BIRZEIT UNIVERSITY

DEPARTMENT OF ELECTRICAL AND COMPUTER ENGINEERING

Digital Signal Processing DSP – ENCS4310

Matlab Assignment Solution

Student:

- **Name:** Rasha Daoud **ID:** 1210382

Instructor: Dr. Abualseoud Hanani

Section: 2

Date: August 29th, 2024

Table Of Contents

Table Of Figures:

Question 1: For the following Signal x[n] Calculate and plot the Spectrum:

$$x[n] = \begin{Bmatrix} 1, n = 1 \dots .10 \\ 0, Otherwise \end{Bmatrix}$$

```
DSP_Q1.m
/MATLAB Drive/DSP_Q1.m
% Rasha Daoud - 1210382
% Define the signal x[n]
n = 1:100;
x = ones(1, 100);

% Calculate the Fourier Transform (Spectrum)
X = fft(x, 1024); % 1024-point FFT for better resolution
X = fftshift(X);  % Shift zero frequency component to the center

% Create frequency axis with omega (rad/sample) from -pi to pi
w = linspace(-pi, pi, 1024); % Frequency axis in radians per sample

% Plot the time-domain signal
figure;
stem(n, x, 'filled');
title('Time Domain Signal x[n]');
xlabel('n (Sample Index)');
ylabel('x[n]');
grid on;

% Plot the magnitude of the Spectrum
figure;
plot(w, abs(X));
title('Magnitude Spectrum of x[n]');
xlabel('Frequency \omega (rad/sample)');
ylabel('|X(\omega)|');
grid on;
xlim([-0.5 0.5]); % Set x-axis limits to clearly show the range -0.5 to 0.5
```

FIGURE 1:QUESTION #1 CODE ON MATLAB.

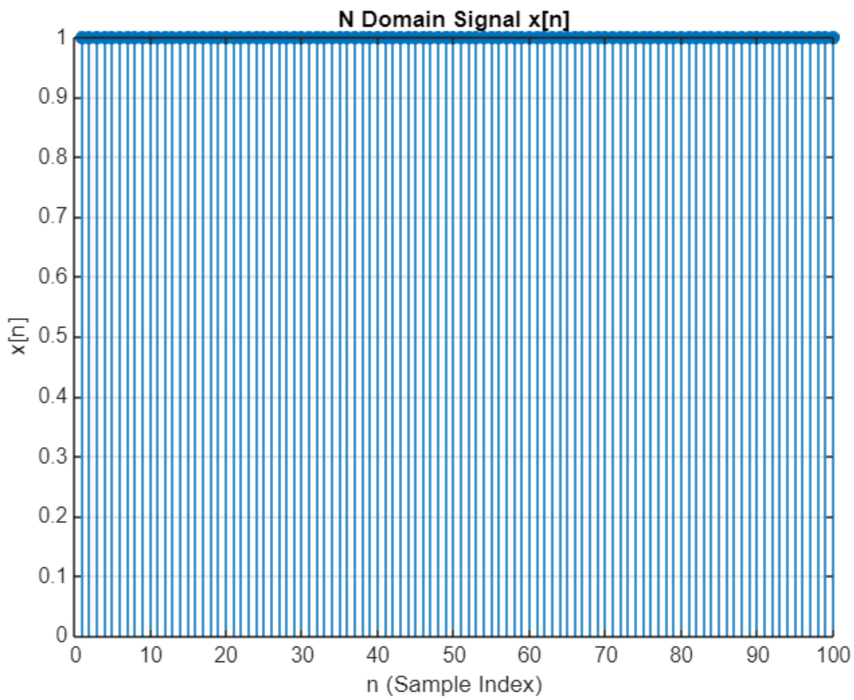

FIGURE 2: X[N] IN N DOMAIN.

