# DIGITAL SIGNAL PROCESSING LAB(BECE301P)

Due Date: 20-1-25 Lab Slot:L35+L36

Name: ISHAAN KEDAR Reg.No:22BEC0153

**Objective:**

To use MATLAB to obtain the output of the LTI systems that are represented using Linear Constant Coefficient Difference Equation (LCCDE).

**NOTES: Solution of an LCCDE Using MATLAB**

The MATLAB built-in function filter can be used to compute the response of LTI systems that are represented using LCCDE. The filter function with the following syntax.

y = filter(b, a, x)

Which filters the input data x using a rational transfer function defined by the numerator and denominator coefficients b and a. It should be noted that the output signal generated from filter function has same length as the input signal.

The MATLAB built-in function conv can also be used to compute the response of LTI systems with impulse response h as follow,

y = conv(x, h)

The length of the output signal using the above regular convolution will be as follow

length(x) + length(h) - 1

**TASK PROBLEMS:**

***1.*** Consider an LTI system whose input $[n]$ and output $y[n]$ are related by the following LCCDE,

$$\boldsymbol{y[n] = \frac{1}{2} y[n-1] + x[n]}$$

a) Determine the impulse response $h$[n] and then determine $y[n]$ for

$$x[n] = u[n] - u[n-15].$$

b) Let $0 \le n \le 50$. Use MATLAB **filter** command to find the output y$[n]$ from part (a). **Note**: plot the input signal and the output signal in the same figure for each part. Adjust the title, xlabel and ylabel accordingly.

**Code:**

**a)**

To find $h[n]$

I/p $x[n] = \delta[n]$

$h[n] = \frac{1}{2} h[n-1] + \delta[n]$

for $n = 0$

$h[0] = \frac{1}{2} h[-1] + 1$

$h(0) = 1$ $\quad$ $(h[-1] = 0$ & $\delta[0] = 1)$

$n = 1$

$h(1) = \frac{1}{2} h(0) + \delta(1) = \frac{1}{2}(1) + 0 = \frac{1}{2}$

$n=2$

$h(2) = \frac{1}{2} h(1) + \delta(2) = \frac{1}{2}(\frac{1}{2}) + 0 = \frac{1}{4} = (\frac{1}{2})^2$

