NAMAL UNIVERSITY MIANWALI
DEPARTMENT OF ELECTRICAL ENGINEERING

EE 345 (L) – Digital Signal Processing (Lab)
LAB # 06
REPORT

Title :

Analysis of Z- Transform, Inverse Z-Transform and Pole Zero Map for Discrete Time systems

Name	Fahim Ur Rehman Shah
Roll No	NIM-BSEE-2021-24
Intructor	Zulaikha Kiran
Lab Engineer	Engr. Faizan Ahmad
Date Performed	03-April-2024
Marks	

Introduction

The purpose of this lab is to revise some MATLAB concepts and implementations including ZTransform, Inverse Z-Transform and how poles and zeros are affected to the stability of the system.

Course Learning Outcomes

CLO1: Develop algorithms to perform signal processing techniques on digital signals using MATLAB and DSP Kit DSK6713

CLO3: Deliver a report/lab notes/presentation/viva, effectively communicating the design and analysis of the given problem

Equipment

- Software
  - MATLAB

Instructions

1. This is an individual lab. You will perform the tasks individually and submit a report.
2. Some of these tasks are for practice purposes only while others (marked as ‘Exercise’) have to be answered in the report.
3. When asked to display an image/ graph in the exercise either save it as jpeg or take a screenshot, in order to insert it in the report.
4. The report should be submitted on the given template, including:
    a. Code (copy and pasted, NOT a screenshot)
    b. Output values (from command window, can be a screenshot)
    c. Output figure/graph (as instructed in 3)
    d. Explanation where required
5. The report should be properly formatted, with easy to read code and easy to see figures.
6. Plagiarism or any hint thereof will be dealt with strictly. Any incident where plagiarism is caught, both (or all) students involved will be given zero marks, regardless of who copied whom. Multiple such incidents will result in disciplinary action being taken.

Z-Transform

The z-transform of signal x[n] can be represented by the formula

$$X(z) = \mathcal{Z}\{x[n]\} = \sum_{n=-\infty}^{\infty} x[n]z^{-n}$$

Where we will assume that z represents any complex number i.e., z is the independent (complex) variable of the z-transform X(z).

We suppose that x[n] is unit impulse sequence

n-domain $\xrightarrow{z}$ z-domain

$$x[n] = \delta[n - no] \xrightarrow{z} X(z) = z^{-no}$$

TASK-01:

Compute the z-transform of following functions

Table of Z – Transforms

	f_n	$F(z)$
1.	$(1)^n$	$\frac{z}{z-1}$
2.	$(-1)^n$	$\frac{z}{z+1}$
3.	a^n	$\frac{z}{z-a}$
4.	n	$\frac{z}{(z-1)^2}$
5.	n^2	$\frac{z^2+z}{(z-1)^3}$

```
% Define symbolic variable
syms n z;

% Define functions
fn1 = 1^n;
fn2 = (-1)^n;
a = sym('a'); % Define symbolic variable 'a'
fn3 = a^n;
fn4 = n;
fn5 = n^2;
% Compute Z-transforms
F1 = ztrans(fn1, n, z);
F2 = ztrans(fn2, n, z);
F3 = ztrans(fn3, n, z);
F4 = ztrans(fn4, n, z);
F5 = ztrans(fn5, n, z);

% Display Z-transform results
disp('Z-transform of f1(n) = 1^n:')
```

```
Z-transform of f1(n) = 1^n:
```

```
disp(F1)
```

$$\frac{z}{z-1}$$

```
disp('Z-transform of f2(n) = (-1)^n:')
```

```
Z-transform of f2(n) = (-1)^n:
```

```
disp(F2)
```

$$\frac{z}{z+1}$$

```
disp('Z-transform of f3(n) = a^n:')
```

```
Z-transform of f3(n) = a^n:
```

```
disp(F3)
```

$$-\frac{z}{a-z}$$

```
disp('Z-transform of f4(n) = n:')
```

```
Z-transform of f4(n) = n:
```

```
disp(F4)
```

$$\frac{z}{(z-1)^2}$$

```
disp('Z-transform of f5(n) = n^2:')
```

```
Z-transform of f5(n) = n^2:
```

```
disp(F5)
```

$$\frac{z\,(z+1)}{(z-1)^3}$$

Explanation:

In the provided MATLAB code, I defined symbolic variables 'n' and 'z' and then defined several functions using these variables. I also defined a symbolic variable 'a' and used it in another function. After that, I computed the Z-transforms of these functions using the 'ztrans' function. Finally, I displayed the Z-transform results for each function using the 'disp' function.

