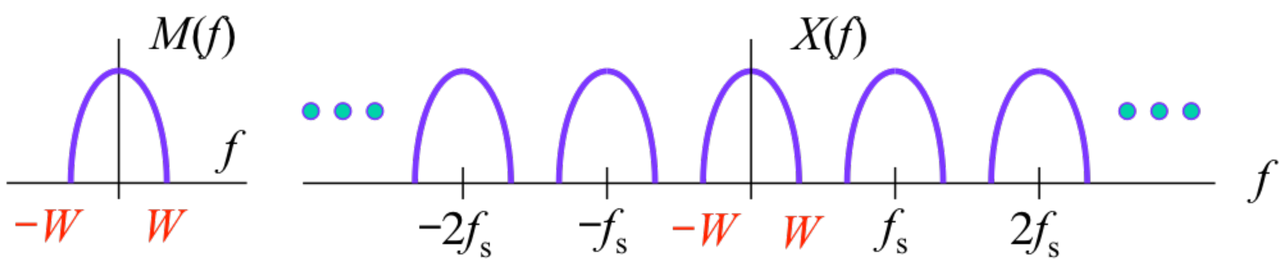
(b) Give the passband and stopband frequencies for the bandpass filter. *3 points*

$f_{pass1} = f_c - W$ and $f_{pass2} = f_c + W$. Width of passband is 2W. 10% of that would be 0.2W.

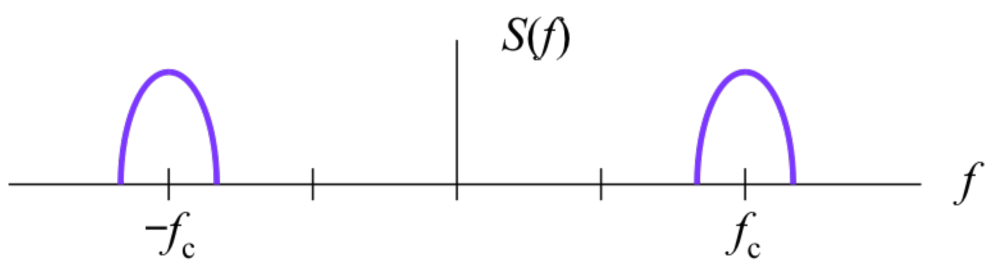
$f_{stop1} = f_{pass1} - 0.2W$ and $f_{stop2} = f_{pass2} + 0.2W$ to give 10% rolloff from passband to stopband.

(c) Draw the spectrum for $m(t)$, $x(t)$, and $s(t)$. You do not need to draw the spectrum for $v(t)$. *9 points.*

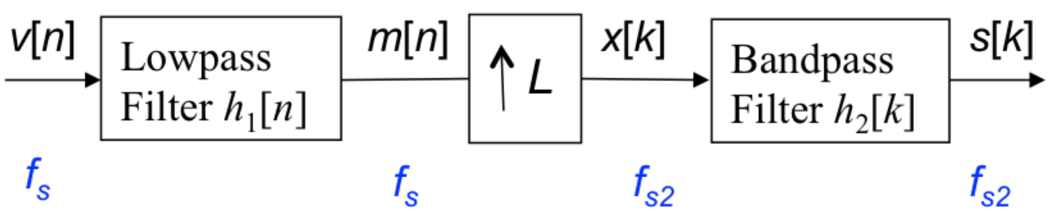
Baseband signal $m(t)$ has bandwidth of W; $X(f)$ will have replicas of $M(f)$ due to sampling.



For a replica of $M(f)$ to be centered at f_c, $f_c = l\, f_s$ where l is an integer. The width of each band of $X(f)$ and $S(f)$ is 2W.



(d) In order to simulate the mixer in discrete-time, e.g. in MATLAB, we use discrete-time filters for the lowpass and highpass filters and replace the sampling block with an upsampling block.

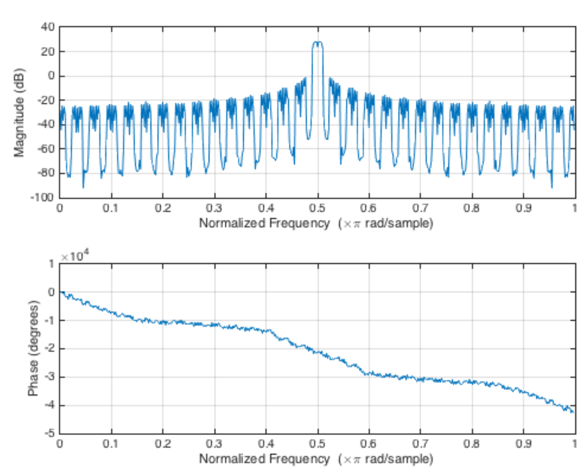


i. Give the constraints on the sampling rate to convert the mixer to discrete time. *6 points.*