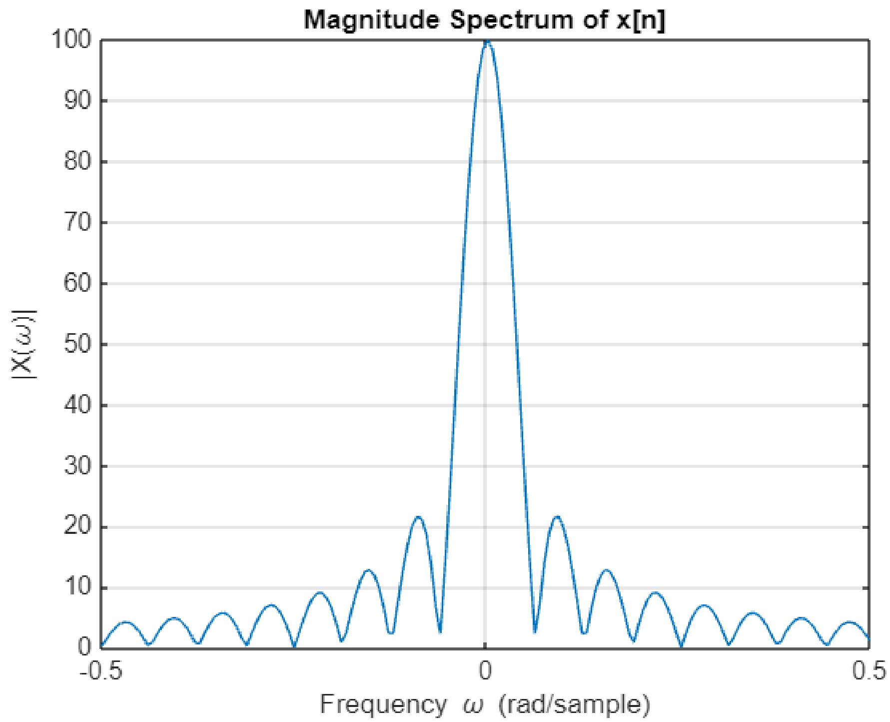

FIGURE 3: MAGNITUDE SPECTRUM OF X[N].

The signal x[n] is a sequence of 100 consecutive ones (x[n]=1 for n=1,2,…,100). This is equivalent to a rectangular pulse of length 100 in the discrete domain.

The spectrum of the signal x[n] shows a big peak at zero frequency because the signal is made of constant values. This tells us that most of the energy is in low frequencies. The graph looks like a wave with a main peak in the center and smaller ripples that fade out. This pattern, called a sinc shape, shows how a simple rectangular signal behaves in the frequency domain, with most of its energy focused near the middle and less at the edges, and having Symmetry about the vertical axis (frequency ω=0).

Question 2: Consider the following signals:

$$x[n] = [0, 0.3, 0.6, 0.8, 1] \text{ and } h[n] = [0.5, 1, 0.5]$$

DSP_Q2.m

/MATLAB Drive/DSP_Q2.m

```
% Rasha Daoud - 1210382
x = [0, 0.3, 0.6, 0.8, 1];
h = [0.5, 1, 0.5];

% Define the length for circular convolution
N = max(length(x), length(h)); % Use the maximum length of x and h

% Compute DFTs using N-point FFT
X = fft(x, N); % N-point FFT
H = fft(h, N); % N-point FFT

% Compute product of DFTs
Y = X .* H;

% Inverse DFT to return to the time domain (circular convolution result)
y_circ = ifft(Y);

% Circular convolution using MATLAB's built-in function (for verification)
y_circ_builtin = cconv(x, h, N);

% Display results
disp('Circular Convolution (Inverse DFT) result:');
disp(y_circ);
disp('Circular Convolution (MATLAB cconv) result:');
disp(y_circ_builtin);

% Plot comparison
figure;
subplot(2,1,1);
stem(y_circ, 'filled');
title('Circular Convolution (Inverse DFT) Result');
xlabel('n');
ylabel('y[n]');
grid on;

subplot(2,1,2);
stem(y_circ_builtin, 'filled');
title('Circular Convolution (MATLAB cconv) Result');
xlabel('n');
ylabel('y[n]');
grid on;
```

FIGURE 4: QUESTION #2 CODE IN MATLAB.

```
>> DSP_Q2
Circular Convolution (Inverse DFT) result:
    1.4000    0.6500    0.6000    1.1500    1.6000

Circular Convolution (MATLAB cconv) result:
    1.4000    0.6500    0.6000    1.1500    1.6000
```

FIGURE 5: RESULT INVERSE DFT AND THE CIRCULAR CONVOLUTION.