$\vdots$

$\Rightarrow h[n] = (\frac{1}{2})^n u[n]$

$y(n) = x[n] * h[n] = \sum_{k=-\infty}^{\infty} x(k) h(n-k)$

$x[k] = 1$ for $0 \le k \le 15$ else 0

$n < 15$

$y(n) = (\frac{1}{2})^n \sum_{k=0}^{n} 2^k = 2 - (\frac{1}{2})^n$

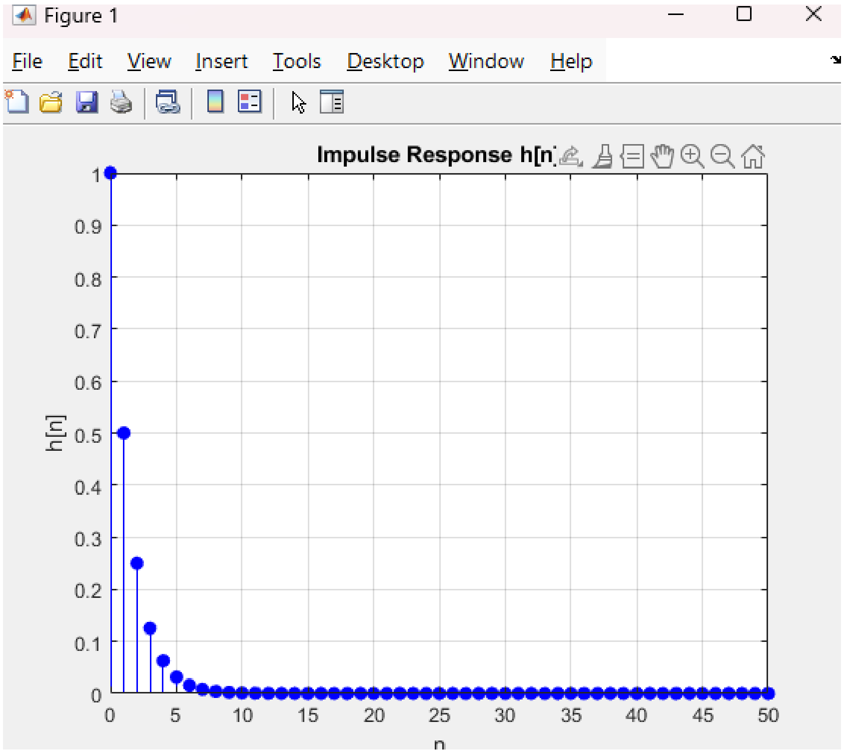
$n > 15$

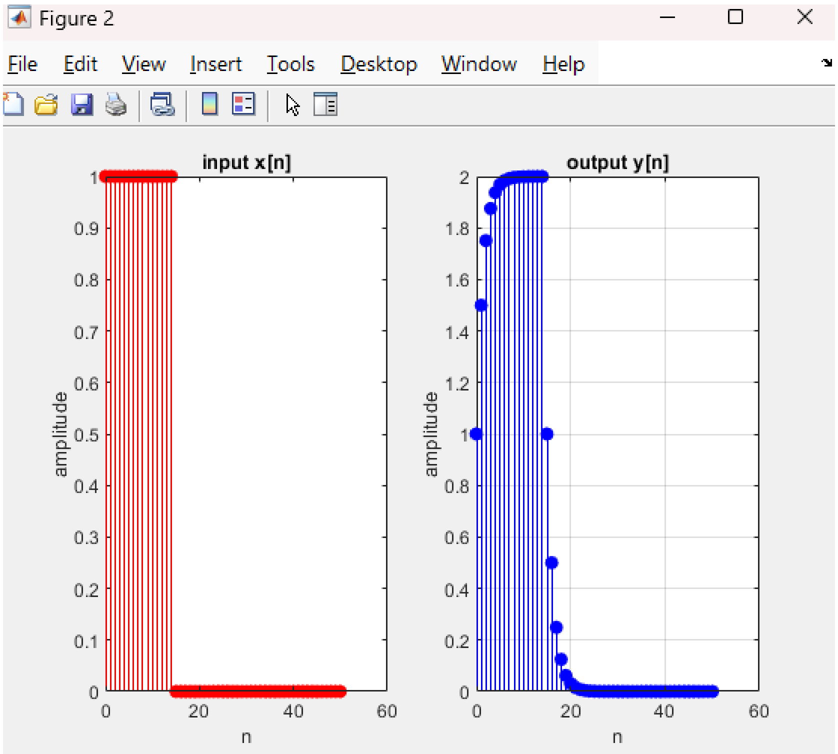
$y(n) = 2 - (\frac{1}{2})^{15}$

**b)**

```
clc;
clear all;
close all;

% Define system coefficients

b = input('enter numerator coefficients = ');
a = input('enter denominator coefficients');

% Compute impulse response
n = 0:50;  % Range of n
impulse = (n == 0);  % Unit impulse signal
h = filter(b, a, impulse);  % Compute impulse response using filter

% Plot the impulse response
figure;
stem(n, h, 'b', 'filled');
xlabel('n');
ylabel('h[n]');
title('Impulse Response h[n]');
grid on;

% Define input x[n] = u[n] - u[n-15]
x = (n >= 0) - (n >= 15);  % Rectangular pulse

% Compute output y[n] using filter
y = filter(b, a, x);

% Plot input x[n] and output y[n]
figure;
subplot(1,2,1);
stem(n, x, 'r', 'filled');
xlabel('n');
ylabel('amplitude');
title('input x[n]')
subplot(1,2,2);
stem(n, y, 'b', 'filled');
xlabel('n');
ylabel('amplitude');
title('output y[n]');
grid on;
```

**Output:**




**Inference:** The impulse response in figure 1 shows an exponential decay indicating a stable system.

The input and output graphs indicate that the system applies h[n] to x[n] so we get a similar response as the input but the signal attenuates over time.

2. Create pole – zero plot of a given transfer function. The function zplane creates a plot of the positions of zeros and poles in the plane of the complex variable z, with the unit circle for reference, starting from the coefficients a and b. where b and a are row vectors. It uses the function roots to calculate the roots of numerator and denominator of the transfer function.

