CT 214 Analog and Digital Communications

Lab # 1: Introduction to LABVIEW

Guidelines:

- **This is your first lab with LabVIEW, please try to explore as much as you can.**
- **Please attach the corresponding code and plots along with your final report.**
- **Ensure that you answer all the questions that have been asked as part of this lab.**

1. **Aim:** The purpose of this lab is to get you started with the software LABVIEW.

2. **Introduction to LABVIEW:** Similar to MATLAB (in case if you already know it) LABVIEW has got a MathScript window that can be used to do scientific computing. Most of the commands used in LABVIEW are similar to those in MATLAB. The MathScript window is a latest addition to the LABVIEW software. Just like in MATLAB, even in the MathScript window, you can create script files (that are stored with .m extension) to implement your code which can be saved and re-used in other programs.

   The MathScript window contains a number of functions in various areas like communications, signal processing, mathematics, etc., that can facilitate scientific computing. Just like most programming languages it even has control structures like (IF...ELSE), (FOR), (WHILE), etc.,.

3. **Procedure to Open MathScript Window in LabVIEW:**

   (a) Start the computer, log in using your student id.

   (b) Double click on the National Instruments LabVIEW 2009 icon on the desktop (in case if the icon is not available on desktop click start, go to all programs, locate the National Instruments LabVIEW 2009 icon and click it to execute the corresponding .exe).

   (c) You should then see a "Getting Started" window open. In that window go to the Tools menu and click on "MathScript window"

   (d) A separate window called "LabVIEW MathScript" should open which consists of the following three sub-windows: Command Window, Output Window and Variables/Script/History Window.

   (e) The command window can be used to perform instantaneous computation, the output of which can be seen in Output Window.

   (f) You can click on the Script tab in the third window and create a new script which can be stored by clicking on the save button.

   (g) Click on the Help menu in the main "LabVIEW MathScript window" and choose "Search the LabVIEW Help" to further explore the various commands.

4. **Few Basic Commands...**

(a) **Declaration and assignment of variables:** In this declaration and assignment of variables can be done in one go as follows: `a=15;`

(b) **Creation of a row vector:** `row_vect=[1 2 3 4 5];`

(c) **Creation of a column vector:** `col_vect=[1;2;3;4;5];`

(d) **Creation of a matrix:** `mat=[1 4 9 16;25 36 49 64];` (of size 2×4).

(e) **Transpose of a matrix:** `mat=[1 4 9 16;25 36 49 64]';`

(f) **Creating an identity matrix:** `identity_matrix = eye(n); (n specifies the size of the matrix;)`

(g) **Value of pi:** `pi;`

(h) **Generating a complex number:** `complex_no=(1.25 + i*3.5);`

(i) **Absolute value of a complex number:** `x=abs(complex_no);`

(j) **Phase of a complex number:** `y=angle(complex_no);`

(k) **Element wise operations:** `c=a.b;d=a.b̂;`

(l) **Inverse of a matrix:** `inv_of_matrix=inv(a);`

(m) **Maximum and minimum element of an array:** `max_in_array=max(a); min_in_array(a); (there are variant of these methods which give you even the location of the maximum/minimum element of the array).`

(n) **Ceil:** `y=ceil(a) (rounds the given real number to the nearest integer above)`

(o) **Floor:** `y=floor(a) (rounds the given real number to the nearest integer down)`

(p) **Sine/Cosine of a value:** `sine_value=sin(x)` (where x is a real number). Likewise the command `cos` computes cosine of the given real number.

(q) **Plot command:** `plot(sin([0:0.1:10]));` (explore various versions of this command; also figure out from help how to insert: legend inside the plot, xlabel, ylabel, title, etc.)

(r) **FFT/IFFT commands:** `fft_of_a=fft(a); ifft_of_d=ifft(d);`

5. **Few Exercises**

(a) In the above list I have mentioned a few important commands that might be useful. Please use the LabVIEW help to explore the entire list of commands and state a minimum of 5 commands (that are not part of the above list) along with a brief description of what they do??

(b) Compute the Fourier transform (FT) of the signal $g(t) = \exp(-2t)u(t)$, where $u(.)$ is the unit step function.

(c) Next Consider the following code which computes the FT of $g(t)$ (given above) using the FFT method in MathScript. Understand what each and every command is doing (there might be a few new ones):

```
clear all;
clc;
t_s=0.01;
t=0:t_s:10;
exp_t=t_s*exp(-2.*t);
fft_exp_t=fftshift(fft(exp_t));
mag_exp_t=abs(fft_exp_t);
phase_exp_t=angle(fft_exp_t);
```

```
figure(1);
stem(mag_exp_t);
title('Plot of Magnitude Response of the given signal')
figure(2);
stem(phase_exp_t);
title('Plot of Phase Response of the given signal')
```

Based on the nature of the magnitude and phase response that the above command generates what can you say about the FT of the given signal (and therefore in general for any real signal).