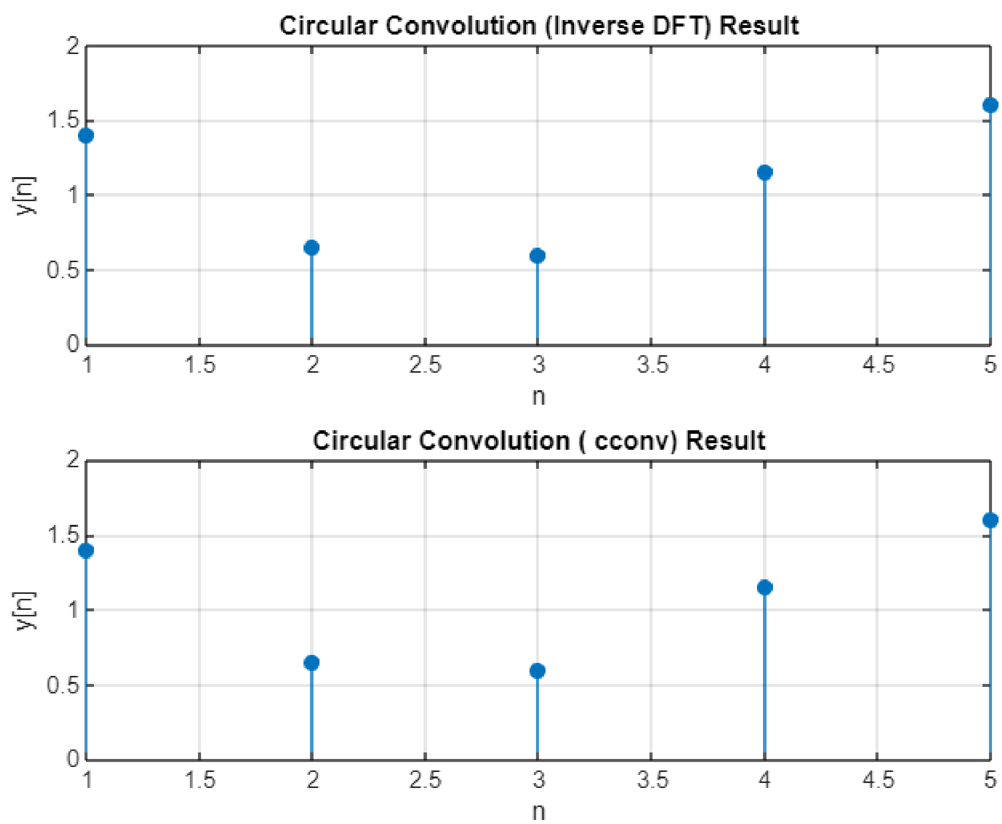

FIGURE 6: CIRCULAR CONVULSION AND THE INVERSE OF IT PLOTS.

The MATLAB code calculates the convolution of two signals x[n] and h[n] using two methods. First, it computes the product of their Discrete Fourier Transforms (DFTs) and then applies the inverse DFT to return to the time domain.
This result is compared to the convolution obtained directly using MATLAB's conv function. Both methods yield the same result, demonstrating that the

convolution computed via the frequency domain approach (DFT and inverse DFT) matches the direct time-domain convolution.