In addition to f_s for the sampler in the mixer, we now have a second sampling rate f_{s2} for converting the mixer to discrete time. The highest frequency of interest in the mixer is $f_c + W$. Hence, $f_{s2} > 2\ (f_c + W)$. The sampling rate f_s will be used for signals $v(t)$ and $m(t)$ as well as the lowpass filter. The upsampler by L will convert an input sampling rate of f_s to an output sampling rate of f_{s2} where $L = f_{s2} / f_s$. Hence, $f_{s2} = L\, f_{s2}$.

ii. Determine the upsampling factor. *3 points.* $L = f_{s2} / f_s$

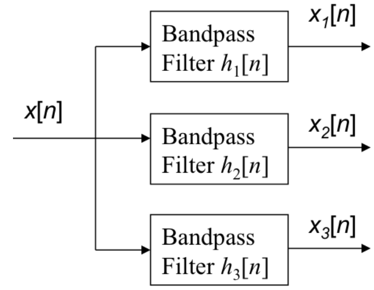
```
% MATLAB code for midterm #1
% problem 1.2(d) for Fall 2017
%
% Programmer: Prof. Brian L. Evans
% The University of Texas at Austin
% Date: October 23, 2017
%
% Mixer
% s(t) = m(t) cos(2 pi fc t)
%
% v(t)     m(t)          x(t)     s(t)
% ---> LPF ---> Sampler ---> BPF --->
%
% System Parameters
% In Continuous Time
W  =  300;     % Message bandwidth Hz
fs = 1000;     % Sampling rate Hz
k  =   15;     % kth replica at fc
fc = k*fs;     % Carrier frequency Hz
% In Discrete Time
L =   4*k;     % Upsampling factor
fs2 = L*fs;    % For x(t) and s(t)
% fs sampling rate for v(t) and m(t)
% fs2 > 2 (fc + W) and fs2 = L fs
% Since fc > W, fs2 >= 4 fc

% Time reference for v(t) and m(t)
Ts = 1/fs;
Tmax = 1;
t1 = Ts : Ts : Tmax;
N1 = length(t1);
n1 = 1 : N1;

% Generate v(t) as chirp to have all
% discrete-time frequencies by
% increasing instantaneous frequency
% from 0 to pi.
fstep = (fs/2) / Tmax;
wstep = pi*fstep*(Ts^2);
v = cos(wstep*n1.^2);

% Lowpass filter
% Truncated sinc pulse
fpass = W;
npass = -100 : 100;
hlpf = sinc(2*(fpass/fs)*npass);
hlpf = hlpf / sum(abs(hlpf).^2);

% Message signal
m = filter(hlpf, 1, v);

% Upsample by L
N2 = L*N1;
x = zeros(1, N2);
x(1:L:N2) = m;

% Bandpass filter centered with
% passband fc - W to fc + W
% 1. Design lowpass prototype
% 2. Modulate by cos(2 pi fc t)
fpass = W;
npass = -4*L : 4*L;
hprot = sinc(2*(fpass/fs2)*npass);
wc = 2*pi*fc/fs2;
hbpf = hprot .* cos(wc*npass);
hbpf = hbpf / sum(abs(hbpf).^2);

s = filter(hbpf, 1, x);
freqz(s);
```



Lectures 1 4 5 & 6

HW 2.1 2.2 2.3 3.1 3.2 3.3 | Lab #3

Problem 1.3 *Filter Design.* 24 points.



Some audio systems split an audio signal in three frequency bands for playback over sub-wolfer, wolfer, and tweeter speaker elements.

The block diagram on the right performs the split in discrete time:

$x_1[n]$ contains sub-wolfer frequencies 20-200 Hz.

$X_2[n]$ contains wolfer frequencies 200-2,000 Hz.

$X_3[n]$ contains tweeter frequencies 2,000-20,000 Hz.

Assume that the sampling rate is 48000 Hz.

Each bandpass filter should have group delay of less than 10 ms.

(a) Give passband ripple and stopband attenuation values for these filters. *6 points.*

A_{pass} = 1 dB and A_{stop} = 80 dB based on homework problems 2.3 and 3.3 on filter design for audio applications. (We'll explore these settings in lecture 8 on quantization. For example, in an A/D converter with B bits, the stopband attenuation would be $6B + 2$ dB.)