**Create a pole – zero plot for the transfer functions**

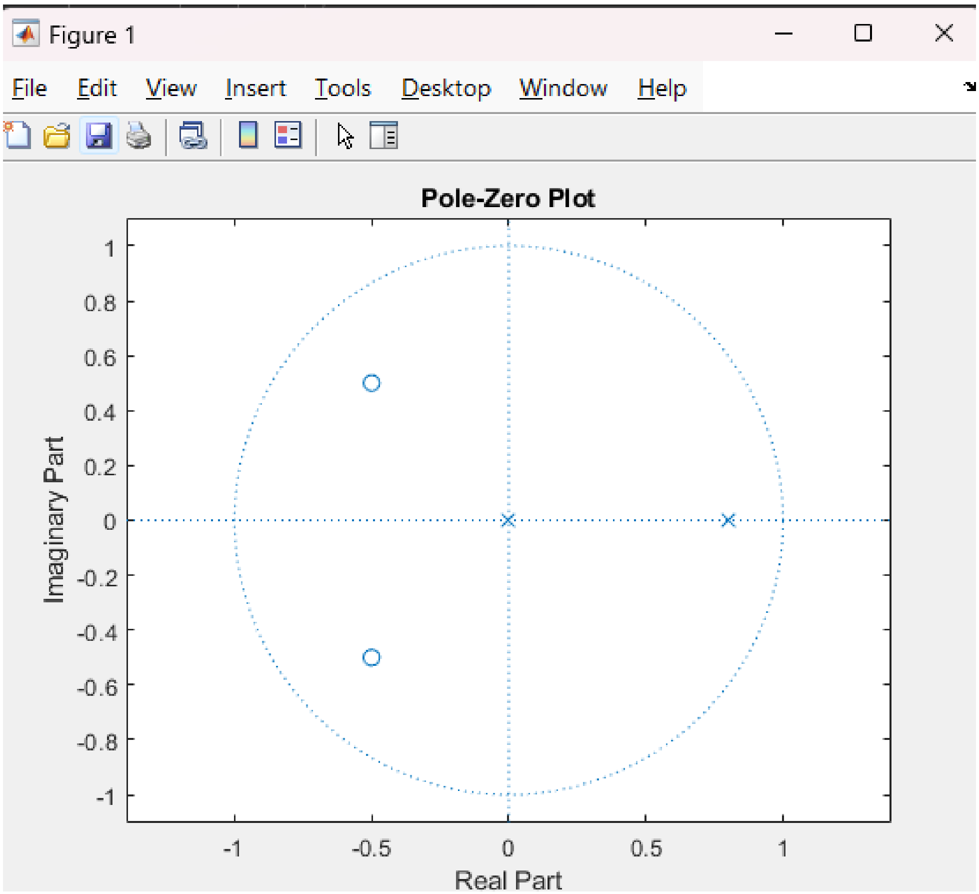
$$H(z) = \frac{2 + 2z^{-1} + z^{-2}}{1 - 0.8z^{-1}}$$

**Code:**

```
% pole zero plot of a given transfer function

clc;
clear all;

num = input('type in the numerator coefficients = ');

den = input('type the denomiator coefficeints = ');


[z,p,k] = tf2zp(num,den);

m = abs(p); %finding the distance from origin of poles

disp('zeros are at;');
disp(z);

disp('poles are at');
disp(p);

disp('gain constant');
disp(k);

disp('radius of poles');
disp(m);

sos = zp2sos(z,p,k);

disp('second order sections');
disp(real(sos));

zplane(num,den);
```

**OUTPUT:**

```
type in the numerator coefficients = [2 2 1]
type the denomiator coefficeints = [1 -0.8 0]
zeros are at;
  -0.5000 + 0.5000i
  -0.5000 - 0.5000i

poles are at
         0
    0.8000

gain constant
     2

radius of poles
         0
    0.8000

second order sections
    2.0000    2.0000    1.0000    1.0000   -0.8000         0
```



**Inference:** the poles occur at 0 and 0.8 indicated by the x while the zeros occur at -0.5 + 0.5i and 0.5 – 0.5i indicated by o.

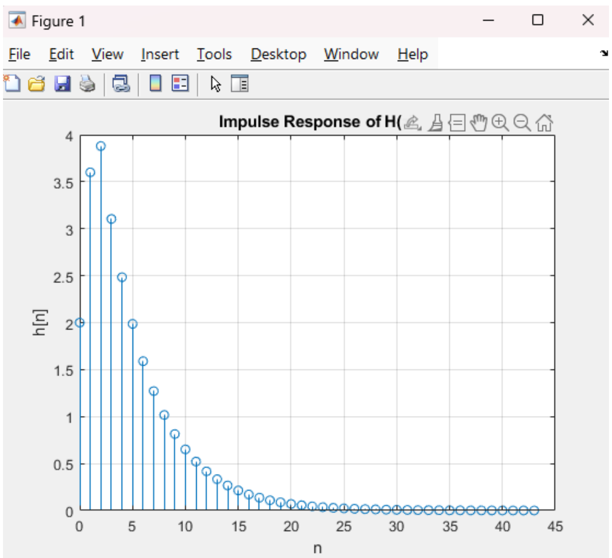
3. Compute the impulse response of a system H(z) given above having coefficients a & b. See ***[h,t]=impz(b,a)*** produces the impulse response in vector h and the time axis in vector t. If the output arguments h and t are omitted, a plot of the impulse response is directly displayed. If the impulse response is of infinite length, only its initial part is computed.

$$H(z) = \frac{2 + 2z^{-1} + z^{-2}}{1 - 0.8z^{-1}}$$

**Code:**

```
% Coefficients of the numerator and denominator
b = input('type in the numerator coefficients = ');

a = input('type the denomiator coefficeints = ');


% Compute the impulse response
[h, t] = impz(b, a);

% Plot the impulse response
stem(t, h);
xlabel('n');
ylabel('h[n]');
title('Impulse Response of H(z)');
grid on;
```

**Output:**



**Inference:** The impulse response shows an exponential decay indicating that the system is stable. Also the system response is non zero indicating that the system is causal.