Question 3: Consider the following impulse response of a digital filter

$$h[n] = \{1, 1.5, 1\}$$

DSP_Q3.m × DSP_Q2.m × +

/MATLAB Drive/DSP_Q3.m

```matlab
% Rasha Daoud - 1210382
% Define the impulse response
h = [1, 1.5, 1];

% Compute the poles and zeros
[z, p, k] = tf2zpk(h, 1);

% Plot Pole-Zero Diagram
figure;
zplane(z, p);
title('Pole-Zero Diagram');
grid on;

% Frequency response H(z) for z = exp(jw) over the whole circle
[H, w] = freqz(h, 1, 1024, 'whole');  % 'whole' computes from 0 to 2*pi

% Adjust the frequency vector for proper centering
w_shifted = fftshift(w) - pi;  % Shift and center the frequency range

% Shift the frequency response to match the frequency vector
H_shifted = fftshift(H);

% Plot the magnitude response
figure;
plot(w_shifted/pi, abs(H_shifted));
title('Magnitude Response |H(e^{j\omega})|');
xlabel('Normalized Frequency (\times\pi rad/sample)');
ylabel('|H(z)|');
grid on;
```

FIGURE 7: QUESTION #3 CODE IN MATLAB.

A-Plot the Pole zero diagram

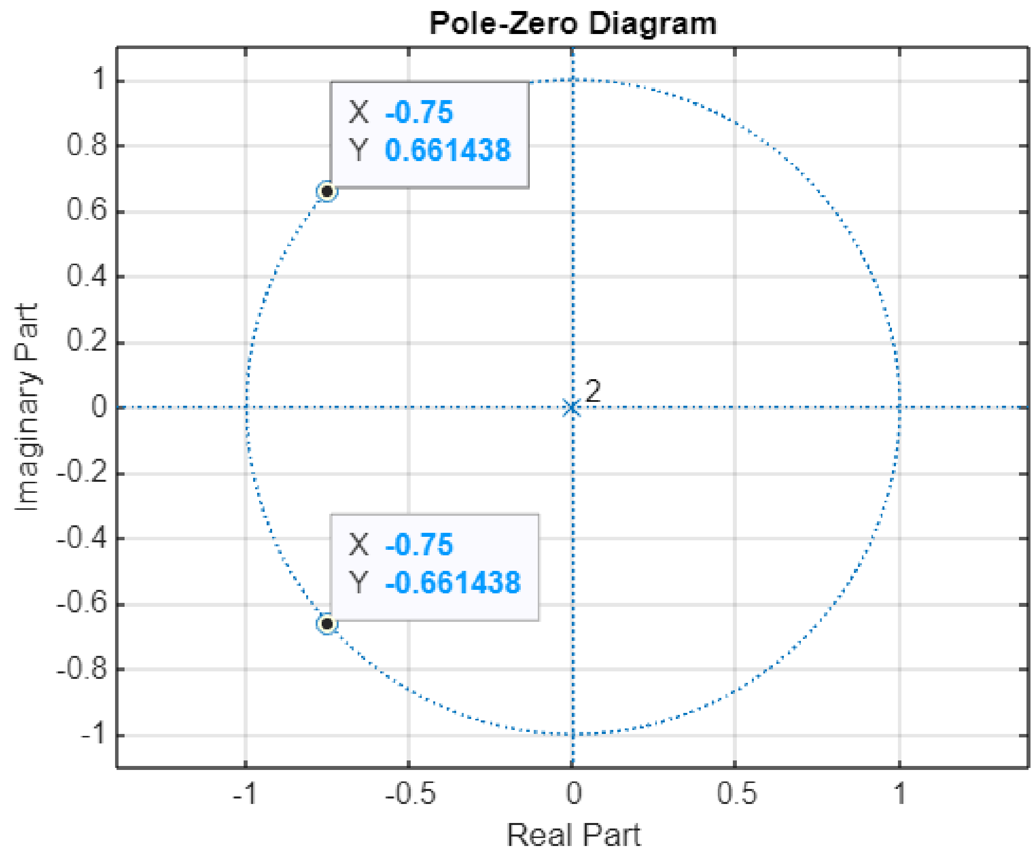

FIGURE 8: POLE-ZERO DIAGRAM.