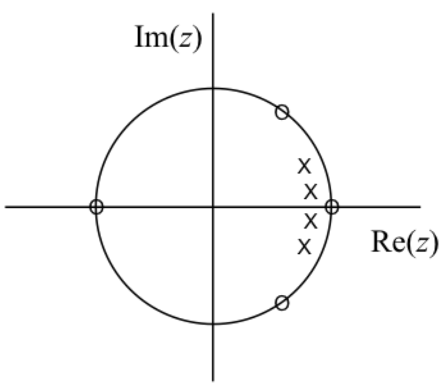
(b) Give passband and stopband <u>discrete-time</u> frequencies to design the bandpass filter $h_2[n]$. *6 points.*

Seek 10% rolloff from stopband1 to passband1, and from passband2 to stopband2. Convert continuous-time frequency f_0 in Hz to discrete-time frequency $\omega_0 = 2\pi f_0 / f_s$.

<u>Answer #1</u>: f_{stop1} = 20 Hz, f_{pass1} = 200 Hz, f_{pass2} = 2000 Hz, f_{stop2} = 2180 Hz. This would give $\omega_{stop1} = 0.000833\pi$, $\omega_{pass1} = 0.00833\pi$, $\omega_{pass2} = 0.0833\pi$, $\omega_{stop2} = 0.0908\pi$, all in rad/sample.

<u>Answer #2</u>: $f_{cutoff1}$ = 200 Hz and $f_{cutoff2}$ = 2000 Hz, which could mean f_{stop1} = 110 Hz, f_{pass1} = 290 Hz, f_{pass2} = 1910 Hz, f_{stop2} = 2090 Hz to give slightly more than 10% rolloffs. This would give $\omega_{stop1} = 0.00458\pi$, $\omega_{pass1} = 0.0121\pi$, $\omega_{pass2} = 0.0796\pi$, $\omega_{stop2} = 0.0871\pi$, all in rad/sample.

(c) Draw the pole-zero diagram for a fourth-order infinite impulse response (IIR) bandpass filter $h_2[n]$. *6 points.*



**Poles near the unit circle indicate the passband(s).
Zeros on/near the unit circle indicate stopband(s).
Poles occur in conjugate symmetric pairs or are real-valued; same goes for zeros.
Try to keep zeros and poles away from eachother.
Many possible answers.
<u>Answer</u>: Four poles at passband frequencies and their negatives. $p_0 = r \exp(j\,\omega_{pass1})$; $p_1 = r \exp(-j\,\omega_{pass1})$; $p_2 = r \exp(j\,\omega_{pass2})$; $p_3 = r \exp(-j\,\omega_{pass2})$. Use $r = 0.9$.
Put zero at $z = 1$ to enforce stopband1, and other three zeros to enforce stopband2. SEE MATLAB CODE BELOW.**

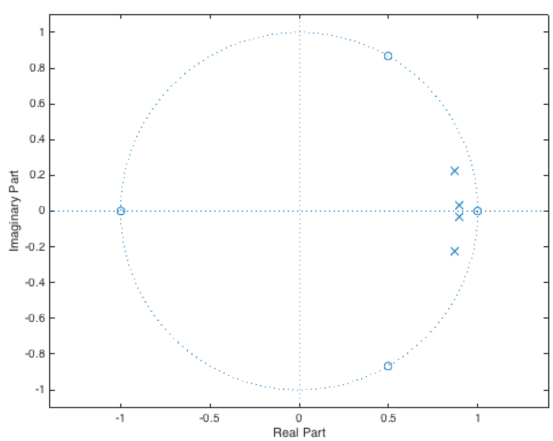
(d) For a linear phase finite impulse response (FIR) filter, indicate the maximum length that would still meet the group delay constraint. The maximum length would apply to all three filters. *6 points.*

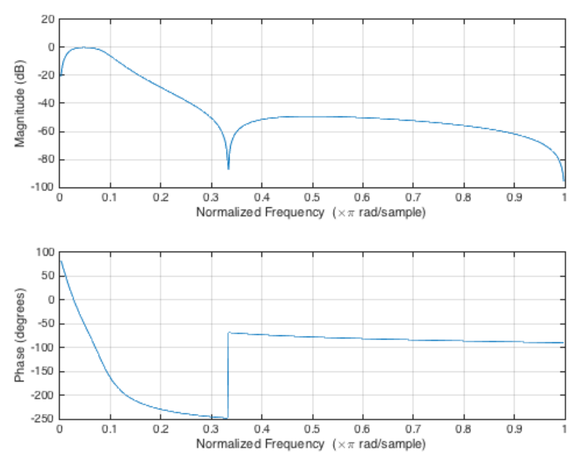
**A group delay of 10ms means (0.01s)(48000 samples/s) = 480 samples.
A linear phase FIR filter with N coefficients has a group delay of $(N-1)/2$ samples.**

So, $N = 961$ samples. (For lower group delay, use discrete-time IIR filters.)