4. Compute the frequency response of system expressed by difference equation or rational transfer equation. The function **"freqz"** is used to compute the frequency response. Refer ***[H,w]=freqz(b,a,N);*** where N is a positive integer, returns the frequency response H and the vector w with the N angular frequencies at which H has been calculated (i.e. N equispaced points on the unit circle, between 0 and π). If N is omitted, a default value of 512 is assumed. If no output argument is specified, the amplitude plot and the phase plot of the frequency response are directly displayed.

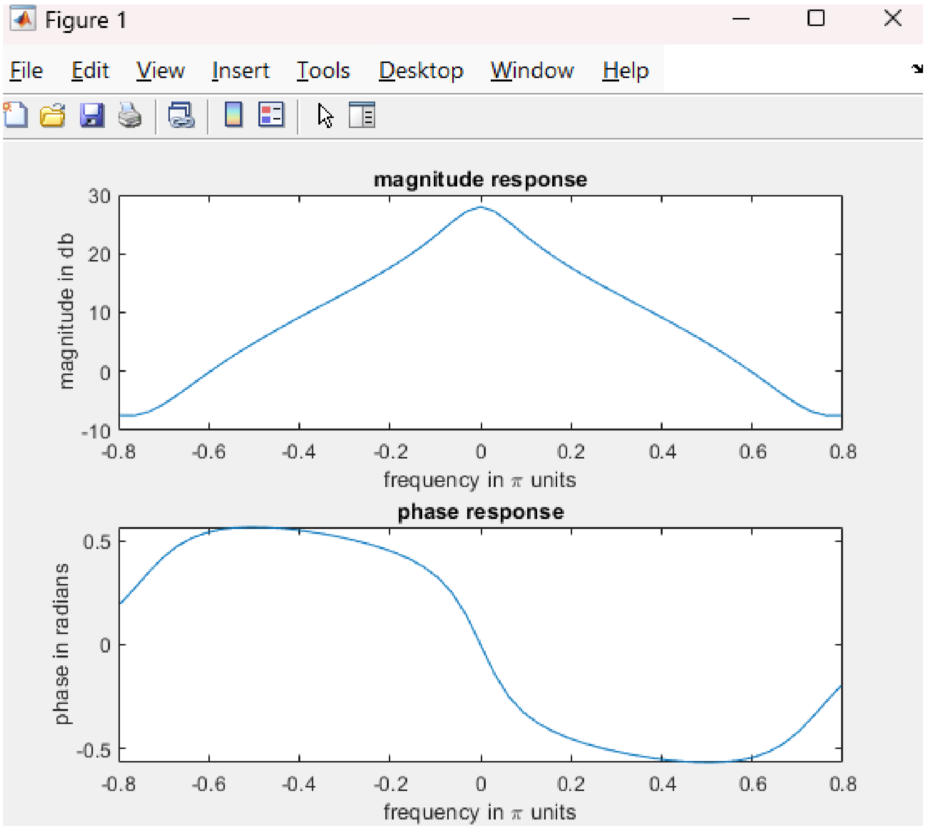
   **Plot the magnitude and phase response for the transfer function H(z) given above.**

$$H(z) = \frac{2 + 2z^{-1} + z^{-2}}{1 - 0.8z^{-1}}$$

**Code:**

```
clc;
clear all;

b = input('enter numerator coefficients = ');
a = input('enter denominator coefficients');

w = [-2.5:1/10:2.5];
[h,w] = freqz(b,a,w);
subplot(211)
plot(w/pi,20*log10(abs(h)));

xlabel('frequency in \pi units');ylabel('magnitude in db');
title('magnitude response');
subplot(212);
plot(w/pi,angle(h)/pi);
xlabel('frequency in \pi units');ylabel('phase in radians');
title('phase response');
```

**Output:**




**Inference:** The magnitude response indicates how the magnitude of the function varies with frequency. Shows attenuation for higher and lower frequencies relative to the peak.

Phase response indicates the variation of phase of the function with frequency. The phase shift occurs continuously with frequency smoothy transitioning through zero indicating a stable system.