School of Engineering

ELC321-02 **Spring 2020**

Assignment 5

Date: April 30th, 2020

Submitted by: Jeffrey Blanda

Instructor: John MacDonald

Problem 1.

A digital pre-emphasis filter is often used to adjust the frequency spectrum of a signal prior to transmission or storage in a continuous time format. One such example is shown in the block diagram below.

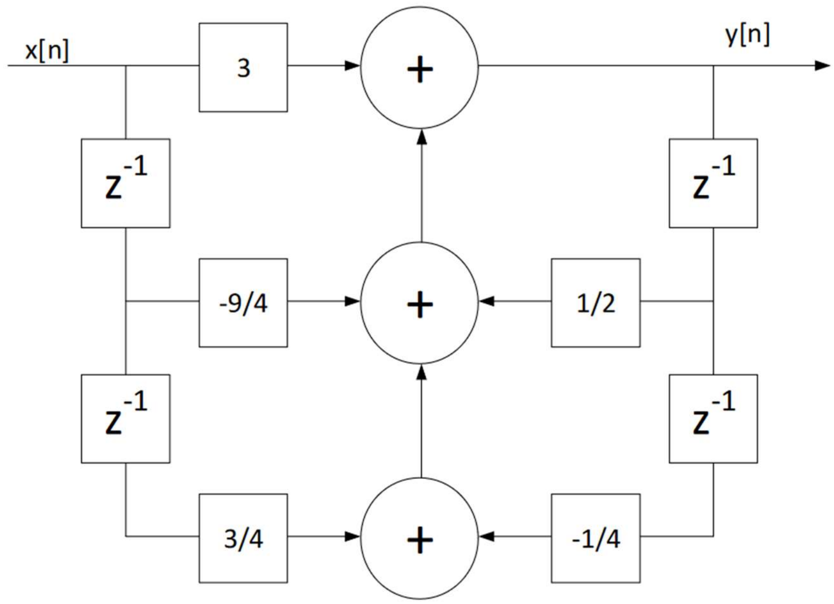

Figure 1. Provided Block Diagram

a) Determine the Difference Equation for the system
b) Determine the Transfer Function, H(z).
c) Determine the filter Step Response, s[n], when the input x[n] = u[n] by multiplying in the z-domain then finding the inverse z-transform.
d) Verify s[n] by direct numerical evaluation of the Difference Equation from n = 0 to steady-state and compare the sequences from c and d.
e) Plot the Frequency Response of the filter, both Magnitude and Phase, assuming a sampling frequency of 40kHz.

a) Determine the difference Equation for the system: The Difference equation is
   a. $y[n] = 3 * x[n] + \frac{1}{2} * y[n-1] - \frac{9}{4} * x[n-1] + \frac{3}{4} * x[n-2] - \frac{1}{4} * y[n-2]$ in the n domain or
   b. $Y(z) = 3X(z) + \frac{1}{2}Y(z)z^{-1} - \frac{9}{4}X(z)z^{-1} + \frac{3}{4}X(z)z^{-2} - \frac{1}{4}Y(z)z^{-2}$ in the z domain
b) To determine the transfer function, we need to use the difference equation in the z domain.

a. $Y(z) = 3X(z) + \frac{1}{2}Y(z)z^{-1} - \frac{9}{4}X(z)z^{-1} + \frac{3}{4}X(z)z^{-2} - \frac{1}{4}Y(z)z^{-2}$
b. $Y(z) - \frac{1}{2}Y(z)z^{-1} + \frac{1}{4}Y(z)z^{-2} = 3X(z) - \frac{9}{4}X(z)z^{-1} + \frac{3}{4}X(z)z^{-2}$
c. $Y(z)[1 - \frac{1}{2z} + \frac{1}{4z^2}] = X(z)[3 - \frac{9}{4z} + \frac{3}{4z^2}]$
d. $H(z) = \frac{3-\frac{9}{4z}+\frac{3}{4z^2}}{1-\frac{1}{2z}+\frac{1}{4z^2}} = \frac{12z^2-9z+3}{4z^2-2z+1}$

c) Finding the step response (where x[n]=u[n]) in the z domain, first we need to find the conversion for u[n] to the z domain.