MATLAB Code and Plots for 1.3(c)

```
% Four poles
wpass1 = 0.0121*pi;
wpass2 = 0.0796*pi;
r = 0.9;
p0 = r * exp(j*wpass1);
p1 = r * exp(-j*wpass1);
p2 = r * exp(j*wpass2);
p3 = r * exp(-j*wpass2);
denom1 = [1 –(p0+p1) p0*p1];
denom2 = [1 –(p2+p3) p2*p3];
denom = conv(denom1, denom2);

% Four zeros
zeroAngle = pi/3;
z0 = exp(j*zeroAngle);
z1 = exp(-j*zeroAngle);
numer1 = [1 –(z0+z1) z0*z1];
z2 = 1;      %% at 0 rad/sample
z3 = -1;     %% at pi rad/sample
numer2 = [1 –(z2+z3) z2*z3];
numer = conv(numer1, numer2);

%%% Normalize response in middle
%%% of the passband
w0 = (wpass1 + wpass2)/ 2;
Hresp = freqz(numer, denom, [w0 w0]);
C = 1 / abs(Hresp(1));

%%% Plot pole-zero diagram
figure; zplane(C*numer, denom);

%%% Plot frequency response
figure; freqz(C*numer, denom);
```





Group delay is ~15 samples in the passband.

Lectures 1 3 4 5 & 6
HW 2.1 2.3 3.1 3.2 3.3	Lab #3

Problem 1.4. *Potpourri.* 24 points.

(a) Consider a linear time-invariant (LTI) system that has bounded-input bounded-output stability. To measure its frequency response, one could input a discrete-time unit impulse $\delta[n]$ for $-\infty < n < \infty$, find the output signal $h[n]$, and take the discrete-time Fourier transform of $h[n]$. In practice, we cannot go back to $n = -\infty$ or wait until $n = \infty$. Give a practical method using a finite-length discrete-time input signal to estimate the frequency response of the LTI system. *12 points.*

For an unknown BIBO stable LTI system, we seek to find its frequency response $H_{freq}(\omega)$.

In the frequency domain, $Y_{freq}(\omega) = H_{freq}(\omega)\, X_{freq}(\omega)$.

We seek to compute $H_{freq}(\omega) = Y_{freq}(\omega) \,/\, X_{freq}(\omega)$.

Since the discrete-time frequency domain is periodic with period 2π, we need a finite-length signal $x[n]$ whose frequency response $X_{freq}(\omega)$ has all frequencies from -π to π in it and does not equal zero at any frequency value so as to avoid a division by zero error.

A two-sided discrete-time cosine signal $\cos(\omega_0\, n)$ has frequency components at ω_0 and $-\omega_0$.

Use a chirp signal that linearly sweeps all frequencies from 0 to π.

(b) Consider the following method to compute a cosine value by using a Taylor series at $\theta = 0$:

$$\cos(\theta) = \sum_{n=0}^{\infty} \frac{(-1)^n}{(2n)!} \theta^{2n} = 1 - \frac{1}{2}\theta^2 + \frac{1}{24}\theta^4 - \ldots$$

Suppose that 10 non-zero terms were kept in the series expansion (i.e. $n = 0, 1, 2, \ldots 9$).

i. How would you minimize the number of multiplications? *6 points.*

Factor the polynomial into Horner's form (*lecture slide 1-12*) to minimize the number of multiplications:

$$a_{18}\, x^{18} + a_{16}\, x^{16} + a_{14}\, x^{14} + \ldots + a_0 = (\ \ldots\ (((a_{18}\, x^2 + a_{16})\, x^2 + a_{14})\, x^2\ \ldots\ )\, x^2 + a_0$$

Number of multiplications reduces from 90 to 9.

Number of additions remains the same at 9.

ii. Please complete the last row of entries for the new method. *6 points.*

Method	Multiplication-Add Operations	ROM (words)	RAM (words)	Quality in floating point
C math library call	30	22	1	Second best
Difference equation	2	2	3	Worst
Lookup table	0	L	0	Best
Taylor series	**9**	**10**	**1**	**Second best**

L is the smallest discrete-time period for the cosine signal.