TASK-02:

Find Inverse z-transform of:

$$H(z) = \frac{\left(20z^2 - 30z + 11.2\right)}{z^3 - 0.3z^2 - 0.58z + 0.24}$$

```
clc;
clear all;

% Define symbolic variable
syms z;

% Define transfer function H1
H1 = ((20*z^2) - (30*z) + 11.2) / (z^3 - 0.3*z^2 - 0.58*z + 0.24);

% Find inverse z-transform for H1
inv_H1 = iztrans(H1, z);

% Display the inverse z-transform result for H1
disp('Inverse z-transform of H1(z):');
```

```
Inverse z-transform of H1(z):
```

```
disp(inv_H1);
```

$$\frac{100\left(\frac{3}{5}\right)^z}{21} - \frac{240\left(\frac{1}{2}\right)^z}{13} - \frac{3000\left(-\frac{4}{5}\right)^z}{91} + \frac{140\,\delta_{z,0}}{3}$$

$$H(z) = \frac{z^2}{(z-\frac{1}{2})(z-\frac{1}{4})}$$

```
% Define transfer function H2
H2 = z^2 / ((z - 1/2)*(z - 1/4));

% Find inverse z-transform for H2
inv_H2 = iztrans(H2, z);

% Display the inverse z-transform result for H2
disp('Inverse z-transform of H2(z):');
```

```
Inverse z-transform of H2(z):
```

```
disp(inv_H2);
```

$$2\left(\frac{1}{2}\right)^z - \left(\frac{1}{4}\right)^z$$

$$H(z) = \frac{20(z-0.8)(z-0.7)}{(z-0.6)(z+0.8)(z-0.5)}$$

```
% Define transfer function H3
H3 = (20*(z - 0.8)*(z - 0.7)) / ((z - 0.6)*(z + 0.8)*(z - 0.5));

% Find inverse z-transform for H3
inv_H3 = iztrans(H3, z);

% Display the inverse z-transform result for H3
disp('Inverse z-transform of H3(z):');
```

```
Inverse z-transform of H3(z):
```

```
disp(inv_H3);
```

$$\frac{100\left(\frac{3}{5}\right)^z}{21} - \frac{240\left(\frac{1}{2}\right)^z}{13} - \frac{3000\left(-\frac{4}{5}\right)^z}{91} + \frac{140\,\delta_{z,0}}{3}$$

Explanation:

In the provided MATLAB code snippet, I started by clearing the command window and workspace. Then, I defined a symbolic variable 'z' and proceeded to define three transfer functions, H1, H2, and H3, in terms of 'z'. Next, I found the inverse Z-transform for each transfer function using the 'iztrans' function. Finally, I displayed the results of the inverse Z-transform for each transfer function using the 'disp' function.

TASK-03:

Consider the following difference equation

$$y[n] = 1/3\, x[n] - 1/3\, x[n-1] + 1/3\, x[n-2]$$

- Determine H(z)
- Determine the impulse response h[n] using MATLAB

  By using ***impz (b, a)***

  Where b is numerator

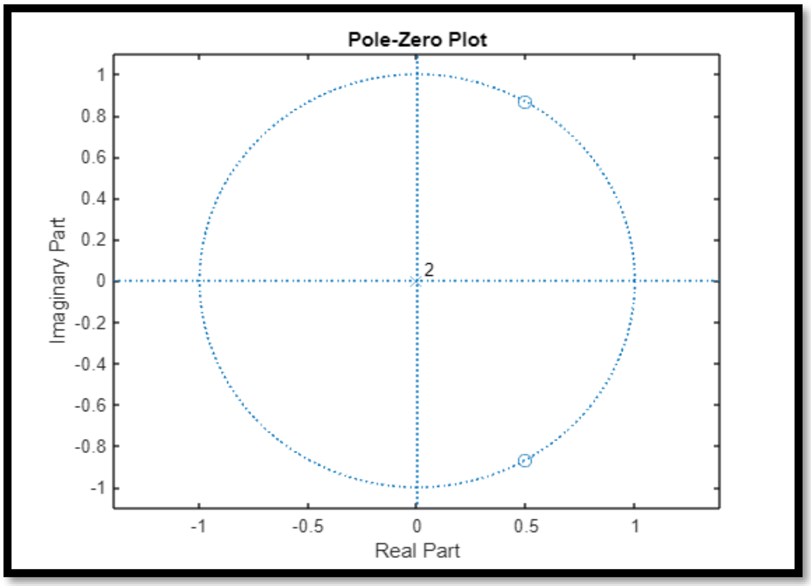
  a is denominator
- The pole-zero diagram using [**z**, p] =tf2zp(b,a) and zplane(**z**,p). Note, to get complete pole-zero plots, you have to match the lengths of a and b by zero-padding the shorter vector.
- Comment about the stability of the above system

```
% Define the coefficients of the difference equation
b = [1/3, -1/3, 1/3];
a = [1, 0, 0]; % Denominator coefficients

% Determine H(z) from the difference equation coefficients
Hz = tf(b, a, 1); % Create transfer function

% Determine the impulse response h[n] using MATLAB
h = impz(b, a); % Get impulse response

% Zero-pad the shorter vector to match lengths
if length(b) < length(a)
    b = [b, zeros(1, length(a) - length(b))];
elseif length(a) < length(b)
    a = [a, zeros(1, length(b) - length(a))];
end
% Get the pole-zero diagram
[z, p] = tf2zp(b, a); % Get zeros and poles
figure;
zplane(z, p); % Plot pole-zero diagram
```



```
% Comment about the stability of the system
if all(abs(p) < 1) % Check if all poles are inside the unit circle
    disp('The system is stable.');
else
    disp('The system is unstable.');
end
```

```
The system is stable.
```