a. $X(z) = \mathcal{Z}\{x[n]\} = \mathcal{Z}\{u[n]\} = \frac{z}{z-1}$
b. Given the transfer function found in the previous step,
c. $H(z) = \frac{Y(z)}{X(z)} = \frac{12z^2-9z+3}{4z^2-2z+1}$
d. Substituting for X(z)
e. $Y(z) = \frac{12z^2-9z+3}{4z^2-2z+1}\left(\frac{z}{z-1}\right)$
f. $Y(z) = \frac{12z^3-9z^2+3z}{4z^3-4z^2-2z^2+2z+z-1}$
g. $Y(z) = \frac{12z^3-9z^2+3z}{4z^3-6z^2+3z-1}$
h. $\frac{Y(z)}{z} = \frac{12z^2-9z+3}{4z^3-6z^2+3z-1}$
i. $\frac{Y(z)}{z} = \frac{2}{z-1} + \frac{0.5}{z-(0.25+0.433i)} + \frac{0.5}{z-(0.25-0.433i)}$
j. $Y(z) = \frac{2z}{z-1} + \frac{0.5z}{z-(0.25+0.433i)} + \frac{0.5z}{z-(0.25-0.433i)}$
k. $Y(z) = \frac{2z}{z-1} + \frac{0.5z}{z-(0.25+0.433i)} + \frac{0.5z}{z-(0.25-0.433i)}$
l. The residue and poles for the second and third functions:,

$$r_1 = 0.5, \qquad p_1 = 0.25 + 0.433i$$
$$r_2 = 0.5, \qquad p_2 = 0.25 - 0.433i$$

m. Our poles can be rewritten as:
$e^{\frac{j\pi}{4}} and\ e^{-\frac{j\pi}{4}}$
n. $Y(z) = \frac{z(0.125+0.2165i)}{z-e^{\frac{j\pi}{4}}} + \frac{z(0.125-0.2165\ )}{z-e^{-\frac{j\pi}{4}}}$
o. Using the Identity $\left(Ap + A^*(p^*)\right)^n = \frac{Az}{z-p} + \frac{A^*z}{z-p^*}$,
p. $y[n] = 2u[n] + 0.5^n\left(e^{\frac{jn\pi}{4}} + e^{-\frac{jn\pi}{4}}\right)u[n]$
q. $y[n] = 2u[n] + 0.5^n \cos\left(\frac{n\pi}{4}\right)u[n]$

d) With the difference equation previously found:

$$y[n] = 3 * x[n] + \frac{1}{2} * y[n-1] - \frac{9}{4} * x[n-1] + \frac{3}{4} * x[n-2] - \frac{1}{4} * y[n-2]$$

evaluating for all values of n until steady state response:

Since we're using the step function, x[n]=1 when n>=0

$$y[0] = 3 * x[0] = 3$$

$$y[1] = 3 + \frac{1}{2}(3) - \frac{9}{4} = \frac{9}{4} = 2.25$$

$$y[2] = 3 + \frac{1}{2}\left(\frac{9}{4}\right) - \frac{9}{4} + \frac{3}{4} - \frac{1}{4}(3) = \frac{15}{8} = 1.875$$

$$y[3] = 3 + \frac{1}{2}\left(\frac{15}{8}\right) - \frac{9}{4} + \frac{3}{4} - \frac{1}{4}\left(\frac{9}{4}\right) = \frac{15}{8} = 1.875$$

$$y[4] = 3 + \frac{1}{2}\left(\frac{15}{8}\right) - \frac{9}{4} + \frac{3}{4} - \frac{1}{4}\left(\frac{15}{8}\right) = \frac{63}{32} = 1.968$$

$$y[5] = 3 + \frac{1}{2}\left(\frac{63}{32}\right) - \frac{9}{4} + \frac{3}{4} - \frac{1}{4}\left(\frac{15}{8}\right) = \frac{129}{64} = 2.016$$

$$y[6] = 3 + \frac{1}{2}\left(\frac{129}{64}\right) - \frac{9}{4} + \frac{3}{4} - \frac{1}{4}\left(\frac{63}{32}\right) = \frac{129}{64} = 2.016$$

The function will approach 2 at steady state.