Since h[n]={1,1.5,1} defines a finite impulse response (FIR) filter, the transfer function H(z) in the Z-domain can be represented as:

$H(z)=1+1.5^{z-1}+1^{z-2}$

$H(z)=0 \Rightarrow z2+1.5z+1=0$

$z = -0.75 \pm j \cdot 0.935$

The zeros are complex conjugates: $z \approx -0.75 \pm 0.935j$

- Two zeros located at $z = -0.75 \pm 0.935j$, plotted as o markers.
- No poles (as this is an FIR filter and the denominator is 1, indicating no poles).
- The zeros are symmetrically placed in the complex plane about the real axis.

B- Plot function $H(z)$ for $z = e^{jw}$

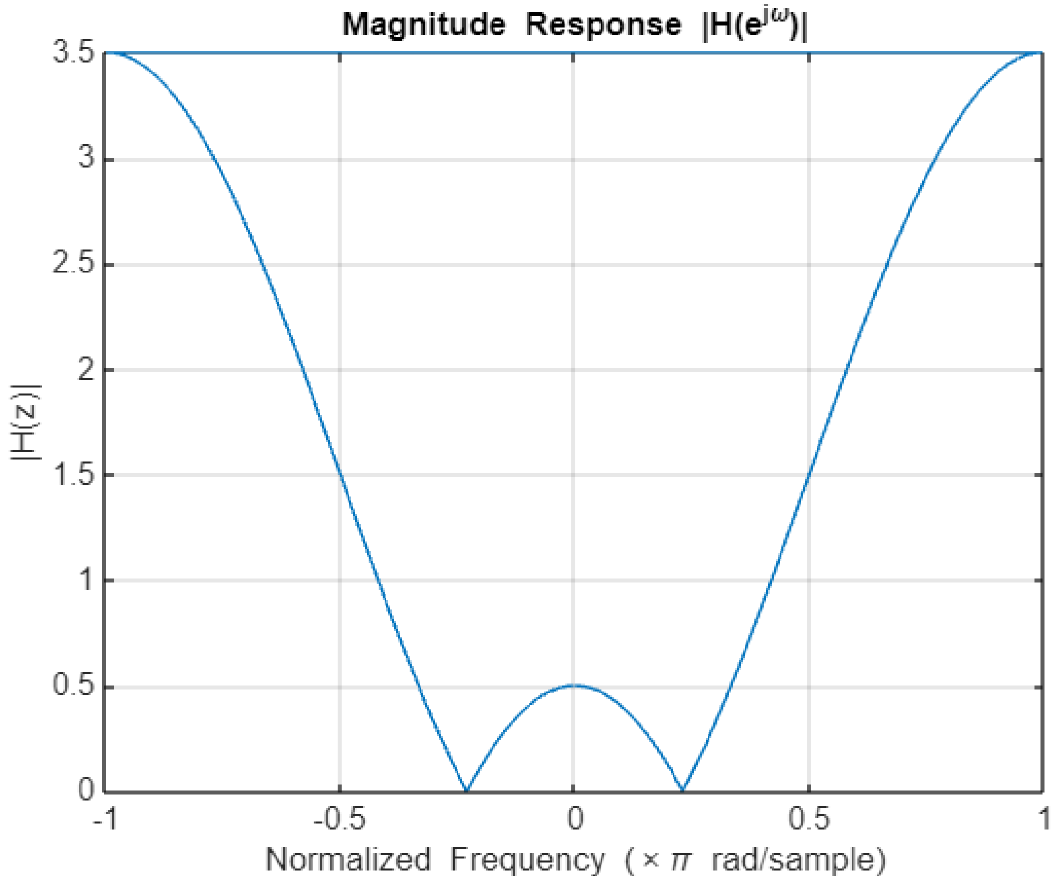

FIGURE 9: MAGNITUDE RESPONSE FOR H(Z) WHERE Z=E^JW.

- The magnitude response reaches its maximum at ω=0 (DC component), which is typical for a low-pass filter.
- The response shows symmetry around the center, indicating that it is plotted over a normalized frequency range from -π to π radians/sample.
- The dips or nulls occur at frequencies corresponding to the zeros of the filter, which can be seen in the lower magnitude near ω = ±π.

The End