Explanation:

In the provided MATLAB code snippet, the coefficients of a difference equation are defined to represent a system. The transfer function 'H(z)' is determined from these coefficients, and the impulse response 'h[n]' of the system is calculated using MATLAB. If the lengths of the numerator and denominator coefficients are different, the shorter vector is zero-padded to match lengths. The pole-zero diagram of the system is then plotted to visualize the distribution of poles and zeros. Finally, a comment is displayed indicating whether the system is stable or unstable based on the location of the poles relative to the unit circle in the complex plane.

TASK-04:

Determine the z-transform from its zero and pole locations.
The zeros are at
z1=0.21, z2=3.14, z3=-0.3+j0.5, z4=-0.3-j0.5;
the poles are at
p1=-0.45, p2=0.67, p3=0.81+j0.72, p4=0.81-j0.72;
and the gain constant k is2.2.
Using command zp2tf

```
% Define the zero and pole locations as column vectors
zeros = [0.21; 3.14; -0.3+0.5i; -0.3-0.5i];
poles = [-0.45; 0.67; 0.81+0.72i; 0.81-0.72i];
k = 2.2; % Gain constant

% Convert zero-pole-gain representation to transfer function
[num, den] = zp2tf(zeros, poles, k);

% Display the transfer function
```

```
disp('The transfer function coefficients are:');
```

```
The transfer function coefficients are:
```

```
disp(['Numerator: ', num2str(num)]);
```

```
Numerator: 2.2        -6.05      -2.2233      -1.6354      0.49323
```

```
disp(['Denominator: ', num2str(den)]);
```

```
Denominator: 1        -1.84       1.2294      0.23004     -0.35411
```

Explanation:

In the provided MATLAB code, zero locations and pole locations are defined as column vectors, along with a gain constant 'k'. These values are used to convert the zero-pole-gain representation to a transfer function using the 'zp2tf' function. The resulting transfer function coefficients are then displayed, showing the numerator and denominator coefficients of the transfer function.

Conclusion:

In the Digital Signal Processing (DSP) lab Tasks, I worked on various MATLAB scripts to explore different aspects of signal processing. In the first set of codes, I focused on symbolic computation and Z-transform analysis. I defined symbolic variables, created functions, and computed Z-transforms for each function. I then displayed the results to understand the transformations applied to these functions in the Z-domain. In the second set of codes, I delved into transfer functions, inverse Z-transforms, and stability analysis. I defined transfer functions, calculated inverse Z-transforms, and plotted pole-zero diagrams to visualize the system's characteristics. I evaluated the stability of the systems based on the locations of poles in relation to the unit circle. Additionally, I demonstrated the conversion of zero-pole-gain representations to transfer functions, providing insight into system modeling and analysis in the DSP domain.

Evaluation Rubric

- **Method of Evaluation**: In-lab marking by instructors, Report submitted by students
- **Measured Learning Outcomes**:

CLO1: Develop algorithms to perform signal processing techniques on digital signals using MATLAB and DSP Kit DSK6713
CLO3: Deliver a report/lab notes/presentation/viva, effectively communicating the design and analysis of the given problem

	Excellent 10	Good 9-7	Satisfactory 6-4	Unsatisfactory 3-1	Poor 0	Marks Obtained
Tasks (CLO1)	All tasks completed correctly. Correct code with proper comments.	Most tasks completed correctly.	Some tasks completed correctly.	Most tasks incomplete or incorrect.	All tasks incomplete or incorrect.	
Output (CLO1)	Output correctly shown with all Figures/Plots displayed as required and properly labelled	Most Output/Figures/Plots displayed with proper labels	Some Output/Figures/Plots displayed with proper labels OR Most Output/Figures/Plots displayed but without proper labels	Most of the required Output/Figures/Plots not displayed	Output/Figures/Plots not displayed	
Answers (CLO1)	Meaningful answers to all questions. Answers show the understanding of the student.	Meaningful answers to most questions.	Some correct/ meaningful answers with some irrelevant ones	Answers not understandable/ not relevant to questions	Not Written any Answer	
Report (CLO3)	Report submitted with proper grammar and punctuation with proper conclusions drawn and good formatting	Report submitted with proper conclusions drawn with good formatting but some grammar mistakes OR proper grammar but not very good formatting	Some correct/ meaningful conclusions. Some parts of the document not properly formatted or some grammar mistakes	Conclusions not based on results. Bad formatting with no proper grammar/punctuation	Report not submitted	
Total						