e) To plot the frequency response with a sampling frequency of 40kHz using MatLab, we can combine the second and third equations and simplify to eliminate the imaginary component which results in $\frac{z^2-0.25}{z^2-0.5z-0.25}$. Then using the code supplied in the lecture notes for graphing response magnitude and phase:

```
Samplfreq = 40000;
% Plots the frequency response of H(z)using parametric evaluation
T=1/Samplfreq;
% sample interval
omega=0:0.001:pi;
% wT goes from 0 to pi
w=omega./T;
% radian frequency (x-axis)
z=exp(1j.*omega);
% z goes along the unit circle
g=(2.*z)/(z-1)+(((z.^2)-0.25)./((z.^2)-(0.5.*z)+(0.25)));
% This is the function H(z)
subplot(2,1,1);
semilogx(w,20*log10(abs(g)));
%Mag in dB
grid;
title('H(z) Magnitude (dB)');
subplot(2,1,2);
semilogx(w,angle(g).*180./pi);
% Phase in degrees
grid;
title ('H(z) Phase (deg)');
```

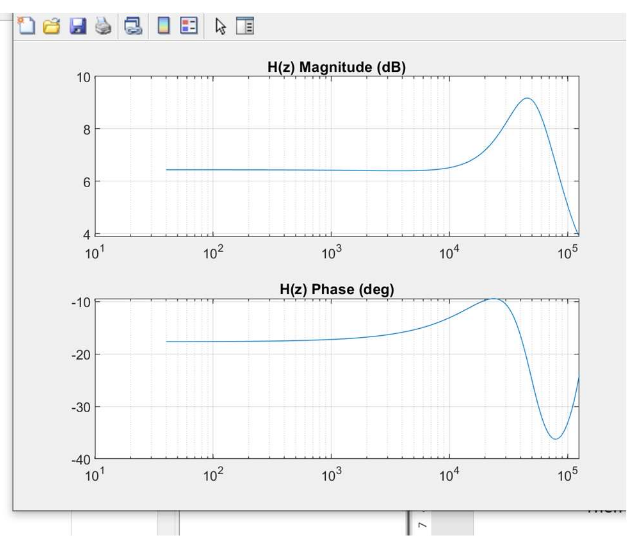

Figure 2. MatLab code and plot for frequency response magnitude and phase.

Problem 2.

Given x[n] = [1 1 1 1 0 1 1 1]

Find the 8-point FFT sequence, X[N].

To find the 8 Point Fast Fourier Transform sequence, we need to solve the summation

$$X_k \sum_{n=0}^{7} x_n W_8^{nk} \quad where\ W_N = e^{-j2\pi/N}$$

$$X_k = x_0 + x_1 W_8^k + x_2 W_8^{2k} + x_3 W_8^{3k} + x_4 W_8^{4k} + x_5 W_8^{5k} + x_6 W_8^{6k} + x_7 W_8^{7k}$$

Then using Euler's identity to solve for W values

$$W_8^k = \left[\frac{\sqrt{2}}{2}(1-j)\right]^k,\ W_8^{2k} = (-j)^k,\ W_8^{3k} = \left[\frac{\sqrt{2}}{2}(-1-j)\right]^k, W_8^{4k} = (-1)^k,$$

$$W_8^{5k} = \left[\frac{\sqrt{2}}{2}(-1+j)\right]^k, W_8^{6k} = j^k, W_8^{7k} = \left[\frac{\sqrt{2}}{2}(1+j)\right]^k$$

A general formula for the 8-point FFT sequence can be formed:

$$X_k = x_0 + x_1\left[\frac{\sqrt{2}}{2}(1-j)\right]^k + x_2(-j)^k + x_3\left[\frac{\sqrt{2}}{2}(-1-j)\right]^k + x_4(-1)^k + x_5\left[\frac{\sqrt{2}}{2}(-1+j)\right]^k + x_6 j^k + x_7\left[\frac{\sqrt{2}}{2}(1+j)\right]^k$$

Simplifying the equation:

$$X_k = \left[\frac{\sqrt{2}}{2}\right]^k [x_1(1-j)^k + x_3(-1-j)^k + x_5(-1+j)^k + x_7(1+j)^k] + x_0 + x_2(-j)^k + x_4(-1)^k + x_6 j^k$$

Solving for each index:

$$X_0 = [x_1 + x_3 + x_5 + x_7] + x_0 + x_2 + x_4 + x_6 = 7$$

$$X_1 = \left[\frac{\sqrt{2}}{2}\right][x_1 - jx_1 - x_3 - jx_3 - x_5 + jx_5 + x_7 + jx_7] + x_0 - jx_2 - x_4 + jx_6$$

$$= \left[\frac{\sqrt{2}}{2}\right][1 - j - 1 - j - 1 + j + 1 + j] + 1 - j - 0 + j$$

$$X_1 = \left[\frac{\sqrt{2}}{2}\right][0] + 1 = 1$$

$$X_2 = \left[\frac{2}{4}\right][x_1(1-2j-1) + x_3(1+2j-1) + x_5(1-2j-1) + x_7(1+2j-1)] + x_0 - x_2 + x_4 - x_6$$

$$X_2 = \left[\frac{1}{2}\right][1 - 2j - 1 + 1 + 2j - 1 + 1 - 2j - 1 + 1 + 2j - 1] + 1 - 1 + 0 - 1$$

$$X_2 = \left[\frac{1}{2}\right][0] - 1 = -1$$

To continue the calculations by hand is possible, however with polynomials with raising exponents the work would inevitably take many pages. Instead this formula can be plugged into MatLab and computed using a for loop.

```
x = [1,1,1,1,0,1,1,1];
for i=0:7
    y(i+1)=((sqrt(2)/2)^i)*((1-1j)^i+(-1-1j)^i+(-1+1j)^i+(1+1j)^i)+1+(-1j)^i+1j^i;
end
y
```

Figure 3. MatLab code to solve the Fast Fourier Transform using a for loop and the derived equation

```
y =

    7.0000    1.0000   -1.0000    1.0000   -1.0000    1.0000   -1.0000    1.0000

fx >>
```

Figure 4. Output of the code from Figure 2.

```
Command Window
>> x = [1,1,1,1,0,1,1,1];
>> y = fft(x)

y =

     7     1    -1     1    -1     1    -1     1

fx >>
```

Figure 5. Checking the results of the FFT with MatLab's built in FFT function.

Problem 3.

12.24. The signal $x(t) = 5\cos(8\pi t)$ is sampled eight times starting at $t=0$ with a sampling period $T=0.1$s.

1) Compute the DFT of this sequence.

2) Use MATLAB to confirm the results of Part a.

3. Determine the Fourier transform of x(t) and compare it with the results of Parts a and b. Explain the differences.

1) With a sampling period of T=0.1 seconds and a starting time of t=0s, we can sample the values using t= 0, 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, and 0.7
   a. Using the equation found in Problem 2, we can write two for loops in MatLab.
      i. Sample the signal 8 times using the time values given
      ii. Compute the fast Fourier transform using the equation found in problem 2.

```
for j=0:7
x(j+1) = 5*cos(8*pi*(0.1*j));
end
for i=0:7
    y(i+1)=((sqrt(2)/2)^i)*(x(2)*(1-1j)^i+x(4)*(-1-1j)^i+x(6)*(-1+1j)^i+x(8)*(1+1j)^i)
end
y
```

Figure 6. MatLab code for sampling the signal and finding the fast Fourier transform.

```
y =

  Columns 1 through 4

   2.5000 + 0.0000i   2.6492 + 0.8057i   3.4549 + 2.1353i  15.4409 +11.9860i

  Columns 5 through 8

  -5.5902 + 0.0000i  15.4409 -11.9860i   3.4549 - 2.1353i   2.6492 - 0.8057i
```

Figure 7. Output for the fast Fourier transform.

2) Using the MatLab FFT function to validate our result

```
for j=0:7
x(j+1) = 5*cos(8*pi*(0.1*j));
end
y = fft(x);
y
```

Figure 8. MatLab code using the fft function.

```
y =

  Columns 1 through 4

   2.5000 + 0.0000i   2.6492 + 0.8057i   3.4549 + 2.1353i  15.4409 +11.9860i

  Columns 5 through 8

  -5.5902 + 0.0000i  15.4409 -11.9860i   3.4549 - 2.1353i   2.6492 - 0.8057i
```

Figure 9. Output using the FFT function

3) Using Table 5.2 from the text,
    a. $\mathcal{F}\{\cos(\omega_0 t)\} = \pi[\delta(\omega - \omega_0) + \delta(\omega + \omega_0)]$
    b. $5\mathcal{F}\{\cos(8\pi t)\} = 5\pi[\delta(\omega - 8\pi) + \delta(\omega + 8\pi)]$
    c. As parts 1 and 2 yielded the same results, comparisons between the Fourier transform and 1 will be the same as the Fourier transform and 2, The actual Fourier transform has impulses at ±8π while the Fast Fourier transform has a train of impulses at different frequencies from 1radian to 8radians. This is due to the fact the Fast Fourier Transform of the cosine function is sampled at less than a full period, so the Fast Fourier Transform is not taking the transform of the cosine function, rather it's taking the transform of the repeated sampled signal.

Problem 4.

Text Problem 12.26:

Repeat problems 12.24 a and b after multiplying the sequence x[n] by an eight-point Hanning window. Discuss the differences between this DFT and that found in problem 12.24.

The Hanning window is defined as:

$$w[n] = 0.50 - 0.50\cos\left(\frac{2\pi n}{N-1}\right), 0 \le n \le N-1$$

So for an 8-point Hanning window, N=8

$$w[n] = 0.50 - 0.50\cos\left(\frac{2\pi n}{7}\right), 0 \le n \le 7$$

Knowing that x(t)=5cos(8πt) and we can sample from points t=0, 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, and 0.7 a MatLab script can be used to find our x[n]w[n]

```
for j=0:7
x(j+1) = 5*cos(8*pi*(0.1*j));
end
for k=0:7
w(k+1) = 0.5-0.5*cos(2*pi*k/7);
end
y = x.*w;
```

Figure 10. MatLab code to find x[n]w[n]

```
>> y

y =

         0   -0.7615    0.9444    1.4686   -3.8448    3.0563   -0.7615         0

>>
```

Figure 11. MatLab output for x[n]w[n]

Adding our fast Fourier transform equation to the script,

```
for j=0:7
x(j+1) = 5*cos(8*pi*(0.1*j));
end
for k=0:7
w(k+1) = 0.5-0.5*cos(2*pi*k/7);
end
y = x.*w;
z = fft(y);
for k=0:7
    E(k+1) = y(1)+y(3)*(-1i)^k+y(5)*(-1)^k+y(7)*(j^k);
   O(k+1) = ((sqrt(2)/2)^k)*(y(2)*(1-1i)^k+y(4)*(-1-1i)^k+y(6)*(-1+1i)^k+y(8)*(1+j)^k
end
FFT = E + O;
```

Figure 12. MatLab code to find the FFT of the signal multiplied by the Hanning window

```
FFT =

   1.0e+05 *

  Columns 1 through 4

   0.0000 + 0.0000i  -0.0001 + 0.0000i  -0.0004 - 0.0000i  -0.0025 + 0.0000i

  Columns 5 through 8

  -0.0184 + 0.0000i  -0.1279 - 0.0000i  -0.8960 + 0.0000i  -6.2713 - 0.0000i
```

Figure 13. MatLab output of Fig. 11

The reason the results of the transform are different is due to the way the transform works in assuming the input is one period of a continuous signal, so the results of problem 3 includes spectral leakage while using the Hanning window effectively eliminates spectral leakage.

Using the built-in MatLab FFT function will yield the same results.

```
z =

  Columns 1 through 4

   0.1015 + 0.0000i   0.1067 - 0.0448i  -4.0277 - 0.8262i   7.5828 + 3.3671i

  Columns 5 through 8

  -7.4252 + 0.0000i   7.5828 - 3.3671i  -4.0277 + 0.8262i   0.1067 + 0.0448i
```

Figure 14. MatLab output using the built-in fft function.

Problem 5.

12.29. The DFT of the analog signal f(t) = 12cos(120πt)cos(20πt) is to be computed.

a. What is the minimum sampling frequency to avoid aliasing?

b. If a sampling frequency of $\omega_s = 300\pi$ rad/s is used, how many samples must be taken to give a frequency resolution of 1 rad/s?

a. To avoid aliasing, we must sample higher than $2f_{max}$

$$f_s > 2f_{max}$$

Since our function has a frequency of 120π we must sample higher than 240π

b. For a frequency resolution of 1 rad/s
Using the general formula:

$$\Delta f = \frac{f_s}{N}$$

for a resolution of 1 rad/s $\Delta f = 2\pi$ and given $\omega_s = 300\pi \frac{rad}{s},\ f_s = 150\ Hz$

$$2\pi = \frac{150}{N}, N = \frac{150}{2\pi} = \frac{75}{\pi} = 23.87$$

You would need to take 24 samples.